NIST Special Publication 1136

2012 Proceedings of the Performance Metrics for Intelligent Systems (PerMI '12) Workshop

Rajmohan Madhavan
Elena R. Messina
Brian A. Weiss

http://dx.doi.org/10.6028/NIST.SP.1136

National Institute of Standards and Technology
U.S. Department of Commerce

NIST Special Publication 1136

2012 Proceedings of the Performance Metrics for Intelligent Systems (PerMI '12) Workshop

Rajmohan Madhavan
Elena R. Messina
Brian A. Weiss
Intelligent Systems Division
Engineering Laboratory

http://dx.doi.org/10.6028/NIST.SP.1136

November 2012

U.S. Department of Commerce
Rebecca Blank, Acting Secretary

National Institute of Standards and Technology
Patrick D. Gallagher, Under Secretary of Commerce for Standards and Technology and Director

Certain commercial entities, equipment, or materials may be identified in this document in order to describe an experimental procedure or concept adequately. Such identification is not intended to imply recommendation or endorsement by the National Institute of Standards and Technology, nor is it intended to imply that the entities, materials, or equipment are necessarily the best available for the purpose.

National Institute of Standards and Technology Special Publication 1136
Natl. Inst. Stand. Technol. Spec. Publ. 1136, 243 pages (November 2012)
http://dx.doi.org/10.6028/NIST.SP.1136
CODEN: NSPUE2

PERFORMANCE METRICS
FOR
INTELLIGENT SYSTEMS (PERMIS) WORKSHOP

The Marriott Inn & Conference Center, College Park, MD USA
March 20 - 22, 2012

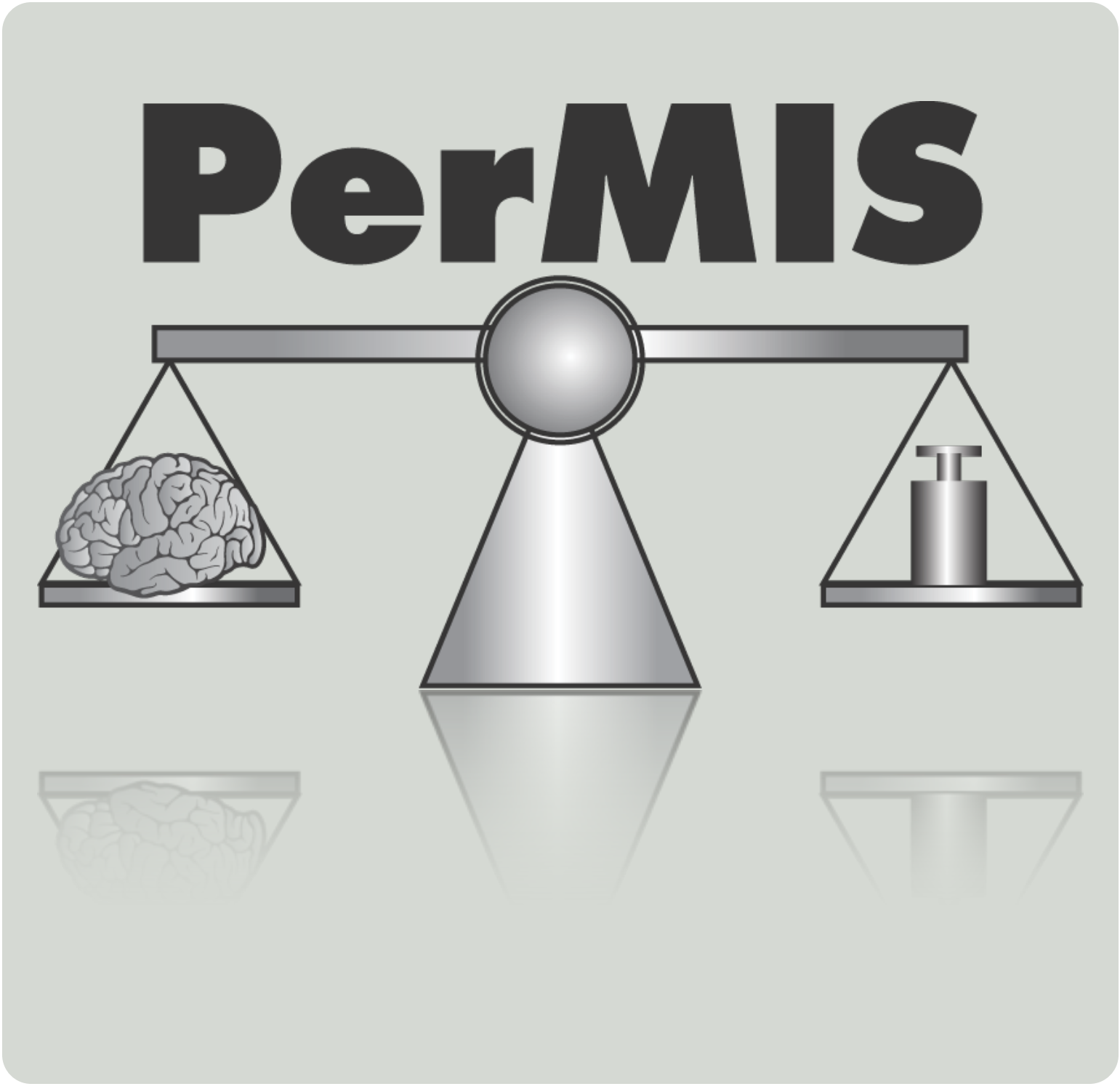


Table of Contents

Foreword.....v

Program Committee.....vi

Plenary Speakers.....vii

Workshop Program at a Glance.....xii

Workshop Program..... xiii

Author Index.....xix

Sponsors.....xx

Acknowledgements xx

Technical Sessions

TUE-AM1 Performance Evaluation

Performance Evaluation of Robotic Knowledge Representations (PERK)
[Craig Schlenoff, Sebti Foufou, Stephen Balakirsky].....1

A Hybrid Approach to 2D Robotic Map Evaluation
[Ross Creed, Kristiyan Georgiev, Rolf Lakaemper] 9

An Overview of Robot-Sensor Calibration Methods for Evaluation of Perception Systems
[Mili Shah, Roger Eastman, Tsai Hong].....15

On the Performance Evaluation of a Vision-Based Human-Robot Interaction Framework
[Junaed Sattar, Gregory Dudek].....21

Functional Requirements of a Model for Kitting Plans
[Stephen Balakirsky, Zeid Kootbally, Thomas Kramer, Raj Madhavan, Craig Schlenoff, Michael Shneier] 29

TUE-AM2 Special Session I: Session Honoring the Legacy of Jim Albus and Alex Meystel

Army-NIST Robotics Teaming: A Thirty Year Retrospective
*[Charles Shoemaker]**

ATR's DoD programs with RoboCrane Technologies
*[Jackson Yang]**

Mind, Brain, and Intelligence: A Krasnow Institute Perspective on the Jim Albus Legacy
*[Kenneth DeJong, Alexei Samsonovich, James Olds]**

Understanding 'Intelligence': in Pursuit of Multi-Resolutional Hierarchies
*[Predrag Filipovic]**

Measurable and Scalable Methods for Modern Knowledge Processing
*[Michael Meystel]**

Birth of an Architecture
*[Alberto Lacaze]**

TUE-PM1 Performance Measures and Metrics

The New Method for Measuring Absolute Threshold of Haptic Force Feedback
[Michal Baczynski] *37*

Approach for Defining Intelligent Systems Technical Performance Metrics
[JoWael Hafez] *41*

Metrics for Planetary Rover Planning & Scheduling Algorithms
[Juan Delfa Victoria, Nicola Policella, Marc Gallant, Oskar von Stryk,
Alessandro Donati, Yang Gao] *47*

Measures for UGV to UGV Collaboration
[Michael Del Rose, Anthony Finn, Robert Kania] *53*

TUE-PM2 Special Session II: Performance Evaluation and Advanced Algorithms for Static and Dynamic 6DOF

2011 Solutions in Perception Challenge Performance Metrics and Results
[Jeremy Marvel, Tsai Hong, Elena Messina] *59*

Development of an Apparatus for Characterizing the Measurement Latency of a Dynamic 3D Tracking System
*[Kamel Saidi]**

Shape-based Pose Estimation Evaluation using Expectivity Index Artifacts
[Chad English, Galina Okouneva, Aradhana Choudhuri] *64*

Ground Truth for Evaluating 6 Degrees of Freedom Pose Estimation Systems
[Jeremy Marvel, Joe Falco, Tsai Hong] *69*

Performance Measurement with 6DOF Laser Tracker Technologies
*[Zach Ryan, Aaron Sabino]**

WED-AM1 Human-Robot Collaboration and Interaction

A Proxemic-Based HRI Testbed
[Zachary Henkel, Robin Murphy, Vasant Srinivasan, Cindy Bethel] 75

Synergistic Methods for Using Language in Robotics
[Ching Teo, Yezhou Yang, Cornelia Fermuller, Yiannis Aloimonos] 82

Reusable Semantic Differential Scales for Measuring Social Response to Robot
[Lilia Moshkina] 89

Levels of Human and Robot Collaboration for Automotive Manufacturing
[Jane Shi, Glenn Jimmerson, Tom Pearson, Roland Menassa] 95

Towards Measuring the Quality of Interaction: Communication through Telepresence Robots
[Katherine Tsui, Munjal Desai, Holly Yanco] 101

WED-AM2 Special Session III: Panel Discussion: Technology Readiness for Randomized Bin Picking Solutions

Technology Readiness Levels for Randomized Bin Picking
[Jeremy Marvel, Roger Eastman, Geraldine Cheok, Kamel Saidi, Tsai Hong, Elena Messina] 109

WED-PM1 Performance Characterization

Characterizing Performance Guarantees for Multiagent, Real-Time Systems Operating in Noisy and Uncertain Environments
[Damian Lyons, Ronald Arkin, Stephen Fox, Shu Jiang, Prem Nirmal, Munzir Zafar] 114

Design, Fabrication and Characterization of the Single-Layer Out-of-Plane Electrothermal Actuator for a MEMS XYZ Stage
[Yong-Sik Kim, Nicholas Dagalakis, Satyandra Gupta] 122

Intelligent Energy Management: Impact of Demand Response and Plug-in Electric Vehicles in a Smart Grid Environment
[Seshadri Raghavan, Alireza Khaligh] 129

Characterization of Forward Rectilinear-Gait Performance for a Snake-Inspired Robot
[James Hopkins, Satyandra Gupta] 136

Emergency Response Robot Evaluation Exercise
[Adam Jacoff, Hui-Min Huang, Ann Virts, Anthony Downs, Raymond Sheh] 145

WED-PM2 Field Testing and Standard Test Methods

Test Method for Measuring Station-Keeping With Unmanned Marine Vehicles Using Sonar or Optical Sensors
[Asish Ghoshal, Avinash Parnandi, Robin Murphy].....*155*

Standard Test Procedures and Metric Development for Automated Guided Vehicle Safety Standards
[Roger Bostelman, William Shackleford, Geraldine Cheok, Richard Norcross].....*160*

Integrating Occlusion Monitoring into Human Tracking for Robot Speed and Separation Monitoring
[William Shackleford, Sandor Szabo, Richard Norcross, Jeremy Marvel].....*168*

Robotics Collaborative Technology Alliance (RCTA) 2011 Baseline Assessment
[Barry Bodt, Richard Camden, Marshal Childers] *174*

Using Competitions to Advance the Development of Standard Test Methods for Response Robots
[Adam, Jacoff, Raymond Sheh, Ann Virts, Tetsuya Kimura, Johannes Pellenz, Soren Schwertfeger, Jackrit Suthakorn].....*182*

THU-PM Performance Testing and Validation

Validation of the Dynamics of an Humanoid Robot in USARSim
[Sander van Noort, Arnoud Visser].....*190*

Evaluation of Robotic Minimally Invasive Surgical Skills using Motion Studies
[Seung-Kook Jun, Madusudanan Sathianarayanan, Abeer Eddib, Pankaj Singhal, Sudha Garimella, Venkat Krovi].....*198*

Multi-Relationship Evaluation Design: Modeling an Automatic Test Plan Generator
[Brian Weiss, Linda Schmidt] *206*

An IEEE 1588 Performance Testing Dashboard for Power Industry Requirements
[Julien Amelot, Ya-Shian Li-Baboud, Clement Vasseur, Jeffrey Fletcher, Dhananjay Anand, James Moyne].....*216*

*Note: * Presentation Only*

FOREWORD

Welcome to PerMIS'12!

As software and hardware become increasingly interwoven, new opportunities and challenges emerge. The field of Cyber-Physical Systems (CPS) – hybrid networked cyber and engineered physical elements co-designed to create adaptive and predictive systems for enhanced performance – focuses on the technology gaps and research challenges that cross cut many new highly-advanced products and processes, such as intelligent transportation systems, autonomous robots, the smart grid, and smart manufacturing systems. Given the importance of ensuring that the resulting products and processes are intelligent, reliable, safe, and secure, cyber-physical systems that people can bet their lives on, performance metrics and evaluation become especially important. Therefore, the 2012 Performance Metrics for Intelligent Systems workshop's theme of *methodologies and techniques of performance measurement for developing and engineering the next generation of cyber physical systems that facilitate seamless human-machine collaboration is both timely and necessary*.

The plenary speakers address cyber-physical systems as well as related topics, particularly robotics, which is a salient example of a CPS. We are fortunate to have George Arnold, SK Gupta, Edward Lee, Jim Overholt, Mark Rice, and Holly Yanco give plenary talks this year. A special session is devoted to discussing Cyber-Physical Systems, with panelists from academia and federal agencies.

Spread over three days, PerMIS'12, the eleventh iteration of the series, features technical presentations organized into two parallel tracks on each day. We thank the special session organizers for proposing interesting topics and assembling researchers related to their sessions. With one of the special sessions, we honor the memory of two of the prime forces that helped forge the PerMIS series and were so influential to the general field of intelligent systems: Jim Albus and Alex Meystel. Our gratitude goes out to the Program Committee members for publicizing the workshop and the reviewers for providing feedback to the authors, and for helping us to put together an interesting program.

PerMIS'12 is sponsored by the National Institute of Standards and Technology (NIST), the Defense Advanced Research Project Agency (DARPA), the National Science Foundation (NSF) and the Maryland Robotics Center, with technical co-sponsorship of the IEEE Washington Section Sensors Council Chapter, and in cooperation with the Association for Computing Machinery (ACM) Special Interest Group on Artificial Intelligence (SIGART). The Defense Advanced Research Projects Agency Information Processing Technology Office graciously provided funding to help support the workshop. Special thanks are due to the National Science Foundation for providing funding to allow undergraduate and graduate students to participate in a special poster session this year. We also thank Professor Ani Hsieh of Drexel University for organizing the NSF new student poster grants program and Professor Holly Yanco of the University of Massachusetts – Lowell for facilitating support for some alumni of prior student poster sessions to return as mentors. We gratefully acknowledge the support of all of our sponsors. The proceedings of PerMIS will be indexed by INSPEC and Compendex and will be available through ACM's Digital Library, as well as being released as a NIST Special Publication.

It is our sincere hope that you will enjoy the presentations, the social programs, renew old relationships, and forge new ones at PerMIS'12!

Elena Messina
NIST Intelligent Systems Division
General Chair

Raj Madhavan
University of Maryland Institute for Systems Research
Program Chair

Disclaimer: This publication consists of workshop proceedings containing technical papers, recommendations, and other materials contributed by participants of this workshop. This publication provides the material as presented and discussed at the workshop in its original form, without modification by the National Institute of Standards and Technology.

PROGRAM COMMITTEE

General Chair:

Elena Messina (Intelligent Systems Division, NIST, USA)

Program Chair:

Raj Madhavan (UMD-CP/NIST, USA)

Publication Chair:

Brian Weiss (Intelligent Systems Division, NIST, USA)

Poster Session Chair:

Ani Hsieh (Drexel University, USA)

Program Committee:

S. Balakirsky, NIST USA
B. Bodt, ARL USA
G. Berg-Cross, EM & I USA
G. Blankenship, UMD-CP USA
F. Bonsignorio, UC3M Spain
M. Childers, ARL USA
A. Godil, NIST USA
J. Gunderson, GammaTwo USA
L. Gunderson, GammaTwo USA
S.K. Gupta, UMD-CP USA
T-H. Hong, NIST USA
A. Hsieh, Drexel U USA
M. Lewis, U Pitt USA
L. Moshkina, Georgia Tech USA
D. Prokhorov, Toyota USA
C. Schlenoff, NIST USA
M. Shneier, NIST USA
N. Tomatis, Bluebotics Switzerland
E. Tunstel, JHU-APL USA

PLENARY SPEAKERS

Prof. Holly Yanco, University of Massachusetts Lowell

Evaluate Early, Evaluate Often: A Design Process for Creating Better Robot Systems

Tue. 08:30 am

ABSTRACT
System evaluations have been conducted in robotics and human-robot interaction for many years. These evaluations usually take place after a robot system has been designed and built as a way to validate the completed system. However, by performing evaluation only at the end of the development cycle, we lose opportunities to create systems with even better performance. Taking inspiration from human-computer interaction, we can design more effective robot systems for human-robot interaction by incorporating user feedback in the initial design phase. This talk will present a number of such formative evaluations from a variety of robotics domains, including assistive robotics and telepresence robot systems.

BIOGRAPHY
Dr. Holly Yanco is Professor and Associate Chair of Computer Science at the University of Massachusetts Lowell. Her research interests include human-robot interaction, multi-touch computing, interface design, robot autonomy, fostering trust of autonomous systems, evaluation methods for human-robot interaction, and the use of robots in K-12 education to broaden participation in computer science. Her research has been funded by the National Science Foundation, including a Career Award, the Army Research Office, Microsoft, and the National Institute of Standards and Technology. Dr. Yanco is the General Chair of the 2012 ACM/IEEE International Conference on Human-Robot Interaction. She served on the Executive Council of the Association for the Advancement of Artificial Intelligence (AAAI) from 2006-2009 and was the Symposium Chair for AAAI from 2002-2005. She was awarded senior membership in AAAI in 2011. Dr. Yanco has a PhD and MS in Computer Science from the Massachusetts Institute of Technology (MIT) and a BA in Computer Science and Philosophy from Wellesley College.

Mark Rice, Maritime Applied Physics Corporation

Geographic Information Systems (GIS) as an Environment for Intelligent Systems Performance Measurement

Tue. 14:00 pm

ABSTRACT
Many intelligent systems work in spatial and temporal environments where Geographic Information Systems (GIS) provide the environment for enabling control and measuring performance. Whether the application involves an automated highway, an unmanned marine vessel, or an unmanned air vehicle, there are GIS based options for the intelligent system designer. This talk will review recent examples of GIS use in unmanned systems where control and performance measurement are enabled by GIS.

BIOGRAPHY
Mark is President of the Maritime Applied Physics Corporation (MAPC, www.mapcorp.com). He has a BA in Physics from the University of Maine and is a licensed Professional Engineer. Mark's first experience with unmanned systems occurred in 1978 when he was the operations officer for the Navy's first 20,000 foot unmanned submersible. Since that time, he has worked as an engineer on various unmanned land, air, and sea systems. Mark formed Maritime Applied Physics Corporation in 1986 and has overseen its growth from a 1 person company to its current 75-person staff. MAPC has both R&D and production work with offices in Baltimore, Maryland, Arlington, Virginia and Brunswick, Maine. MAPC currently designs and manufactures electro-mechanical systems that range from submarine and surface ship components to unmanned systems. Mark is a member of the Maryland/D.C. District Export Council and is the Chair of the National Advisory Board to the NIST Manufacturing Extension Partnership.

Jim Overholt, U.S. Army Tank Automotive Research Development and Engineering Center

Practical to Tactical: Making the Case for a Shift in Ground Vehicle Robotics

Wed. 08:30 am

ABSTRACT

Army ground robotics has been a strategic research and development focus for well over 20 years. In the past 10 years, over 8,000 robotic systems (at its peak in 2010) have been fielded in Southwest Asia. This figure is impressive, especially when you consider that in 2004 it required 5separate vendors to provide 162 robots for only a few select missions. Currently, these systems are used for a variety of critical combat activities but mobile robots are rarely (if ever)used state-side for CONUS operations. In addition, as much as robots have contributed to the War fighters success in various scenarios (most notably in Explosive Ordnance Disposal (EOD) activities in Iraq and Afghanistan), the primary mode of operation of our current robot fleet is still either RemoteControl or Tele-Operation. This is in stark contrast to the intelligent navigation capabilities being shown at our leading universities and other robot Original Equipment Manufacturers (OEMs). So where is the disconnect?

This talk will focus on addressing this very question from various points of view; including new efforts to heavily leverage DOT programs and commercial automotive S&T to facilitate robotics on military base and installations, and to segment the potential robotics mission work-space into 2 simple classifications of environmental features and human intent of the indigenous population. This will lead to some interesting findings in the minimum barriers of technology entry and whether or not advanced autonomy is really needed at all.

BIOGRAPHY

In March 2010, James L. Overholt, Ph.D. was appointed to a Scientific and Professional service position (ST), a system equal to Senior Executive Service, designed for specifically qualified scientific and professional personnel engaged in research and development. As the Senior Research Scientist in Robotics for the Department of Defense, Department of the Army, Dr. Overholt is responsible for defining the strategic vision for robotics science and technology and for conducting, mentoring, and sponsoring cutting edge robotics research. In his nearly 30 years of service to the Army, Dr. Overholt has held numerous lead research positions. Dr. Overholt was the U.S. co-chair of the Multi Autonomous Ground-robotics International Challenge (MAGIC) event held in Australia in November 2010. In 2009 Dr. Overholt was appointed Director of the Office of the Secretary of Defense (OSD) Joint Ground Robotics Enterprise (JGRE), where he was responsible for providing science and technology guidance to the OSD with an emphasis on closing gaps between war fighter requirements and technology, and coordinating efforts between Services to ensure interoperability and commonality among unmanned systems and supporting the strategic goals of the OSD and the Office of the Undersecretary of Defense for Acquisition, Technology and Logistics (AT&L). From 2007 to May 2009, Dr. Overholt served as the Director of the Joint Center for Robotics (JCR) at the U.S. Army Research Development and Engineering Command (RDECOM) Tank Automotive Research and Development Center (TARDEC). He was responsible for establishing a portfolio of programs that strived to rapidly transition robotics technology into the hands of the Soldier, leveraging industry and academia. From October 2006 to May 2007, Dr. Overholt was detailed to the Army Research Office (ARO) as the acting PM for all academic extra-mural robotics and intelligent controls research programs.

Dr. Overholt earned a BS in Physics from the Lawrence Institute of Technology, and a MS in Systems Engineering from Oakland University. He earned his Ph. D. from Oakland University in 1999, emphasizing the development of neural-fuzzy sensor fusion behavioral architectures for unmanned vehicles. His current research interests are machine intelligence and high-speed mobile robot navigation and control. Dr. Overholt is the co-author of more than 50 scientific papers, and was awarded the Bronze Medal at the 2006 Army Science Conference for his contributions in writing “High Speed Hazard Avoidance for Unmanned Ground Vehicles in Emergency Situations.”

Satyandra K. Gupta, Maryland Robotics Center Mechanical Engineering Department and Institute for Systems Research University of Maryland, College Park

Simulation-Based Design and Evaluation of Physics-Aware Planners for Robotic Operations in Challenging Environments

Wed. 14:00 pm

ABSTRACT

Physically challenging environments require robots to be able to negotiate around dynamically moving objects, cope with significant uncertainties in the outcome of action execution, sensor limitations, and the presence of intelligent adversaries. Physics-aware planners are needed in such environments. Unfortunately, exhaustive evaluation of planners using only physical tests is not possible in these applications. This presentation describes how simulations can be successfully used to design and evaluate physics-aware planners. I plan to cover the following four topics. First, I will describe a physics-aware planner that integrates task planning, behavior selection, and trajectory planning in a seamless manner to successfully handle physically challenging environments. This approach provides the right balance between deliberative planning and reactive behaviors during the execution of complex tasks in a dynamic uncertain environment. Second, I will describe our work in the area of physically accurate computationally efficient simulations to enable physics-aware planning and evaluate planners. Third, I will describe computational synthesis techniques for automatically generating sophisticated reactive behaviors using simulations. Finally, the following applications will be used to illustrate simulation-based design and evaluation of planners: (1) guarding of a valuable asset by autonomous unmanned sea surface vehicles, (2) assembly of micro particles in a fluidic medium using holographic optical tweezers, and (4) supply mission on a rugged terrain by unmanned ground vehicles.

BIOGRAPHY

Dr. Satyandra K. Gupta is a Professor in the Mechanical Engineering Department and the Institute for Systems Research at the University of Maryland, College Park. He is the director of the Maryland Robotics Center. Prior to joining the University of Maryland, he was a Research Scientist in the Robotics Institute at Carnegie Mellon University. He received a Bachelor of Engineering (B.E.) degree in Mechanical Engineering from the University of Roorkee (currently known as Indian Institute of Technology, Roorkee) in 1988, a Master of Technology (M. Tech.) degree in Production Engineering from Indian Institute of Technology, Delhi in 1989, and a Ph.D. in Mechanical Engineering from the University of Maryland in 1994.

Dr. Gupta's interest is broadly in the area of automation. He is specifically interested in automation problems arising in Engineering Design, Manufacturing, and Robotics. His current research focus is mainly on simulation-based computational synthesis and automated planning. He is a fellow of the American Society of Mechanical Engineers (ASME). He has served as an Associate Editor for IEEE Transactions on Automation Science and Engineering, ASME Journal of Computing and Information Science in Engineering, and SME Journal of Manufacturing Processes.

Dr. Gupta has authored or co-authored more than two hundred forty articles in journals, conference proceedings, and book chapters. Awards received by Dr. Gupta include a Best Paper Award in 1994 ASME International Conference on Computers in Engineering, a Best Paper Award in 1999 ASME Design for Manufacturing Conference, a Young Investigator Award from Office of Naval Research in 2000, a Robert W. Galvin Outstanding Young Manufacturing Engineer Award from Society of Manufacturing Engineers in 2001, a CAREER Award from National Science Foundation in 2001, a Presidential Early Career Award for Scientists and Engineers (PECASE) in 2001, a Best Paper Award in 2006 ASME Computers and Information in Engineering Conference, and a Best Paper Award in 2010 ASME Mechanism and Robotics Conference. He received Kos Ishii-Toshiba Award from ASME in 2011.

Edward Lee, UC Berkeley

Time for High-Confidence Cyber-Physical Systems

Thu. 08:30 am

ABSTRACT
All widely used software abstractions lack temporal semantics. The notion of correct execution of a program written in every widely-used programming language today does not depend on the temporal behavior of the program. But temporal behavior matters in almost all systems, particularly in networked systems. Even in systems with no particular real-time requirements, timing of programs is relevant to the value delivered by programs, and in the case of concurrent and distributed programs, also affects the functionality. In systems with real-time requirements, including most embedded systems, temporal behavior affects not just the value delivered by a system but also its correctness.

This talk will argue that time can and must become part of the semantics of programs for a large class of applications. It will argue that temporal behavior is not always just a performance metric, but is often rather a correctness criterion. To illustrate that this is both practical and useful, we will describe recent efforts at Berkeley in the design and analysis of timing-centric software systems. In particular, we will focus on two projects, PRET, which seeks to provide computing platforms with repeatable timing, and PTIDES, which provides a programming model for distributed real-time systems.

BIOGRAPHY
Edward A. Lee is the Robert S. Pepper Distinguished Professor in the Electrical Engineering and Computer Sciences (EECS) department at U.C. Berkeley. His research interests center on design, modeling, and analysis of embedded, real-time computational systems. He is a director of Chess, the Berkeley Center for Hybrid and Embedded Software Systems, and is the director of the Berkeley Ptolemy project. From 2005-2008, he served as chair of the EE Division and then chair of the EECS Department at UC Berkeley. He is co-author of nine books (counting second and third editions) and numerous papers. He has led the development of several influential open-source software packages, notably Ptolemy and its various spinoffs. He received the B.S. degree in Computer Science from Yale University, New Haven, CT, in 1979, the S.M. degree in EECS from the Massachusetts Institute of Technology (MIT), Cambridge, in 1981, and the Ph.D. degree in EECS from the University of California Berkeley, Berkeley, in 1986. From 1979 to 1982 he was a member of technical staff at Bell Telephone Laboratories in Holmdel, New Jersey, in the Advanced Data Communications Laboratory. He is a co-founder of BDTI, Inc., where he is currently a Senior Technical Advisor, and has consulted for a number of other companies. He is a Fellow of the IEEE, was an NSF Presidential Young Investigator, and won the 1997 Frederick Emmons Terman Award for Engineering Education.

George Arnold, NIST

Performance and New Paradigms for the Electric Power System

Thu. 14:00 pm

ABSTRACT
The structure of the world's power system has not changed much since the era of Thomas Edison: it is characterized by the one-way flow of electricity from controllable carbon-producing centralized power generation plants to users who have little awareness of how much energy they consume and how they can be more efficient. This talk will describe how the Smart Grid will eventually enable a new paradigm - the dynamic, two-way flow of electricity and information that will support growing use of distributed green generation sources (such as wind and solar), widespread use of electric vehicles, and ubiquitous intelligent appliances and buildings that can dynamically adjust power consumption in response to conditions on the grid. Modeling, forecasting, and control strategies that reflect new dynamic operational paradigms will be essential to realizing the environmental and energy efficiency benefits enabled by the smart grid.

BIOGRAPHY
George Arnold was appointed National Coordinator for Smart Grid Interoperability at the National Institute of Standards and Technology (NIST) in April 2009. He is responsible for leading the development of standards underpinning the nation's Smart Grid. In October 2011 he assumed an additional responsibility as Director, Smart Grid and Cyber-Physical Systems Program Office in the NIST Engineering Laboratory. Dr. Arnold joined NIST in September 2006 as Deputy Director, Technology Services, after a 33-year career in the telecommunications and information technology industry.

Dr. Arnold served as Chairman of the Board of the American National Standards Institute (ANSI), a private, non-profit organization that coordinates the U.S. voluntary standardization and conformity assessment system, from 2003 to 2005. He served as President of the IEEE Standards Association in 2007-2008 and was Vice President-Policy for the International Organization for Standardization (ISO) during 2006-2009, where he is responsible for guiding ISO's strategic plan.

Dr. Arnold previously served as a Vice-President at Lucent Technologies Bell Laboratories where he directed the company's global standards efforts. His organization played a leading role in the development of international standards for Intelligent Networks and IP-based Next Generation Networks. In previous assignments at AT&T Bell Laboratories he had responsibilities in network planning, systems engineering, and application of information technology to automate operations and maintenance of the nationwide telecommunications network.

Dr. Arnold received a Doctor of Engineering Science degree in Electrical Engineering and Computer Science from Columbia University in 1978. He is a Fellow of the IEEE.

Time	Tuesday March 20		Wednesday March 21		Thursday March 22
8:00-8:30	Welcome/Overview		Overview		Overview
8:30-9:30	**Plenary 1: Holly Yanco**		**Plenary 3: Jim Overholt**		**Plenary 5: Edward Lee**
9:30-10:00	*Coffee Break*		*Coffee Break*		*Coffee Break*
10:00 -12:30	TUE-AM1: *Performance Evaluation*	TUE-AM2: *Session Honoring the Legacy of Jim Albus & Alex Meystel*	WED-AM1: *Human-Robot Collaboration & Interaction*	WED-AM2: *Technology Readiness for Randomized Bin Picking Solutions*	THU-AM: *CPS Panel Discussion*
12:30-14:00	*Lunch*		*Lunch*		*Lunch*
14:00 - 15:00	**Plenary 2: Mark Rice**		**Plenary 4: SK Gupta**		**Plenary 6: George Arnold**
15:00-15:30	*Coffee Break*		*Coffee Break*		*Coffee Break*
15:30 -17:30	TUE-PM1: *Performance Measures & Metrics*	TUE-PM2: *Performance Evaluation and Advanced Algorithms for Static & Dynamic 6DOF*	WED-PM1: *Performance Characterization* (15:30-18:00*)*	WED-PM2: *Field Testing & Standard Test Methods* (15:30-18:00*)*	THU-PM: *Performance Testing & Validation*
	(18:00 – 20:00) ***Reception & Poster Session***		*(18:30 – 20:00)* ***Banquet***		

March 20

PROGRAM

PERMIS

TUESDAY

08:00	**Welcome & Overview - Brian Darmody, Associate Vice President for Research & Economic Development, University of MD**
08:30	***Plenary Presentation:*** ***Holly Yanco*** ***Evaluate Early, Evaluate Often: A Design Process for Creating Better Robot Systems***
09:30	**Coffee Break**
10:00	**TUE-AM1 Performance Evaluation** *Chairs: Greg Dudek and Craig Schlenoff* • Performance Evaluation of Robotic Knowledge Representations (PERK) [Craig Schlenoff, Sebti Foufou, Stephen Balakirsky] • A Hybrid Approach to 2D Robotic Map Evaluation [Ross Creed, Kristiyan Georgiev, Rolf Lakaemper] • An Overview of Robot-Sensor Calibration Methods for Evaluation of Perception Systems [Mili Shah, Roger Eastman, Tsai Hong] • On the Performance Evaluation of a Vision-Based Human-Robot Interaction Framework [Junaed Sattar, Gregory Dudek] • Functional Requirements of a Model for Kitting Plans [Stephen Balakirsky, Zeid Kootbally, Thomas Kramer, Raj Madhavan, Craig Schlenoff, Michael Shneier]
12:30	**Lunch**
14:00	***Plenary Presentation:*** ***Mark Rice*** ***Geographic Information Systems (GIS) as an Environment for Intelligent Systems Performance Measurement***
15:00	**Coffee Break**
15:30	**TUE-PM1 Performance Measures and Metrics** *Chairs: Rolf Lakaemper and Michael Del Rose* • The New Method for Measuring Absolute Threshold of Haptic Force Feedback [Michal Baczynski] • Approach for Defining Intelligent Systems Technical Performance Metrics [Wael Hafez] • Metrics for Planetary Rover Planning & Scheduling Algorithms [Juan Delfa Victoria, Nicola Policella, Marc Gallant, Oskar von Stryk, Alessandro Donati, Yang Gao] • Measures for UGV to UGV Collaboration [Michael Del Rose, Anthony Finn, Robert Kania]
18:00	**Reception & Poster Session**

08:00	**Welcome & Overview - Brian Darmody, Associate Vice President for Research & Economic Development, University of MD**
08:30	***Plenary Presentation:*** ***Holly Yanco*** ***Evaluate Early, Evaluate Often: A Design Process for Creating Better Robot Systems***
09:30	**Coffee Break**
10:00	**TUE-AM2 Special Session I: Session Honoring the Legacy of Jim Albus and Alex Meystel** *Organizers: Alberto Lacaze and Elena Messina* • Army-NIST Robotics Teaming: A Thirty Year Retrospective [Charles Shoemaker, U. S. Army Communications-Electronics Research, Development, and Engineering] • ATR's DoD programs with RoboCrane Technologies [Jackson Yang, Advanced Technology and Research Corporation] • Mind, Brain, and Intelligence: A Krasnow Institute Perspective on the Jim Albus Legacy [Kenneth DeJong, Alexei Samsonovich, James Olds, Krasnow Institute for Advanced Study, GMU] • Understanding 'Intelligence': in Pursuit of Multi-Resolutional Hierarchies [Predrag Filipovic, Agora Creative Solutions Inc.] • Measurable and Scalable Methods for Modern Knowledge Processing [Michael Meystel, The Vanguard Group and Cognisphere Inc.] • Birth of an Architecture [Alberto Lacaze, Robotic Research LLC.]
12:30	**Lunch**
14:00	***Plenary Presentation:*** ***Mark Rice*** ***Geographic Information Systems (GIS) as an Environment for Intelligent Systems Performance Measurement***
15:00	**Coffee Break**
15:30	**TUE-PM2 Special Session II: Performance Evaluation and Advanced Algorithms for Static and Dynamic 6DOF** *Organizers: Chad English and Jane Shi* • 2011 Solutions in Perception Challenge Performance Metrics and Results [Jeremy Marvel, Tsai Hong, Elena Messina] • Development of an Apparatus for Characterizing the Measurement Latency of a Dynamic 3D Tracking System [Kamel Saidi] • Shape-based Pose Estimation Evaluation using Expectivity Index Artifacts [Chad English, Galina Okouneva, Aradhana Choudhuri] • Ground Truth for Evaluating 6 Degrees of Freedom Pose Estimation Systems [Jeremy Marvel, Joe Falco, Tsai Hong] • Performance Measurement with 6DOF Laser Tracker Technologies [Zach Ryan, Aaron Sabino]
18:00	**Reception & Poster Session**

08:15	**Overview**
08:30	***Plenary Presentation:*** ***Jim Overholt*** ***Practical to Tactical: Making the Case for a Shift in Ground Vehicle Robotics***
09:30	**Coffee Break**
10:00	**WED-AM1 Human-Robot Collaboration and Interaction** *Chairs: Jane Shi and Kate Tsui* • A Proxemic-Based HRI Testbed [Zachary Henkel, Robin Murphy, Vasant Srinivasan, Cindy Bethel] • Synergistic Methods for Using Language in Robotics [Ching Teo, Yezhou Yang, Cornelia Fermuller, Yiannis Aloimonos] • Reusable Semantic Differential Scales for Measuring Social Response to Robots [Lilia Moshkina] • Levels of Human and Robot Collaboration for Automotive Manufacturing [Jane Shi, Glenn Jimmerson, Tom Pearson, Roland Menassa] • Towards Measuring the Quality of Interaction: Communication through Telepresence Robots [Katherine Tsui, Munjal Desai, Holly Yanco]
12:30	**Lunch**
14:00	***Plenary Presentation:*** ***Satyandra K. Gupta*** ***Simulation-Based Design and Evaluation of Physics-Aware Planner for Robotic Operations in Challenging Environments***
15:00	**Coffee Break**
15:30	**WED-PM1 Performance Characterization** *Chairs: Damian Lyons and Hui-Min Huang* • Characterizing Performance Guarantees for Multiagent, Real-Time Systems Operating in Noisy and Uncertain Environments [Damian Lyons, Ronald Arkin, Stephen Fox, Shu Jiang, Prem Nirmal, Munzir Zafar] • Design, Fabrication and Characterization of the Single-Layer Out-of-Plane Electrothermal Actuator for a MEMS XYZ Stage [Yong-Sik Kim, Nicholas Dagalakis, Satyandra Gupta] • Intelligent Energy Management: Impact of Demand Response and Plug-in Electric Vehicles in a Smart Grid Environment [Seshadri Raghavan, Alireza Khaligh] • Characterization of Forward Rectilinear-Gait Performance for a Snake-Inspired Robot [James Hopkins, Satyandra Gupta] • Emergency Response Robot Evaluation Exercise [Adam Jacoff, Hui-Min Huang, Ann Virts, Anthony Downs, Raymond Sheh]
18:30	**Banquet**

08:15	**Overview**
08:30	***Plenary Presentation:*** ***Jim Overholt*** ***Practical to Tactical: Making the Case for a Shift in Ground Vehicle Robotics***
09:30	**Coffee Break**
10:00	**WED-AM2 Special Session III: Panel Discussion: Technology Readiness for Randomized Bin Picking Solutions** *Organizers: Jeremy Marvel, Tsai Hong, Gerry Cheok, Elena Messina* *Moderator: Roger Eastman* • The NASA-developed TRL Methodology [Karen McNamara] • Challenges of bin-picking [Jeremy Marvel] • A round-table panel discussion moderated by Roger Eastman from NIST/Loyola University, featuring James Wells (General Motors), Joyce Guthrie (US Postal Service), Bob Bollinger (Procter & Gamble), Eric Hershberger (Cognex), Carlos Martinez (ABB Inc.), Paul Evans (Southwest Research Institute), and Karen McNamara (NASA)
12:30	**Lunch**
14:00	***Plenary Presentation:*** ***Satyandra K. Gupta*** ***Simulation-Based Design and Evaluation of Physics-Aware Planner for Robotic Operations in Challenging Environments***
15:00	**Coffee Break**
15:30	**WED-PM2 Field Testing and Standard Test Methods** *Chairs: Barry Bodt and Roger Bostelman* • Test Method for Measuring Station-Keeping With Unmanned Marine Vehicles Using Sonar or Optical Sensors [Asish Ghoshal, Avinash Parnandi, Robin Murphy] • Standard Test Procedures and Metric Development for Automated Guided Vehicle Safety Standards [Roger Bostelman, William Shackleford, Geraldine Cheok, Richard Norcross] • Integrating Occlusion Monitoring into Human Tracking for Robot Speed and Separation Monitoring [William Shackleford, Sandor Szabo, Richard Norcross, Jeremy Marvel] • Robotics Collaborative Technology Alliance (RCTA) 2011 Baseline Assessment [Barry Bodt, Richard Camden, Marshal Childers] • Using Competitions to Advance the Development of Standard Test Methods for Response Robots [Adam, Jacoff, Raymond Sheh, Ann Virts, Tetsuya Kimura, Johannes Pellenz, Soren Schwertfeger, Jackrit Suthakorn]
18:30	**Banquet**

THURSDAY

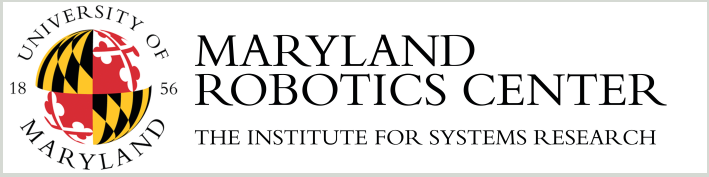
08:15	**Overview**
08:30	***Plenary Presentation:*** ***Edward Lee*** ***Time for High-Confidence Cyber-Physical Systems***
09:30	**Coffee Break**
10:00	**THU-AM Cyber-Physical Systems Panel Discussion** *Organizers: Richard Voyles, NSF and Elena Messina, NIST* *Moderator: Albert Wavering, NIST* • Clare Allocca, NIST • Panos Antsaklis, U. of Notre Dame • George Arnold, NIST • Edward Lee, U. of California-Berkeley • Suzanne Lightman, NIST • Rahul Mangharam, U. of Pennsylvania
12:30	**Lunch**
14:00	***Plenary Presentation:*** ***George Arnold*** ***Performance and New Paradigms for the Electric Power System***
15:00	**Coffee Break**
15:30	**THU-PM Performance Testing and Validation** *Chairs: Brian Weiss and Venkat Krovi* • Validation of the Dynamics of an Humanoid Robot in USARSim [Sander van Noort, Arnoud Visser] • Evaluation of Robotic Minimally Invasive Surgical Skills using Motion Studies [Seung-Kook Jun, Madusudanan Sathianarayanan, Abeer Eddib, Pankaj Singhal, Sudha Garimella, Venkat Krovi] • Multi-Relationship Evaluation Design: Modeling an Automatic Test Plan Generator [Brian Weiss, Linda Schmidt] • An IEEE 1588 Performance Testing Dashboard for Power Industry Requirements [Julien Amelot, Ya-Shian Li-Baboud, Clement Vasseur, Jeffrey Fletcher, Dhananjay Anand, James Moyne]
17:30	**Adjourn**

08:15	**Overview**
08:30	***Plenary Presentation:*** ***Edward Lee*** ***Time for High-Confidence Cyber-Physical Systems***
09:30	**Coffee Break**
10:00	**THU-AM Cyber-Physical Systems Panel Discussion** *Organizers: Richard Voyles, NSF and Elena Messina, NIST* *Moderator: Albert Wavering, NIST* • Clare Allocca, NIST • Panos Antsaklis, U. of Notre Dame • George Arnold, NIST • Edward Lee, U. of California-Berkeley • Suzanne Lightman, NIST • Rahul Mangharam, U. of Pennsylvania
12:30	**Lunch**
14:00	***Plenary Presentation:*** ***George Arnold*** ***Performance and New Paradigms for the Electric Power System***
15:00	**Coffee Break**
15:30	**THU-PM Performance Testing and Validation** *Chairs: Brian Weiss and Venkat Krovi* • Validation of the Dynamics of an Humanoid Robot in USARSim [Sander van Noort, Arnoud Visser] • Evaluation of Robotic Minimally Invasive Surgical Skills using Motion Studies [Seung-Kook Jun, Madusudanan Sathianarayanan, Abeer Eddib, Pankaj Singhal, Sudha Garimella, Venkat Krovi] • Multi-Relationship Evaluation Design: Modeling an Automatic Test Plan Generator [Brian Weiss, Linda Schmidt] • An IEEE 1588 Performance Testing Dashboard for Power Industry Requirements [Julien Amelot, Ya-Shian Li-Baboud, Clement Vasseur, Jeffrey Fletcher, Dhananjay Anand, James Moyne]
17:30	**Adjourn**

AUTHOR INDEX

Aloimonos, Y. WED-AM1
Amelot, J. THU-PM1
Anand, D. THU-PM1
Arkin, R. WED-PM1
Baczynski, M. TUE-PM1
Balakirsky, S. TUE-AM1
Balakirsky, S. TUE-AM1
Bethel, C. WED-AM1
Bodt, B. WED-PM2
Bostelman, R. WED-PM2
Camden, R. WED-PM2
Cheok, G. WED-AM2
Cheok, G. WED-PM2
Childers, M. WED-PM2
Choudhuri, A. TUE-PM2
Creed, R. TUE-AM1
Dagalakis, N. WED-PM1
Delfa Victoria, J. TUE-PM1
Del Rose, M. TUE-PM1
Desai, M. WED-AM1
Donati, A. TUE-PM1
Downs, A. WED-PM1
Dudek, G. TUE-AM1
Eastman, R. TUE-AM1
Eastman, R. WED-AM2
Eddib, A. THU-PM1
English, C. TUE-PM2
Falco, J. TUE-PM2
Fermuller, C. WED-AM1
Finn, A. TUE-PM1
Fletcher, J. THU-PM1
Foufou, S. TUE-AM1
Fox, S. WED-PM1
Gallant, M. TUE-PM1
Garimella, S. THU-PM1
Gao, Y. TUE-PM1
Georgiev, K. TUE-AM1
Ghoshal, A. WED-PM2
Gupta, S. WED-PM1
Gupta, S. WED-PM1
Hafez, W. TUE-PM1
Henkel, Z. WED-AM1
Hong, T. TUE-AM1
Hong, T. TUE-PM2
Hong, T. TUE-PM2
Hong, T. WED-AM2
Hopkins, J. WED-PM1
Huang, H-M. WED-PM1
Jacoff, A. WED-PM1
Jacoff, A. WED-PM2
Jiang, S. WED-PM1
Jimmerson, G. WED-AM1
Jun, S-K. THU-PM1
Kania, R. TUE-PM1
Khaligh, A. WED-PM1
Kim,Y-S. WED-PM1
Kimura, T. WED-PM2
Kootbally, Z. TUE-AM1
Kramer, T. TUE-AM1
Krovi, V. THU-PM1
Lakaemper, R. TUE-AM1
Li-Baboud, Y-S. THU-PM1
Lyons, D. WED-PM1
Madhavan, R. TUE-AM1
Marvel, J. TUE-PM2
Marvel, J. TUE-PM2
Marvel, J. WED-AM2
Marvel, J. WED-PM2
Menassa, R. WED-AM1
Messina, E. TUE-PM2
Messina, E. WED-AM2
Moshkina, L. WED-AM1
Moyne, J. THU-PM1
Murphy, R. WED-AM1
Murphy, R. WED-PM2
Nirmal. P. WED-PM1
Norcross, R. WED-PM2
Norcross, R. WED-PM2
Okouneva, G. TUE-PM2
Parnandi, A. WED-PM2
Pearson, T. WED-AM1
Pellenz, J. WED-PM2
Policella, N. TUE-PM1
Raghavan, S. WED-PM1
Saidi, K. WED-AM2
Sathianarayanan, M. ..THU-PM1
Sattar, J. TUE-AM1
Schlenoff, C. TUE-AM1
Schlenoff, C. TUE-AM1
Schmidt, L. THU-PM1
Schwertfeger, S. WED-PM2
Shackleford, W. WED-PM2
Shackleford, W. WED-PM2
Shah, M. TUE-AM1
Sheh, R. WED-PM1
Sheh, R. WED-PM2
Shi, J. WED-AM1
Shneier, M. TUE-AM1
Singhal, P. THU-PM1
Srinivasan, V. WED-AM1
Suthakorn, J. WED-PM2
Szabo, S. WED-PM2
Teo, C. WED-AM1
Tsui, K. WED-AM1
van Noort, S. THU-PM1
Vasseur, C. THU-PM1
Virts, A. WED-PM1
Virts, A. WED-PM2
Visser, A. THU-PM1
von Stryk, O. TUE-PM1
Weiss, B. THU-PM1
Yang, Y. WED-AM1
Yanco, H. WED-AM1
Zafar, M. WED-PM1

SPONSORS



ACKNOWLEDGMENTS

These people provided essential support to make this event happen. Their ideas and efforts are very much appreciated.

Website and Proceedings
Debbie Russell

Local Arrangements
Debbie Russell
Jeanenne Salvermoser
Sarah Standifer

Intelligent Systems Division
Engineering Laboratory
National Institute of Standards and Technology
100 Bureau Drive, MS 8230
Gaithersburg, MD 20899-8230
http://www.nist.gov/el/isd/permis2012.cfm

Performance Evaluation of Robotic Knowledge Representation (PERK)

Craig Schlenoff[a,b]
[a]National Institute of Standards and Technology
100 Bureau Drive, Stop 8230
Gaithersburg, MD 20899, USA
craig.schlenoff@nist.gov

Sebti Foufou[b,c]
[b]University of Burgundy,
LE2i Lab., Dijon, France.
sfoufou@u-bourgogne.fr
and
[c]Computer Science and Eng.
Qatar University, Doha, Qatar
sfoufou@qu.edu.qa

Stephen Balakirsky[a]
[a]National Institute of Standards and Technology
100 Bureau Drive, Stop 8230
Gaithersburg, MD 20899,USA
stephen.balakirsky@nist.gov

ABSTRACT

In this paper, we explore some ways in which symbolic knowledge representations have been evaluated in the past and provide some thoughts on what should be considered when applying and evaluating these types of knowledge representations for real-time robotics applications. The emphasis of this paper is that the robotic applications require real-time access to information, which has not been one of the aspects measured in traditional symbolic representation evaluation approaches.

Categories and Subject Descriptors

I.2.4. **[Computer Methodologies]:** Artificial Intelligence: Knowledge Representation Formalisms and Methods – Representation languages

General Terms

Measurement, Performance

Keywords

Robotics, knowledge representation, performance metrics, real-time, ontologies

1. INTRODUCTION

A robot can only perform tasks based on what it knows, which is often captured within the robot's internal knowledge representation. This representation can take many forms and knowledge can be captured at various levels of specificity. With the growing complexity of behaviors that robots are expected to perform, the need to measure the knowledge representation, in terms of coverage, the ability to reason to infer new knowledge, and the ability to successfully complete complex tasks, is becoming more evident.

Knowledge representations have historically been evaluated using metrics such as completeness (Is all necessary knowledge represented?), expressiveness (Can all necessary knowledge be represented?), accuracy (Is the represented knowledge correct?), and consistency (Are there contradictory facts represented?)[1, 2]. While these metrics are important in a theoretical sense, knowledge representation for robotics introduces a series of additional metrics, such as performance (real-time access), flexibility (ability to constantly update knowledge as new information becomes available), and relevance (is information represented at a level of resolution that can be used by planning systems). In addition, the way that the representations are evaluated must change when introducing these new metrics. For example, while running a consistency checker can help to identify contradictory knowledge, it does not assess the representation's ability to respond to an ever-changing environment. Successful measures for these types of metrics may include the ability (and time) to answer what-if questions, the ability to support real-time planning, etc.

In this paper, we explore some ways in which knowledge representations have been evaluated in the past and provide some thoughts on what should be considered when evaluating knowledge representation for real-time robotics applications. This paper is organized as follows:

- Section 2 discusses current knowledge representation approaches in the robotics domain
- Section 3 describes an ontology standardization effort that will serve as the basis for future research efforts
- Section 4 describes some previous efforts that have explored how to measure the performance of symbolic knowledge representations with an emphasis on ontologies
- Section 5 attempts to categorize the types of metrics that have been used in the past along with some thoughts on their applicability to the robotics domain
- Section 6 concludes the paper by discussing the relationship between ontology metrics and traditional robotics knowledge representation approaches and where the current gaps lie.

2. KNOWLEDGE REPRESENTATIONS FOR ROBOTICS

Traditionally, robots use a wide array of knowledge representations. Some of these include parametric knowledge, spatial knowledge, and symbolic knowledge. A good overview of these types of knowledge and how they have been applied

(c) 2012 Association for Computing Machinery. ACM acknowledges that this contribution was authored or co-authored by a contractor or affiliate of the U.S. Government. As such, the Government retains a nonexclusive, royalty-free right to publish or reproduce this article, or to allow others to do so, for Government purposes only. PerMIS'12, March 20-22, 2012, College Park, MD, USA. Copyright © 2012 ACM 978-1-4503-1126-7-3/22/12...$10.00

to the robotics domain can be found in [3]. An overview of these types of knowledge is described below.

2.1 Parametric Knowledge

The lowest levels of any control system, whether for an autonomous robot, a machine tool, or a refinery, are at the servo level, where knowledge of the value of system parameters is needed to provide position and/or velocity and/or torque control of each degree of freedom by appropriate voltages sent to a motor or a hydraulic servo valve. The control loops at this level can generally be analyzed with classical techniques and the "knowledge" embedded in the world model is the specification of the system functional blocks, the set of gains and filters that define the servo controls for a specific actuator, and the current value of relevant state variables. These are generally called the system parameters, so we refer to knowledge at this level as parametric knowledge.

Figure 1shows a traditional PD (Proportional Derivative) servo control for a motor of a robot arm. All six or seven motors that drive the arm will have basically the same servo control, but each will have different parameters because there are different size motors driving different loads at different points in the arm. Any errors that deal with a single degree of freedom, such as ball screw lead errors, contact instabilities, stiction, and friction are best compensated for at this level.

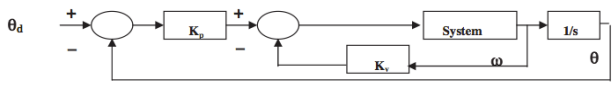


Figure 1: PD Servo Control

2.2 Spatial Knowledge

Above the servo level are a series of control loops that coordinate the individual servos and that require what can be generally called "geometric knowledge," "iconic knowledge," "metrical maps," or "patterns." This knowledge is spatial in nature and can be defined as 2D or 3D array data in which the dimensions of the array correspond to dimensions in physical space. The value of each element of the array may be Boolean data or real number data representing a physical property such as light intensity, color, altitude, range, or density. Each element may also contain spatial or temporal gradients of intensity, color, range, or rate of motion. Each element may also contain a pointer to a geometric entity (such as an edge, vertex, surface, or object) to which the pixel belongs.

Examples of iconic knowledge include digital terrain maps, sensor images, models of the kinematics of the machines being controlled, and knowledge of the spatial geometry of parts or other objects that are sensed and with which the machine interacts in some way. This is where objects and their relationship in space and time are modeled in such a way as to represent and preserve those spatial and temporal relationships, as in a map, image, or trajectory.

For industrial robots, machine tools, and coordinate measuring machines, the first level above the servo level deals with the kinematics of the machine, relating the geometry of the different axes to allow coordinated control. Linear, circular and other interpolation and motion in world or tool coordinates is enabled by such coordination. The "knowledge" here may be the kinematic equations or Jacobian coefficients that define the geometric relationships of the axes, or the mathematical routines for interpolation or coordinate transformations. It is at this level that systematic multi-dimensional geometric errors such as non-orthogonality of axes of a machine tool are considered.

For mobile autonomous robots, there are two main categories of spatial knowledge representation that are useful. These are sometimes referred to as metrical maps in the literature. One captures what the sensors see (the view "out the windshield"). This may be two-dimensional images, as is the case for CCD (Charge Coupled Device) cameras, or three-dimensional images, in the case of range sensors such as LADARs (laser Detection and Ranging). Some mobile robots successfully accomplish their goals by planning based on a world model derived purely from the sensor image view.

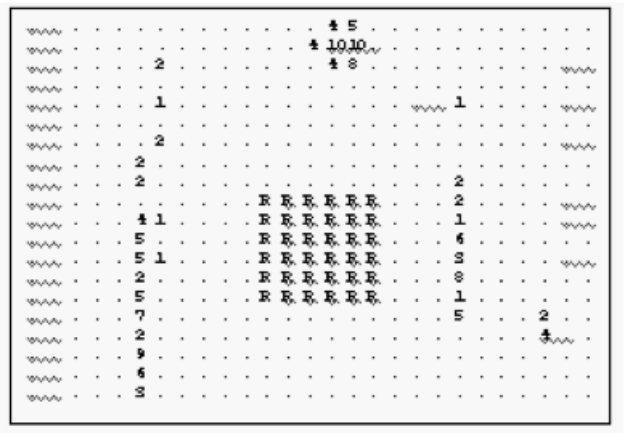


Figure 2: Occupancy Grid Map for Mobile Robot

Figure 2 shows a typical local map from a mobile robot navigating through an indoor environment. The robot's position at the center is indicated by marking the occupied cells with "R". The numbers in certain cells indicate the degree of confidence that there is an obstacle occupying that cell.

The second type of spatial representation is akin to the "bird's-eye-view." Figure 3 shows a higher level map for path planning for outdoor navigation. This map contains several feature layers, including elevation, vegetation, roads, buildings, and obstacles. Digital maps are a natural way of representing the environment for path planning and obstacle avoidance, and provide a very powerful mechanism for sensor fusion since the data from multiple sensors can be represented in a common format. Digital terrain maps are essentially two-dimensional grid structures that are referenced to some coordinate frame tied to the ground or earth. A map may have multiple layers that represent different "themes" or attributes

at each grid element. For instance, there may be an elevation layer, a road layer, a hydrology layer, and an obstacle layer. The software can query if there is a road at grid location [x, y] and similarly query for other attributes at the same [x, y] coordinates.

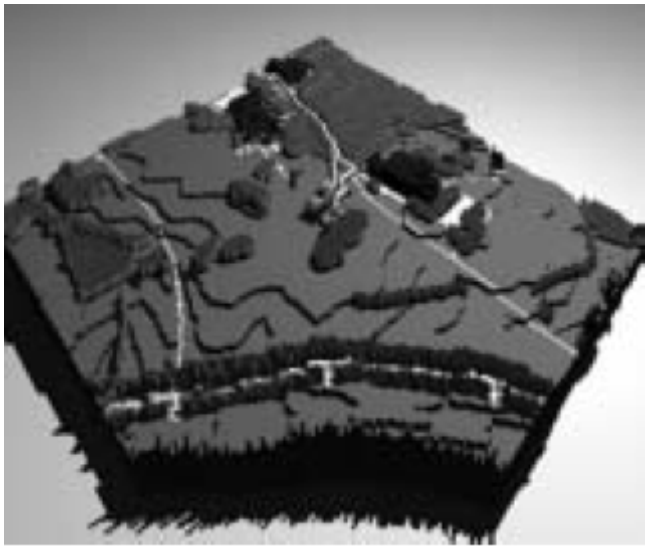


Figure 3: Multi-Terrain Digital Terrain Map

2.3 Symbolic Knowledge

At the highest levels of control, knowledge will be symbolic, whether dealing with actions or objects. It is at this level that a large body of relevant work exists in knowledge engineering for domains other than real-time control, such as formal logic systems or rule-based expert systems. Whether the knowledge is represented in terms of mathematical logic, rules, frames, or semantic nets, there is a formal linguistic structure for defining and manipulating and using the knowledge.

An example of a formal description of a solid model of a part is shown in Figure 4. A block is being described using International Standards Organization Standard for the Exchange of Product Model Data (STEP) Part 21 [4]. Note that this representation can be linked by pointers to a geometric representation where, for example, a block might be represented by equations of six planes with bounding curves and a coordinate transformation matrix to position the block within a given coordinate system.

Linguistic representations provide ways of expressing knowledge and relationships, and of manipulating knowledge, including the ability to address objects by property. Tying symbolic knowledge back into the geometric levels provides symbol grounding, thereby solving a serious problem inherent to purely symbolic knowledge representations. It also provides the valuable ability to identify objects from partial observations and then extrapolate facts or future behaviors from the symbolic knowledge. In the manufacturing domain, using a feature-based representation (which is symbolic) is reasonable at the generative planning level (Figure 5a). Graphical primitives (Figure 5b) that relate to the geometry can be tied to features to let users easily pick a feature (such as a pocket) by selecting on a portion of it on the screen. The geometric representation of each edge and surface that comprise a feature (Figure 5c) can be tied to the feature definition in order to facilitate calculations for generating the tool paths.

```
DATA;
#10 =
BLOCK_BASE_SHAPE(#20,#30,#70,#80);
#20 = NUMERIC_PARAMETER('block Z
dimension',50.,'mm');
#30 = ORIENTATION(#40,#50,#60);
#40 = DIRECTION_ELEMENT((0.,0.,1.));
#50 = DIRECTION_ELEMENT((1.,0.,0.));
#60 = LOCATION_ELEMENT((62.5,37.5,0.));
#70 = NUMERIC_PARAMETER('block Y
dimension',75.,'mm');
#80 = NUMERIC_PARAMETER('block X
dimension',125.,'mm');
#90 = SHAPE((),#10,());
#100 = PART('out','rev1','','simple
part','insecure',(),#90,(),(),(),$,(),
(#110),(),());
#110 = MATERIAL('aluminum','soft
aluminum',$,(),());
```

Figure 4: STEP Representation of a Block

Another type of symbolic representation for representing rules is ontological. Ontologies are definitions and organizations of classes of facts and formal rules for accessing and manipulating (and possibly extending) those facts. [5] There are two main approaches to creating ontologies, one emphasizing the organizational framework, with data entered into that framework, and the other emphasizing large scale data creation with relationships defined as needed to relate and use that data. Cyc [6] is an example of the latter, an effort to create a system capable of common sense, natural language understanding, and machine learning.

An ontology may be designed to make it easy for reasoning systems to reason using the ontology. This includes being able to infer information that may not be explicitly represented, as well as the ability to pose questions to the knowledge base and receive answers in return. One way of enabling this functionality is to represent the symbolic information in the world model in a logic-based, computer-interpretable format, such as in the Knowledge Interface Format (KIF) representation [7] and using a logic programming tool such as Prolog. [8]

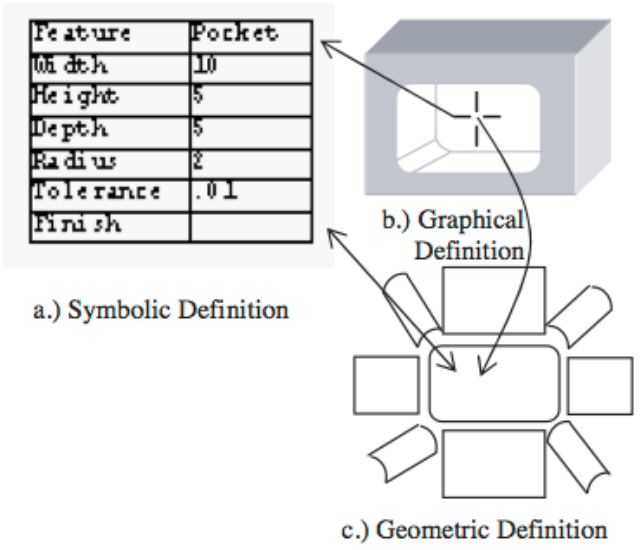


Figure 5: Pocket Feature

Through the use of an inference engine or theorem prover, information represented in this format could be queried, and logically-proven answers could be returned. As an example, a manufacturer may want to know whether a given set of fixture positions is suitable to fully inspect a part. Assuming that the necessary inspection points, access volumes, and machine capabilities are represented in KIF, the manufacturer could enter in the fixture positions and the system could logically-prove whether those positions are sufficient to fully inspect the part.

The focus of the remainder of this paper will be on symbolic representation, as it will be the focus of future research efforts.

3. IEEE ROBOTICS AND AUTOMATION SOCIETY (RAS) ONTOLOGIES FOR ROBOTICS AND AUTOMATION (ORA) WORKING GROUP

For the research effort described later in this paper, a standard knowledge representation (ontology) is needed. IEEE had formed a working group to explore the development of a standard robot ontology. It is anticipated that this ontology will serve as the basis for this work and is described below.

In October 2011, IEEE approved a new working group called Ontologies for Robotics and Automation (ORA) [9]. The goal of this working group is to develop a standard ontology and associated methodology for knowledge representation and reasoning in robotics and automation, together with the representation of concepts in an initial set of application domains. The standard provides a unified way of representing knowledge and provides a common set of terms and definitions, allowing for unambiguous knowledge transfer among any group of humans, robots, and other artificial systems. To date, the working group is made up of over 115 members containing a cross-section of industry, academia, and government and representing over twenty countries.

The working group defines an ontology as a knowledge representation approach that represents key concepts, their properties, their relationships, and their rules and constraints. [10] Whereas taxonomies usually provide only a set of vocabulary and a single type of relationship between terms (usually a parent/child type of relationship), an ontology provides a much richer set of relationships and also allows for constraints and rules to govern those relationships. In general, ontologies make all pertinent knowledge about a domain explicit and are represented in a computer-interpretable format that allows software to reason over that knowledge to infer additional information.

The working group acknowledges that it would be extremely difficult to develop an ontology that could cover the entire space of robotics and automation. As such, the working group is structured in such a way as to take bottom-up and top-down approaches to addressing this broad domain. From a top-down approach, a sub-group entitled "Upper Ontology/Methodology". (UpOM) is exploring the identification or development of an upper ontology on which to hang more detailed concepts. In addition to this upper ontology, a methodology is being developed that would allow interested colleagues to propose additional concepts and reconcile any differences between the new concepts and those that already exist.

From a bottom-up perspective, three sub-groups have been formed which will take a detailed look at three sub-domains in the robotics and automation area. Those sub-domains are Autonomous Robots (AuR), Service Robots (SeR), and Industrial Robots (InR). Each of those subgroups will deeply explore their respective areas by identifying key concepts, along with their definitions, that need to be represented. The group's structure is shown in Figure 6. These concepts and definitions will then be modeled more formally in an ontology.

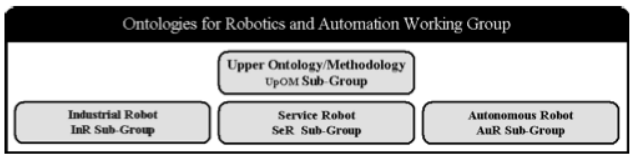


Figure 6: IEEE ORA Group Structure

The sub-domain ontologies will serve as a test case to validate the upper ontology and the methodology. The sub-domains were determined in such a way to ensure that there would be overlap amongst them. Once initial versions of the ontologies are completed, they will be integrated into the overall ontology. During the integration process, as overlapping concepts are identified, a process will be formalized to accurately determine if these concepts should be merged, if they should be separated into two separate concepts, or if some other approach should be explored to reconcile them.

For this effort, the working group has decided to use OWL (Web Ontology Language) [11] as the knowledge representation language. OWL is a family of knowledge

representation languages for authoring ontologies and is endorsed by the World Wide Web Consortium (W3C). It is characterized by formal semantics and RDF/XML-based serialization for the Semantic Web. OWL was chosen by the group because of its popularity among the ontology development community, its endorsement by the W3C, as well as the number of OWL tools and reasoning engines that are available.

4. RELATED WORK

Performance evaluation of symbolic knowledge representation is not a new area; research has been explored for many years. Most of these research efforts have focused on the application of symbolic representations (specifically, ontologies) to domains that do not require real-time access and have primarily focused on the structure and consistency of the ontology as opposed to how it is applied to the domain.

In [12], Bhattacharya and Ghosh describe a generalized method for comparatively evaluating different knowledge representation schemes. They use expressiveness and performance as the primary metrics. Expressiveness is defined as the capability to correctly express the information appearing in one scheme in terms of the other scheme. Performance is defined as how resource "hungry" a knowledge representation scheme is with respect to processing, memory consumption, errors involved, etc. They evaluate systems based on criteria such as time complexity, space complexity, accuracy, relational capacity, maintainability, and user friendliness. As test examples, they use these metrics to perform pair-wise comparisons of rule-based schemes, object-oriented schemes, relational schemes, and hybrid schemes. They determined that hybrid schemes are best for the representation of zonation of landslide hazards, which is the domain they used for their study.

In [13], Aruna et. al. propose an evaluation framework made up of a number of different existing tools including OntoAnalyser [14], OntoGenerator, OntoClean [15], ONE-T, and S-OntoEval [16]. The supposition is that all of these tools provide different functionalities and benefits and that a combination of all of them is needed to perform a thorough ontology evaluation. The criteria that are proposed for evaluation include:

- Ontology properties
  - language conformity (syntax)
  - consistency (semantics)
- Technology properties
  - interoperability
  - turn around ability
  - performance
  - memory allocation
  - scalability
  - integration into frameworks
  - connectors and interfaces

The paper explains why this is important, but never goes into detail about how these tools can be combined into a common framework. It simply describes each tool without any conclusions.

In [17], Brank, Grobelnik, and Mladenic perform a survey of various ontology evaluation techniques. They describe evaluation approaches at various "levels," including lexical/vocabulary/data layer, hierarchy/taxonomy (and other semantic relationships), context/application level, syntactic level, and structure/architecture/design. They also describe various evaluation approaches and classify them as (1) comparing to a golden standard, (2) using ontologies in specific applications, (3) comparing ontologies with source data (e.g., collection of documents), and (4) evaluations performed by humans. They do not give opinions on which is best or worst… they simply try to classify the different approaches.

In [18], Gruninger and Fox describe the concept of competency questions to help evaluate ontologies. They start by defining scenarios that are relevant to the domain for which the ontology is being developed, and then develop competency questions that capture the questions that the ontology is intended to be able to answer. From these questions, concepts are identified and defined. There should be a direct mapping from the competency questions and the concepts, such that all of the concepts are present that allow the competency questions to be answered and no concepts are present that do not contribute to the answer to the questions. This approach focuses more on evaluating the concepts that are represented in the ontology as opposed other metrics such as performance related issues.

In [1], Vrandecic presents a theoretical framework and several methods for ontology evaluation with a focus on the Semantic Web. He focuses on the following three scenarios as relevant for ontology evaluation:

- Mistakes and omissions in ontologies can lead to the inability of applications to achieve the full potential of exchanged data. Good ontologies lead directly to a higher degree of reuse of data and a better cooperation over the boundaries of applications and domains.
- People constructing an ontology need a way to evaluate their results and possibly to guide the construction process and any refinement steps. This will make the ontology engineers feel more confident about their results, and thus encourage them to share their results with the community and reuse the work of others for their own purposes.
- Local changes in ontology development and maintenance processes may affect the work of others who are using the ontology. Ontology evaluation technologies allow a system to automatically check if constraints and requirements are fulfilled, in order to automatically reveal usability and compatability problems.

5. EXISTING METRICS FOR EVALUATING ONTOLOGIES

There are many different aspects of ontologies that one can analyze and measure. There are at least five significant additional research efforts that have attempted to capture some of these metrics. An excellent overview of ontology evaluation efforts is described in [1] and many of the descriptions below are adapted from this work. A superset of all of these metrics are listed below in alphabetical order, with pointers to the publications from which they arose. Some liberty was taken and assumptions applied to cluster metrics when significant overlap was perceived.

- Clarity/Understandability: The ontology should effectively communicate the intended meaning of defined terms. Definitions should be objective. When a definition can be stated in logical axioms, it should be. Where possible, a definition is preferred over a description. All entities should be documented with natural language. [19] [20] [21]
- Competency: The goals and purpose of the ontology is described using competency questions and the ontology has the concepts (and only the concepts) necessary to successfully answer the questions. [18]
- Completeness/Coverage: All the knowledge that is expected to be in the ontology is either explicitly stated or can be inferred from the ontology. [2] [20]
- Computational Integrity and Efficiency: the principle characteristics of an ontology that can be successfully/easily processed by a reasoner (inference engine, classifier, etc.). These could include logical consistency, disjointness ratio, etc, [21]
- Conciseness / Minimal Ontological Commitment: The ontology should specify the weakest theory (i.e., allowing the most models) and defining only those terms that are essential to the communication of knowledge consistent with that theory. [2] [19]
- Consistency/Coherence: capturing both the logical consistency (i.e., no contradictions can be inferred) and the consistency between the formal and the informal descriptions (i.e., the comments and the formal descriptions match) [2] [19] [20]
- Expandability/Extendibility: An ontology should offer a conceptual foundation for a range of anticipated tasks, and the representation should be crafted so that one can extend and specialize the ontology monotonically. New terms can be introduced without the need to revise existing axioms. [2] [19]
- Mappability to upper level and other ontologies [20]
- Minimal encoding bias: An encoding bias results when representation choices are made purely for the convenience of notation or implementation. Encoding bias should be minimized, because knowledge-sharing agents may be implemented with different libraries and representation styles. [19]
- Relevance: Evaluation against specific use cases, scenarios, requirements, applications, end-user knowledge, and data sources the ontology was developed to address [20]
- Reusability/Flexibility: How easily the developed ontologies can be applied to unanticipated domains that require the same sort of knowledge or lend itself to various views. [20] [21]
- Sensitivity: relates to how small changes in an axiom alter the semantics of the ontology. [2]
- Soundness: Free from error [20] [21]
- Types of inferences that can be used [20]
- Usability/Organization Fitness: Compliance to procedures for extension, integration, adaptation, and access for effective application. Can it be easily deployed within an organization? [21]

This information in tabular form is included below:

Table 1: Ontology Evaluation Metrics

Metric	Gangemi [21]	Gomez-Perez [2]	Gruber [19]	Gruninger [18]	Obrst [20]
Clarity / Understandable	x		x		x
Competency				x	
Completeness / Coverage		x			x
Computational Integrity and Efficiency	x				
Conciseness / Minimal Ontological Commitment		x	x		
Consistency / Coherence		x	x		x
Expandability / Extendability		x	x		
Mappability					x
Minimal Encoding Bias			x		
Relevance					x
Reusability / Flexibility	x				x
Sensitivity		x			
Soundness	x				x
Types of Inferencing					x
Usability / Organization Fitness	x				

It is interesting to note the relatively minimal overlaps between the metrics mentioned in each of the papers. There is no metric that shows up on more than three of the research papers and this only happens two times. In addition, eight of the metrics only show up once in the five research papers. This could be due to a number of factors:

1. There is not broad agreement in the community about the metrics that should be used to evaluate ontologies.
2. There is some overlap among the requirements such that the same things are evaluated but are categorized differently. This could be due to the liberties that were taken by this paper's author to categorize the metric descriptions in the respective papers or from different sets of terminologies used by each paper's author.
3. The authors focused on specific aspects of ontology evaluation and did not try to take a comprehensive view of all of the aspects involved.

It is likely that this lack of overlap is due to some combination of all three items above, though it is the authors' belief that item #1 (lack of broad agreement) is the most substantial.

6. WHERE ARE THE GAPS?

Robots are innately real-time systems. However, real-time is a relative word. At the servo level, real-time can mean tens or hundreds of cycles per second. At the higher-level planning level, real-time can be on the order of tens of seconds or minutes (or even longer). The trick is to figure out where symbolic representations like ontologies play a role, both in the usefulness of the information that they provide and in the representation's ability to work within a system to deliver information at the rate necessary.

Many of the lower-level real-time aspects have been removed from the symbolic representation realm and applied to other types of representations that are better suited for them (e.g., parametric and spatial knowledge levels, as discussed in Section 2). While this has worked in the past, symbolic representations provide a level of information that would be valuable to real-time applications, including the ability to reason over existing knowledge at a level deeper than what is possible in other types of representations. As can be seen in Section 5, almost all of the metrics focus on the structure of the ontology, including clarity, completeness, relevance, sensitivity, soundness, etc. Almost none of the metrics focused on the functionality that the ontology supports, such as how quickly it is able to work within a system to process new data or how rapidly it is able to work within a system to provide useful data back to the application. This is alluded to in the metric "computational integrity and efficiency" but this was just presented as a concept in the literature without details of how one would go about analyzing it and how one would determine if the resulting metrics are suitable for the application of interest.

One area that will be explored in the future is coupling the ontology with other types of symbolic representations, such as databases, that may be able to handle real-time applications more efficiently at lower levels in the control hierarchy. In concept, there are several data structures in the ontology which would not need to be updated in real time and would likely stay static throughout an entire ontology application. This may include the names of certain objects, their capabilities, and in the case of static items, their locations.

For example, in a manufacturing plant performing automated kitting operations, the names of the machines, their locations, and their capabilities may stay the same during the entire operation. However, the exact location of their robotic arm, what kit they are working on at the time, and the parts that are being manipulated may change by the minute or second. The idea is that these "dynamic" concepts would have a link from their instances and structures in the ontologies to a database that would be dynamically updated as new information is made available from the sensor systems (or entered by a human).

Information can either be "pushed" from the database to the ontology instances when some criterion is reached (e.g., an object's location is moved by over a predefined distance, the state of the overall system reaches a milestone, an error state is detected, etc.), or can be "pulled" from the database to the ontology at certain time intervals or just before reasoning is about to be performed. With this approach, a system would rely on the database structures for the real-time access and updating functions but would still get the benefit of ontology reasoning through the links between the database and the ontology.

Another advantage of this approach is the reusability and semantics that the ontology provides that may not be available through the database alone. Databases are very good at representing concepts and their characteristics, but do not provide detailed semantics about what the concepts and characteristics mean. By coupling the database fields with the ontology instances, detailed semantics can be captured in the ontology while not slowing down the processing of the information in the database.

Once the application is concluded (e.g., a kitting operation), the resulting database information can be written back to the ontology and easily shared with other applications. This could include scheduling systems, process planning systems, or other management-type applications that have a need to see and understand the state of the factory at any given time. Ontologies are often developed to be highly reusable, thus providing another benefit of the database-ontology integration.

7. CONCLUSION

In this paper, we discuss some of the ways that knowledge is represented in robotic applications, describe an IEEE effort to standardize symbolic representation in robot systems, look at some metrics that have been applied to measuring the quality of symbolic representations, and provide thoughts on what other types of metrics and procedures may be necessary to measure the performance of symbolic representations (with an emphasis on ontologies) in robotic applications. This is the first paper in what is expected to be a series of papers detailing ways to measure and apply symbolic representations to the robotics field. With much of the research in this area not yet started, the purpose of this paper is to describe some related efforts and some preliminary thoughts that will set the stage for future work.

DISCLAIMER

The name of commercial products or vendors does not imply NIST endorsement or that this product is necessarily the best for the purpose.

REFERENCES

[1] D. Vrandecic, "Ontology Evaluation," PhD, Institute of Applied Informatics and Formal Description Methods, Karlsruher Instituts fur Technologie (KIT), 2010.

[2] A. Gomez-Perez, "Ontology evaluation," in *Handbook on Ontologies*. vol. First Edition, S. Staab and R. Studer, Eds., ed: Springer, 2004, pp. 251-274.

[3] E. Messina, J. Albus, C. Schlenoff, and J. Evans, "Knowledge Engineering for Real Time Intelligent Control," *Journal of Intelligent & Fuzzy Systems,* vol. 14, 2003.

[4] "ISO 10303-21, Industrial automation systems and integration - Product data representation and exchange - Part 21: Clear Text Encoding of the Exchange Structure," Geneva, Switzerland1994.

[5] T. Gruber, "A Translation Approach to Portable Ontology Specification," *Knowledge Acquisition,* vol. 5, p. 22, 1993.

[6] D. Lenat, R. Guha, K. Pittman, D. Pratt, and M. Shephard, "CYC: Toward Programs with Common Sense," *Communications of the ACM,* vol. 33, pp. 30-49, 1990.

[7] M. Genesereth and R. Fikes, "Knowledge Interchange Format," Stanford University.

[8] W. F. Clocksin and C. S. Mellish, *Programming in Prolog*. New York: Springer-Verlag, 2003.

[9] R. Madhavan, Yu. W, Biggs, G., Schlenoff, C., Huang, H.M., "IEEE RAS Standing Committee for Standards Activities: History and Status Update," *Journal of the Robotics Society of Japan, Special Issue on "Activities of International Standards for Robot Technologies",* vol. 29, May 2011.

[10] C. Schlenoff, E. Prestes, R. Madhavan, P. Goncalves, H. Li, S. Balakirsky, T. Kramer, and E. Miguelanez, "An IEEE Standard Ontology for Robotics and Automation," in *to be published in Bridges Between the Methodological and Practical Work of the Robotics and Cognitive Systems Communities - From Sensors to Concepts*, A. Chibani, Ed., ed: Springer-Verlag, 2012.

[11] (2009). *W3C OWL Working Group, OWL 2 Web Ontology Language Document Overview*. Available: http://www.w3.org/TR/owl2-overview/

[12] D. Bhattacharya and J. K. Ghosh, "Evaluation of Knowledge Representation Schemes as a Prerequisite towards Development of a Knowledge-Based System," *Journal of Computing in Civil Engineering,* November/December 2008 2008.

[13] T. Aruna, K. Saranya, and C. Bhandari, "A Survey of Ontology Evaluation Tools," presented at the International Conference on Process Automation, Control and Computing (PACC), Coimbatore, India, 2011.

[14] D. Rogozan and G. Paquette, "Managing Ontology Changes on the Semantic Web," presented at the IEEE/WIC/ACM International Conference on Web Intelligence (WI'05), France, 2005.

[15] A. Oltramari, A. Gangemi, N. Guarino, and C. Masolo, "Restructuring WordNet's Top=Level: The OntoClean approach.," presented at the LREC2002 (OntoLex workshop), Las Palmas, Spain, 2002.

[16] R. Dividino, M. Romanelli, and D. Sonntag, "Semiotic-based Ontology Evaluation Tool (S-OntoEval)," presented at the Sixth International Conference on Language Resources and Evaluation (LREC'08), Marrakech, Morocco, 2008.

[17] J. Brank, M. Grobelnik, and D. Mladenic, "A survey of ontology evaluation techniques," presented at the Conference on Data Mining and Data Warehouses (SikDD), Ljubljana, Slovenia, 2005.

[18] M. Gruninger and M. Fox, "Methodology for the Design and Evaluation of Ontologies," presented at the IJCAI'95 Workshop on Basic Ontological Issues in Knowledge Sharing, Montreal Quebec, 1995.

[19] T. Gruber, "Towards principles for the design of ontologies used for knowledge sharing," *International Journal of Human-Computer Studies,* vol. 43, 1995.

[20] L. Obrst, W. Ceusters, I. Mani, S. Ray, and B. Smith, "The evaluation of ontologies," in *Revolutionizing Knowledge Discovery in the Life Sciences*, C. J. O. Baker and K.-H. Cheung, Eds., ed: Springer, 2007, pp. 139-158.

[21] A. Gangemi, C. Catenacci, M. Ciaramita, and J. Lehmann, "Ontology evaluation and validation: an integrated formal model for the quality diagnostic task," Rome, Italy, 2005.

A Hybrid Approach to 2D Robotic Map Evaluation

Ross Creed
Temple University
1801 N. Broad St
Philadelphia, PA
+1 215-204-3949
ross.creed@temple.edu

Kristiyan Georgiev
Temple University
1801 N. Broad St
Philadelphia, PA
+1 215-204-3949
georgiev@temple.edu

Rolf Lakaemper
Temple University
1801 N. Broad St
Philadelphia, PA
+1 215-204-7996
lakamper@temple.edu

ABSTRACT

This article introduces the Temple Map Evaluation Toolkit (TMET), which is a tool for evaluating robotic maps produced by existing mapping algorithms. The toolkit performs ground truth based evaluation, i.e. it compares similarities between a map defined as ground truth and a target map. TMET allows for hybrid evaluation, since methods for pose based as well as grid based evaluation are implemented. For pose based evaluation, the user can define regions on the ground truth map which are handled as transformable submaps. TMET allows for evaluation of grid based maps as well as segment based maps, and therefore covers most of the representations of maps for existing mapping algorithms. The paper introduces the toolkit and the underlying design principles and algorithms. Experiments with maps from simulated as well as real world data are presented, demonstrating that the tool can be used to evaluate the quality of a map in a quantitative way.

Categories and Subject Descriptors

D.2.8 [**Software Engineering**]: Metrics—*performance measures*

General Terms

Standardization

Keywords

robotic mapping, map evaluation

1. INTRODUCTION

In recent years, the problem of simultaneous localization and mapping (SLAM) has been advanced to a state where many consider the problem to be solved in a two dimensional planar environment (i.e. a one story building). Measuring the performance of these solutions requires a scientifically sound and statistically significant metric, and evaluation methodologies and tools for quantifying a solution's performance. Because of the many different uses for robotic maps, a metric has yet to be found which is both significant and fair for a map while considering its entire usage space. For example: an Autonomous Ground Vehicle (AGV) requires accurate geometric map to avoid obstacles, where on the other hand a rescue worker searching for victims is concerned with the topological correctness of the map, as to see if the victims are currently reachable from the rescuers current position. Additionally, created maps lack standardization, and are presented in several different formats (segment-based map, point based maps, and image/grid based maps).

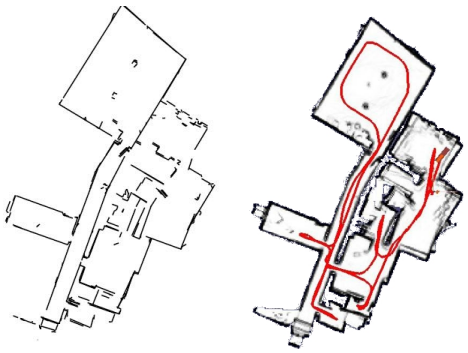
The solution requires a framework for generating accurate representations that take into account the multifaceted nature of the operational domain. Part of the difficulty in comparing maps comes from the lack of a concrete measurement ubiquitous through all maps produced from different mapping algorithms. The end result of most mapping algorithms is either positional data of each point from the range data acquired by the robot, or an image created from similar data. Both of these results can be extended to 2D mid-level geometry, in the form of line segments. Line segments offer more information when dealing with spacial data, and they accurately represent the contextual data of the map. Another benefit is less storage cost, since usually ~100 points/pixels are represented by a single segment.

This paper presents the Temple Map Evaluation Toolkit (TMET), a tool to aid in designing experiments and test methods to enable performance evaluation and benchmarking towards characterizing constituent components of navigation and world modeling systems. The benchmarks given by the system provide statistically significant results and quantifiable performance data. The encountered challenges and methodologies involved in creating TMET are discussed and applied to robotic maps used for a wide variety of applications.

Currently there is no established standard tool for benchmarking and quantitatively evaluating the performance of robotic mapping systems against user defined requirements. The most widely used indicator of the quality of a map is visual inspection. This visual inspection method does not aid in understanding what specific errors systems are prone to, but it has become common practice in the literature to compare newly developed mapping algorithms with former methods by presenting images of generated maps (Figure 1). This procedure of course offers no standard way for evaluation and becomes even infeasible, particularly when applied to large-scale maps. TMET offers tools to quantify map differences, based on a hybrid measure using an underlying segment representation that can handle segment, point, and image map representations.

The rest of this paper is organized as follows: related projects are discussed in section 2. The methods used by the toolkit will then be laid out, starting with the theory between the hybrid measure in section 3, followed by implementation details in section 4. Finally, experiments, conclusions and future work will be presented

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. PerMIS'12, March 20-22, 2012, College Park, MD, USA.
Copyright 2012 ACM 1-4503-1126-7-3/22/12 ...$10.00.



(a) Map 1 (b) Map 2

Figure 1: Two maps generated by different algorithms using the same data. Using the prior technique of visual inspection, it is difficult to determine which map is 'better'. Map 1 is taken from [5] and Map 2 is produced by the RSLAM algorithm [12].

in sections 5, and 6 respectively.

2. RELATED WORK

Learning robotic maps has been a frequently studied problem in robotics literature. From this literature, several areas of though have become prominent. The extended Kalman filter (EKF) [8] [11], particle filters [13], grid maps [6] and least square error minimization approaches [3] are some of the most commonly used approaches. These approaches have long been without an unbiased means of comparison, until recent developments in map evaluation. Two approaches of map evaluation have emerged, grid-based evaluation [14] and pose-based [9] evaluation.

The extended Kalman filter is a successful approach for mapping because of the full estimation of the posterior probabilities of the map elements and robot poses. The weakness associated with the approach is the assumptions made both in the motion of the robot and sensor noise. These assumptions, if not made correctly, can lead to drifting in the alignment process and an insufficient map.

GraphSLAM [10] is a technique used to improve upon EKFs by creating a graph of all robot pose relationships, and relaxing the graph when a cycle is present in the graph. This amendment has led GraphSLAM to be one of the leading techniques for map creation.

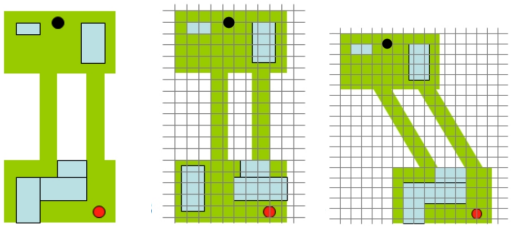
Particle filter approaches keep track of many hypothesis of the location and path of the robot while doing mapping. By finding the best trajectory among the candidate particles, a map can be formed based on the scans taken along the chosen trajectory.

In these cited papers, each technique claims to be better than previous techniques. The method used to show this is a presentation of maps using similar datasets. There are no quantifiable results presented, and the lack of these have lead to an interest in evaluating maps. In [14] a measure is presented by comparing the distance images created by map images. This measure accurately compares the geometric correctness of a map, but fails to consider other potential uses. In [17] an evaluation is presented based on the corrected trajectories of the robot. Although this method claims to need no 'ground truth' data, the reference poses of the robot are needed, and these are often unavailable. The proposed metric in this paper is a hybrid of these two approaches, comparing a map to a ground truth, and penalizing for both geometric and pose-based errors.

3. HYBRID MAPPING METHODOLOGY

Mapping, in general, is spatial analysis of environmental features of interest. Inherent to this process is its task dependency, hence there is no 'optimal general mapping'. Mapping can be divided into two classes: topographic and topological mapping. Topographic mapping is concerned with detailed and correct geometry, while topological mapping is concerned only with the correct spatial relation between features. They are often referred to as 'global correctness' vs. 'local accuracy' and can be related to grid and pose based approaches in map evaluation.

Figure 2 presents a normal scenario in robotics, where a robot is in a start position represented by the black dot. It wants to reach a goal state, represented by the red dot using one of the three maps. Since the map in Figure 2(b) is geometrically more similar to the ground truth shown in Figure 2(a) a geometric approach would prefer this map. In completing the desired task using this map, the robot would choose to navigate down the left hallway, and would never be able to reach the goal state. In contrast, a pose based approach would favor the map seen in Figure 2(c), and although it would choose the correct hallway to go down in order to reach the goal, the diagonal orientation would prove troublesome while attempting to drive to the goal. Thus, neither approach on it's own performs satisfactorily in this scenario.



(a) (b) (c)

Figure 2: Example maps for a robot trying to get from the start point (black dot) to a goal state (red dot). 2(a) represents the ground truth, and 2(b) and 2(c) are sample robotic maps demonstrating flaws in geometry and topology, respectively.

3.1 Grid-Based Evaluation

The RoboCup Rescue competitions [1] have proved to be a good forum to evaluate task-based performance of robots. An image similarity metric and a cross entropy metric are outlined in [4] to measure the quality of occupancy grid maps. The metric gives an indication of distortion of the map with respect to a ground truth map in the presence of noise and pose errors. This metric is embedded in the Jacobs Map Analysis Toolkit [14] and has been tested for comparing maps in the RoboCup Rescue context.

The Jacobs Map Analysis Toolkit, recently extended to evaluate maps using fiduciary markers (objects added to the environment for evaluation purposes), is purely tailored to perform evaluation of geometric precision, which limits its versatility to be applied to evaluation of maps under different aspects. TMET provides a different approach to mapping evaluation based on the principles of an algorithm which first allows for a topological alignment then considers the geometric precision.

3.2 Pose-Based Evaluation

The numerous applications of robotic maps that don't need correct geometry and the lack accurate ground truth data in datasets led to the development of pose-based methods. In this method the rel-

ative differences between robot poses are evaluated without the use of a global reference frame. This measure is not publicly available as a software suite.

Using differences between robot poses as an evaluation measure has several drawbacks as well. Even though no ground truth map is needed, an accurate measurement of the robots position is needed, which is often unavailable due to sensor noise and wheel slippage of the robot. Also, this evaluation doesn't take into account the noise of the vision sensor. Even if the robot is perfectly positioned, errors in a visual sensor can contribute to errors in the map.

3.3 Hybrid Evaluation

Since neither of the above approaches seem to work in all cases, but both have convincing arguments, a mixture between the two is appropriate in most situations. By first correcting for pose-based errors, then examining the geometry of the created map, a measure is created that is both examines topological and topographic accuracy.

In addition to this criteria, TMET can also measure other factors important in mapping. For instance, the question of local accuracy vs. global coverage can be examined. After performing pose-based alignment, geometric comparisons can be performed over the scope of the local map to find local accuracy, and over the scope of the ground truth map to determine completeness. This can be useful in sever scenarios when it was infeasible for a robot to map the entire space it was in.

4. IMPLEMENTATION

When implementing TMET, several design aspects were accomplished. The tool is easily accessible, intuitive to use, flexible, and provides an accurate and intuitive response. This section will describe the design of the tool and the process flow for how the tool can be used.

The implementation of TMET was written in the JavaFX [16] language. This makes it portable across all platforms, and easily accessible through the java web start environment. No installations and only a small download (~1Kb) is required, making the program easily accessible to any Java based web-capable device.

To make the program as easily understandable by the user, the process flow was done in a 'wizard' format. This means the program has several states, and at each state the user is presented with the tools they need and instructed to complete an action. By following the defined actions, the user ends at the result state, where the map evaluation metric is displayed and the user is able to retrace their steps and tweak parameters until a desirable result is achieved.

4.1 Process Flow

First, the user is asked to load a ground truth map, or a map in which to compare the map in question against (query map). The program supports several types of map inputs. Segment based maps can be loaded, where a segment based map is stored in a file with a .seg extension, and the file contains a list of the 2D start and end points of each segment. After loading a segment map it is displayed to show accuracy. Also, the user can load maps in image formats, where allowable image types are JPEG, GIF, and PNG. Once the file is loaded, the user is asked to select a binary threshold for the image (binarization is necessary to perform the morphological operations involved in converting the image into segment representation) as seen in Figure 3(a). After a threshold is chosen, the image is converted to segments and displayed for the user.

In the next step, the user is asked to 'chop' the ground truth map into pieces, seen in Figure 3(b). In this step, the user should identify areas that need to be geometrically precise and consistent in the query map, and isolate these in separate regions. A typical example is to separate rooms from a hallway: geometric precision in the rooms might be needed, while topological consistency between rooms and hallway might be sufficient. Please note that the ground truth map is separated into regions, not the target map. Therefore, this step can be performed before the actual evaluation process. Additionally, region separation (i.e. identification of topological structure of the ground truth map) is performed by the user only once, and stays the same for evaluation of different query maps.

The query map is loaded using the same process as for the ground truth map, and the two maps are superimposed. No chopping is performed on the query map. The regions of the ground truth map are then aligned to the query map. This can be done either automatically or, if needed, with manually (in the case of strongly distorted query map, a manual pre-alignment might be needed to assist the automatic alignment). The toolkit records the parameters for the underlying region transformations (rotation and translation) and derives a quality measure for it. The transformation step, which leads to topological evaluation, is followed by geometric evaluation using the measure described in [14].

The final measure displayed by the toolkit is computed as follows:

$$EM(qm,gt) = \alpha \sum_{i=0}^{n} Rotation(i) * \beta \sum_{j=0}^{n} Translation(j) * \gamma * G \quad (1)$$

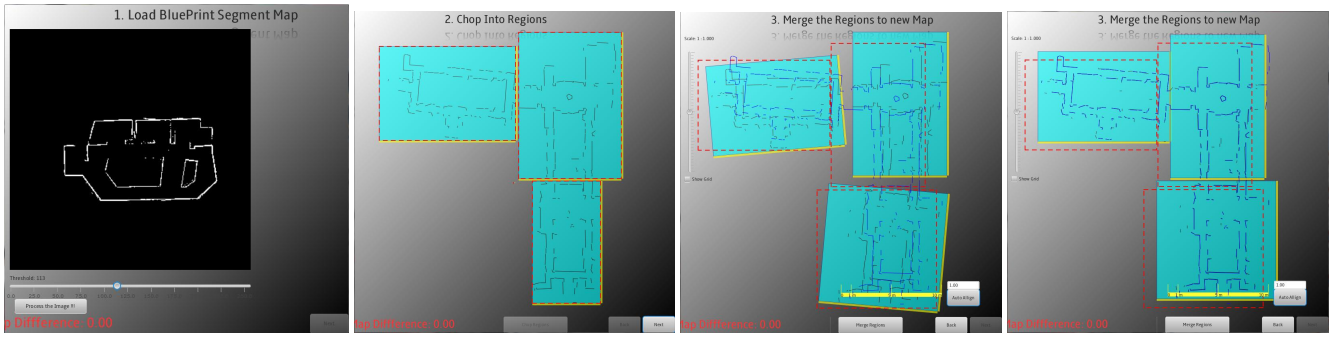
where G is the geometric evaluation of the two maps, $Rotation(x)$ and $Translation(x)$ are the rotation and translation of piece x of the chopped map, and α, β, and γ are user defined weight parameters. These parameters offer flexibility for the user, who can determine the weights for grid evaluation (geometric precision) or pose evaluation (topological precision).

5. EXPERIMENTS

To show the applicability of TMET, two experiments were conducted. First, three maps of the 'Stanford gates' dataset, found in [7], were compared. Second, several maps were created using a dataset of the tenth floor of Wachman Hall at Temple University. These maps were created using different parameters, and the best performing map is found.

Figure 4: The chose chopping scheme for the first experiment.

For the first experiment, the 'gates' data set was chosen from the Radish repository, and two implementations using this data set



(a) Load Map (b) Chop Map (c) Align Maps (d) Auto-Aligned Maps

Figure 3: The steps of the Hybrid Toolkit. In 3(a) the load map screen and binarization of images are shown. In 3(b) the ground truth map is chopped into regions. In 3(c) the user manually aligns the two maps, and in 3(d) the auto-aligned maps are shown along with the corresponding error metric.

were found and evaluated. These two maps ([15][2]) were chosen because of the qualitative, visible differences between them. The maps can be seen in Figure 5. The division used for chopping can be seen in Figure 4.

The results from the first experiment can be seen in Table 5. From this table we can see that Map 1 is more topographically correct than Map 2 (by examining the first and the seventh row where $\alpha = \beta = 0$), but Map 2 is more topologically correct (examine rows six and twelve where $\gamma = 0$). These results are consistent with visual examination of the maps.

	α	β	γ	Result
Map 1	0	0	1	6.09
	0.1	0.1	0.8	35.76
	0.2	0.2	0.6	65.43
	0.3	0.3	0.4	95.11
	0.4	0.4	0.2	124.78
	0.5	0.5	0	154.46
Map 2	0	0	1	8.40
	0.1	0.1	0.8	20.55
	0.2	0.2	0.6	32.69
	0.3	0.3	0.4	44.83
	0.4	0.4	0.2	56.98
	0.5	0.5	0	69.12

Table 1: The results of of experiment 1 using different weights of topological and topographical features.

In the second experiment, a sample data set was taken from the tenth floor of Wachman Hall at Temple University. In contrast to the previous experiment, the input data here are segment based maps. Four maps were created from this data set using different parameters in a Kalman Filter Based SLAM algorithm. The results can be seen in Figure 6. The query maps were compared to the ground truth (we defined the visually best map as the ground truth, which is sufficient to demonstrate the TMET functionalities), using the same chopping scheme, and the results for evaluation can be seen in Table 5. These results support the visual claims that Map 2 is more geometrically equivalent to the ground truth because of a lack of noise in the map. When the hybrid measure is weighted equally, Map 2 is still better, but the discrepancy between the two is much less when topological factors are considered.

	α	β	γ	Result
Map 1	0	0	1	112.09
	0.25	0.25	.5	74.25
	0.5	0.5	0	36.42
Map 2	0	0	1	4.12
	0.25	0.25	.5	58.41
	0.5	0.5	0	112.69

Table 2: The results of of experiment 2 using different weights of topological and topographical features.

6. CONCLUSIONS AND FUTURE WORK

The Temple Map Evaluation Toolkit allows for hybrid evaluation of grid and segment based maps. It is intuitive, and easy to use. The user defined parameters allow for different target functions, which makes TMET applicable to different tasks in map evaluation. The automatic alignment allows fast pose based evaluation. TMET's implementation in JavaFX enables cross platform usage, TMET runs on smart phones and tablets. In the future, TMET will be extended to handle the evaluation of 3D maps.

7. REFERENCES

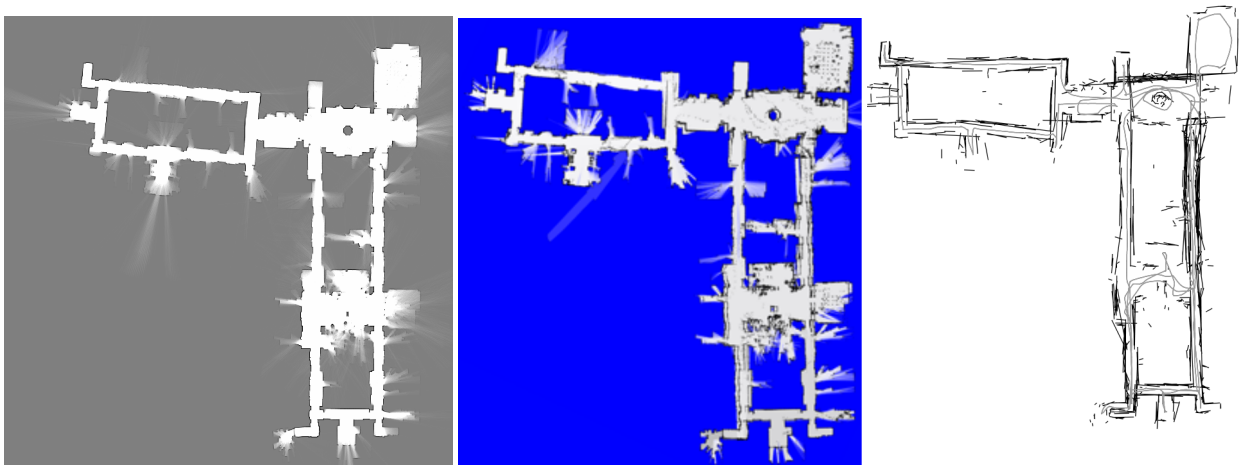
[1] J. Baltes, M. G. Lagoudakis, T. Naruse, and S. S. Ghidary, editors. *RoboCup 2009: Robot Soccer World Cup XIII [papers from the 13th annual RoboCup International Symposium, Graz, Austria, June 29 - July 5, 2009]*, volume 5949 of *Lecture Notes in Computer Science*. Springer, 2010.

[2] K. Beevers and W. Huang. Slam with sparse sensing. In *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, pages 2285–2290, may 2006.

[3] P. Besl and N. McKay. A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14:239–256, 1992.

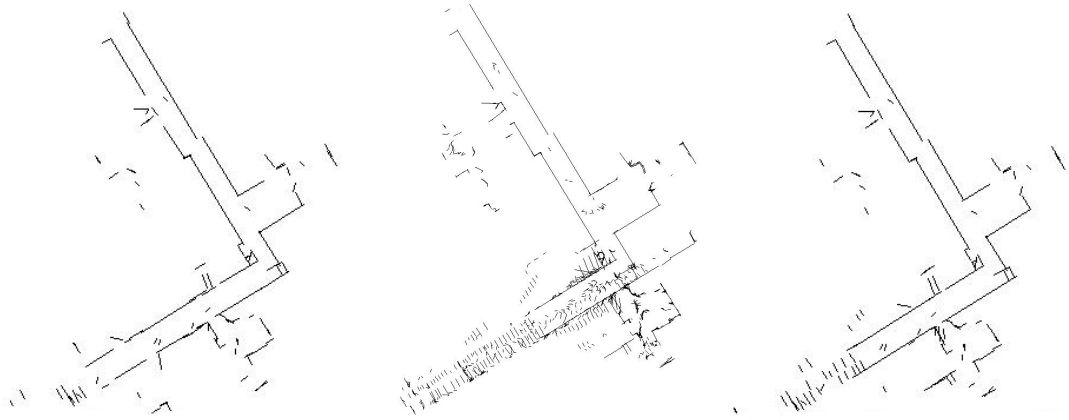
[4] A. Birk. A quantitative assessment of structural errors in grid maps. *Autonomous Robots*, 28:187–196, 2010. 10.1007/s10514-009-9159-2.

[5] J. Elseberg, R. T. Creed, and R. Lakaemper. A line segment based system for 2d global mapping. *IEEE International Conference on Robotics and Automation (ICRA)*, pages 3924–3931, 2010.



(a) Ground Truth Map (b) Map 1 (c) Map 2

Figure 5: From Right to Left: the ground truth and two query maps used in the first experiment.



(a) Ground Truth (b) Map 1 (c) Map 2

Figure 6: From Right to Left: the ground truth and two segment-based query maps used in the second experiment.

[6] G. Grisetti, C. Stachniss, and W. Burgard. Improving grid-based slam with rao-blackwellized particle filters by adaptive proposals and selective resampling. pages 2443–2448, Barcelona (Spain), 2005. ISBN: 0-7803-8914-X.

[7] A. Howard and N. Roy. The robotics data set repository (radish), 2003.

[8] R. E. Kalman. A new approach to linear filtering and prediction problems. 1960.

[9] R. Kümmerle, B. Steder, C. Dornhege, M. Ruhnke, G. Grisetti, C. Stachniss, and A. Kleiner. On measuring the accuracy of SLAM algorithms. *Autonomous Robots*, 27(4):387–407, 2009.

[10] F. Lu and E. Milios. Globally consistent range scan alignment for environment mapping. *Autonomous Robots*, 4:333–349, 1997.

[11] B. S. Rao and H. F. Durrant-Whyte. Fully decentralised algorithm for multisensor kalman filtering. *Iee Proceedings D Control Theory And Applications*, 138(5):413–420, 1991.

[12] D. Sun, A. Kleiner, and T. M. Wendt. *Multi-robot Range-Only SLAM by Active Sensor Nodes for Urban Search and Rescue*, pages 318–330. Springer-Verlag, Berlin, Heidelberg, 2009.

[13] S. Thrun. Particle filters in robotics. In *Proceedings of the 17th Annual Conference on Uncertainty in AI (UAI)*, 2002.

[14] I. Varsadan, A. Birk, and M. Pfingsthorn. Determining map quality through an image similarity metric. In L. Iocchi, H. Matsubara, A. Weitzenfeld, and C. Zhou, editors, *RoboCup 2008: Robot Soccer World Cup XII*, volume 5399 of *Lecture Notes in Computer Science*, pages 355–365. Springer Berlin / Heidelberg, 2009. 10.1007/978-3-642-02921-9_31.

[15] A. Visser, B. A. Slamet, and M. Pfingsthorn. Robust weighted scan matching with quadtrees. In *Proc. of the Fifth International Workshop on Synthetic Simulation and Robotics to Mitigate Earthquake Disaster (SRMED 2009)*, 2009.

[16] J. L. Weaver. *JavaFX Script: Dynamic Java Scripting for Rich Internet/Client-side Applications*.

[17] O. Wulf, A. Nuchter, J. Hertzberg, and B. Wagner. Ground truth evaluation of large urban 6d slam. In *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pages 650–657, 29 2007-nov. 2 2007.

An Overview of Robot-Sensor Calibration Methods for Evaluation of Perception Systems

Mili Shah
Loyola University Maryland
4501 North Charles Street
Baltimore, MD, 21210
1-410-617-2724
mishah@loyola.edu

Roger D. Eastman
Loyola University Maryland
4501 North Charles Street
Baltimore, MD, 21210
1-410-617-2281
reastman@loyola.edu

Tsai Hong
National Institute of Standards and Technology
100 Bureau Drive
Gaithersburg, MD, 20899
1-301-975-3444
hongt@nist.gov

ABSTRACT

In this paper, an overview of methods that solve the robot-sensor calibration problem of the forms $\mathbf{AX} = \mathbf{XB}$ and $\mathbf{AX} = \mathbf{YB}$ is given. Each form will be split into three solutions: separable closed-form solutions, simultaneous closed-form solutions, and iterative solutions. The advantages and disadvantages of each of the solutions in the case of evaluation of perception systems will also be discussed.

Categories and Subject Descriptors

C.4 [**Performance of Systems**]: Performance attributes; B.8.2 [**Performance and Reliability**]: Performance Analysis and Design Aids; G.1.6 [**Optimization**]: Global optimization; I.4.8 [**Scene Analysis**]: Motion, Tracking; I.5.4 [**Applications**]: Computer Vision

General Terms

Computer Vision, Robot-Sensor Calibration, Hand-Eye Calibration, Performance Evaluation

1. INTRODUCTION

Robot-sensor calibration has been an active area of research for many decades. The most common mathematical representations for the robot-sensor calibration problem consist of two forms: $\mathbf{AX} = \mathbf{XB}$ and $\mathbf{AX} = \mathbf{YB}$. Examples for each of the forms can be seen in Figure 1. Specifically in Figure 1a, $\mathbf{A}_i$ represents robot motion, $\mathbf{B}_i$ represents camera motion, and the unknown $\mathbf{X}$ represents the fixed homogeneous transformation between the robot base and camera. Following the arrows, it can easily be seen that

$$\mathbf{A}_i\mathbf{X} = \mathbf{X}\mathbf{B}_i \Rightarrow \mathbf{AX} = \mathbf{XB},$$

where $\mathbf{A} = \mathbf{A}_i$ and $\mathbf{B} = \mathbf{B}_i$. Similarly in Figure 1b, $\mathbf{A}_i$ represents the transformation from robot base to gripper, $\mathbf{B}_i$ represents the transformation from camera to object, and the unknown $\mathbf{X}$ represents the fixed homogeneous transformation between gripper and camera. Following the arrows

$$\mathbf{A}_1\mathbf{X}\mathbf{B}_1 = \mathbf{A}_2\mathbf{X}\mathbf{B}_2 \Leftrightarrow \mathbf{A}_2^{-1}\mathbf{A}_1\mathbf{X} = \mathbf{X}\mathbf{B}_2\mathbf{B}_1^{-1} \Rightarrow \mathbf{AX} = \mathbf{XB},$$

where $\mathbf{A} = \mathbf{A}_2^{-1}\mathbf{A}_1$ and $\mathbf{B} = \mathbf{B}_2\mathbf{B}_1^{-1}$. Finally in Figure 1c, $\mathbf{A}_i$ represents the transformation from target to sensor, $\mathbf{B}_i$ represents the transformation from camera to object, the unknown $\mathbf{X}$ represents the fixed homogeneous transformation between sensor and object, and the unknown $\mathbf{Y}$ represents the fixed homogeneous transformation between target and camera. Following the arrows

$$\mathbf{A}_i\mathbf{X} = \mathbf{Y}\mathbf{B}_i \Rightarrow \mathbf{AX} = \mathbf{YB},$$

where $\mathbf{A} = \mathbf{A}_i$ and $\mathbf{B} = \mathbf{B}_i$.

In this paper, we will give an overview of methods to solve $\mathbf{AX} = \mathbf{XB}$ and $\mathbf{AX} = \mathbf{YB}$. Notice that for

$$\begin{aligned}
\mathbf{AX} &= \mathbf{XB} \\
\begin{pmatrix} \mathbf{R_A} & \mathbf{t_A} \\ 0 & 1 \end{pmatrix}\begin{pmatrix} \mathbf{R_X} & \mathbf{t_X} \\ 0 & 1 \end{pmatrix} &= \begin{pmatrix} \mathbf{R_X} & \mathbf{t_X} \\ 0 & 1 \end{pmatrix}\begin{pmatrix} \mathbf{R_B} & \mathbf{t_B} \\ 0 & 1 \end{pmatrix} \\
\begin{pmatrix} \mathbf{R_A}\mathbf{R_X} & \mathbf{R_A}\mathbf{t_X} + \mathbf{t_A} \\ 0 & 1 \end{pmatrix} &= \begin{pmatrix} \mathbf{R_X}\mathbf{R_B} & \mathbf{R_X}\mathbf{t_B} + \mathbf{t_X} \\ 0 & 1 \end{pmatrix},
\end{aligned}$$

Thus,

$$\mathbf{R_A}\mathbf{R_X} = \mathbf{R_X}\mathbf{R_B},$$

which we will define as the *orientational component*, and

$$\mathbf{R_A}\mathbf{t_X} + \mathbf{t_A} = \mathbf{R_X}\mathbf{t_B} + \mathbf{t_X}.$$

which we will define as the *positional component* for $\mathbf{AX} = \mathbf{XB}$. The orientational component

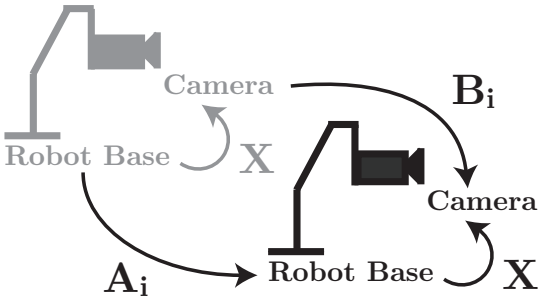
$$\mathbf{R_A}\mathbf{R_X} = \mathbf{R_Y}\mathbf{R_B},$$

and positional component

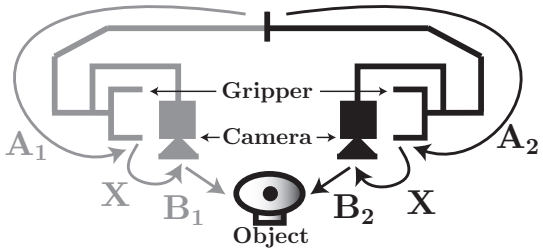
$$\mathbf{R_A}\mathbf{t_X} + \mathbf{t_A} = \mathbf{R_Y}\mathbf{t_B} + \mathbf{t_Y}$$

for $\mathbf{AX} = \mathbf{YB}$ can similarly be constructed. The methods to solve $\mathbf{AX} = \mathbf{XB}$ and $\mathbf{AX} = \mathbf{YB}$ consist of three forms: separable closed-form solutions, simultaneous closed-form solutions, and iterative closed-form solutions. The separable closed-form solutions arise from solving the orientational component separately from the positional component, the simultaneous closed-form solutions arise from simultaneously solving the orientational component and the positional component, while the iterative solutions arise from solving both the orientational component and positional component iteratively using optimization techniques. Details of each of the

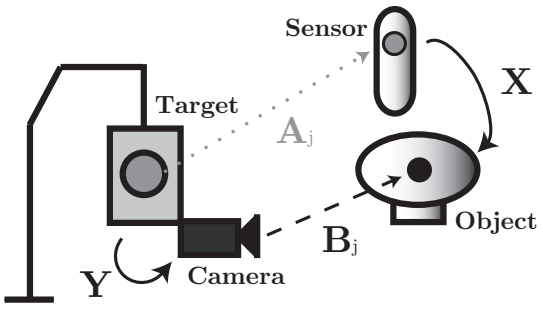
(c) 2012 Association for Computing Machinery. ACM acknowledges that this contribution was authored or co-authored by a contractor or affiliate of the U.S. Government. As such, the Government retains a nonexclusive, royalty-free right to publish or reproduce this article, or to allow others to do so, for Government purposes only.
PerMIS '12, March 20-22, 2012, College Park, MD, USA.
Copyright © 2012 ACM 978-1-4503-1126-7-3/22/12 ...$10.00.



(a) $\mathbf{AX} = \mathbf{XB}$ where the unknown $\mathbf{X}$ represents the transformation from robot base to camera.



(b) $\mathbf{AX} = \mathbf{XB}$ where the unknown $\mathbf{X}$ represents the transformation from gripper to camera.



(c) $\mathbf{AX} = \mathbf{YB}$ where the unknown $\mathbf{X}$ represents the transformation from sensor to object and the unknown $\mathbf{Y}$ represents the transformation from target to camera.

Figure 1: Different experimental setups for robot-sensor calibration.

solutions will be discussed in the following sections. Specifically for $\mathbf{AX} = \mathbf{XB}$, separable closed-form solutions will be discussed in Section 2.1, simultaneous closed-form solutions will be discussed in Section 2.2, and iterative solutions will be discussed in Section 2.3. Following, in Section 3, will be a section discussing the different solutions for $\mathbf{AX} = \mathbf{YB}$. Finally, concluding remarks, which will include the advantages and disadvantages for each of the solutions in the evaluation of perception systems, will be discussed in Section 4.

2. AX=XB SOLUTIONS

2.1 Separable Solutions for AX=XB

The robot-sensor calibration problem of the form $\mathbf{AX} = \mathbf{XB}$ was introduced in the work of Shiu and Ahmad [21]. In this paper, they solve the robot-sensor calibration problem by separating the problem into its orientational component

$$\mathbf{R_A R_X} = \mathbf{R_X R_B}$$

and positional component

$$\mathbf{R_A t_X} + \mathbf{t_A} = \mathbf{R_X t_B} + \mathbf{t_X}.$$

They solve the orientational component by utilizing the angle-axis formulation of rotation; i.e., let $\mathbf{R} = \mathrm{Rot}(k_\mathbf{R}, \theta)$, where $k_\mathbf{R}$ is the axis of rotation of $\mathbf{R}$ and θ is the angle. Specifically, they state that the general solution

$$\mathbf{R_X} = \mathrm{Rot}(k_{\mathbf{A}_i}, \beta_i)\mathbf{R}_{\mathbf{X}_{P_i}},$$

where

$$\begin{aligned} \mathbf{R}_{\mathbf{X}_{P_i}} &= \mathrm{Rot}(\mathbf{v}, \omega) \\ \mathbf{v} &= k_{\mathbf{B}_i} \times k_{\mathbf{A}_i} \\ \omega &= \mathrm{atan2}(|k_{\mathbf{B}_i} \times k_{\mathbf{A}_i}|, k_{\mathbf{B}_i} \cdot k_{\mathbf{A}_i}) \end{aligned}$$

and β_i is calculated by solving a $9 \times 2n$ linear system of equations where the number of frames $n \geq 2$. They also prove for uniqueness at least two of the axes of rotation of $\mathbf{R}_{\mathbf{A}_i}$ cannot be parallel. Once $\mathbf{R_X}$ is formulated, the positional component

$$\begin{pmatrix} \mathbf{R}_{A_1} - \mathbf{I} \\ \vdots \\ \mathbf{R}_{A_n} - \mathbf{I} \end{pmatrix} \mathbf{t_X} = \begin{pmatrix} \mathbf{R_X t}_{\mathbf{B}_1} - \mathbf{t}_{\mathbf{A}_1} \\ \vdots \\ \mathbf{R_X t}_{\mathbf{B}_n} - \mathbf{t}_{\mathbf{A}_n} \end{pmatrix}$$

can be solved using standard linear system techniques. This is the general technique of separable solutions for $\mathbf{AX} = \mathbf{XB}$: first calculate $\mathbf{R_X}$ using some technique and then use that $\mathbf{R_X}$ to solve for $\mathbf{t_X}$ using standard linear system techniques. Thus, for the rest of this section concentration will be placed solely on calculating the optimal rotation $\mathbf{R_X}$.

A problem with the Shiu and Ahmad method is that the size of the linear system doubles each time a new frame is added to the system. An alternative method by Tsai and Lenz [23] solves the robot-sensor calibration method using a fixed size linear system. The derivation is simpler than the Shiu and Ahmad method and computationally more efficient. Specifically, Tsai and Lenz solve the orientational component by again considering the angle-axis formulation $\mathbf{R} = \mathrm{Rot}(k_\mathbf{R}, \theta)$ for rotation. They find the axis of rotation $k_{\mathbf{R_X}}$ for $\mathbf{R_X}$ by solving

$$\begin{aligned} \mathrm{Sk}\left(k_{\mathbf{R}_{A_i}} + k_{\mathbf{R}_{B_i}}\right) k'_{\mathbf{R_X}} &= k_{\mathbf{R}_{A_i}} - k_{\mathbf{R}_{B_i}} \qquad (1) \\ k_{\mathbf{R_X}} &= \frac{2k'_{\mathbf{R_X}}}{\sqrt{1 + \left|k'_{\mathbf{R_X}}\right|^2}} \end{aligned}$$

where the skew-symmetric matrix

$$\mathrm{Sk}(\mathbf{x}) = \begin{pmatrix} 0 & -\mathbf{x}(3) & \mathbf{x}(2) \\ \mathbf{x}(3) & 0 & -\mathbf{x}(1) \\ -\mathbf{x}(2) & \mathbf{x}(1) & 0 \end{pmatrix},$$

and the angle of rotation θ for $\mathbf{R_X}$ by setting

$$\theta = 2\mathrm{atan}\left|k'_{\mathbf{R_X}}\right|.$$

Another formulation that utilizes the angle-axis formulation was presented by Wang in [24]. They solve the orientational component by considering the properties of the axes of rotation of $\mathbf{R}_{\mathbf{A}_i}$, $\mathbf{R}_{\mathbf{B}_i}$, $\mathbf{R}_{\mathbf{A}_{i+1}}$, and $\mathbf{R}_{\mathbf{B}_{i+1}}$ for $i = 1, 2, \ldots n-1$. Wang compares his method with the Shiu and

Ahmad method [21] and the Tsai and Lenz method [23]. He concludes that of the three methods, the Tsai and Lenz method is the best on average.

The angle-axis methods for calculating the solution of the robot-sensor calibration problem up to this point can be cumbersome. In order to simplify the problem, Park and Martin formed a solution for $\mathbf{R_X}$ by taking advantage of Lie group theory to transform the orientational component into a linear system [17]. Specifically, they take advantage of the property that for a given rotation matrix $\mathbf{R}$

$$\log \mathbf{R} = \frac{\theta}{2\sin\theta}\left(\mathbf{R} - \mathbf{R}^T\right) = \mathrm{Sk}(\mathbf{r}).$$

Here, $\mathbf{r} = \theta k_{\mathbf{R}}$ where θ is the angle of rotation of $\mathbf{R}$ and $k_{\mathbf{R}}$ is the axis of rotation of $\mathbf{R}$. For this paper, $\mathbf{r}$ is the shorthand notation of $\log \mathbf{R}$. Using this formulation,

$$\mathbf{R_{A_i}}\mathbf{R_X} = \mathbf{R_X}\mathbf{R_{B_i}} \quad \Leftrightarrow \quad \mathbf{R_X}\mathbf{a}_i = \mathbf{b}_i$$

where $\mathbf{a}_i$ and $\mathbf{b}_i$ are the shorthand logarithms of $\mathbf{A}_i$ and $\mathbf{B}_i$, respectively. In the presence of noise, Park and Martin calculate the solution of the robot-sensor problem by solving

$$\min_{\mathbf{R_X}} \sum_{i=1}^{n} \|\mathbf{R_X}\ \mathbf{a}_i - \mathbf{b}_i\|^2,$$

whose closed-form solution can be calculated efficiently as

$$\mathbf{R_X} = \mathbf{U}\mathbf{V}^{-1/2}\mathbf{U}^{-1}\mathbf{M}^T$$

where $\mathbf{M} = \sum_{i=1}^{n} \mathbf{b}_i\mathbf{a}_i^T$ and the eigendecomposition of $\mathbf{M}^T\mathbf{M} = \mathbf{U}\mathbf{V}\mathbf{U}^{-1}$.

Chou and Kamel introduce quaternions into the robot-sensor calibration problem in [4, 5]. They notice that the orientational component

$$\mathbf{R_A}\mathbf{R_X} = \mathbf{R_X}\mathbf{R_B} \quad \Leftrightarrow \quad \mathbf{q_A} * \mathbf{q_X} = \mathbf{q_X} * \mathbf{q_B}$$

where $\mathbf{q_X}$ is the quaternion representation of the rotation matrix $\mathbf{R_X}$. Using the matrix form of quaternion multiplication, the orientational component can be restructured into a linear system

$$\begin{aligned} \mathbf{q_A} * \mathbf{q_X} - \mathbf{q_X} * \mathbf{q_B} &= \mathbf{q_A} * \mathbf{q_X} - \overline{\mathbf{q_B}} * \mathbf{q_X} \\ &= (\mathbf{q_A} - \overline{\mathbf{q_B}}) * \mathbf{q_X} = 0 \end{aligned}$$

since

$$\begin{aligned} \mathbf{q_X} * \mathbf{q_B} &= \begin{pmatrix} x_0 & -\mathbf{x}^T \\ \mathbf{x} & (x_0\mathbf{I} + \mathrm{Sk}(\mathbf{x})) \end{pmatrix} \begin{pmatrix} b_0 \\ \mathbf{b} \end{pmatrix} \\ &= \begin{pmatrix} x_0 b_0 - \mathbf{x}^T\mathbf{b} \\ \mathbf{x}b_0 + (x_0\mathbf{I} + \mathrm{Sk}(\mathbf{x}))\,\mathbf{b} \end{pmatrix} \\ &= \begin{pmatrix} b_0 x_0 - \mathbf{b}^T\mathbf{x} \\ \mathbf{b}x_0 + (b_0\mathbf{I} - \mathrm{Sk}(\mathbf{b}))\,\mathbf{x} \end{pmatrix} \\ &= \begin{pmatrix} b_0 & -\mathbf{b}^T \\ \mathbf{b} & (b_0\mathbf{I} - \mathrm{Sk}(\mathbf{b})) \end{pmatrix} \begin{pmatrix} x_0 \\ \mathbf{x} \end{pmatrix} \\ &= \overline{\mathbf{q_B}} * \mathbf{q_X}. \end{aligned}$$

Chou and Kamel solve the linear system using the singular value decomposition.

Horaud and Dornaika form another closed-form solution for $\mathbf{R_X}$ via quaternions in [11]. Specifically, they find that the quaternion representation $\mathbf{q_X}$ for $\mathbf{R_X}$ can be found as the eigenvector associated with the smallest (positive) eigenvalue of

$$\mathcal{A} = \sum_{i=1}^{n} \mathcal{A}_i^T\mathcal{A}_i$$

where

$$\mathcal{A}_i = \begin{pmatrix} 0 & -\mathbf{a}_x^{(i)} + \mathbf{b}_x^{(i)} & -\mathbf{a}_y^{(i)} + \mathbf{b}_y^{(i)} & -\mathbf{a}_z^{(i)} + \mathbf{b}_z^{(i)} \\ \mathbf{a}_x^{(i)} - \mathbf{b}_x^{(i)} & 0 & -\mathbf{a}_z^{(i)} - \mathbf{b}_z^{(i)} & \mathbf{a}_y^{(i)} + \mathbf{b}_y^{(i)} \\ \mathbf{a}_y^{(i)} - \mathbf{b}_y^{(i)} & \mathbf{a}_z^{(i)} + \mathbf{b}_z^{(i)} & 0 & -\mathbf{a}_x^{(i)} - \mathbf{b}_x^{(i)} \\ \mathbf{a}_z^{(i)} - \mathbf{b}_z^{(i)} & -\mathbf{a}_y^{(i)} - \mathbf{b}_y^{(i)} & \mathbf{a}_x^{(i)} + \mathbf{b}_x^{(i)} & 0 \end{pmatrix}$$

and $\mathbf{a}^{(i)} = (\mathbf{a}_x^{(i)}, \mathbf{a}_y^{(i)}, \mathbf{a}_z^{(i)})^T$ is the axis of rotation for $\mathbf{A}_i$ and $\mathbf{b}^{(i)} = (\mathbf{b}_x^{(i)}, \mathbf{b}_y^{(i)}, \mathbf{b}_z^{(i)})^T$ is the axis of rotation for $\mathbf{B}_i$.

Zhuang and Roth also apply quaternions to the robot-sensor calibration problem in [28] to get a closed-form solution that is very similar in formulation to the angle-axis formulation (1) of Tsai and Lenz [23].

Liang et al. apply the Kronecker product to the orientational component of the robot-sensor problem to solve for $\mathbf{R_X}$ in [14]. As a result, the orientational component becomes the linear system

$$\underbrace{\begin{pmatrix} \mathbf{R_{A_1}} \otimes \mathbf{I} - \mathbf{I} \otimes \mathbf{R}_{\mathbf{B}_1}^T \\ \vdots \\ \mathbf{R_{A_n}} \otimes \mathbf{I} - \mathbf{I} \otimes \mathbf{R}_{\mathbf{B}_n}^T \end{pmatrix}}_{\mathbf{L}} \mathrm{vec}\,(\mathbf{R_X}) = 0. \qquad (2)$$

Here the Kronecker product

$$\mathbf{A} \otimes \mathbf{B} = \begin{pmatrix} a_{1,1}B & \cdots & a_{1,n}B \\ \vdots & \ddots & \vdots \\ a_{m,1}B & \cdots & a_{m,n}B \end{pmatrix},$$

where $a_{i,j}$ is the (i,j)-th element of $\mathbf{A}$, and $\mathrm{vec}(\mathbf{A})$ vectorizes a matrix $\mathbf{A}$ column-wise. Liang et al. solve system (2) by

1. Calculating the eigenvector $\mathbf{y}$ corresponding to the smallest eigenvalue of $\mathbf{L}$
2. Forming $\mathbf{Y} = \mathrm{vec}^{-1}(\mathbf{y})$
3. Setting $\mathbf{R_X} = |\mathbf{U}\mathbf{V}^T|$ where the singular value decomposition of $\mathbf{Y} = \mathbf{U}\mathbf{S}\mathbf{V}^T$

Here,

$$|\mathbf{A}| = \begin{cases} \mathbf{A} & \text{if } \det(\mathbf{A}) \geq 0 \\ -\mathbf{A} & \text{if } \det(\mathbf{A}) < 0. \end{cases}$$

For all these separable solutions, errors in the calculation of the optimal rotation $\mathbf{R_X}$ get carried into the calculations of the optimal translation $\mathbf{t_X}$. In order to minimize these errors, simultaneous solutions for $\mathbf{AX} = \mathbf{XB}$ were created. However, these solutions have their own problems as will be discussed.

2.2 Simultaneous Solutions for AX=XB

Chen in [3] believes that separating the orientational component from the positional component, which implies that one has nothing to do with the other, is invalid. Thus, Chen creates a new solution, based on screw theory, that simultaneously solves the orientational component with the positional component. Specifically, he finds that the $\mathbf{AX} = \mathbf{XB}$ problem can be reduced to an absolute orientation problem of finding the best rigid transformation ($\mathbf{R_X}$ and $\mathbf{t_X}$) that transforms the camera screw axis to the robot screw axis.

Daniilidis and Bayro-Corrochano describe an algebraic interpretation of Chen's screw theory method via dual quaternions in [6, 7]. Specifically, they use the vector portions from

the dual-quaternion representations $\mathbf{a}_i + \mathbf{a}'_i$ and $\mathbf{b}_i + \mathbf{b}'_i$ of $\mathbf{A}_i$ and $\mathbf{B}_i$ respectively to create the matrix

$$\mathbf{T} = \begin{pmatrix} \mathbf{S}_1^T & \mathbf{S}_2^T & \dots & \mathbf{S}_n^T \end{pmatrix}^T$$

$$\mathbf{S}_i = \begin{pmatrix} \overrightarrow{\mathbf{a}_i} - \overrightarrow{\mathbf{b}_i} & \mathrm{Sk}\left(\overrightarrow{\mathbf{a}_i} + \overrightarrow{\mathbf{b}_i}\right) & 0 & 0 \\ \overrightarrow{\mathbf{a}'_i} - \overrightarrow{\mathbf{b}'_i} & \mathrm{Sk}\left(\overrightarrow{\mathbf{a}'_i} + \overrightarrow{\mathbf{b}'_i}\right) & \overrightarrow{\mathbf{a}_i} - \overrightarrow{\mathbf{b}_i} & \mathrm{Sk}\left(\overrightarrow{\mathbf{a}_i} + \overrightarrow{\mathbf{b}_i}\right) \end{pmatrix}$$

Using the singular value decomposition on $\mathbf{T}$, Daniilidis and Bayro-Corrochano show that the dual-quaternion representation for the unknown $\mathbf{X}$ can be calculated as a linear combination of the last two right singular vectors of $\mathbf{T}$. It should be noted that the authors developed a similar method through the use of Clifford Algebra in [2]. Zhao and Liu also develop a similar method through the algebraic properties of screw theory in [27].

Lu and Chou [15] apply the quaternions via the eight step method to solve the robot-sensor calibration problem simultaneously. Specifically, by the use of quaternions, they can simplify the problem to a single linear system which they solve using Gaussian elimination and Schur decomposition.

Andreff et al. are the first to apply the Kronecker product to simultaneously solve the robot-sensor problem in [1]. They reformulate the robot-sensor problem into a linear system of the form

$$\begin{pmatrix} \mathbf{I} - \mathbf{R}_{\mathbf{B}_i} \otimes \mathbf{R}_{\mathbf{A}_i} & 0 \\ \mathbf{t}_{\mathbf{B}_i}^T \otimes \mathbf{I} & \mathbf{I} - \mathbf{R}_{\mathbf{A}_i} \end{pmatrix} \begin{pmatrix} \mathrm{vec}(\mathbf{R}_{\mathbf{X}}) \\ \mathbf{t}_{\mathbf{X}} \end{pmatrix} = \begin{pmatrix} 0 \\ \mathbf{t}_{\mathbf{A}_i} \end{pmatrix}.$$

Andreff et al. prove that at least two independent general motions with non-parallel axes are needed to have a unique solution to the linear system. A problem with this method is that due to noise the solution for $\mathbf{R}_{\mathbf{X}}$ may not necessarily be an orthogonal matrix. Thus, an orthogonalization step for the orientational component has to be taken. However, the corresponding positional component is not recalculated, which causes errors in the solution. Therefore, Andreff et al. suggest separating the orientational and positional components as was shown in the work of Liang et al. (see Section 2.1) in [14].

2.3 Iterative Solutions for AX=XB

Simultaneous solutions were developed to solve the problem of orientational errors propagating into the positional errors. Another option to solve this problem is to create an iterative solution for $\mathbf{AX} = \mathbf{XB}$. Zhuang and Shiu propose a one-step iterative method, based on minimizing $\|\mathbf{AX}-\mathbf{XB}\|$ with the Levenberg-Marquardt algorithm in [30]. The iterative method solves both the orientational and positional components simultaneously. Furthermore, the method is not dependent on robot orientation $\mathbf{R}_{\mathbf{B}_i}$ information. Fassi and Legnani propose a similar algorithm in [9]. This paper also provides a geometric interpretation of the hand-eye calibration problem. Wei et al. [25] create an efficient iterative method that is optimized by the sparse structure of the corresponding normal equations.

Horaud and Dornaika in [11] also propose to solve the orientational and positional components simultaneously using an iterative method. However, their method is based on using the quaternion representation for the orientational component.

Mao et al. [16] apply the Kronecker product in their iterative formulation. An issue with the Mao et al. optimization problem is that the solution is based on the initial condition. Therefore, different initial conditions could result in varying solutions. A remedy to this problem is to use convex optimization as shown in the work of Zhao [26]. Zhao claims that his Kronecker product algorithm is very fast and not dependent on an initial condition. However, their setup gives no guarantee that the orientational component $\mathbf{R}_{\mathbf{X}}$ of the solution is a rotation matrix. Therefore, his algorithm may cause errors that are similar to the errors of Andreff et al. [1]. Shi et al. [20] have a similar formulation to Zhao (thus similar problems), but their iterative algorithm optimizes motion selection to improve accuracy and to avoid degenerate cases.

Strobl and Hirzinger create an iterative method that is based on a parameterization of a stochastic model in [22]. This iterative method is novel since it creates an inherent algorithm to weight the orientational and positional components to optimize the accuracy of the method. Kim et al. extend this formulation in [12] with the use of the Minimum Variance method.

These iterative methods get rid of the propagation of orientational errors into the positional component. However, solving the robot-sensor calibration method in this manner can be computationally taxing since these methods often contain complex optimization routines. In addition, as the number of equations (n) gets larger, the differences between iterative solutions and closed-form solutions often get smaller. Thus, one has to decide whether the accuracy of an iterative solution is worth the computational costs.

3. AX=YB SOLUTIONS

In this section we will give an overview of techniques to solve $\mathbf{AX} = \mathbf{YB}$. The methods for solving this system are very similar to the $\mathbf{AX} = \mathbf{XB}$ problems, i.e., the methods can be organized into three groups: separable solutions, simultaneous solutions, and iterative solutions.

Wang proposes the $\mathbf{AX} = \mathbf{YB}$ problem in [24], though he assumes that one of the unknowns is given. Zhuang et al. were the first to give a separable closed-form solution via quaternions in [29]. Dornaika and Horaud extend Zhuang et al.'s separable solution to give a more accurate separable closed-form solution via quaternions in [8]. Shah creates a formulation based on Kronecker product in [19].

Li et al. look at simultaneous closed-form solutions via dual-quaternions and Kronecker products in [13]. Their formulations follow the methodology of the $\mathbf{AX} = \mathbf{XB}$ formulation of dual quaternions of Daniildis [7] and the formulation of Kronecker product of Andreff et al. [1].

Iterative solutions for the $\mathbf{AX} = \mathbf{YB}$ problem were first introduced in the work of Remy et al. [18]. Here they define a nonlinear optimization problem and use the Levenberg-Marquardt method to solve it. Hirsh et al. develop an iterative method in [10] that optimizes the orientational and positional components separately, while Strobl and Hirzinger create an iterative method [22] that simultaneously solves the orientational and positional components. Their method is based on a parameterization of a stochastic model which is identical to their $\mathbf{AX} = \mathbf{XB}$ model. Kim et al. also use a model [12] identical to their $\mathbf{AX} = \mathbf{XB}$ model to simultaneously solve $\mathbf{AX} = \mathbf{YB}$ using the Minimum Variance method.

4. CONCLUSION

In this paper, we give an overview of methods to solve the

robot-sensor calibration problem of the forms $\mathbf{AX} = \mathbf{XB}$ and $\mathbf{AX} = \mathbf{YB}$ for the evaluation of perception systems. Each form's solutions can be split into three categories: separable solutions, simultaneous solutions, and iterative solutions. The separable solutions are simple and fast solutions; however, errors calculated from the orientational component get carried over to the positional component. As a result, simultaneous solutions were developed. However, these solutions produce variable results depending on the scaling of the positional component. To weight the orientational and positional components, iterative methods were created. However, though these solutions are often more accurate, the solutions are often complex and generally depend on starting criteria. In addition, there is generally no guarantee that the convergent solution is the optimal solution. Thus, users must decide which type of method to use for evaluation which is dependent on their desired accuracy and complexity.

5. REFERENCES

[1] N. Andreff, R. Horaud, B. Espiau On-line Hand-Eye Calibration. In *Second International Conference on 3-D Digital Imaging and Modeling (3DIM'99)*, pages 430–436, 1999.

[2] E. Bayro-Corrochano, K. Daniilidis, G. Sommer Motor Algebra for 3D Kinematics: The Case of the Hand-Eye Calibration. In *Journal of Mathematical Imaging and Vision*, 13: 79–100, 2000.

[3] H. H. Chen A screw motion approach to uniqueness analysis of head-eye geometry. In *IEEE Proceedings of Computer Vision and Pattern Recognition (CVPR'91)*, pages 145–151, 1991.

[4] J. C. K. Chou, M. Kamel Quaternions approach to solve the kinematic equation of rotation, $A_aA_x = A_xA_b$, of a sensor-mounted robotic manipulator. In *IEEE International Conference on Robotics and Automation*, pages 656–662, 1988.

[5] J. C. K. Chou, M. Kamel Finding the Position and Orientation of a Sensor on a Robot Manipulator Using Quaternions. In *The International Journal of Robotics Research*, 10(3): 240–254, 1991.

[6] K. Daniilidis, E. Bayro-Corrochano The dual quaternion approach to hand-eye calibration. In *Proceedings of the 13th International Conference on Pattern Recognition*, pages 318–322, 1996.

[7] K. Daniilidis Hand-Eye Calibration Using Dual Quaternions. In *The International Journal of Robotics Research*,18(3): 286 – 298, 1999.

[8] F. Dornaika, R. Horaud, Simultaneous robot-world and hand-eye calibration, IEEE Transactions on Robotics and Automation, 14(4): 617 – 622, 1998.

[9] I. Fassi, G. Legnani Hand to Sensor Calibration: A Geometrical Interpretation of the Matrix Equation AX=XB. In *Journal of Robotic Systems*, 22(9): 497 – 506, 2005.

[10] R. L. Hirsh, G. N. DeSouza, A. C. Kak An Iterative Approach to the Hand-Eye and Base-World Calibration Problem. In *Proceedings of the 2001 IEEE International Conference on Robotics and Automation*, 3: 2171 – 2176, 2001.

[11] R. Horaud, F. Dornaika Hand-Eye Calibration In *International Journal of Robotics Research*, 14(3): 195 –210, 1995.

[12] S. Kim, M. Jeong, J. Lee, J. Lee, K. Kim, B. You, S. Oh Robot Head-Eye Calibration Using the Minimum Variance Method. In *Proceedings of the 2010 IEEE International Conference on Robotics and Biomimetics*, pages 1446 – 1451, 2010.

[13] A. Li, L. Wang, D. Wu Simultaneous robot-world and hand-eye calibration using dual-quaternions and kronecker product. In*International Journal of the Physical Sciences*, 5(10): 1530 –1536, 1995.

[14] R. Liang, J. Mao Hand-Eye Calibration with a New Linear Decomposition Algorithm. In *Journal of Zhejiang University*, 9(10):1363–1368, 2008.

[15] Y. Lu, J. C. K. Chou Eight-Space Quaternion Approach for Robotic Hand-eye Calibration. In *Systems, IEEE International Conference on Man and Cybernetics*, 4: 3316 – 3321, 1995.

[16] J. Mao, X. Huang, L. Jiang A Flexible Solution to AX=XB for Robot Hand-Eye Calibration. In *Proceedings of the 10th WSEAS International Conference on Robotics, Control and Manufacturing Technology*, pages 118–122, 2010.

[17] F. Park, B. Martin Robot Sensor Calibration: Solving AX = XB on the Euclidean Group. In *IEEE Transactions on Robotics and Automation*, 10(5): 717–721, 1994.

[18] S. Remy, M. Dhome, J. M. Lavest, N. Daucher Hand-eye Calibration. In *Proceedings of the 1997 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2:1057 – 1065, 1997.

[19] M. I. Shah. Solving the Robot-World/Hand-Eye Calibration Problem using the Kronecker Product. Technical report, Department of Mathematics and Statistics, Loyola University in Maryland, 2012.

[20] F. Shi, J. Wang, Y. Liu An Approach to Improve Online Hand-Eye Calibration. In *Pattern Recognition and Image Analysis*, 3522: 539 – 567, 2005.

[21] Y. Shiu, S. Ahmad Calibration of Wrist-Mounted Robotic Sensors by Solving Homogeneous Transform Equations of the Form AX = XB. In *IEEE Transactions on Robotics and Automation*, 5(1):16–29, 1989.

[22] K. H. Strobl, G. Hirzinger Optimal Hand-Eye Calibration. In *Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4647 – 4653, 2006.

[23] R. Tsai, R. Lenz A New Technique for Fully Autonomous and Efficient 3D Robotics Hand/Eye Calibration. In *IEEE Transactions on Robotics and Automation*, 5(3):345-358, 1989.

[24] C. Wang Extrinsic Calibration of a Vision Sensor Mounted on a Robot. In *IEEE Transactions on Robotics and Automation*, 8(2): 161–175, 1992.

[25] G. Wei, K. Arbter, G. Hirzinger Active self-calibration of robotic eyes and hand-eye relationships with model identification. In *IEEE Transactions on Robotics and Automation*, 14(1): 158 – 166, 1998.

[26] Z. Zhao Hand-Eye Calibration Using Convex Optimization. In *2011 IEEE International Conference on Robotics and Automation*, pages 2947–2952, 2011.

[27] Z. Zhao, Y. Liu Hand-Eye Calibration Based on Screw

Motions. In *18th International Conference on Pattern Recognition(ICPR'06)*, pages 1022 – 1026, 2006.

[28] H. Zhuang, Z. S. Roth Comments on "Calibration of Wrist-Mounted Robotic Sensors by Solving Homogeneous Transform Equations of the Form AX = XB." In *IEEE Transactions on Robotics and Automation*, 7(6): 877–878, 1991.

[29] H. Zhuang, Z. S. Roth, R. Sudhakar Simultaneous Robot/World and Tool/Flange Calibration by Solving Homogeneous Transformation Equations of the form AX=YB. In *IEEE Transactions on Robotics and Automation*, 10(4):549 – 554, 1994.

[30] H. Zhuang, Y . C. Shiu A Noise-Tolerant Algorithm for Robotic Hand-Eye Calibration with or without Sensor Orientation Measurement. In *IEEE Transactions on Systems, Man, and Cybernetics*, 23(4): 1168 – 1175, 1993.

On the Performance Evaluation of a Vision-based Human-Robot Interaction Framework

Junaed Sattar
University of British Columbia
Vancouver, BC, Canada V6T 1Z4
junaed@cs.ubc.ca

Gregory Dudek
McGill University
Montreal, QC, Canada H3A 0E9
dudek@cim.mcgill.ca

ABSTRACT

This paper describes the performance evaluation of a machine vision-based human-robot interaction framework, particularly those involving human-interface studies. We describe a visual programming language called RoboChat, and a complimentary dialog engine which evaluates the need for confirmation based on utility and risk. Together, RoboChat and the dialog mechanism enable a human operator to send a series of complex instructions to a robot, with the assurance of confirmations in case of high task-cost or command uncertainty, or both. We have performed extensive human-interface studies to evaluate the usability of this framework, both in controlled laboratory conditions and in a variety of outdoors environments. One specific goal for the RoboChat scheme was to aid a scuba diver to operate and program an underwater robot in a variety of deployment scenarios, and the real-world validations were thus performed on-board the Aqua amphibious robot [4], in both underwater and terrestrial environments. The paper describes the details of the visual human-robot interaction framework, with an emphasis on the RoboChat language and the confirmation system, and presents a summary of the set of performance evaluation experiments performed both on- and off-board the Aqua vehicle.

1. INTRODUCTION

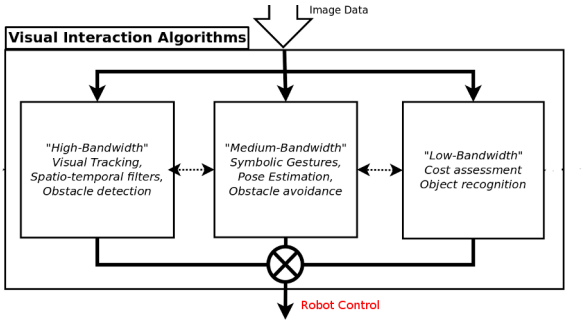
With the rapidly increasing adoption of robotic technologies in society, human-robot interaction frameworks and schemes are becoming more and more ubiquitous. "Traditional" interface devices and paradigms in computing and robotics are being replaced by more intuitive means of interaction, with "intuition" being broadly applied in the human interaction context. Speech, vision, tactile sensing etc. are but a few examples of such novel classes of interaction modalities. Our research explores the use of machine vision as a modality for human-machine interaction, building on the intuitive nature of visual gestures as a means of communication. In a wider scale, our vision-interaction framework also includes algorithms for person detection and tracking (in the underwater domain) and a learning-based tracker to robustly track objects with spatially complex color distributions. Algorithms of this nature, while directly not communicating with the human operator, assist mobile robots to exhibit "human-aware" behaviors, and are we label them as *implicit interaction* algorithms. The focus of this paper, however, will be on the more *explicit interaction* algorithms for human-robot interaction. Specifically, we look at a visual programming language for mobile robot programming called *RoboChat* [5] and a dialog management algorithm that evaluates the need for interaction based on risk and uncertainty [20]. We focus on the algorithms and the experimental validations performed to evaluate the usability of these techniques. The experiments were designed to assess usability through timing and accuracy measurements across a variety of task scenarios, and also included qualitative feedback from users as they operated the Aqua underwater robot [4] using these methods. However, we limit our discussion to the quantitative user studies for this paper, and briefly discuss the various issues and experimental outcomes of the robot field trials.

Validation of our research has focused on the usability of the individual algorithms, and the HRI framework as a whole, as a collection of disparate algorithms. Quantitatively measuring performance of the individual components are mostly straightforward, as the experimental setups can be arranged off-board in a laboratory setting, under controlled environments. Under such circumstances, measurements can be obtained minimizing error and experiments can be repeated arbitrary number of times (except for those involving human participants). In field trials, specially those involving robots operating in challenging environments, as is the case of our underwater vehicle, such measurements can be exceedingly difficult. Several aspects contribute to this hardship, including but not limited to the constraints of human endurance, finding participants with required skill levels (which often are beyond those required by the norms of operational certification, such as those held by scuba divers), requirements of special equipments – both experimental and for measurements, inability to reproduce experimental conditions and the resulting lack of repeatability of obtained results, and cognitive loading of participants, often resulting in incomplete and/or biased measurements. Keeping such issues in mind, we turn our attention to reporting measurements of coarser scale – number of successful trials, application of novel commands or behaviors, distances traveled, and confirmations requested are to name a few of these metrics. Finer levels of quantitative measurements are reserved for off-board experiments held in controlled settings.

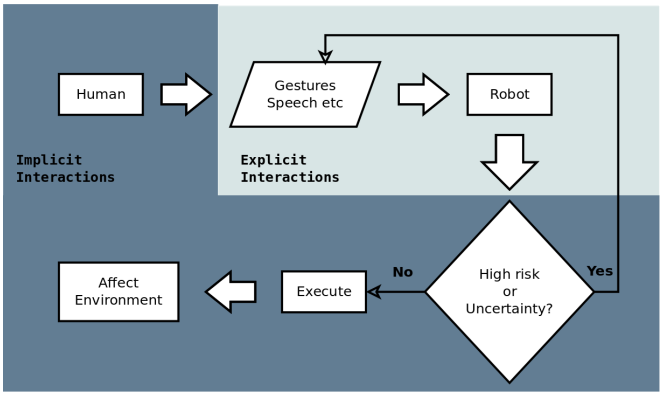
2. RELATED WORK

This paper presents, along with quantitative evaluations, of a human-robot interaction system that uses vision algorithms to communicate with and detect the presence of human operators (and other humans in the surrounding). Along with machine vision, this work spans the domains of gesture recognition, robot control, dialog management, and robot software architecture. In this Section,

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage, and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
PerMIS'12, March 20-22, 2012, College Park, MD, USA.
Copyright © 2012 ACM 978-1-4503-1126-7-3/22/12 ...$10.00.



(a) Classification of algorithms in the described three-layer visual HRI framework.



(b) Explicit versus implicit interaction in the framework depicted in Fig. 1(a).

Figure 1: Core concepts of a vision-based HRI framework.

we present, albeit briefly, a summary of related previous work in these domains.

Our previous work looked at using visual communications, and specifically visual servo-control with respect to a human operator, to handle the navigation of an underwater robot [21]. In that work, the robot is able to follow a scuba diver, or any arbitrary target, to maneuver, but the diver accompanying the robot can only modulate the robot's activities by making hand signals that are interpreted by a second human operator sitting on a tethered robot control unit. Visual communication has also been used by several authors to allow communication between systems, for example in the work of Dunbabin *et al*. [6]

The work of Waldherr, Romero and Thrun [24] exemplifies the explicit communication paradigm in which hand gestures are used to interact with a robot and lead it through an environment. Tsotsos *et al*. [23] considered a gestural interface for non-expert users, in particular disabled children, based on a combination of stereo vision and keyboard-like input. As an example of implicit communication, Rybski and Voyles [16] developed a system whereby a robot could observe a human performing a task and learn about the environment. Such class of "learning-by-demonstration" tasks are part of a richly growing field, and are particularly attractive to human-robot interaction problems, where robot-human coexistence and coordination are of utmost importance.

Fiducial marker systems, as mentioned in the previous section, are efficiently and robustly detectable under difficult conditions. Apart from the ARTag toolkit mentioned previously, other fiducial marker systems have been developed for use in a variety of applications. The ARToolkit marker system [15] consists of symbols very similar to the ARTag flavor in that they contain different patterns enclosed within a square black border. The April Tag class of fiducials [**?**] also rely on square black-and-white markers, and have been used in vision-guided robotic tasks. Circular markers are also possible in fiducial schemes, as demonstrated by the Fourier Tags [17] fiducial system.

Gesture-based robot control has been considered extensively in Human-Robot Interaction (HRI). This includes explicit as well as implicit communication frameworks between human operators and robotics systems. Several authors have considered specialized gestural behaviors [9] or strokes on a touch screen to control basic robot navigation. Skubic *et al.* have examined the combination of several types of human interface components, with special emphasis on speech, to express spatial relationships and spatial navigation tasks [22].

Vision-based gesture recognition has long been considered for a variety of tasks, and has proven to be a challenging problem examined for over 20 years with diverse well-established applications [7][14]. The types of gestural vocabularies range from extremely simple actions, like simple fist versus open hand, to very complex languages, such as the American Sign Language (ASL). ASL allows for the expression of substantial affect and individual variation, making it exceedingly difficult to deal with in its complete form. For example, Tsotsos *et al.* [1] considered the interpretation of elementary ASL primitives (*i.e.*, simple component motions) and achieved 86 to 97 *per cent* recognition rates under controlled conditions. While such rates are good, they are disturbingly low for open-loop robot-control purposes.

While our current work looks at interaction under uncertainty in any input modality, researchers have investigated uncertainty modeling in human-robot communication with specific input methods. For example, Pateras *et al*. applied fuzzy logic to reduce uncertainty to reduce high-level task descriptions into robot sensor-specific commands in a spoken-dialog HRI model [13]. Montemerlo *et al*. have investigated risk functions for safer navigation and environmental sampling for the Nursebot robotic nurse in the care of the elderly [12]. Bayesian risk estimates and active learning in POMDP formulations in a limited-interaction dialog model [2] and spoken language interaction models [3] have also been investigated in the past. Researchers have also applied planning cost models for efficient human-robot interaction tasks [10] [11].

3. A FRAMEWORK FOR VISUAL HRI

In this work, an algorithm that enables a mobile robot to interact with a human, both through explicit and implicit communications, is labeled as an *Interaction Algorithm*. Algorithms belonging to the class of explicit interactions require an operator to give instructions to a mobile robot directly, for example through gestures or some direct input method. By using implicit interaction algorithms, a mobile robot can execute commands given by explicit instructions, particularly those that enable it to accompany the operator and assess task safety (from both the human and robot's perspective). Both classes of algorithms can further be categorized in a three-layer architecture, according to how frequently the functions are invoked by the robot. This three layer breakdown is demonstrated in Fig. 1(a), while a categorization of explicit versus implicit algorithms can be seen in Fig. 1(b). As we go from left-to-right in

Fig. 1(a), the rates of invocation for the algorithms in each box decreases; consequentially, the computation costs increase along the same directions, highlighting a natural inverse relationship between convocation rate and computational complexity.

The current implementation includes four major algorithmic components that facilitate vision-based human-robot interaction. These components are enumerated below:

1. A human-robot dialog model that evaluates the need for interaction based on utility and risk [20].
2. A visual language for programming robotic systems using gestures, and the human-interface studies towards quantifying its performance [5].
3. A visual biometric system for detecting and tracking multiple scuba divers in the underwater domain [19].
4. A machine learning algorithm to learn spatial color distribution of objects to achieve robust tracking under variable lighting and color distortion [18].

A comprehensive treatment of the framework is beyond the scope of this paper; instead, we limit the discussion on the core principles of the first two components, and present experimental setups and validation results.

3.1 Visual Programming

To visually program a robot, we use a set of engineered markers, called fiducials, to form simple geometric gestures that are interpreted by the robot as input commands. The underlying language, called *RoboChat*, enables the user to program the robot to carry out a large variety of tasks, both simple and complex in nature. RoboChat has a core set of basic tokens, including numerical digits, arithmetic operators, and relational operators. Additionally, RoboChat defines a limited number of variables, including command parameters, as well as some general-purpose variable names. RoboChat features two control flow constructs – the if-else statement, and the indexed iterator statement. The former construct allows the user to implement decision logic, while the latter immensely cuts down on the required number of tokens for repeated commands. The user can encapsulate a list of expressions into a numerically tagged macro, which can then be called upon later. This feature allows the reuse of code, which is essential when trying to minimize the number of tokens needed to specify behavior. Every construct is designed to minimize the number of tokens needed to express that construct. Reverse Polish notation (RPN) is heavily exploited to achieve this minimization – operators and operands are presented using RPN, eliminating the need for both an assignment operator and an end-of-command marker, while still allowing one-pass "compilation". Additionally, the use of RPN notation in the more abstract control flow constructs eliminate the need of various delimiters common to most programming languages. RoboChat interprets the tokens in real time, but only executes the commands upon detection of the `EXECUTE` token. This feature allows for batch processing, and also enables the error recovery element, using the `RESET` token.

As evident from the previous paragraph, the language is well-formed, governed by a strict grammar. This ability to instruct the robot with the aid of visual markers provides a first-order method for human-robot communication. By using fiducials, we also obviate the need for an error-free, robust gesture recognition algorithm. To compensate for errors in programming, RoboChat provides built-in syntax checking, and further error checking and uncertainty reduction is provided by a risk assessment engine, as described below.

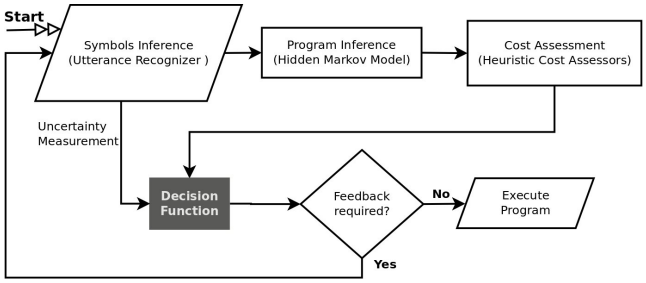


Figure 2: System flowchart for the confirmation dialog system.

3.2 Risk Assessment

In almost all real-world human-robot interfaces, there remains non-trivial uncertainty in input. If not accounted for in a robust manner, such uncertainty could lead to unsafe and potentially hazardous consequences for the robot and also cause harm to the operating environment. To minimize risk in the presence of uncertainty, we use a *Decision Function*, which takes into account belief states over a set of likely inputs, and the corresponding task execution costs. By using a Hidden Markov Model for belief tracking, and assessment of task costs through task simulation, the decision function requests confirmation of the high-risk input commands. That is, expensive tasks are executed if and only if they are truly requested by the user. An outline of the algorithmic flow for our system can be seen in Fig. 2.

4. USER STUDIES

We present a summary of user interface results for both RoboChat and the dialog system in this section. A substantial number of user interface studies were performed to evaluate the usability of these schemes, and an implementation of these systems are currently deployed on-board the Aqua family of underwater robots. The usability studies were performed across a wide range of users, and a number of representative tasks for our particular vehicle and its operating domains.

4.1 Experiments with RoboChat

We performed two sets of studies using the proposed marker-based input scheme in combination with the RoboChat language, to assess their usability. In both studies, the ARTag mechanism is compared to a hand gestures system, as competing input devices, particularly for environments unsuitable for the use of conventional input interfaces. The first study investigated the performance of the two systems with the user under significant stress, similar to the one scuba divers must face underwater. The second study compared the two input mechanisms in the presence of different vocabulary sizes. The main task in both studies is to input a sequence of action commands, with the possibility of specifying additional parameters, as accurately and efficiently as possible. The RoboChat format is used with both input devices, although in the case of the hand signal system, the gestures are interpreted by an expert human operator remotely, who subsequently validates the correctness of the input using the RoboChat syntax. This setup is realistic because in the case of our particular application, the diver's hand signals are interpreted by an operator on land, who then takes control of the robot. Also, the operator is not forced to be unbiased when interpreting gestures, because realistically the robot operator will guess and infer at what the diver is trying to communicate, if the hand gestures are ambiguously perceived.

1	TURN RIGHT, REVERSE, EXECUTE
2	FORWARD, TURN LEFT, FORWARD, EXECUTE
3	REVERSE, TURN RIGHT, FORWARD, REVERSE, EXECUTE
4	REVERSE, TURN RIGHT, REVERSE, TURN LEFT, REVERSE, EXECUTE
5	TURN RIGHT, SURFACE, TURN LEFT, SURFACE, REVERSE, TURN LEFT, STOP, EXECUTE
6	STOP, FORWARD, SURFACE, TURN RIGHT, SURFACE, EXECUTE
7	REVERSE, STOP, SURFACE, FORWARD, STOP, SURFACE, TURN RIGHT, EXECUTE
8	FORWARD, STOP, FORWARD, SURFACE, STOP, EXECUTE
9	TURN RIGHT, TURN LEFT, SURFACE, EXECUTE
10	FORWARD, REVERSE, FORWARD, STOP, EXECUTE
11	FORWARD, REVERSE, TURN RIGHT, EXECUTE

Table 1: Tasks used in Study A.

4.1.1 *Study A*

In the first study, the ARTag markers are provided to the participants, and they are allowed to place them in any configuration in the provided work area, particularly in a manner so that the tags can be easily accessible. The hand gestures in this study are predetermined, and are visually demonstrated to the participants, who are then asked to remember all the gestures. During the experiment session, the participants must rely on memory alone to recall the gestures, much like the case for the scuba divers.

The stress factor in the first study is introduced by asking participants to play a game of Pong (a classical 1970's table tennis video game [8]) during the experimental sessions. A suitable distractor task must be fairly accessible to all users, continually demanding of attention, yet still allow the core task to be achievable. Pong was decided to be closely fulfilling such requirements, and was chosen as a distractor task. This particular implementation of Pong uses the mouse to control the user's paddle. As such, participants are effectively limited to using only one hand to manipulate the markers and to make out gestures, while constantly controlling the mouse with the other hand. But since some of the predefined hand gestures require the use of both hands, this distraction introduces additional stress for the participants in terms of the alternatively showing gestures and playing Pong. Also, the experiment rules make it mandatory to inform the participant when the entered command is incorrect, and proceed onto the next command only after receiving the previous one correctly.

4.1.2 *Study B*

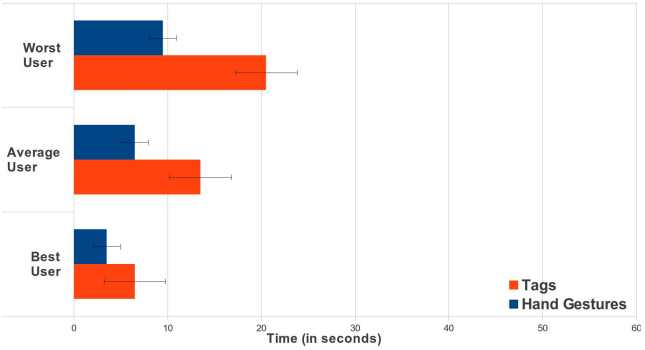
For the second study, the parameter of interest is the performance difference using different vocabulary sizes. Two vocabulary sets are used in this study – the first set contains only 4 action commands, while the second includes 32. This distinction is mentioned to every participant so that they can use this information to their advantage. As it is unrealistic to ask participants to remember more than 50 different hand gestures under the experiment's tight time constraints, a gesture lookup sheet is given to each participant. The subjects are encouraged to familiarize themselves with this cheat sheet during the practice sessions, to ensure that they spend minimal time searching for particular hand signals. The ARTag markers are also provided in the form of 'flip-books' to facilitate fast lookup and easy access. There is no distraction factor in this second study, but at the same time, the system accepts incorrect commands without informing the participants or making them re-enter the commands. The users are informed of this criterion, and are recommended to constantly keep track of the entered tokens and try to make as few mistakes as possible.

FORWARD, TURN LEFT, FORWARD, TURN RIGHT, REVERSE, STOP, FORWARD, SURFACE, TURN RIGHT, SURFACE, STOP, FORWARD, DEPTH, 10, TURN RIGHT, FORWARD, STOP, TAKE PICTURE, SURFACE, GPSFIX, DEPTH, 15, FORWARD, TURN RIGHT, FORWARD, TAKE PICTURE, 10, TURN LEFT, FORWARD, SURFACE, STOP, EXECUTE

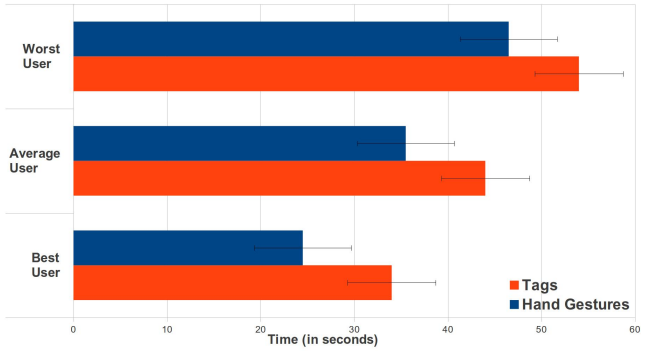
Table 2: Example of a long command used in Study B.

4.1.3 *Criteria*

Two criteria are used to compare the performance of the two input interfaces. The first criterion is speed; *i.e.*, the average speed it takes to enter a command. A distinction is made between the two studies regarding this metric: in the first study, the input time per command is measured from the time a command is shown on screen until the time the command is *correctly* entered by the participant, whereas in the second study, the command speed does not



(a) Study A: Average time taken per command using ARTag markers (in red) and using hand gestures (in dark blue).



(b) Study B: Average time taken per command using ARTag markers (in red) and using hand gestures (in dark blue).

Figure 3: Timing data for programs: Hand gestures vs RoboChat.

take into consideration the correctness of the command. The second study also uses the average time per individual token as a comparison metric. This metric demonstrates the raw access speeds of both input interfaces outside the context of RoboChat or any other specific environment.

The second criterion used to compare the two systems is the error rate associated to each input scheme. Once again, due to the distinction between how incorrect commands are treated between the two studies, results from this metric cannot be compared directly between studies. This criterion is used to look at whether the two schemes affect the user's performance in working with RoboChat differently.

In total, 12 subjects participated in study A, whereas 4 subjects participated in study B. One of the participants present in both studies has extensive experience with ARTag markers, RoboChat, and the hand gesture system. This expert user is introduced in the dataset to demonstrate the performance of a well-trained user. However, this user has no prior knowledge of the actual experiments, therefore is capable of exhibiting similar performance improvements throughout the sessions.

4.1.4 Results: Study A

One obvious observation we can make from the performance data is that the gesture system allows for faster communication than the marker system. The ratio between the two input techniques for some users surpasses 3:1 favoring hand gestures, while data from other users (including those from the expert user) show ratios of lower than 2:1. Since all users have experience with primitive hand gestures, we can infer that it may simply be that those users who did almost equally well with markers as gestures adapted to the marker system more quickly. Thus, the data suggest that the ARTag markers are capable of matching half the speed of the hand gestures, even given only limited practice. It is worth noting that contrary to the hand gestures which are chosen to have intuitive and natural mappings to their corresponding tokens, the mappings between the ARTag markers and tokens are completely arbitrary.

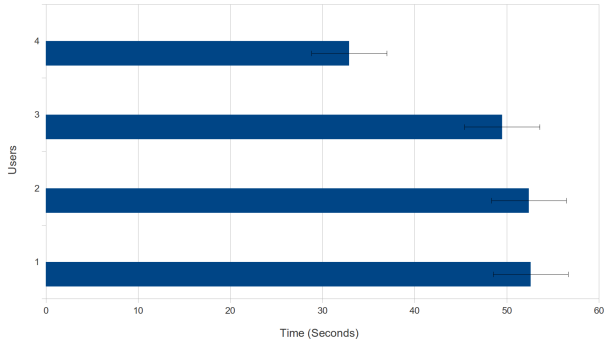
To further substantiate the hypothesis that the enhanced performance of hand gestures is due to familiarity, note that Fig. 3 indicates that the spread of the average time per command using gestures ($\pm$ 3 seconds) is much smaller than that for markers ($\pm$ 8 seconds). Arguably the more sporadic spread for the markers is due to unfamiliarity with this new input interface.

The distraction task (playing Pong) also plays an important role in increasing the performance disparity between the two systems. For each token, the participants need to search through the entire ARTag vocabulary set for the correct marker, whereas the associated hand gesture can be much easily recalled from memory. Since the Pong game requires the participant's attention on an ongoing basis, the symbol search process was repeatedly disrupted by the distraction task, amplifying the marker search time.

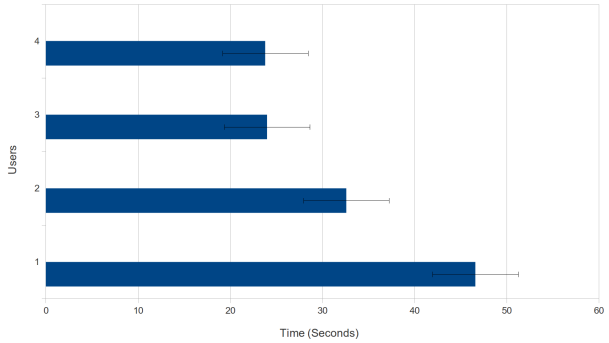
In terms of the error rate associated with each system, all the participants displayed error rates of roughly 5 *per cent* for both systems. This finding is surprising and interesting, because even though the symbolic system is harder to learn, it does not seem to generate more errors than the gesture system, even for inexperienced users.

4.1.5 Results: Study B

The data from study B suggests that the two input interfaces have very similar performances under the new constraints. Major contributing factors include the increase in the vocabulary size and the inclusion of many abstract action tokens (such as `RECORD_VIDEO` and `POWER_CYCLE`). This variation takes away the crucial advantage gestures had in the former study, and participants are now forced to search through the gesture sheet rather than remembering the many hand gestures. Essentially, in this study, the command speed criterion boils down to the search speed for each input device, and therefore depends on the reference structure, whether it is the ARTag flipbook or the gesture cheat sheet. And using the two engineered reference structures, the data of the experiments show that the speed performance of both input systems are actually very similar. Interestingly enough, the data spread between systems are actually reversed, as shown in Fig. 4. With the exception of the expert user, the average command and token speeds for all the participants using ARTag markers are almost identical, whereas the same speeds using gestures are now erratic between individuals. This result can be attributed to the fact that since the gestures are not kept in memory, different subjects adapt to the cheat sheet setup at different speeds.



(a) Study B: Average time taken per command across users using ARTag markers.



(b) Study B: Average time taken per command across users using hand gestures.

Figure 4: Study B: Average time taken per command using ARTag markers and hand gestures. In both plots, user 4 is the "expert user".

4.2 Experiments with the Dialog System

We performed a set of user studies to collect quantitative performance measures of our algorithm. When operating as a diver's assistant in underwater environments, the system uses fiducials to engage in a dialog with the robot. However, in the off-board bench trials, we employed a simplified "gesture-only language", where the users were limited to using mouse input. We used a vocabulary set of 18 tokens defined by oriented mouse gestures, and as such

each segment is bounded by a 20^o-wide arc. The choice for using mouse gestures stemmed from the need to introduce uncertainty in the input modality, while keeping the cognitive load roughly comparable to that experienced by scuba divers. We could not use the ARTag scheme for these experiments, as the ARTag library neither provides a confidence factor for tag detection, nor does it have a significant false positive detection rates. Using ARTags would have provided insufficient data to throughly validate the algorithm for an arbitrary input modality.

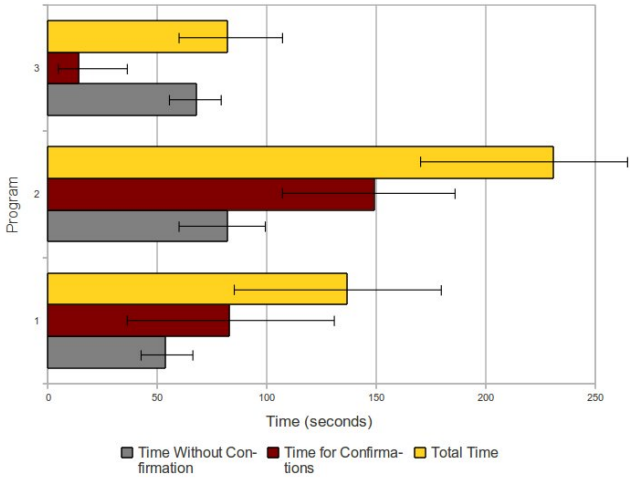
4.2.1 Experimental Setup

To calculate uncertainty in input, we trained a Hidden Markov Model using commonly used programs given to the robot (such as those used in previous experiments and field trials; *i.e.*, real operational data). To estimate task costs, we simulated the programs using a custom-built simulation engine and used a set of assessors that takes into account the operating context of an autonomous underwater vehicle. The simulator has been designed to take into account the robot's velocity, maneuverability and propulsion characteristics to accurately and realistically simulate trajectories taken by the robot while executing commands such as those used in our experiments. In choosing assessors for the user studies, we considered factors that directly affect underwater robot operations. For example, the distance traveled by the robot (and the farthest distance it travels from the start point) often has a direct bearing on the outcome of the mission, as the probability of robot recovery is inversely proportional to these factors. That is because energy consumption is directly proportional to the distance traveled. Robot safety (*e.g.*, chance of collisions) is also significantly compromised by traveling large distances. In particular, we applied four assessors during the user studies, which assessed total distance, furthest distance, execution time and average distance traveled.

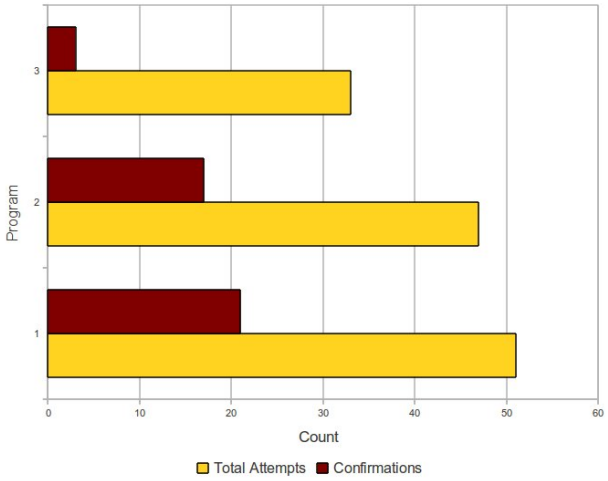
Each user was given three programs to send to the system, and each program was performed three times. A total of 10 users participated in the trials, resulting in 30 trials for each program, and 90 programs in all; Except for mistakes that created inconsistent programs, users did not receive any feedback about the correctness of their program. When a user finished writing a program, she either received feedback notifying her of program completion, or a confirmation dialog was generated based on the output of the Decision Function. The users were informed beforehand about the estimated cost of the program; *i.e.*, whether to expect to receive a feedback or not. In case of a confirmation request for Programs 1 and 3, the users were instructed to redo the program. For Program 2, the users were informed of the approximate values of the outputs of the assessors. In all cases, users were required to conduct the programming task until the output of the system (*i.e.*, either quantitative values from assessor outputs or confirmation dialogs) was consistent with the expected behavior. It is worth noting, however, that this does not necessarily indicate correctness of the programming, but merely indicates that the Decision Function has judged the input program (and likely alternatives of that) to be sufficiently safe (*i.e.*,"inexpensive") and thus safe for execution.

4.2.2 Results

From the user studies, it was observed that in cases where the programs were correctly entered, the system behaved consistently in terms of confirmation requests. Program 2 was the only one that issued confirmations, while Programs 1 and 3 only confirmed that the task would be executed as instructed. As mentioned, the users were not given any feedback in terms of program correctness. Thus, the programs sent to the robot were not accurate in some trials; *i.e.*, the input programs did not match exactly the programs given to the users. In case of mistakes, the Decision Function evaluated the input program and most likely alternatives, and only allowed a program to be executed (without confirmation) if and only if the task was evaluated to be less costly.



(a) Programming times, all users combined.



(b) Programming attempts and generated confirmations, all users combined.

Figure 5: Results from user studies, timing 5(a) and confirmations 5(b).

The cost of feedback, not unexpectedly, is the required time to program the robot. As seen in Figure 5(a), all three programs took more time to program on average with confirmations (top bar in each program group). From the user studies data, we see that the use of confirmations increases total programming time by approximately 50%. Although the users paid a penalty in terms of programming time, the absence of safety checks meant a greater risk to the system and higher probability of task failures. This was illustrated in all cases where the system issued a confirmation request; an example of which is demonstrated in a trial of program 3 by user 2. The input to the system was given as

"`LEFT` **`9`** `RIGHT 3 MOVIE 3` **`FOLLOW`** `FOLLOW` **`9`** `UP GPSFIX EXECUTE`"

where the mistakes are in bold. The system took note of the change in duration from $\mathbf{6 \times 3 = 18}$ seconds to $\mathbf{9 \times 3 = 27}$ sec-

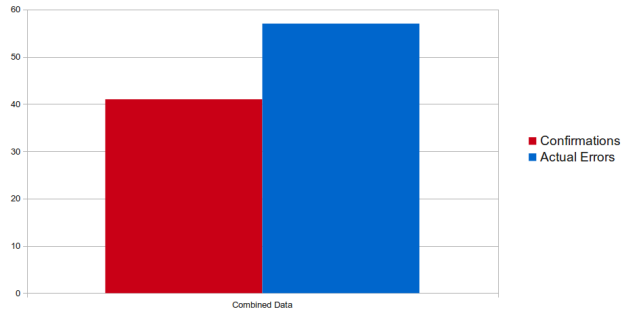


Figure 6: Error filter rate plot over all user studies data.

onds on two occasions, but more importantly, the `FOLLOW` command was issued without a `TUNETRACKER` command. This, and the change in parameters to the higher values, prompted the system to generate a confirmation request, which helped the user realize that mistakes were made in programming. A subsequent reprogramming fixed the mistakes and the task was successfully accepted without a confirmation. The distribution of confirmation requests and total number of attempts to program is shown in Figure 5(b).

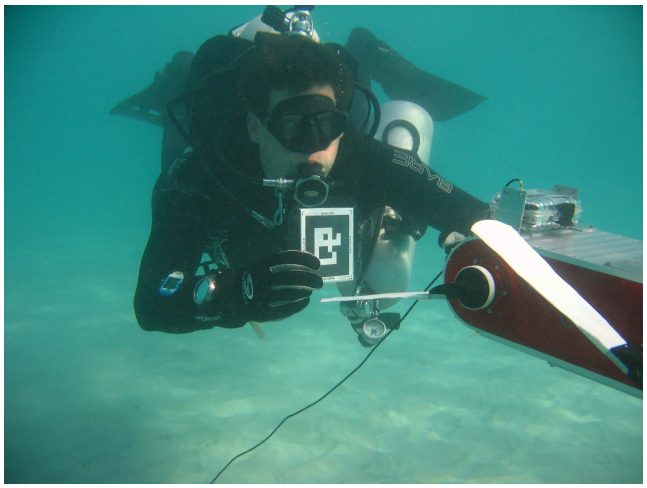
To further establish the benefits of this approach, we introduce a metric termed the *Error Filter Rate* (EFR). The EFR is a measure of the number of confirmations compared to the number of mistakes made by users during a programming task; *i.e.*,

$$\text{EFR} = \frac{\text{Confirmations}}{\text{Total Errors}}$$

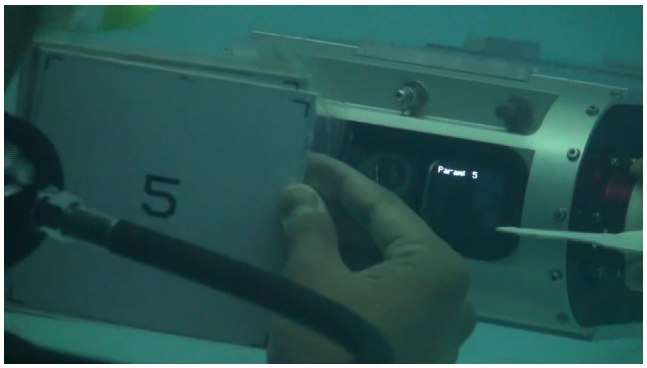
The EFR indicates the percentage of erroneous inputs which the system deemed to be dangerous; in other words, a low EFR value does not necessarily indicate a low error rate in programming, but indicates that most of the commands to the robot are interpreted as low-risk. In our studies, we achieved an EFR of approximately 72.8 *per cent*, as can be seen in Fig. 6, indicating the system interpreted roughly 72% of the erroneous commands as high-risk and intervened (with confirmation dialogs) to ensure the user's true desire.

5. FIELD TRIALS

We performed field trials of our system on-board the Aqua underwater robot, in both open-ocean and closed-water (controlled))environments. In both trials, the robot was visually programmed using RoboChat with the same language set used for the user studies, with ARTag and ARToolkitPlus [15] fiducials used as input tokens. The assessors used for the dialog user studies were also used in the field trials; in addition, we provided an assessor to take into account the depth of the robot during task execution. Because of the inherent difficulty in operating underwater (as discussed in Sec. 1), the trials were not timed. Users were asked to do each program once. Unlike in the user study, where there was no execution stage, the robot performed the tasks that it was programmed to do, when given positive confirmation to do so. In all experimental cases, the robot behaved consistently, asking confirmations when required, and executing tasks immediately when the tasks were inexpensive to perform. Unlike the user study, where the users had no feedback, the field trial participants were given limited feedback in the form of symbol acknowledgement using an micro-Organic-LED (Light Emitting Diode) or μOLED display at the back of the robot. Also unlike the user studies, the field trial participants were given access to a command to delete the program and start from the beginning, in case they made a mistake. A pictorial demonstration of our system in action during field trials can be seen in Fig. 7, which demonstrates the visual programming, and command feedback through the μOLED screen.



(a) A diver programming Aqua during ocean trials. The trailing cable is for a floating GPS antenna buoy on the ocean surface.



(b) Example of command acknowledgement given on the LED screen of the the Aqua robot during field trials.

Figure 7: Field trials of the proposed algorithm on board the Aqua robot.

6. CONCLUSIONS

This paper describes the experimental validations of two algorithms for *explicit* visual human-robot interaction which are components of a larger visual-HRI framework. The results focus primarily on the user studies, and also discusses key observations and findings from our field trials. From the results, it can be seen that individually both RoboChat and the dialog system increase efficiency and robustness in human-robot communication, particularly in areas where more traditional means of communication is not viable. The combination of both, however, is the most effective mean to increase fault-tolerance arising from mistakes in instructions, and communication uncertainty. RoboChat provides users with a expressive yet compact method for instructing a mobile robot, and the dialog engine, in a complimentary manner, ensures task and user safety, which is a much sought-after design goal of HRI systems. The implicit interaction algorithms, though not discussed for the sake of topic coherence, helps to create an effective scheme to

compliment these explicit interaction mechanisms.

Future research will aim to quantify system performance in real-world scenarios. As we discussed in the paper, a number of significant issues prevent accurate measurements of performance metrics in the field, particularly those that relate to human-centric systems. An open issue is the trade-off between expressivity, ease of use, flexibility and the minimization of coding errors. It seems that for different applications, different language subsets may be best and, in fact, this is what we have sometimes done in some practical deployments. One of our future goals is to design instrumentation capabilities as part of the framework itself, such that measurements of human operational data (along with robot performance) happens in an integrated manner. By design, the framework should be applicable to arbitrary robots across a variety of operating domains, irrespective of actual algorithms or modalities being used for human interaction. This remains a difficult challenge, and an open problem. Future goals also include adapting robot behaviors based on user input and feedback, such that more streamlined dialogs can be presented to the user, for example. As before, the challenge remains in appropriately quantifying operational parameters in a human context, such that operator preference and robot capabilities can be closely correlated towards creating a seamless man-machine interface.

7. REFERENCES

[1] K. Derpanis, R. Wildes, and J. Tsotsos. Hand gesture recognition within a linguistics-based framework. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 282–296, 2004.

[2] F. Doshi, J. Pineau, and N. Roy. Reinforcement learning with limited reinforcement: Using Bayes risk for active learning in POMDPs. In *Proceedings of the 25th international conference on Machine learning*, pages 256–263. ACM New York, NY, USA, 2008.

[3] F. Doshi and N. Roy. Spoken language interaction with model uncertainty: an adaptive human-robot interaction system. *Connection Science*, 20(4):299–318, 2008.

[4] G. Dudek, M. Jenkin, C. Prahacs, A. Hogue, J. Sattar, P. Giguère, A. German, H. Liu, S. Saunderson, A. Ripsman, S. Simhon, L. A. Torres-Mendez, E. Milios, P. Zhang, and I. Rekleitis. A visually guided swimming robot. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3604–3609, Edmonton, Alberta, Canada, August 2005.

[5] G. Dudek, J. Sattar, and A. Xu. A visual language for robot control and programming: A human-interface study. In *Proceedings of the International Conference on Robotics and Automation ICRA*, pages 2507–2513, Rome, Italy, April 2007.

[6] M. Dunbabin, I. Vasilescu, P. Corke, and D. Rus. Data muling over underwater wireless sensor networks using an autonomous underwater vehicle. In *International Conference on Robotics and Automation, ICRA 2006*, Orlando, Florida, May 2006.

[7] R. Erenshteyn, P. Laskov, R. Foulds, L. Messing, and G. Stern. Recognition approach to gesture language understanding. In *13th International Conference on Pattern Recognition*, volume 3, pages 431–435, August 1996.

[8] S. L. Kent. *The ultimate history of video games: from Pong to Pokémon and beyond : the story behind the craze that touched our lives and changed the world*. Prima, 2001.

[9] D. Kortenkamp, E. Huber, and R. P. Bonasso. Recognizing and interpreting gestures on a mobile robot. In *Proceedings of the thirteenth national conference on Artificial intelligence - Volume 2*, AAAI'96, pages 915–921. AAAI Press, 1996.

[10] K. Krebsbach, D. Olawsky, and M. Gini. An empirical study of sensing and defaulting in planning. In *Artificial intelligence planning systems: proceedings of the first international conference, June 15-17, 1992, College Park, Maryland*, page 136. Morgan Kaufmann Pub, 1992.

[11] D. Kulic and E. Croft. Safe planning for human-robot interaction. In *2004 IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA'04*, volume 2, 2004.

[12] M. Montemerlo, J. Pineau, N. Roy, S. Thrun, and V. Verma. Experiences with a mobile robotic guide for the elderly. In *Proceedings of the 18th National Conference on Artificial Intelligence AAAI*, pages 587–592, 2002.

[13] C. Pateras, G. Dudek, and R. D. Mori. Understanding referring expressions in a person-machine spoken dialogue. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing, 1995. (ICASSP '95)*, volume 1, pages 197–200, May 1995.

[14] V. Pavlovic, R. Sharma, and T. S. Huang. Visual interpretation of hand gestures for human-computer interaction: A review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):677–695, 1997.

[15] I. Poupyrev, H. Kato, and M. Billinghurst. *ARToolkit User Manual Version 2.33*. Human Interface Technology Lab, University of Washington, Seattle, Washington, 2000.

[16] P. E. Rybski and R. M. Voyles. Interactive task training of a mobile robot through human gesture recognition. In *IEEE International Conference on Robotics and Automation*, volume 1, pages 664–669, 1999.

[17] J. Sattar, E. Bourque, P. Giguère, and G. Dudek. Fourier tags: Smoothly degradable fiducial markers for use in human-robot interaction. In *Proceedings of the Fourth Canadian Conference on Computer and Robot Vision*, pages 165–174, Montréal, QC, Canada, May 2007.

[18] J. Sattar and G. Dudek. Robust servo-control for underwater robots using banks of visual filters. In *Proceedings of the IEEE International Conference on Robotics and Automation, ICRA*, pages 3583–3588, Kobe, Japan, May 2009.

[19] J. Sattar and G. Dudek. Underwater human-robot interaction via biological motion identification. In *Proceedings of the International Conference on Robotics: Science and Systems V, RSS*, pages 185–192, Seattle, Washington, USA, June 2009. MIT Press.

[20] J. Sattar and G. Dudek. Towards quantitative modeling of task confirmations in human–robot dialog. In *Proceedings of the IEEE International Conference on Robotics and Automation, ICRA*, pages 1957–1963, Shanghai, China, May 2011.

[21] J. Sattar, P. Giguère, G. Dudek, and C. Prahacs. A visual servoing system for an aquatic swimming robot. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1483–1488, Edmonton, Alberta, Canada, August 2005.

[22] M. Skubic, D. Perzanowski, S. Blisard, A. Schultz, W. Adams, M. Bugajska, and D. Brock. Spatial language for human-robot dialogs. *IEEE Transactions on Systems, Man and Cybernetics, Part C*, 34(2):154–167, May 2004.

[23] J. K. Tsotsos, G. V. S. Dickinson, M. Jenkin, A. Jepson, E. Milios, F. Nuflo, S. Stevenson, M. B. adn D. Metaxas, S. Culhane, Y. Ye, , and R. Mannn. PLAYBOT: A visually-guided robot for physically disabled children. *Image Vision Computing*, 16(4):275–292, April 1998.

[24] S. Waldherr, S. Thrun, and R. Romero. A gesture-based interface for human-robot interaction. *Autonomous Robots*, 9(2):151–173, 2000.

Functional Requirements of a Model for Kitting Plans

Stephen Balakirsky
NIST MS 8230
100 Bureau Drive
Gaithersburg, MD 20899, USA
301-975-4791
stephen.balakirsky@nist.gov

Zeid Kootbally
Dept. of Mechanical Engineering
University of Maryland
College Park, MD 20742, USA
301-975-3428
zeid.kootbally@nist.gov

Thomas Kramer
Dept. of Mechanical Engineering
Catholic University of America
Washington, DC 20064, USA
301-975-3518
thomas.kramer@nist.gov

Raj Madhavan
Maryland Robotics Center
Inst. for Systems Research, UMD
College Park, MD 20742, USA
301-975-2865
madhavan@umd.edu

Craig Schlenoff
NIST MS 8230
100 Bureau Drive
Gaithersburg, MD 20899, USA
301-975-3456
craig.schlenoff@nist.gov

Michael Shneier
NIST MS 8230
100 Bureau Drive
Gaithersburg, MD 20899, USA
301-975-3421
michael.shneier@nist.gov

NIST = National Institute of Standards and Technology

ABSTRACT

Industrial assembly of manufactured products is often performed by first bringing parts together in a kit and then moving the kit to the assembly area where the parts are used to assemble products. Kitting, the process of building kits, has not yet been automated in many industries where automation may be feasible. Consequently, the cost of building kits is higher than it could be. We are addressing this problem by building models of the knowledge that will be required to operate an automated kitting workstation. A first pass has been made at modeling non-executable information about a kitting workstation that will be needed, such as information about a robot, parts, kit designs, grippers, etc. A model (or models) of executable plans for building kits is also needed. The plans will be used by execution systems that control robots and other mechanical devices to build kits. The first steps in building a kitting plan model are to determine what the functional requirements are and what model constructs are needed to enable meeting those requirements. This paper discusses those issues.

Categories and Subject Descriptors

D.3.3 [**Programming Languages**]: Language Constructs and Features – *frameworks, data types and structures, classes and objects, control structures.*

General Terms

Design, Standardization, Languages

Keywords

assembly, functional requirements, kitting, language, model, planning, process planning,

Permission to make digital or hard copies of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. ACM acknowledges that this contribution was authored or co-authored by a contractor or affiliate of the U.S. Government. As such, the Government retains a nonexclusive, royalty-free right to publish or reproduce this article, or to allow others to do so, for Government purposes.

PerMIS '12 March 20-22 2012 College Park, MD, USA

Copyright 2012 ACM 978-1-4503-1126-7-3/22/12...$10

1. INTRODUCTION

Industrial assembly of manufactured products is often performed by first bringing parts together in a kit and then moving the kit to the assembly area where the parts are used to assemble products. Kitting, the process of building kits, has not yet been automated in many industries where automation may be feasible. Consequently, the cost of building kits is higher than it could be. We are addressing this problem by building models of the knowledge that will be required to operate an automated kitting workstation. A first pass has been made at modeling non-executable information about a kitting workstation, such as information about a robot, parts, kit designs, grippers, etc. The model is written in Web Ontology Language (OWL) [5]. A model (or models) of executable plans for building kits is also needed. Thus far, we have only a mock-up of a sample plan that includes a natural language description of the elements of a plan model. We intend to build that model in OWL, also. The plans will be used by execution systems that control robots and other mechanical devices to build kits. The first steps in building a kitting plan model are to determine what the functional requirements are and what model constructs are needed to support those requirements.

We are working towards developing standard representations of both kitting workstations and process plans for kitting. Kitting is accomplished by discrete processes, so we consider only process plans for discrete processes. We are also committed to using hierarchical control. In the case of a kitting workstation, there are at least two control levels, the workstation level and the robot level. The robot controller will take commands from the workstation controller. In this paper we deal only with planning models for the workstation level.

In section 2 we introduce existing process plan models. Section 3 discusses functional requirements for the language used to build a process plan model. Section 4 presents constructs often found in process plan models, relates them to the functionality they serve, and describes the extent to which we are currently planning on using them in the plan model for kitting. Section 5 discusses planning considerations that affect the need for various types of functionality. Section 6 describes how we plan to evaluate the adequacy of the model for kitting process plans and gives suggestions for further work.

This paper does not deal with process models or with the connection between process plans and processes.

2. EXISTING PROCESS PLAN MODELS

A few standard discrete process plan models and many non-standard models have been developed.

2.1 Standard Process Plan Models

Since we are interested in standards, we have examined existing standards for discrete process plans. These include the following.

ISO 10303, generally known as STEP (STandard for the Exchange of Product model data), includes a Part 49, "Process structure and properties" [1]. The model is built in the EXPRESS language (as are all ISO 10303 models). It is a very general, domain-independent model. The central object of the model is "action_method." Other than describing itself as specifying "the elements of a process plan," which is "the specification of instructions for a task," the document containing the model provides no description of the functionality of a process plan that it is intended to provide. The concepts defined in STEP Part 49 are used in Part 240 of STEP, which is focused on process plans for machined products [3].

Part 10 of ISO 14649 [2] "specifies the process data which is generally needed for NC-programming within all machining technologies." This includes the definition of a general process plan model that might be used outside of the machining domain as well as inside. A central element of the model is "Executable." Instances of Executable "initiate actions on a machine" when executed. Part 10 of ISO 14649 is remodeled in STEP terms in STEP Part 238 [4].

Languages for programming machine tool controllers may be regarded as process plan models. The Dimensional Measuring Interface Specification (DMIS) is the only standard language for writing programs to be executed by the controller of coordinate measuring machines [6]. DMIS has many of the plan constructs described in section 4.

2.2 Other Process Plan Models

Since we are using OWL to model the environment of a kitting workstation, we looked at the Process.owl section of the most recent version of OWL-S [7]. OWL-S was originally developed for the World Wide Web Consortium (W3C), but was never adopted as a W3C recommendation.

There is an enormous body of literature regarding planning, particularly planning on state spaces. The book *Automated Planning* [9], for example, includes over 500 references. There is a correspondingly wide variety of plan models. We do not attempt to describe them here.

The language *A Language for Process Specification* (ALPS) was developed at the National Institute of Standards and Technology (NIST) and has been used in a few NIST projects [10].

3. LANGUAGE AND STRUCTURE FUNCTIONAL REQUIREMENTS

The language used to represent a process plan model should make it possible to use plan instances easily. Specifically, the model should be written in a language that is automatically processable into computer code that (1) has data structures for representing a plan, (2) can read a plan file and save it in terms of the automatically defined structures, (3) has access functions for getting data out of the structures and putting data into them, and (4) can write a plan file from the structures. Languages for which mature software exists that can generate computer code as just described include EXPRESS and XML schema. Software exists that can process OWL that way, but it is at an early stage of development and is not yet widely used [11].

Even if a code generator for a language can do the four things listed above, if the structure of the model is too general, the information available by using the access functions may be too atomic for even an expert in both programming and process planning to use readily. This is the case with both Part 238 of STEP and Part 240 of STEP. A set of computer code written by a STEP expert is required to extract meaningful process plan data usable by a programmer building a process planner or a process plan executor. On the other hand, automatically generated code built by processing ISO 14649 (which, like the two STEP parts, is written in EXPRESS) using the same code generator is readily usable by an application builder [17].

It is not sufficient, of course, to have a plan model. It must be possible to represent plans that are instances of the model. For EXPRESS models, there is more than one standard way in which this may be done, the most commonly used of which is a "physical file" [11]. For XML schema, instances are built in XML files that conform to the schema [18]. For OWL, instances and structures may be put into the same file. It is more convenient, however, to have a fixed structures file and build instance files that use the structure file via an "import" statement. Curiously, while C++ is a widely used standard programming language [16] and would be entirely adequate for building class models of many planning domains, there is no standard textual data file representation for C++ class instances.

4. PLAN CONSTRUCTS

The plan model needs to be rich enough to represent all aspects of the kit building process. This process includes operations ranging from selecting the appropriate gripper for moving kit trays or parts to iterating through a list of steps that place parts in a kit. In order to meet these requirements, we have examined techniques for representing parameters, variables, resources, and actions.

An abbreviated example kitting plan is shown in Fig. 1 using XML format. The example uses the constructs described in the remainder of this section.

```
<ProcessPlan>
 <About>
  <PlanId>kitABCPlan</PlanId>
  <PlanVersion>1.0</PlanVersion>
  ...
  <TargetSKU>kitABC</TargetSKU>
 </About>
 <PlanRequirements>
  <PlanRequirement>
   <Name>boxOfEmptyTrays</Name>
   <Type>LargeContainer</Type>
   <SkuRef>Box1</SkuRef>
   <ContentsType>
    <SkuRef>KitTrayX</SkuRef>
   </ContentsType>
  </PlanRequirement>
  ...
 </PlanRequirements>
```

```
  <PlanParameters>
  <PlanParameter>
    <Name>NumberOfKitsToMake</Name>
    <Type>positiveInteger</Type>
  </PlanParameter>
 </PlanParameters>
 <InternalVariables>
  <InternalVariable>
    <Name>BoxWithEmptyTrays</Name>
    <Type>LargeContainer</Type>
    <Requirement>boxOfEmptyTrays</Requirement>
  </InternalVariable>
  <InternalVariable>
    <Name>CurrentKitTray</Name>
    <Type>KitTray</Type>
    <SkuRef>KitTrayX</SkuRef>
    <InitialValue>NULL</InitialValue>
  </InternalVariable>
  ...
 </InternalVariables>
 <ToDo>
  <Start></Start>
  <DoInGivenOrder>
    <DoInAnyOrder>
      <Bind>
        <Variable>BoxWithEmptyTrays</Variable>
        <WhichOne>ANY</WhichOne>
        <ErrorAction>QUIT</ErrorAction>
      </Bind>
      <Bind>
        <Variable>BoxForFullTrays</Variable>
        <WhichOne>ANY</WhichOne>
        <ErrorAction>QUIT</ErrorAction>
      </Bind>
      ...
    </DoInAnyOrder>
    <Set>
      <Variable>n</Variable>
      <Value>0<Value>
    </Set>
    <LoopInGivenOrderWhile>
      <Test>n LessThan NumberOfKitsToMake</Test>
      ...
      <RobotMoveAbove>CurTrayPose</RobotMoveAbove>
      <RobotPickUp>CurrentKitTray</RobotPickUp>
      <RobotMoveAbove>KitTrayPose</RobotMoveAbove>
      <RobotPutDown>
        <What>CurrentKitTray</What>
        <Where>KitTrayPose</Where>
      </RobotPutDown>
      ...
      <Set>
        <Variable>n</Variable>
        <Value>n+1<Value>
      </Set>
    </LoopInGivenOrderWhile>
  </DoInGivenOrder>
  <Stop></Stop>
 </ToDo>
</ProcessPlan>
```

Fig. 1 Kitting Process Plan Example (abbreviated)

4.1 Plan Parameters

Plans can have parameters, such as the name of a file of decision rules to use or the number of kits to be put together. If plan parameters are used, structures to support their use may be needed in the plan model. Having a parameter for the number of kits, for example, requires some structure that implements looping. Plan parameters are set in the command to execute the plan. Typically, plan parameters are not reset during plan execution. Plan parameters serve the function of allowing execution time specification of what to do or how to do it. The current kitting process plan model has a plan parameters section.

4.2 Plan Variables

Plan variables are variables set in the course of executing a plan, not in the command to run the plan. It is useful if plan variables have specific data types. A given variable may represent different objects of the same type during plan execution. The current kitting process plan model has an InternalVariables section that contains plan variables.

4.3 Resources

The current kitting process plan model has a PlanRequirements section that gives required resources.

4.3.1 Resource Requirements

A process plan that is intended to be executable should make it easy for a user to determine if the resources required to execute the plan are available. The straightforward way to do this is to have a separate section of the plan that lists the required resources. Each step of a plan should identify each resource it requires beyond what the plan as a whole assumes is available. It is not sufficient, however, to mention resources only as they are associated with steps of the plan, since if only that is done, it may be difficult to determine the total set of required resources.

A plan model for a specific domain (a kitting workstation, for example) may assume the availability of fixed resources in the environment (a robot, for example). The resource section of a plan does not need to include those resources. If a plan is intended to be usable in several different environments of the same type (different kitting workstations, for example), then the resources section of the plan will need to include specific values applicable to the fixed resources in that type of environment (the extent of a robot work volume, for example).

Where plans include alternative actions and those actions use different resources, it may be hard for the user to determine if available resources are adequate. Where one resource may be substituted for another and at least one of a set of alternative resources must be used, there is no difficulty. The list of required resources simply contains sets of mutually substitutable resources (three alternative grippers, for example). If alternative ways of executing the plan require different sets of resources, there is a problem. On the one hand, it is counterproductive to force the user to assemble all the resources that might be required. The user should have to assemble only a minimal set of required resources. On the other hand, until an execution of the plan is performed, it is not known which resources will be used. Where decisions on which alternative to use are made on the basis of environmental conditions that change slowly, one way to deal with this is to run a simulation of executing the plan. Then the resource requirements can be pared down to those resources used in the simulation. At the same time, the plan would be pruned of those branches that are not used. The reduced plan would be usable as

long as the conditions under which the reduced plan is executed are close enough to those under which the reduced plan was generated. Simulation before execution is also useful when a user has a set of resources and a complex plan and needs to determine if the plan can be executed with that set of resources.

4.3.2 Resource Descriptions

The description of a resource might be given at any of three levels of abstraction.

- a description of the capabilities of the resource (for example, the lifting capacity and maximum opening of a gripper)
- a specification of a resource in a catalog (for example, GripCo model 123)
- a specific instance of a resource (for example, GripCo model 123 with serial number ABC).

Which resource description level to use, if any, depends on the level of abstraction a plan is intended to have. Section 5 discusses levels of abstraction.

4.4 Actions

The actions section of a plan specifies what to do. This is by definition a functionality every sort of process plan must have. The actions section must always include tasks. The actions section may also include explicit control structures, or information to be used for control may be contained in the task description. In any event, some method of controlling the order in which tasks are performed is required. The current kitting process plan model has a ToDo section that contains the actions.

4.4.1 Explicit Control Structures

The current kitting process plan model includes all of the following types of control structure except for synchronous operation, DoSimultaneously, and DoSome.

4.4.1.1 Do In Given Order

In many process planning models and most computer programming languages, the default rule for execution order is to do things in the order in which they are listed in the file, and there is no explicit control structure for doing things in that order.

The functionality of being able to execute plan steps in the order in which they are given in a file is very convenient. Unless a process plan model uses implicit control structures throughout the actions section, the model should include a default rule or an explicit DoInGivenOrder control structure. If the plan model includes explicit commands for ordering, such as described immediately below, then having an explicit DoInGivenOrder will help avoid confusion.

4.4.1.2 Do In Any Order

In theory, an extraordinarily simple process plan language might specify in its natural language execution rules that the steps of all plans may be executed in any order. In any realistic plan model, however, if the ability to say that some set of steps may be performed in any order is needed, then an explicit control structure implementing this functionality is needed.

A DoInAnyOrder functionality is desirable if it is expected that there will be circumstances in which no particular task order is required and the system executing the plan is either capable of multitasking or is expected to have better information available for setting the order than is available at the time the plan is made.

The DoInAnyOrder control structure might have subtypes that allow or disallow simultaneous execution of tasks. If the execution system is known be able to perform operations in parallel, then the plan model should include a DoSimultaneously control structure that requires parallel execution.

4.4.1.3 Do One

The DoOne control structure is followed by a list of alternatives. The execution system picks one alternative and executes it. The execution system is free to pick any of the alternatives. It may pick one at random, or it may evaluate the goodness of the alternatives by whatever criteria it prefers and pick the best one. The alternatives will usually have the same primary effect but may have different secondary effects. For example, in kitting, if it is necessary to get at box A which is underneath box B, the plan might include a DoOne with the alternatives of putting box A on the table or putting box A on box C.

Some languages include a DoSome control structure that specifies that any N of a set of alternatives should be executed. This is more powerful than DoOne, since when N is 1, it is equivalent to DoOne, but occasions when N is not 1 will probably be rare – remove three of the six boxes on the table, for example.

4.4.1.4 Branch on Condition

Another type of control structure includes a condition to be tested followed by a specification of what to do if the condition is met. In common computer languages, these are called if or switch or select. All of them may be combined with else, which specifies what to do if none of the explicit conditions is met. Switch and select have cases. Implementing condition testing requires that the plan language include variables and (usually) expressions, for example, "(x+y) > 3" is a condition that is a Boolean expression using a less than operator to compare an arithmetic expression containing variables and an addition operator with a numerical constant. The Boolean test may be implicit rather than explicit, but variables are always needed.

Branching on a condition is a functionality that is hard to do without whenever a plan model includes plan parameters and/or variables.

4.4.1.5 Loop

When a set of steps must be repeated a number of times or as long as a condition holds, a control structure that implements looping (iteration) is needed. The simplest form of loop simply states that a set of steps must be executed N times, and there is no explicit test (the execution system is expected to keep track), but in most of the many varieties of loop structure ([15] has a 40-page chapter on looping), a condition is tested at some point in the loop that stops the looping.

For kitting plans, our model includes LoopInAnyOrderWhile and LoopInGivenOrderWhile. In these control structures, a condition is tested before any step in the list of conditional steps is executed. The rest of the action of these loops is as implied by their names.

4.4.1.6 Synchronous Operation

If two devices must operate together to accomplish something (such as two robot arms picking up opposite ends of a pipe), a control structure for synchronization is needed in the plan.

4.4.1.7 Create and Destroy Instances

Control structures that are able to create and destroy instances will be useful in the plan model for any activity in which instances come into existence or go out of existence during plan execution. In kitting, for example, kits come into existence that did not exist before plan execution was started, and part supplies go out of existence when they are empty (the empty container that remains is no longer a part supply).

4.4.1.8 Bind Resources

A model of a step that binds a plan variable to an instance of a resource is useful in the plan model. When a "bind" step is executed, a plan variable representing a resource is set to a specific object in the workstation matching the description of a resource. This requires being able to obtain information about what is in the workstation. Such information would reside in a dynamic knowledge base, so implementing resource binding requires that a dynamic knowledge base be available to the plan executor. Resource binding might be combined with resource allocation. For example, when a bind command is executed, the resource might be marked as unavailable as the value of a set command or another bind command.

4.4.1.9 Set Variables

A model of a step that sets a plan variable to a value is useful in the plan model. When a "set" step is executed, the value of a plan variable is set. The value to which the variable is set may be obtained by a straightforward knowledge base inquiry (such as the location of a solid object) or it may be obtained by evaluating an expression (for example, (a + b)) or making a function call (for example, a call to a function that returns the first item in a list). The last two methods, of course, require that the plan model include an expression model and a function model.

4.4.1.10 Start and Stop

Because explicit start and stop control structures simplify executing plans, the plan model should include Start and Stop. Only one Start step is allowed in a plan, and it must be the first step. Either multiple Stop steps or only one might be allowed. If only one is allowed, it must be the last step.

4.4.2 Implicit Control Structures

The order in which steps of a plan are executed may be controlled implicitly by putting a list of predecessor (and/or successor) steps into each step. In some implementations of this ([10], for example), only "join" steps, which are steps that join threads coming from a matching "split" step may have more than one predecessor. In other implementations, any step may have multiple predecessors, and the control rule is that all the predecessors of a step must be executed before the step may be executed. The two approaches may be combined using split/join pairs that enable/disable the use of multiple predecessors. This was implemented in [13]. The use of multiple predecessors allows the plan to be executed in multiple orders that would otherwise be allowed only by including a combinatorial explosion of split/join pairs.

4.4.2.1 Do In Precondition Order

Enabling the use of multiple predecessors for a portion of a plan may be implemented by the DoInPreconditionOrder control structure. A DoInPreconditionOrder step is followed by a list of steps, each of which has a sequence number and a list of the sequence numbers of other steps that must be executed previously. All the steps in the list must eventually be executed.

4.4.3 Support Structures

Where steps or conditions in a plan require numbers or Boolean values, it is convenient if plan parameters, plan variables, object properties, operator expressions, and functions are used. These all may be classed as subtypes of expression. Some plan models, such as STEP part 49 and ALPS, observe that an expression model is required without modeling one. Other plan models, such as DMIS, include explicit models of expressions.

For kitting, in order to deal with location information and do geometric reasoning, all of the support structures just listed are required. For example, in order to take a part out of a part supply, a function that finds the first part remaining in the part supply is needed, and the location property of that part must be found in order to generate an instruction telling the robot where to go to pick up the part. As another example, if there is a stack of empty trays in a box and we want to pick up the one on top (which is not necessarily the first one in the list of trays in the box), a function that finds the tray on top is needed.

4.4.4 Kitting Actions

A set of task types specific to kitting is required in a kitting process plan model. The last subsection of this subsection presents the task types we intend to use first. The stage is set by brief descriptions of the objects in a kitting workstation, the scenario our plan model must support, and the execution model we intend to follow.

4.4.4.1 Objects in a Kitting Workstation

Our initial kitting workstation model is relatively simple. A kitting workstation contains some fixed equipment: a robot, a work table, a part gripper, a tray and kit gripper, and a gripper changing station. Items that enter the workstation include empty kit trays, boxes in which to put finished kit trays or empty part supply trays, and part supplies. A part supply may be a tray or box with parts inside in known or unknown locations or a box containing trays with parts. Items that leave the workstation may be boxes with finished kits inside, empty part trays, empty boxes, or boxes with empty part trays inside.

4.4.4.2 Scenario

In our kitting project, the first version of the plan model is designed to support the following scenario. An external agent (which we call the factotum) sets up the workstation by putting into it:

- a box of empty kit trays (may be only partially full)
- a box for finished kits (may have some kits in it already)
- a box for empty part supply trays
- several part supply trays

The knowledge base for the workstation includes descriptions of the designs of kits, parts, and trays involved. The knowledge base also has descriptions of where all the objects in the workstation are. The factotum that sets up the workstation fills in the knowledge base so that it describes the setup correctly. After the initial setup, objects are expected to move only if the robot or factotum moves them. Whenever an object is moved by the robot or factotum, its location is updated. The workstation control system builds kits by:

- telling the robot to take an empty kit tray out of the box of empty kit trays and to put it on the work table
- telling the robot several times to take a part out of a part supply and put it in the kit being built

- telling the robot, whenever a kit is finished, to put the finished kit in the finished kit box
- telling the robot to change its gripper as necessary for handling either parts or part trays
- telling the factotum, whenever necessary, to remove empty parts trays, to put part supplies in, to put boxes of empty kit trays in, or to remove full boxes of finished kits.

4.4.4.3 Kitting Task Execution

The scenario is carried out by having the workstation controller execute a workstation level process plan. The workstation controller can, by itself, execute steps that set variables, choose among alternatives, etc. To move things, however, the workstation controller requires the robot or the factotum to execute instances of specific types of kitting tasks. This is expected to be accomplished by having the workstation controller send a command to the robot controller or the factotum. The robot controller or factotum will carry out the command and report back whether command succeeded or failed. If the command succeeds, the workstation controller will execute the next step in the plan. If the command fails, the workstation controller will either just stop executing the plan or deal with the error condition outside of executing the process plan and then resume executing the plan. As currently envisioned, resuming plan execution after an error will be feasible only if the error condition can be corrected and the workstation environment can be set to the state it would have been in if the command that failed had succeeded.

Currently, the kitting process plan model contains no error handling tasks. The workstation controller is expected to deal with error conditions independently from executing the process plan.

4.4.4.4 Types of Kitting Tasks

The task types that have been defined to enable writing a plan that follows the scenario include the following.

- FactotumRefill - This is followed by a variable representing a requirement. When the statement is executed, the factotum puts an object of the required type in the workstation and updates the workstation model.
- FactotumRemove - This is followed by a variable representing the object to remove. When the statement is executed, the factotum removes the object and updates the workstation model.
- FactotumReplace - This is followed by a variable representing the object to replace. When the statement is executed, the factotum removes the object, puts another object of the same type in the same place, and updates the workstation model. The new object should be different from the old one in an appropriate way.
- RobotChangeEndEffector - This is followed by the name of an EndEffector to change to. When the statement is executed, if the robot is not already holding the named EndEffector, the robot moves to the changing station, puts down the EndEffector it has (if it has one) and picks up the named EndEffector. If the robot is already holding the named EndEffector, no action is taken.
- RobotMoveAbove - This is followed by a Pose. When this statement is executed, the controlled point on the robot's end effector moves to a point that has the same X and Y values of the location of the Pose but has a greater Z value by some amount the executor thinks will be sufficient so that the robot will not collide with anything near the location point. This statement is not particularly well defined and might be modified.
- RobotPickUp - This is followed by a variable whose value is the object to pick up. When the statement is executed, the robot moves its gripper down into position for grasping the object, the gripper grasps the object, and the robot moves up so that the height of the lowest point of the object is the same as what the height of the lowest point of the gripper was previously.
- RobotPutDown - This is followed by a variable representing the object to put down and a variable representing the Pose of the object at which the object should be released. When the statement is executed, the robot moves the object into the given Pose and releases the gripper's grip on the object. Then the robot moves up so that the lowest point of the gripper is clear of the object that was put down.

4.5 Other Plan Contents

A process plan file needs to include information that may be used to keep track of the document. This information is not used by the process plan execution system at execution time, though it may be used immediately before execution starts to verify that the right plan is being used. The current kitting process plan model includes an About section with subsections for PlanId, PlanVersion, PlanDateAndTime, PlanAuthor, PlanWorkstation, Description, and TargetSKU (an identifier for the stock keeping unit data that is a detailed description of the type of kit to be made).

5. PLANNING CONSIDERATIONS

The ways in which plans are intended to be generated and used is a major consideration in deciding what constructs to include in the plan.

5.1 Abstraction

The most abstract (or high-level) plan may specify only the intended effects of the plan. For a kitting workstation, a high-level plan might state that a number of kits of a particular type are to be made. For a quality control system, a high-level plan might state that parts of a particular type are to have the tolerances on a particular set of features checked.

If a plan is intended to be executable, the plan should include resources, executable operations, and whatever degree of ordering is required for executing the operations.

In many industrial settings, it is useful if a process plan can be refined in stages. The NIST Manufacturing Systems Integration (MSI) project, for example, identified three stages, which were called (1) process plans (2) production-managed plans, and (3) production plans [12]. As used in the MSI project, "A production-managed plan is an expansion of a process plan which supports the production of a required number of products using a given factory configuration. A production plan is a refinement of a production-managed plan which identifies specific resources for each step and the times of

their usage for that step." As described, a production plan is a combination of a plan and a schedule. Scheduling, in our view, is beyond the scope of a process plan, but supporting the other types of refinement described by the MSI project, as well as refinement by pruning branches of a plan, is a functionality that may be required of a process plan model. With this functionality, a single plan model will support both a plan and any refinements of it (possibly in a chain of successive refinements). This implies, for example, that all levels of action abstraction and resource description should be supported by the plan model.

Plan refinement was implemented in the Feature Based Inspection and Control System at NIST [13] and was discussed in [14]. Refining a plan may require generating a separate document containing the refined plan. A plan model that supports representing both a plan and its refinement in a single document may be unnecessarily complex. Regardless of the way in which refinement is handled, there must be a link from any refinement back to the plan it refines.

5.2 Decision-making Responsibilities

Planning decisions might be made in either the planner or the plan executor. Depending on the assignment of planning responsibilities to the planner or the plan executor, the functional requirements of the plan model may be very different.

At one extreme, if the planner knows enough to make all the decisions, a plan format may suffice that is simply an ordered list of tasks to perform. In this case, since no decisions need to be made at execution time, no Boolean expressions, if-thens, or structures that allow alternatives are needed in the plan. In addition, since the natural form of a file is an ordered list, no ordering structures are needed. All that is necessary is to be able to tell where one step ends and the next begins. Because a file is an ordered list by nature, the most abstract plans will require using a structure such as DoInAnyOrder that is able to disorder the steps.

At the other extreme, if there may be foreseen but random changes in the environment in which the plan is executed (e.g., the robot is apt to drop things) or if the conditions of the environment are not known at planning time (e.g., the location of the part supply is not set until execution time), the plan will need to include items such as variables, if-thens, sets of alternatives, and Boolean expressions.

5.3 Extendible Generic Plan Model

It is extremely desirable to have a generic model of process plans that may be extended into specific domains. If models for different domains build on a common core, people who understand the plan model for one domain can gain understanding of other plan models much more easily than if there is no common core. Similarly, it will be possible to use the core software of a system that executes plans in one domain when building a plan execution system for a new domain.

Because the target level of plan abstraction varies from application to application, the generic model must be built so as to support different levels of abstraction efficiently and clearly. It may be possible to support different levels of abstraction by using optional elements. This notion needs further examination since items that are optional at a high level may be required at lower levels.

A generic plan model might specify the sections of the plan, control structures, some aspects of resource description, and a generic task. Specializations of the generic plan for specific domains would have specialized resource and task descriptions that are subtypes of generic tasks and resources.

5.4 Human Comprehensibility

With a human in the loop during plan generation (always or as needed), the range of good plans that can be generated expands greatly. Thus, one functional requirement is that the semantics of the plan model should be readily understandable to trained humans. The syntax does not need to be human-friendly since user-friendly interfaces can be built to generate syntax from user actions that convey the semantics. Since computers can handle a wide variety of syntax, however, it should be possible to design a syntax that is friendly to both humans and computers. That will be helpful when no user-friendly interface is available and a human needs to do planning.

6. CONCLUSION

We plan to build:

- an OWL model of kitting workstation process plans
- example process plans conforming to the model
- C++ software for representing, reading, writing, and accessing the plans
- a C++ kitting workstation plan executor
- a simulated kitting workstation
- an actual kitting workstation

Using the simulated and actual workstations, we plan to evaluate the performance of the kitting process plan model. Where we find a need for additional functionality in the model, it will be added. If we discover plan functionality that is not used in our example plans and does not appear likely to be used in any plans, it will be removed.

We intend to include sensory processing in the kitting workstation. Some of this, such as a switch that detects whether a gripper is seated properly in a gripper changer, might be used only by the robot controller. Other sensory data will be reported to the workstation's knowledge base. For example, we might have fixed cameras that feed into a system that computes the observed locations of objects in the workstation. For any observed object, the observed location data might be fused with the location data that is a priori or entered in the course of plan execution. A large difference between the stored and observed values might trigger an error signal.

The sensory processing described in the previous paragraph requires nothing from the contents of a process plan or from a process planner. The only thing it requires from a process plan executor is the ability to receive error signals and react to them. Other elements of the system would handle sensory processing and knowledge base maintenance. Hence, we currently do not deal with sensory processing in the process plan model.

If we find that the robot needs to help with sensory processing or that sensory devices need explicit instructions that are coordinated with robot actions, then the process plan model will need to be expanded to include tasks for sensory processing devices or robot tasks that serve sensory processing. For example, a camera end effector might be defined and used. If a part were dropped and could not be found by fixed sensors, the robot would change to

the camera end effector and move it into position to see where fixed cameras cannot see. As another example, if a box with one part in it is dropped and the part cannot be located, the robot might be commanded to move the box in order to determine if the part is now under the box.

As mentioned earlier, there are currently no kitting workstation tasks in the process plan model designed specifically for error recovery. If it is found that error recovery tasks are needed in the process plan model, they will be added.

7. REFERENCES

[1] ISO 1998. *Industrial automation systems and integration – Product data representation and exchange – Part 49: Integrated generic resources: Process structure and properties*, 1998, International Organization for Standardization.

[2] ISO 2004. *Industrial automation systems and integration – Physical device control – Data model for computerized numerical controllers -- Part 10: General process data*, 2004, International Organization for Standardization.

[3] ISO 2005. *Industrial automation systems and integration – Product data representation and exchange – Part 240: Application protocol: Process plans for machined products*, 2005, International Organization for Standardization.

[4] ISO 2007. *Industrial automation systems and integration – Product data representation and exchange – Part 238: Application protocol: Application interpreted model for computerized numerical controllers*, 2007, International Organization for Standardization.

[5] W3C 2009. *OWL 2 Web Ontology Language Structural Specification and Functional-Style Syntax 4.1*, W3C Editor's Draft 21 September 2009, W3C, http://www.w3.org/2007/OWL/draft/ED-owl2-syntax-20090921.

[6] Dimensional Measurement Standards Consortium 2009. *Dimensional Measuring Interface Standard Part 1 Revision 5.2*, ANSI/DMIS 105.2 Part 1-2009, Dimensional Measurement Standards Consortium.

[7] SRI 2008. *OWL-S 1.2 Release*, http://www.ai.sri.com/daml/services/owl-s/1.2.

[8] Nof, S. Y., Wilhelm, W. E., Warnecke, H-J., 1997. *Industrial Assembly* , Chapman & Hall, London, UK.

[9] Ghallab, M., Nau, D., Traverso, P., 2004. *Automated Planning Theory and Practice*, Morgan Kaufmann, San Francisco, USA.

[10] Catron, B., Ray, S., 1991. *ALPS – A Language for Process Specification*, International Journal of Computer Integrated Manufacturing, Vol. 4, No. 2, pp 105-113.

[11] http://sourceforge.net/apps/mediawiki/owl-cpp

[12] Wallace, S., Senehi, M. K., Barkmeyer, E., Ray, S., Wallace, E., 1993. *Manufacturing Systems Integration Control Entity Interface Specification*, NISTIR 5272, National Institute of Standards and Technology, Gaithersburg, MD, USA.

[13] Kramer, T. R., Horst, J. A., Huang, H. M., Messina, E., Proctor, F. M., Scott, H. A., 2004. *Feature-Based Inspection and Control System*, NISTIR 7098, National Institute of Standards and Technology, Gaithersburg, MD, USA.

[14] Jasthi, S. R. K., Rao, P. N., Tewari, N. K., 1995. *Studies on Process Plan Representation in CAPP systems*, Computer Integrated Manufacturing Systems, Vol. 8, No. 3, pp 173-184.

[15] Steele, G. L, 1990. *Common LISP the Language*, Second Edition, Digital Equipment Corporation.

[16] ISO/IEC 14882:2011. *Information Technology – Programming Languages – C++*, 2011, International Organization for Standardization.

[17] Kramer, T. R., Proctor, F., Xu X., Michaloski, J. L., 2006. *Run-time interpretation of STEP-NC: implementation and performance*; International Journal of Computer Integrated Manufacturing, Volume 19, Issue 6, pp 495 – 507.

[18] W3C 2004. XML Schema Part 1: Structures Second Edition, W3C Recommendation 28 October 2004, http://www.w3.org/TR/xmlschema-1.

The new method for measuring absolute threshold of haptic force feedback

Michal Baczynski, Ph.D.
Samsung Electronics Polska
Poland R&D Center
Polna 11
00-633 Warsaw, Poland
+48 22 377 8114
m.baczynski@samsung.com

ABSTRACT
This paper describes new fast and accurate method of measuring absolute threshold for haptic force feedback. The classic, widely published methods applied to measure force absolute threshold are time consuming because of time consumed by measurement procedure and time necessary for user training. The proposed method of measurement is very intuitive, thus it does not require trainings. The author has done researches using different methods and as a result stated that new method is not worse in terms of accuracy than classic ones.

Categories and Subject Descriptors
J.2.2 [**Computer Applications**]: Physical Sciences And Engineering – Engineering

General Terms
Algorithms, Measurement, Performance, Experimentation, Human Factors, Verification.

Keywords
Force feedback, absolute threshold, haptic, kinesthetic sense, Flexible Wall Technique.

1. INTRODUCTION
It is obvious that teleoperations that involve telerobotic systems have a number of advantages over traditional approach when humans perform tasks manually. It is possible to work from a distance, isolated from possibly dangerous materials, more precise and many others. One of the important drawbacks, however, is the fact that operator can't feel any of the resistance put up by the manipulated objects and obstacles in the workspace – essentially, the controls provide no sense of touch. To address this problem more and more often modern robots provide force feedback from workspace to the operator. The desire for natural, intuitive means of human-machine interactions, and for multi-modal sensory feedback to users has resulted in the design of machines which allow users to generate control inputs using hand motion, and at the same time experience forces or resistance on their hands which create interesting and useful perceptions. These machines are called haptic interfaces and are used as user interface parts of telerobotic systems.

The experiments show that human kinesthetic force perception is limited, that means that people cannot detect forces of any values. Detection is the problem of determining whether given exerted force is present or not, and relates to the absolute sensitivity of human sensory systems. The absolute threshold of detection is the value of a force that is just noticeable to an observer. That minimal detectable force value is called "absolute threshold for haptic force feedback".

The main difficulty in determining thresholds of perception is that when people are presented with identical stimuli on different occasions, they do not always respond in identical ways. For instance, a signal which may be detected on one occasion may not be detected on another, so that the transition from stimulus intensities that are never detected to those that are always detected is not perfectly sharp. One reason for this is presumably that the neurosensory system is somewhat noisy. Other reasons include attention differences, learning, and adaptation. There are the three classical psychophysical methods for determining absolute threshold: constant stimuli, limits, and adjustment. The experiments have been conducted with all of the methods to gather the reference results for proposed new method.

2. New experimental procedure
The new experimental method, called "Flexible Wall Technique" has been proposed. To conduct the experiment the commercially available force feedback interface Sensable Haptic Omni has been involved, see figure 1.

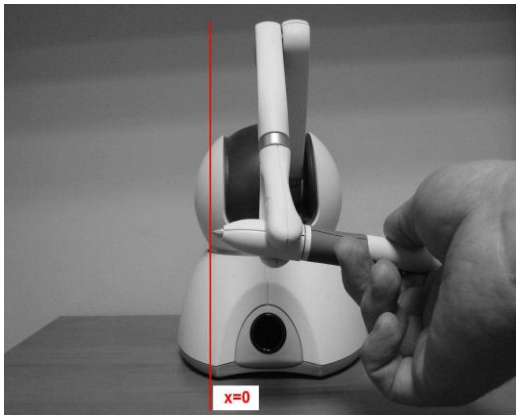


Figure 1. The haptic device used in the experiment.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage, and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
PerMIS'12, March 20-22, 2012, College Park, MD, USA.
Copyright © 2012 ACM 978-1-4503-1126-7-3/22/12...$10.00

The application that controls the device generates virtual haptic wall in workspace as presented on figure 1. The user is not able to see that flat obstacle because it is only generated by force feedback haptic. Operator is able to tacitly feel it and slide the tip of pen on the surface. The algorithm of generating the wall is defined as presented on figure 2.

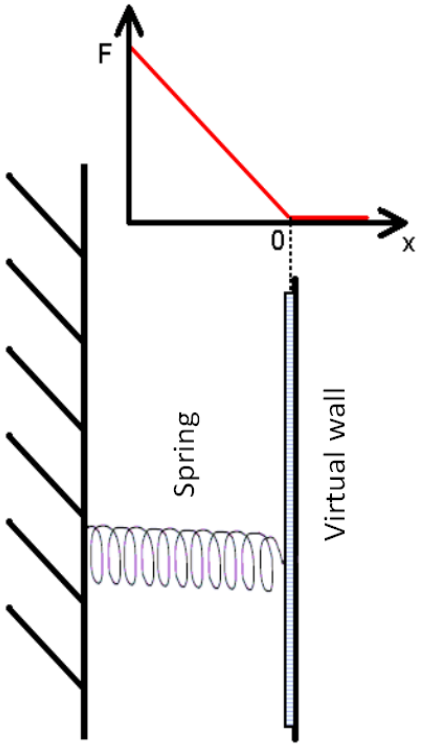


Figure 2. The algorithm that is used to generate the flexible wall, the profile of exerted force.

The wall is produced by force F exerted along X-axis, see figure 1. The force is generated according to the following formula, see figure 3.

$$\begin{cases} \mathbf{F} = [-k \cdot p_x \,;\, 0 \,;\, 0]^T & p_x < 0 \\ \mathbf{F} = [0 \,;\, 0 \,;\, 0]^T & p_x \geq 0 \end{cases}$$

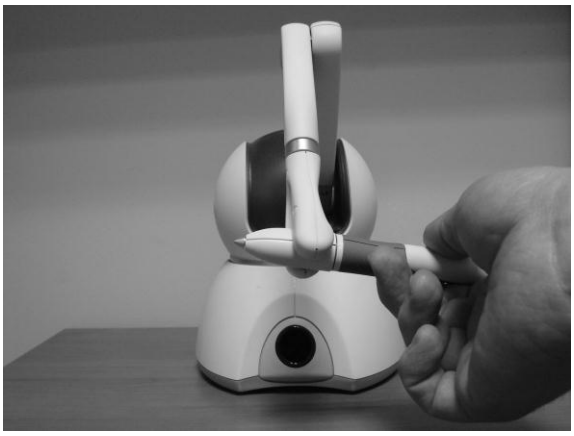
Figure 3. Formulas that define force generation.

In practice the wall is not 100% stiff but it is flexible. The generated force that represents stiffness of the wall is proportional to deflection of wall and constant value k equals to 0.125 N/mm.

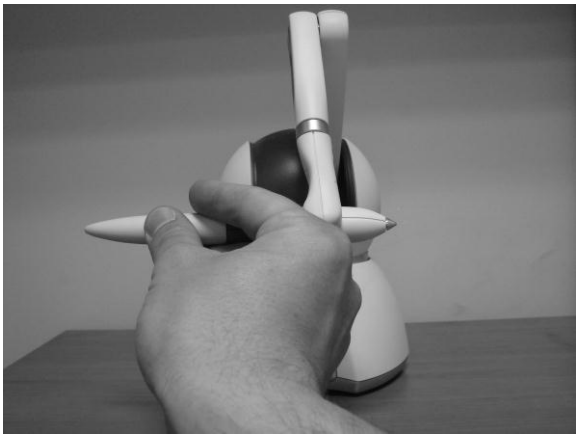
Examined person is asked to move the tool of haptic device – pen on the surface of the virtual wall for 5 seconds. The goal is to slide the pen exactly on the surface in circular manner and not to deflect the wall. In fact the wall is slightly deflected, but the user does not feel it. The computer application gathers the information about deflections and forces during the experiment with the 1000 Hz rate. In that way during 5 seconds experiment 5000 samples are registered. The new described method enables to gather 5000 results in a very short time and by calculating average of values the Absolute Threshold for Haptic Force Feedback is obtained.

3. Results of experiments

Ten persons have been examined. Two variants of hand orientation have been taken into consideration – see figure 4.



Hand orientation – variant A.



Hand orientation – variant B

Figure 4. The experimental trials have been divided into two variants: hand orientation A and B.

In both hand orientation variants the force is exerted along X-axis. The difference is that in variant A the force is generated in positive direction of X-axis, in variant B in negative. The user in one variant is pulled and in the other is pushed by the tool.

The experimental results of measurements performed with proposed method “Flexible Wall Technique” are presented in table 5.

Examined Person	Results variant A	Results variant B
1	0.313	0.482
2	0.319	0.451
3	0.351	0.438
4	0.299	0.464
5	0.302	0.465
6	0.286	0.465
7	0.325	0.478
8	0.297	0.439
9	0.340	0.474
10	0.311	0.439
Average	0.314	0.460
Standard deviation	0.020	0.017

Figure 5. Results obtained by using "Flexible Wall Technique", variant A and B of hand orientation.

4. Discussion

The experimental results of the Absolute Thresholds for Haptic Force Feedback measurements are gathered in the tables presented on figure 6 and 7.

Method	Variant A Result [N]
Constant stimuli method	**0.342**
Limits method	**0.320**
Adjustment technique	**0.351**
Flexible Wall Technique	**0.314**
Average	**0.332**

Figure 6. Results obtained by using different measurement methods – hand orientation variant A.

Method	Variant B Result [N]
Constant stimuli method	**0.459**
Limits method	**0.426**
Adjustment technique	**0.469**
Flexible Wall Technique	**0.460**
Average	**0.453**

Figure 7. Results obtained by using different measurement methods – hand orientation variant B.

Results obtained by using different measurement methods are similar. The huge different is in time and simplicity of the measurement technique. The users have judged clearly that "Flexible Wall Technique" is the simplest and the most intuitive way of conducting the experiment.

5. ACKNOWLEDGMENTS

My thanks go to Sensable Technologies Inc. for allowing me buying the Phantom Omni device in reduced price.

6. REFERENCES

[1] Baczynski M., Baczynski J.: „Simple stereo unit for perception of space depth in teleoperator systems", Proceedings of IEEE International Conference on Industrial Informatics, Berlin 2004, pp. 407-410

[2] Baczynski M., Baczynski J.: „Simple computer system for transferring and gamma radioactivity scanning of neutron-activated samples", Proceedings of IEEE International Conference on Industrial Technology, Slovenia, Maribor 2003, pp. 213-216

[3] Feygin D., Keehner M., Tendick F.: „Haptic Guidance: Experimental Evaluation of a Haptic Training Method for a Perceptual Motor Skill", Proceedings of the 10th Symposium on Haptic Interfaces For Virtual Environments & Teleoperator Systems, USA, Orlando 2002, pp. 40-47

[4] Carignan C. R., Cleary K. R.: „Closed losed-loop force control for haptic simulation of virtual environments", Haptics-e The Electronic Journal of Haptic Research, 1(2), www.haptics-e.org, pp. 1-14

[5] Colgate J. E.: „Power and impedance scaling in bilateral manipulation", Proceedings of the 1991 IEEE International Conference on Robotics and Automation, USA, Sacramento 1991, pp. 2292–2297 Vol.3

[6] Fasse E.D., Hogan N.: „Quantitative measurement of haptic perception", Proceedings of IEEE International Conference on Robotics and Automation, USA, San Diego 1994, pp. 3199-3204 vol.4

[7] Gu, J.H.; de Silva, C.W.: „Interpretation of mechanical impedance profiles for intelligent control of robotic meat processing", Proceedings of the 1996 IEEE IECON 22nd International Conference on Industrial Electronics, Control, and Instrumentation, Tajwan, Tajpej, 1996, pp. 507 512, vol.1

[8] Hogan, N.: „Controlling impedance at the man/machine interface”, Proceedings of IEEE International Conference on Robotics and Automation, USA, Scottsdale 1989, pp. 1626 - 1631 vol.3

[9] Hondori, H. M., Shih-Fu L.: „A method for measuring human arm's mechanical impedance for assessment of motor rehabilitation”, Proceedings of the 3rd International Convention on Rehabilitation Engineering & Assistive Technology, Singapur, 2009, pp. 31-35

[10] Jezierski E.: „Dynamika robotów”, Wydawnictwa Naukowo-Techniczne, Warsaw 2006

[11] Kim K., Youm Y., Chung W. K.: „Human Kinematic Factor for Haptic Manipulation : The Wrist to Thumb”, Proceedings of the 10th Symposium on Haptic Interfaces For Virtual Environments & Teleoperator Systems, USA, Orlando 2002, pp. 319-326

[12] Kirkpatrick A. E, Douglas S. A.: „Application-based evaluation of haptic interfaces”, Proceedings of the 10th Symp. On Haptic Interfaces for Virtual Envir. & Teleoperator Systs., USA, Orlando 2002, pp. 32-39

[13] Millman P.: „Haptic perception of localized features”, PhD Disertation, Nothwestern University, 1995

[14] SensAble Technologies, Inc.: „PHANTOM omni user’s guide”, 2004

[15] Yamakawa S., Fujimoto H., Manabe S., Kobayashi Y.: „The necessary conditions of the scaling ratio in master–slave systems based on human difference limen of force sense”, IEEE Transactions on systems, man and cybernetics – part A: Systems and Humans, vol. 35, no. 2, 2005, pp. 275-282

Approach for Defining Intelligent Systems Technical Performance Metrics

Wael Hafez
WHA Research
Alexandria, VA, USA
(+1) 202 322 8223
w.hafez@wha-research.com

Abstract
Intelligent systems performance is a result of the interaction and cooperation of the system's components with one another and with their environment. In general, those components and subsystems can be fundamentally different in nature, structure and the role each plays in the system's overall activity. It might be possible to measure the performance of each subsystem in its own terms. However, such measures will apply only to the subsystems of the same type. Taking intelligent systems heterogeneity into consideration, the paper argues that to understand and analyze the performance of intelligent systems, it is necessary to develop measures that apply to their different components regardless of their nature. The paper applies communication theory to develop such measures. Accordingly, the activities of an intelligent system and its subsystems are considered to be communication activities. The characteristics of this communication determine the system technical performance. Communication measures are used to define a system's communication state, which reflects the system technical performance regardless of its nature.

Categories and Subject Descriptors
H.1.1 [**Information Systems**]: Systems and Information Theory – *general systems theory, information theory.*
I.2.11 [**Computing Methodologies**]: Distributed Artificial Intelligence – *intelligent agents, multiagent systems.*

General Terms
Measurement, Performance, Design, Standardization, Theory.

Keywords
Intelligent systems, Multiagent systems, Agents, Complex Adaptive Systems, Cyber Physical Systems, Panarchy, Performance Measures.

1. INTRODUCTION
In the general case, intelligent systems are heterogeneous, complex systems made up of several components and subsystems. The performance of an intelligent system is determined by the activities and interactions of its subsystems. Figure 1 is a general representation of an intelligent system. The different subsystems can be computational agents, mechanical components, applications, or even a human operator with a specific role.

The different nature of the involved subsystems is a challenge for designing and analyzing the performance of the system [1] [4] [5]. This is mainly due to the fundamentally different models and laws used to describe the different types of subsystems. The current paper argues that if it is possible to consider the activity of the subsystems as an activity of communication, then their performance can be described in terms of communication, regardless of their different types. That is, if all subsystems can be considered as communication systems, then they all can be analyzed and modeled form the same communication perspective.

This is similar to analyzing the performance of a power network from an energy consumption perspective by identifying the energy consumed by the different network components and systems, regardless of their nature, or the purpose for which they use this energy.

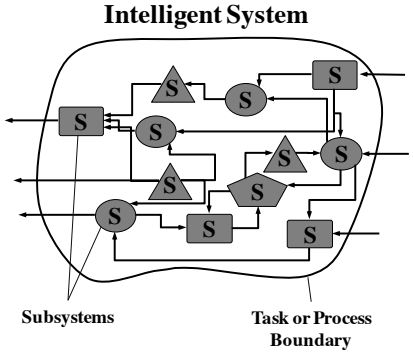


Figure 1. Intelligent system general structure. The boundary defines all subsystems involved in realizing a task or a process performed by the system.

To represent the different subsystems as communication systems, it is required to describe the activities of those subsystems in terms of communication. Communication theory is concerned with "… reproducing at one point either exactly or approximately a message selected at another point" [6]. This is achieved by creating a series of dependencies between the two points. In terms of communication, the two points are the communication source and communication destination. Accordingly, the activity of

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage, and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. PerMIS'12, March 20-22, 2012, College Park, MD, USA. Copyright © 2012 ACM 978-1-4503-1126-7-3/22/12...$10.00

communication stands in establishing and optimizing the dependencies that relate the source and the destination. Communication theory showed that the higher the dependency between the source and destination, the higher the possibility of reproducing at the destination the message that was selected at the source.

If the system's input and output are considered to stand for the source and destination of this communication, then the system's communication activity can be considered to be an activity of establishing a dependency between its input and output [3]. Figure 2 shows the basic architecture of a system from a communication perspective.

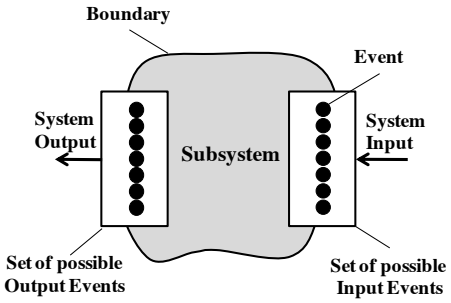


Figure 2. Subsystem architecture

The system input and output are made up of two sets of limited events. Out of either set of events, the system can only build a limited number of inputs or outputs (see Figure 3). Communication theory shows that the more dependency among the events within a set, the fewer inputs (or outputs) that can be constructed out of the set, and vice versa.

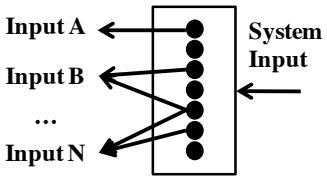


Figure 3. Building inputs out of events

2. SUBSYSTEMS AS COMMUNICATION SYSTEMS

Representing a system (regardless of its nature) as a communication system is realized by identifying the system's input and output events. The subsystem's activity is expressed in reacting to some input with a specific output. By doing this, the subsystem defines dependencies among its inputs and outputs. Those dependencies are either built into the subsystem by its very design, or defined by the subsystem if it has adaptation and learning capabilities. From communication perspective, and regardless of the subsystem's nature, its communication behavior is primarily defined by the size of its input and output event sets, the dependencies within each set and the dependencies between the two sets.

The first step in approaching intelligent systems from the communication perspective is to define the following for each of the subsystems within the system (see Figure 4):

- Clear boundaries defining the subsystem and separating it from the environment and other subsystems,
- A communication source (X): a component representing all possible input events to the system from the environment, including other subsystems.
- A communication Destination (Y): a component representing all possible output events from the subsystem to the environment, including other subsystems.

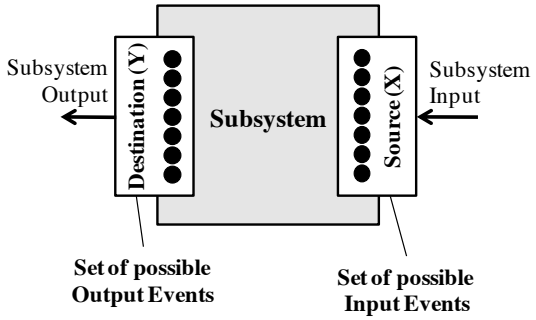


Figure 4. Representation of a subsystem as a communication system.

It should be noted that the source and destination belong to the same subsystem. The subsystem inputs and outputs are made out of the events triggering the subsystem, and triggered by the subsystem during its interaction with the environment or other subsystems. The set of input events is considered to be the system's communication source and the set of output events is considered to be the system's communication destination. From this perspective, the subsystem behavior is a result of receiving inputs from the environment, evaluating them, and sending a response back to the environment in form of specific outputs.

A subsystem might perform very complex activities and use many operations and steps to evaluate the input and select an output response. However, from a communication perspective, and no matter how elaborate or complex those internal operations might be, the system - at the end - is establishing a correlation or dependency between its input and output event sets.

3. SUBSYSTEM COMMUNICATION ENTROPIES

Communication theory measures the activity of communication by obtaining the statistical characteristics (expressed by entropies) for the different quantities involved in the communication activity. Figure 5 shows a representation of the entropies involved in the communication between a source X and a destination Y.

According to communication theory, for a set of symbols, entropy is a measure of the different ways a message can be constructed using the symbols in the set. Entropy is at maximum if there are no constraints or rules for building the messages. However, applying rules and constraints on how to use and relate the symbols in the set reduces the number of possible messages, thus the set's entropy. This is because rules and constraints establish

dependencies among the symbols within the set, thus reducing the number of possible ways to combine them.

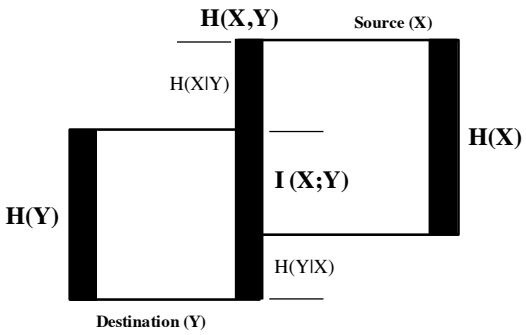


Figure 5. System communication entropies. The X and Y squares represent the source and destination sets respectively. The side length of each square stands for the entropy of the set. The height of the shared side represents the source-destination joint entropy.

In the current context, symbols are considered to be events, and messages are considered to be the inputs or outputs (see Figure 3). The different combinations of the events define the different inputs to, and outputs from the system. Entropy is the number of bits required to define the possible inputs that can be defined out of the source events, or the number of possible outputs from the destination events. The more dependencies there are among the events within a set, the less the number of possible inputs (or outputs) that can be defined for this set, and the less the entropy of the set. On the other hand, the fewer dependencies among the events, the more inputs or outputs that can be constructed, and the higher the entropy of the set.

Communication theory defines the following entropies for describing communication activity between a source and a destination:

3.1 Source and destination entropies

The source and destination entropies are defined as the following:

- Source entropy, H(X): the number of bits required to describe the source different inputs. The number of events in the source and their probability of occurrence determine the source entropy. Each source input represents an input to the system.
- Destination entropy, H(Y): the number of bits required to describe the destination different outputs. The number of events in the destination and their usage probabilities define the destination entropy. Each destination output represents a system output.

The source and destination event sets define the limits to what the system can receive and send as it interacts with other systems. Accordingly, the two sets are considered to make up the system communication resources.

3.2 Joint entropy, H(X,Y)

During the interaction with the environment or other subsystems, the activity of the subsystem is manifest in defining, using, maintaining and redefining dependencies between its inputs and outputs. That is, the system tries to establish a correlation between its input and output event sets.

Communication theory defines Joint entropy as the number of bits required to describe the correlation between two sets X and Y. In the current context, and given an input and output event sets X and Y, joint entropy is the number of bits required to describe all available input-output (I/O) dependencies that can be built for X and Y. This value is at maximum when X and Y are independent (all inputs and outputs are available for building I/O correlations). As the system interacts with the environment and defines more dependencies between its source and destination, the joint entropy decreases because the number of available inputs and outputs also decreases. This is represented in Figure 6.

3.3 Mutual Information, I(X;Y)

In communication theory, mutual information defines the channel capacity: the degree of source and destination dependency. This dependency is defined by the dependent inputs and outputs. From this perspective, mutual information is the number of bits required to describe the actual I/O dependencies used by the system during its communication activity. The number of the dependent inputs and outputs and their corresponding probabilities determines this entropy. As shown in Figure 6, mutual information increases as the system builds and uses more I/O dependencies. That is, the more correlation between the system's input and output, the higher the mutual information.

Communication theory states that due to noise, there is always a degree of uncertainty in any communication. The effect of noise on communication is represented by the conditional entropies H(X|Y) and H(Y|X)[1].

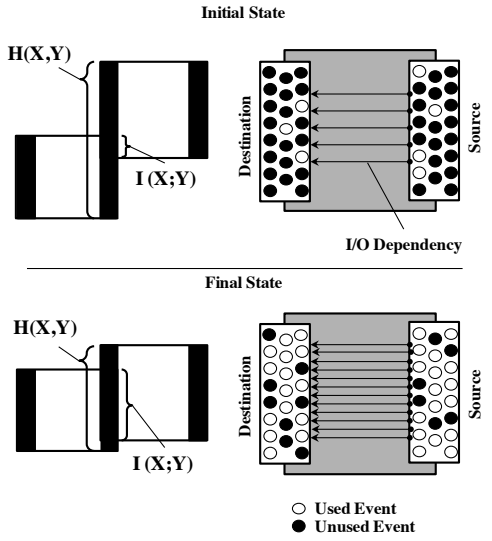


Figure 6. Changes in communication entropies. In the initial state, a system has a low degree of I/O dependency, thus a higher joint entropy value than in the final state. It is the reverse for the mutual information that increases as the system builds more I/O dependencies.

[1] At the current basic level of analysis, conditional entropies are not considered.

4. SYSTEM COMMUNICATION STATE VARIABLES

The interaction of subsystems with their environment (and with each another) results in changes in their inputs and outputs, as well as in the dependencies among them. In terms of entropy and information changes, the interaction results in changes in the subsystem's source and destination entropies, as well as in its mutual information. As all those values are dependent, they are used here to define what will be called the system communication state. The entropy-based variables that define a system communication state are discussed below.

4.1 Communication Capacity

In communication, capacity is a measure of the level of dependency between the source and the destination. For a subsystem, the number of I/O dependencies and their corresponding probabilities define the subsystem communication capacity. As shown in Figure 5, the quantity that defines communication capacity is the mutual information:

$$\text{Communication Capacity} = I(X;Y) \quad (1)$$

Entropy and information are calculated using the same equation and have the same units (bits). However, entropy is used to describe the inputs or outputs that can be built out of the corresponding sets, where information refers to the dependency between those inputs and outputs. In other words, information is used to describe the statistical characteristics between two dependent sets and entropy is used to describe the same characteristics for the sets themselves.

4.2 System Communication Efficiency

Any system – no matter how simple or complex – has limited resources. Efficiency is a measure of how good the system is using those resources to sustain itself and achieve its goals. A subsystem can have different efficiency measures according to the analysis perspective. If the ultimate goal of communication is to create and increase the dependency between the source and destination, then communication efficiency is about how good does the communication system achieves this goal. In other words, communication efficiency is the ratio between the actual I/O dependency to the available or maximum I/O dependency.

In terms of entropies, the actual I/O dependency is defined by the mutual information and the source-destination joint entropy defines the maximum I/O dependency.

$$\text{Communication Efficiency} = \quad (2)$$

$$\text{Channel Capacity / Input-Output Joint Entropy}$$

$$= I(X;Y) / H(X,Y) \quad (3)$$

This ratio thus defines the subsystems communication efficiency. For a fixed size input and output event sets, increasing communication efficiency is achieved by increasing the communication capacity (I/O dependency). On the other hand, increasing the size of the input and output event sets (increasing either H(X) or H(Y) or both, and thus H(X,Y)), will directly reduce the system's communication efficiency, but not necessarily it is communication capacity.

Communication theory states that due to noise, channel capacity will always remain less than the joint entropy. Accordingly, communication efficiency is always < 1.

4.3 Communication Flexibility

Flexibility is the ability of the system to adjust its activity to changes in the operating conditions in order to keep realizing its goals. Changes in operating conditions are expressed by changes in the number of subsystem's input events, their values or probability of occurrence. Such changes might result in new values of existing events, or the presence of completely new events. At any point of time, an intelligent system has a certain level of adaptability to new conditions in its environment. This level depends directly on the resources available to the system to deal with the unexpected and new input from the environment [2].

Flexibility is considered here to be a relative value, and will be defined as the ratio of the system unused communication resources to the system available communication resources. A system with a 20% flexibility means that 20 % of the system's communication resources are still available and can be used for communicating with its environment.

If we consider Figure 5, and as the system communication capacity defines the system used communication resources, the remaining communication resources are equal to the difference between the system available communication resources H(X,Y), and the used resources. In terms of the communication entropies:

$$\text{Unused Communication Resources} = H(X,Y) - I(X;Y) \quad (4)$$

$$\text{Communication Flexibility} = (H(X,Y) - I(X;Y)) / H(X,Y) \quad (5)$$

$$= 1 - \text{Communication Efficiency} \quad (6)$$

Communication flexibility is at its maximum (equals one) when the input and output event sets are completely independent. That is, the system did not define any I/O dependencies yet. It is at zero when the two sets are completely dependent. That is when all possible I/O states are defined. According to communication theory this is not attainable due to the noise present in any communication. That is, the system can never achieve a complete dependency between its source and destination event sets.

5. SYSTEM COMMUNICATION STATE

The diagram in Figure 7 relates the three communication variables. The definition of system communication state diagram is motivated by concepts related to complex adaptive systems (CAS) behavior. CAS are considered to have a lifecycle in which the relationships and dependencies within the system change as the system interacts and adapts to its environment [2].

For a specific subsystem, any point in the chart represents the communication state at which this system can exist.

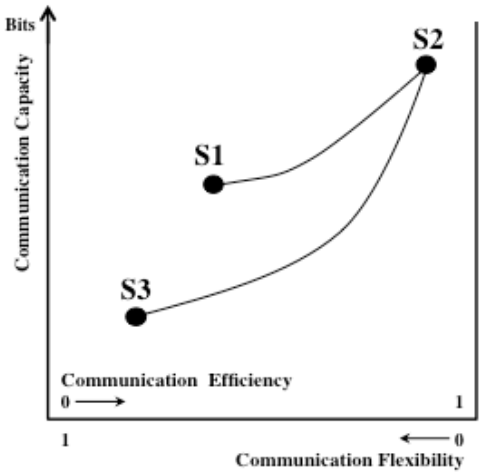


Figure 7. System communication state diagram

The horizontal axis represents the communication efficiency and as such, has a range from zero to one. Communication flexibility is represented on the same axis but as the complement of the efficiency. Changes in the size of the input and output event sets are reflected on this axis as such changes directly impact the system's efficiency and flexibility. Communication capacity is represented in bits on the vertical axis and defines the system level of I/O dependency.

According to this representation, a subsystem that exists at (S1) will have a specific communication capacity and efficiency values. As it interacts with its environment, it might define new I/O dependencies. Such new I/O dependencies increase the subsystems communication capacity and in return communication efficiency. This moves the system to a new communication state such as (S2).

When at (S2), if the subsystem faces new conditions in the environment, or new interaction with other subsystems, its options and reactions are different from the case when faced with the same new conditions while at (S1). This is because at (S1) the subsystem still has more flexibility and more freedom in using its free input and output events to build the required I/O dependencies to face the new conditions.

This is not the case when at (S2). The subsystem here has less available communication resources. In this case it can either decouple some I/O dependencies (or redefine them to adapt to the new changes), or expand its communication resources by extending its input and output event sets. Both options will impact the system communication efficiency and the system might end up at state (S3).

6. DISCUSSION

The communication state describes the activity of a system because it captures this activity as changes in the system's input or output, regardless of the origin or meaning of those changes. Such changes can either be due to new events, or changes in the probability of existing events (how often they happen). As far as a system can be represented as a communication system, the nature of the events is not relevant, only how they relate and depend on one another matters. This enables the definition of the communication state for systems with different nature and makes it possible to model and analyze systems of different natures from the same perspective.

Figure 8 shows an example for representing the interaction among subsystems within an intelligent system from a communication perspective. The shown subsystems are assumed to support a specific process performed by the intelligent system. Representing the subsystems as communication systems indicates the process overall communication behavior and how this behavior relates to the communication behavior of the involved subsystems.

To apply the approach to analyze the process performance, for each of the involved subsystems, the following is defined:

- Input output event sets
- Inputs and outputs with their corresponding probabilities
- I/O dependencies (dependent inputs-outputs) and their corresponding probabilities

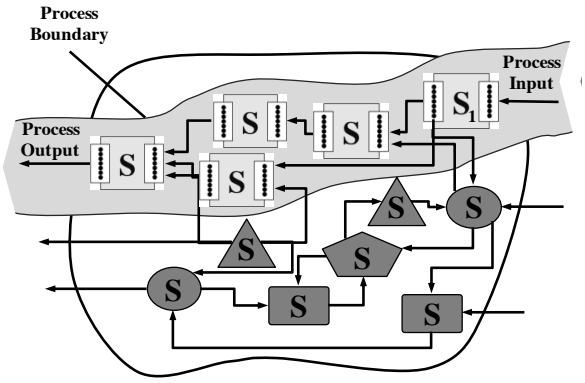


Figure 8. Intelligent system communication Analysis

Based on those values, the communication entropies are determined and the communication state is defined for each subsystem. If there is a process input change, an analysis question could be how the different subsystems react to this change? Such a change can take the form of a new input event to the subsystem S1. Such new input increases S1 source entropy H(X). This in return reduces the efficiency of the subsystem. The communication capacity is first impacted when the new event is used to define a new I/O dependency or included in an existing one. Both changes will impact the subsystem overall communication state. The analysis and changes in dependent subsystems along he process can be traced in the same manner.

In general, and depending on the analysis question, communication states can be defined for a subsystem within the intelligent system, a process that encompasses several subsystems, or for the intelligent system as a whole.

7. CONCLUSION

The paper argued that the system communication state reflects any changes happening to the system as it interacts with its environment. Although the conclusions in the current approach are theoretical and have not been practically verified or tested,

they rely in their validity on the logic and results of the well-established theory of communication.

As defined by Shannon and Weaver [6], communication theory states that the ultimate objective of communication is for the source to influence and change the behavior of the destination in a desired manner. The theory argues that although the achieved influence is due to the meaning of the messages exchanged between the source and destination, the technical aspect of communication (which is concerned with the delivery of messages between the two) is the foundation for this influence to take place. This is because if the technical aspect is not reliable, or efficient, it is uncertain that the desired meaning is transmitted, and thus it is also uncertain how far the messages from the source did influence the behavior of the destination.

The current paper is following the same logic. The subsystems within an intelligent system achieve their objectives by interacting with one another. Although the result of this interaction depends on the content and meaning of the exchanged input and output events, the statistical characteristics of how subsystems use, group and relate those events determines, and directly impacts the subsystem's behavior. In other words, the technical aspect of the system's performance, that is, the number of events it uses as well as their frequency of usage, are the bases for the system's performance. Accordingly, the technical measures developed here are the bases upon which the system's functional performance can be based.

8. REFERENCES

[1] Bogdan, P. 2011. Towards a science of cyber physical systems design. In *Proceedings of the IEEE/ACM Second International conference on Cyber Physical Systems* (Chicago, IL, USA, April 11 - 14, 2011) IEEE/ACM, 99-108. DOI= http://ieeexplore.ieee.org/10.1109/ICCPS.2011.14

[2] Gunderson, L. H., and Holling, C. S. (Eds.) 2002. *Panarchy, Understanding Transformation in Human and Natural Systems*. Island Press, Washington.

[3] Hafez, W., 2010. Intelligent System-Environment Interaction as a Process of Communication. In *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, (Istanbul, Turkey, October 10-13, 2010) IEEE, 3393-3396. DOI=http://ieeexplore.ieee.org/10.1109/ICSMC.2010.5642335

[4] Lee, E. 2008. Cyber Physical Systems: Design Challenges. In *Proceedings of the 11th IEEE Symposium on Object Oriented Real-Time Distributed Computing (ISORC)* (Orlando, Florida, USA, May 05 - 07, 2008) IEEE, 363-369. DOI=http:// 10.1109/ISORC.2008.25

[5] Lee, E. 2010. CPS Foundations. In *Proceedings of the 47th IEEE/ACM Design Automation Conference* (Anaheim, CA, USA, June 13-18, 2010) IEEE/ACM, 737-742.

[6] Shannon, C.E., Weaver, W. 1949. *The Mathematical Theory of Communication.* Univ. of Illinois Press, Illinois.

Metrics for Planetary Rover Planning & Scheduling Algorithms

J.M. Delfa Victoria*
Department of Computer Science
Technische Universität Darmstadt, Germany
+496151903123
delfa@sim.tu-darmstadt.de

N. Policella
ESA-ESOC
European Space Agency
Darmstadt, Germany
+49615192258
nicola.policella@esa.int

M. Gallant
ESA-ESOC
European Space Agency
Darmstadt, Germany
+496151902668
marc.gallant@esa.int

O. von Stryk
Department of Computer Science
Technische Universität Darmstadt, Germany
+496151162513
stryk@sim.tu-darmstadt.de

A. Donati
ESA-ESOC
European Space Agency
Darmstadt, Germany
+496151902574
alessandro.donati@esa.int

Y. Gao
Surrey Space Center
University of Surrey, UK
+441483683446
yang.gao@surrey.ac.uk

ABSTRACT

In addition to its utility in terrestrial-based applications, Automated Planning and Scheduling (P&S) has had a growing impact on space exploration. Such applications require an influx of new technologies to improve performance while not comprimising safety. As a result, a reliable method to rapidly assess the effectiveness of new P&S algorithms would be desirable to ensure the fulfillment of of all software requirements. This paper introduces *RoBen*, a mission-independent benchmarking tool that provides a standard framework for the evaluation and comparison of P&S algorithms. RoBen considers metrics derived from the model (the system on which the P&S algorithm will operate) as well as user input (e.g., desired problem complexity) to automatically generate relevant problems for quality assessment. A thorough description of the algorithms and metrics used in RoBen is provided, along with the preliminary test results of a P&S algorithm solving RoBen-generated problems.

Categories and Subject Descriptors

D.2.8 [**Software Engineering**]: Metrics—*complexity measures, performance measures*

General Terms

Algorithms, Measurement, Performnace

*Also affiliated to Surrey Space Center, University of Surrey, UK and to ESA-ESOC, European Space Agency, Darmstadt, Germany

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage, and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
PerMIS'12, March 20-22, 2012 College Park, MD, USA
Copyright 2012 ACM 978-1-4503-1126-7-3/22/12 ...$10.00.

1. INTRODUCTION

Recently, robotics has been gaining prominence in several space scenarios such as planetary exploration, ISS exploitation and deep space missions. The complexity of these missions requires the infusion of new technologies in order to maximize performance while preserving the safety of the spacecraft. One of these technologies is Automated Planning and Scheduling (P&S), which gives the system the ability to make decisions about the actions to execute while considering its status and the changes in the environment. A number of missions from NASA and ESA have been equipped with different levels of on-board autonomy [12, 13, 6, 1] providing a great improvement in terms of cost savings, science return and safety, among other benefits [4, 7].

A relevant problem when introducing innovation is the assessment of the new technologies in order to provide adequate confidence to the customer that the software satisfies its requirements [9]. This is particularly important in the case of future missions where real test-cases may not be available to prove the adequacy of new solutions [14]. The development and introduction of these new concepts should be supported by ad-hoc methodologies and techniques to measure the final expected performance of the system.

In particular, this paper focuses on the assurance of software product quality, while program processes and implementation are not within the scope of this study. We have identified *Functionality* [8] as the characteristic to be measured, more precisely the *Efficiency* of the product. For this sub-characteristic, we have identified a number of metrics that can be classified in two different groups:

- Problem complexity metrics that analyse the complexity of the generated problem.
- Performance metrics that focus on the performance of the P&S algorithm.

It is worth remarking that problem complexity metrics are crucial to deeply understand the results obtained by using the performance

metrics. In fact, good performance results are valuable only when obtained on (very) complex problem instances. Problem complexity metrics can be further divided between:

- Static metrics that only consider the problem input and initial values (e.g., resource availabilities) to calculate the complexity.
- Dynamic metrics that also consider the estimated behavior of the system during the plan execution.

For example, dynamic metrics can consider the resources available at the specific time each individual task should be executed, the initial amount, the expected consumption for each task already executed, and the expected amount of resources generated during execution.

This paper presents a new mission-independent benchmarking tool called RoBen, which is currently under development. The main objectives of RoBen are to provide a standard framework to evaluate and compare automated planning tools as well as to help future operators validate alternative plans. In particular, RoBen will automatically generate problems to evaluate the quality of P&S algorithms for rover scenarios.

A set of metrics assembled for use with RoBen is introduced in the paper. While the performance of software products has been widely studied, problem complexity in real scenarios such as space robotics remains quite immature and dependant on the specific characteristics of the mission and/or planning tools. Therefore, a combination of metrics both novel and borrowed from the literature [8, 11, 5] have been used to improve the results.

Finally, we present some preliminary results with different autonomously generated problems evaluated by a general purpose planner developed at ESA.

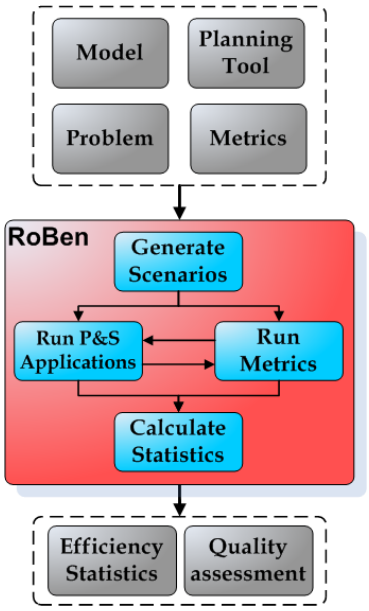


Figure 1: RoBen Architecture

2. MODELLING LANGUAGES

The problem generated via RoBen is based on the DDL3 (Domain Description Language) and the associated PDL (Problem Domain Description) [10]. This language is used in the area of AI

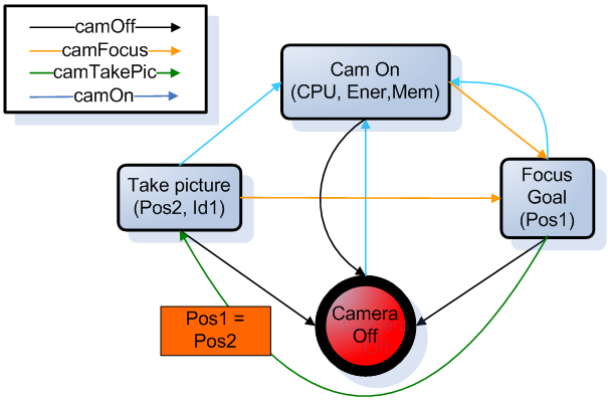


Figure 2: Automaton describing the state transitions of the camera on-board a rover

Timeline Planning and in particular by the APSI planner [2], which is a timeline planning framework that uses Component-Based formalism.

A problem in DDL3/PDL is decomposed into components. Examples of components in the rover scenario might be simple elements (e.g., the camera of a rover), complex elements (e.g., the locomotion system composed of wheels, motors, etc), or external elements (e.g., a rock on the surface). The process of representing a planning domain/problem is then composed of the following steps:

1. Modelling (components): The P&S problem is modelled by identifying the set of relevant features called *components*.
2. Synchronizing components (Domain Theory): Once all the components are created, *synchronisations* between them are created. A synchronisation describes collaborations among the components. A component may require other components to be in a specific state in order to change its state.
3. Problem description: A problem description represents a specific instance of the domain with the initial state of the world (values of the components). The goal is defined as a set of values (called *ValueChoices*) that some components must have in specific instants or periods of time.

A component is represented in DDL3 as a finite automaton (Figure 2) containing the valid state transitions. Each component has an associated *timeline* that represents the evolution of the state of the component over time, limited by a time horizon. Decisions are posted along the timeline of the components either as *ValueChoices* over the set of values of the state variable, or as *consumption/production* activities on a resource.

3. AUTONOMOUS ROVER PROBLEM

A planetary rover scenario will be used throughout the paper as an ongoing example. It is comprised of a rover equipped with a pan-tilt unit (PTU), a stereo camera (mounted on top of the PTU) and an antenna. The rover is able to autonomously navigate the environment, move the PTU, and take pictures and communicate images to a remote orbiter that is not visible for some periods.

To obtain a timeline-based specification of our robotic domain, we consider each of the above elements as a *Component*, each with its own automaton that contains a number of *ValueChoices* that represent the states of the automaton. The states can be described as follows:

- Navigation: Can be in a certain position (At(x, y)) or moving to a certain destination (GoingTo(x, y)).
- PTU: Can assume a PointingAt(pan, tilt) value if pointing in a certain direction, or a MovingTo(pan, tilt) value when it is moving.
- Camera: Can take a picture of a given object in a position <x, y> by requesting a <pan, tilt> for the PTU and a file location in the onboard memory (TakingPicture(file-id, x, y, pan, tilt)). Assumes the value CamIdle() if in an idle state.
- Antenna: Can be transmitting a given file (Communicating(file-id)) or can be idle (CommIdle()).
- Orbiter Visibility: Indicates the visibility of the orbiter. Its possible values (Visible or Not-Visible) represent external constraints for the P&S problem. In particular, these values represent contingent communication opportunities for the rover.

The rover must obey some operative rules for safety reasons. The following *constraints* must hold during the overall mission:

- While the robot is moving the PTU must be in the safe position.
- The robotic platform can take a picture only if the robot is stationary, is in one of the requested locations, and the PTU is pointing in the correct direction.
- Once a picture has been taken, the rover must send the picture to the base station.
- While communicating, the rover must be stationary.
- While communicating, the orbiter must be visible.

The system also has a set of synchronisations between ValueChoices:

- PointingAt(0, 0) value must occur during a GoingTo(x, y) value (C1).
- At(x, y) and PointingAt(pan, tilt) values must occur during a TakingPicture(pic, x, y, pan, tilt) value (C2).
- Communicating(pic) must occur after a TakingPicture(pic, x, y, pan, tilt) (C3).
- At(x, y) value must occur during a Communicating(file) (C4).
- Visible value must occur during a Communicating(file) (C5).

Figure 3 contains a representation of the system, where dotted lines represent synchronisations and normal lines represent transitions.

4. AUTOMATIC PROBLEM GENERATION

The process of generating an automated problem consists of three main steps: extracting relevant information from the model, calculating the number of occurrences of each ValueChoice using linear programming, and generating the problem in the proper format. The following inputs are required to complete these steps:

- A formal model of the system (M)
- The desired resource complexity ($T(R)^{user}$)
- The desired time complexity ($T(t)^{user}$)

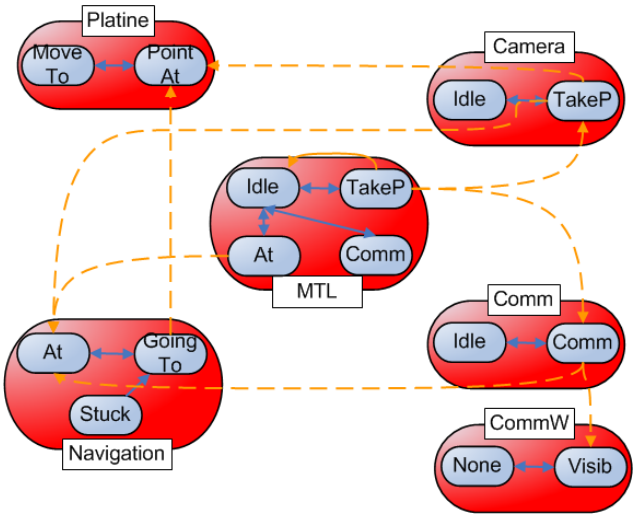


Figure 3: Rover Domain

- Constraints supplied by the user, specifically:
  - The maximum number of occurrences for each ValueChoice ($V_{i,j}^{max(x)}$)
  - The average execution time of each stable ValueChoice ($\bar{V}_{i,j}^{avg}$)
  - The average resource consumption for each ValueChoice for each resource ($V_{i,j}^{R_k}$)

The model represents the system for which the plan will be generated. Several pieces of information are extracted from the model, including its components, component decisions, synchronisations, etc. An inclusive list of these elements are described in Section 4.1. A model of the domain described in Section 3 is used to evaluate RoBen.

The desired resource complexity is given in the range $[0, +\infty)$, where 0 represents the trivial problem, 1 represents full resource consumption, and $T(R) > 1$ represents overconsumption of the resources. The time complexity is also given in the range $[0, +\infty)$, where 0 represents the trivial problem (no goals are assigned), 1 represents full use of the available time, and $T(t) > 1$ represents the assignment of goals whose total time requirement exceeds the available time.

The user can specify constraints that limit the maximum number of occurrences for each ValueChoice in order to generate more realistic problems. Using the model presented in (3) as an example, the user might specify the following constraint:

$$Navigation.StuckAt.numOccurrences = 0 \qquad (1)$$

In this example, the user wanted to prevent the generation of problems in which the rover being stuck at location could be specified as a goal.

Additionally, the user must specify the average execution time for each stable ValueChoice. The worst-case scenario is considered where it is assumed that the execution of any ValueChoice in a component requires the execution of all other ValuesChoices in that component (i.e., cyclic transitions). As a result, the average execution time of each non-trivial ValueChoice is the same for ValueChoices in the same component. In this context, non-stable ValueChoices are called *transitional* ValueChoices. These are ValueChoices whose average execution time cannot be described and

generally has no upper limit. For example, a trivial ValueChoice in the model presented in Section 3 is $Camera.Idle()$. Although a transition within the component $Camera$ may require $Idle()$ to be executed, it should not be considered when calculating the average execution time of the ValueChoices in $Camera$.

Finally, the user must specify the average resource consumption of the ValueChoices. This is to ensure that the number of occurences of the ValueChoices assigned in the generated problem do not consume more resources than are allocated by the resource complexity. The average resource consumption is considered when specifying this constraint, which is described in greater detail in Section 4.2.

The following sections described the three main steps of the automatic problem generator.

4.1 Model Analysis

The analysis of the model consists of two steps: extraction of the model information and an analysis of the synchronisations among the ValuesChoices. The following elements are extracted from the model:

- An automaton for each component showing the transitions among its ValueChoices.
- The resources (consummable and reusable) used by the ValueChoices.
- The synchronisations of ValueChoices among different components.
- The horizon time (H).

In order to consider the fact that the execution time of a ValueChoice increases if it has synchronisations that must also be executed, the propagated time of each stable ValueChoice ($\bar{V}_{i,j}^{prop}$) must be calculated. Because the synchronisations among the ValueChoices may contain cycles, an upper limit on the number times the propagated time of a ValueChoice is updated is taken as the size of the cycle. This value is depicted as $\bar{V}_{i,j}^{up}$ and is calculated using the Tarjan graph cycle algorithm [3]. A description of the algorithm used to calculate the propagated time of every stable ValueChoice is shown in Algorithm 4.1.

Algorithm 1 The time-propagation algorithm

```
1:  procedure MAIN( )
2:      update_list = V̄
3:      for all V̄_{i,j} ∈ V̄ do
4:          V̄^{prop}_{i,j} = V̄^{avg}_{i,j}
5:      while update_list ≠ ∅ do
6:          v = RemoveItem(update_list)
7:          if synchronisations(V̄_{i,j}) ≠ ∅ then
8:              UpdatePropagationTime(v)
9:              for all V̄_{i,j} ∈ V̄ − {v} do
10:                 if (V̄^{up}_{i,j} > 0) & (V̄_{i,j} ∉ update_list) then
11:                     update_list.add(V̄_{i,j})

12: procedure UPDATEPROPAGATIONTIME(V̄_{i,j})
13:     for all v ∈ synchronisations(V̄_{i,j}) do
14:         V̄^{prop}_{i,j} += v^{prop}
15:     V̄^{up}_{i,j} −= 1
```

4.2 ValueChoice Occurrence Assignment

The automata describing the model of the timeline system can be transformed to a non-deterministic Turing machine. By taking this into consideration, an analysis of the time-complexity ($T(t)$) of the model can be performed, which is equal to $2^{O(t(n))}$ [15]. For the model, the length of the chain is equal to the time required to execute a set of goals, divided by the time to the horizon in order to make it proportional to the available time. This formulation is shown in (2).

$$T(t) = 2^{\frac{\sum_{j=i}^{\bar{V}_i^{num}} \bar{V}_{i,j}^{prop} \cdot \bar{V}_{i,j}^{x}}{H}} - 1 \tag{2}$$

where H is the time until the horizon is reached (defined in the model).

The number of times each value of each component must be executed in the generated problem (i.e., $V_{i,j}^{x}$ for every value in every component) was calculated using integer linear programming (ILP). The goal of this approach was to maximise the total propagated time required by the assigned $V_{i,j}^{x}$, subject to constraints derived from the desired value complexity ($T(t)^{user}$) and desired resource complexity ($T(R)^{user}$) input by the user. The special case of ILP is required over standard linear programming (LP) because all $V_{i,j}^{x}$ must belong to the set of natural numbers ($\mathbb{N}$). The formulation of the ILP is shown in (3)–(5).

Maximise:

$$z = \sum_{i=1}^{C^{num}} \sum_{j=1}^{\bar{V}_i^{num}} \bar{V}_{i,j}^{prop} \cdot \bar{V}_{i,j}^{x} \tag{3}$$

Subject to:

$$\sum_{j=1}^{\bar{V}_1^{num}} \bar{V}_{1,j}^{prop} \cdot \bar{V}_{1,j}^{x} \leq \log_2[T(t)^{user} + 1] \cdot H \tag{4}$$

$$\sum_{j=1}^{\bar{V}_2^{num}} \bar{V}_{2,j}^{prop} \cdot \bar{V}_{2,j}^{x} \leq \log_2[T(t)^{user} + 1] \cdot H$$

$$\vdots$$

$$\sum_{j=1}^{\bar{V}_{C^{num}}^{num}} \bar{V}_{C^{num},j}^{prop} \cdot \bar{V}_{C^{num},j}^{x} \leq \log_2[T(t)^{user} + 1] \cdot H$$

and

$$\sum_{i=1}^{C^{num}} \sum_{j=1}^{V_i^{num}} V_{i,j}^{R_1} \cdot V_{i,j}^{x} \leq T(R)^{user} \cdot R_1^{max} \tag{5}$$

$$\sum_{i=1}^{C^{num}} \sum_{j=1}^{V_i^{num}} V_{i,j}^{R_2} \cdot V_{i,j}^{x} \leq T(R)^{user} \cdot R_2^{max}$$

$$\vdots$$

$$\sum_{i=1}^{C^{num}} \sum_{j=1}^{V_i^{num}} V_{i,j}^{R_{R^{num}}} \cdot V_{i,j}^{x} \leq T(R)^{user} \cdot R_{R^{num}}^{max}$$

Where:

$$V_{i,j}^{x} \in \{0, 1, 2, \ldots, V_{i,j}^{max(x)}\}$$

The value complexity constraints in (4) are a reconfiguration of (2) for each component. These constraints prevent the value complexity of each component ($T(t)_i^{prob}$) from exceeding the desired

value complexity input by the user ($T(t)^{user}$). The overall value complexity of the generated problem is simply taken as

$$T(t)^{prob} = \frac{\sum_{i=1}^{C^{num}} T(t)_i^{prob}}{C^{num}}, \quad (6)$$

which is the average complexity over all the components. The resource complexity constraints in (5) prevent the assignment of $\bar{V}_{i,j}^{x}$ that would cause overconsumption for any resource, which is the product of the desired resource complexity input by the user and the maximum capacity of each resource, as defined in the model.

The ILP was solved using the open source GNU Linear Programming Kit (GLPK)[1]. This solver uses an optimized version of the branch and bound method. The output of the solver is passed to the problem generator described in Section 4.3.

4.3 Problem Generation

The output of the ILP system is used to generate the problem in PDL. Each of the occurrences of each ValueChoice, adds a goal to the file and the parameter values are filled in invoking the special procedures. The following paragraph shows a goal generated by RoBen consisting of taking a picture.

```
g20 <goal> Camera.camera.TakingPicture(?file
_id1 = 1, ?x1 = 2, ?y1 = 1, ?pan1 = 10, ?tilt1
= 40) AT [0, +INF] [1, +INF] [1, 100];
```

The initial conditions of the system are also represented in the PDL file via facts that, in opposition to goals, do not need to be justified by the planner. Therefore, RoBen establishes as initial conditions default values chosen from the domain. The PDL file, together with the DDL represent the inputs to be passed to the planner.

5. PROBLEM EVALUATION

In order to validate the heuristic, $T(t)^{prob}$ can be compared with the performance of a planner to understand whether an increasing value of $T(t)^{prob}$ also represents a more complex search space with less solutions. Increasing this value represents a higher number of goals that should reduce the potential number of solutions. We are also interested in understanding the relation between $T(t)^{prob}$ and the final percentage of time demanded for each timeline. We have performed several tests using the APSI planner [2], consisting on the generation of problems with increasing $T(t)^{prob}$ in the range $[0.1 - 1]$, limiting the execution time to 30 minutes for the rover domain presented in Section 3.

The constraints defined for the system are:

- One constraint of type (4) for each component of the domain
- Semantic constraints to define the list of transitional states: $\bar{V}$ = {Navigation: (GoingTo, StuckAt), Platine: (MovingTo), Camera: (CamIdle), Antenna: (CommIdle), CommunicationVW: (All), MissionTimeline: (All)}

The results generated by RoBen are shown in Table 1.

It is important to remark that the error shown is inherent to the linear programing. The fact that the ILP tries to allocate an integer number of tasks that consumes a total time smaller or equal than the maximum time available (the Horizon) limits the number of solutions. This constraint becomes critical in case the time required by a ValueChoice is close to the Horizon time or $T(t)^{user}$ is too small. An example of the last situation can be observed in the table for $T(t)^{user} = 0.1$. The ILP is not able to generate a problem close to this complexity because the allocation of one single $V_{i,j}$ for almost all components makes $T(t)^{prob} > T(t)^{user}$.

Figure 4 represents the number of goals and variables relative to each of the problems generated. The results of the executions displayed in Figure 5 show an increasing complexity in terms of time required by APPlanner to solve the problem directly related to the increase of $T(t)^{prob}$. Notice that the planner was not able to find a solution for $T(t)^{prob} = 0.9$ and $T(t)^{prob} = 1$ in less that 30 minutes. Early analysis of these results lead us to think that the heuristic based on $T(t)$ produce problems with appropriate complexities.

[1]http://www.gnu.org/software/glpk/

$T(t)^{user}$	Navi-At	Platine-PointAt	Camera-TakePic	Antenna-Comm	Error (%)
0	0	0	0	0	0
0.1	0	2	0	0	82
0.2	1	5	0	0	57.75
0.3	1	7	1	1	28.5
0.4	2	9	1	0	30
0.5	2	11	2	2	12.7
0.6	3	13	2	2	14.7
0.7	3	15	2	2	22.85
0.8	4	16	2	2	23.63
0.9	4	18	3	3	10.75
1	5	20	3	3	9.85

Table 1: ValueChoices ocurrence assignment

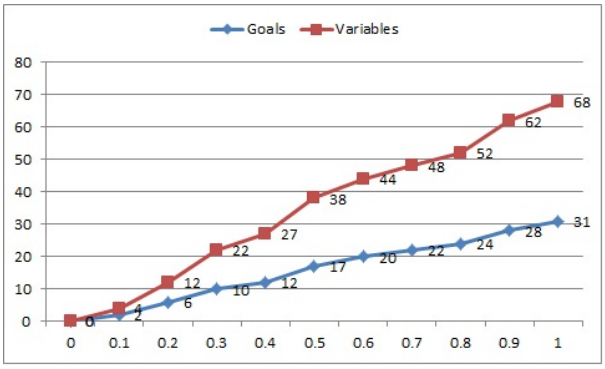


Figure 4: Number of goals and variables respect complexity

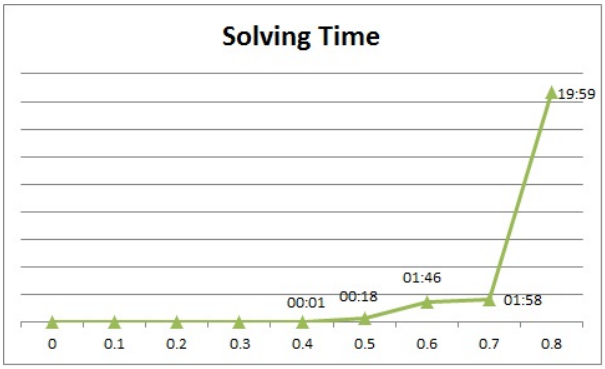


Figure 5: Planner solving time respect complexity

6. CONCLUSIONS AND FUTURE WORK

In this paper we described a benchmarking tool called RoBen. The main objective of RoBen is to provide a means to evaluate and compare automated planners. RoBen, given a specific domain as

input, generates sets of problems which can be used to evaluate the quality of P&S algorithms.

Regarding the results obtained, it is important to remark that it turned out to be a difficult enterprise to generate actual valid plans due to hidden incongruencies in constraints that were preventing the planner in finding a solution. After some tuning of the PDLs, it was possible to find solutions, but it requires some knowledge of the domain that has not yet been automated. It was also difficult to evaluate the results provided by the planner apart from the execution time. Due to the fact that the planner is generating flexible timelines and that the execution of some $V_{i,j}$ cannot be estimated, like it happens for most of the $\tilde{V}$ ValueChoices, it is difficult to compare the plan with $T(t)^{prob}$. Presently, the analysis is focused on the planning time, but random assigments of execution times to all the $V_{i,j}$ in the domain might provide new indicators. However, they must be added carefully in order to avoid incongruencies or 0-solution search spaces.

Future work will consider different alternative evolutions of RoBen. A first direction is to introduce further metrics in order to evaluate the completeness of a benchmark set. In other words, given a domain model, the objective is to verify that all the aspects in the domain are covered and stressed (i.e., according to a set of requirements). Regarding the domain language, the results obtained so far can lead to further evolution of the modeling languages. In fact, as a side effect of the empirical evaluation, we noticed some limitations (and possible extensions) of DDL3 and PDL. As an example, it would be convenient to add semantic information, at least in the domain language, to provide meta-information about states and constraints such as average execution time, type of state (error, stable or transitional), etc.

Acknowledgments. This research has been co-funded by the Networking/Partnering Initiative (NPI) in collaboration between ESA-ESOC and TU Darmstadt. It also receives support from the German Research Foundation (DFG) within the Research Training Group 1362 “Cooperative, adaptive and responsive monitoring in mixed mode environments". The authors would like to thank Simone Fratini for his valuable support

7. REFERENCES

[1] J. Bresina, A. Jónsson, P. Morris, and K. Rajan. Mixed-initiative activity planning for Mars rovers. In *Proceedings of the 19th international joint conference on Artificial intelligence*, IJCAI'05, pages 1709–1710, San Francisco, CA, USA, 2005. Morgan Kaufmann Publishers Inc.

[2] A. Cesta and S. Fratini. The timeline representation framework as a planning and scheduling software development environment. In *In PlanSIG-08, Proceedings of the 27th Workshop of the UK Planning and Scheduling Special Interest Group*, 2008.

[3] B. V. Cherkassky, K. St, and A. V. Goldberg. Negative-cycle detection algorithms. *Mathematical Programming*, 85:349–363, 1996.

[4] S. Chien, R. Doyle, A. Davies, A. Jonsson, and R. Lorenz. The Future of AI in Space. *Intelligent Systems, IEEE*, 21(4):64 –69, july-aug. 2006.

[5] S. Chien, R. Knight, A. Stechert, R. Sherwood, and G. Rabideau. Using Iterative Repair to Improve Responsiveness of Planning and Scheduling. In *Proceedings of the Fifth International Conference on Artificial Intelligence Planning and Scheduling*, pages 300–307, 2000.

[6] S. Chien, R. Sherwood, D. Tran, B. Cichy, G. Rabideau, R. Castaño, A. Davies, D. Mandl, S. Frye, B. Trout, J. D'Agostino, S. Shulman, D. Boyer, S. Hayden, A. Sweet, and S. Christa. Lessons learned from autonomous sciencecraft experiment. In *Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems*, AAMAS '05, pages 11–18, New York, NY, USA, 2005. ACM.

[7] A. G. Davies, S. Chien, T. Doggett, F. Ip, and R. C. no. Improving Mission Survivability and Science Return with Onboard Autonomy. In *In Proc. of 4th International Planetary Probe Workshop (IPPW)*, 2006.

[8] ECSS. *ECSS-Q-80-04 – Guidelines for Software Metrication Programme Definition and Implementation*. European Cooperation for Space Standardization (ECSS), February 2006.

[9] ECSS. *ECSS-Q-ST-80C – Space product assurance - Software product assurance*. European Cooperation for Space Standardization (ECSS), March 2009.

[10] S. Fratini, F. Pecora, and A. Cesta. Unifying Planning and Scheduling as Timelines in a Component-Based Perspective. *Archives of Control Sciences*, 18(2):231–271, 2008.

[11] D. Long and M. Fox. The International Planning Competition Series and Empirical Evaluation of AI Planning Systems, 2006.

[12] N. Muscettola. HSTS: Integrating Planning and Scheduling. In M. Zweben and M. S. Fox, editors, *Intelligent Scheduling*, pages 169–212. Morgan Kaufmann, 1994.

[13] N. Muscettola, P. P. Nayak, B. Pell, and B. C. Williams. Remote Agent: to boldly go where no AI system has gone before. *Artificial Intelligence*, 103:5–47, August 1998.

[14] A. Orlandini, A. Finzi, A. Cesta, S. Fratini, and E. Tronci. Enriching APSI with Validation Capabilities: The KEEN environment and its use in Robotics. In *Proceedings of the 11th ESA Symposium on Advanced Space Technologies in Robotics and Automation (ASTRA)*, 2011.

[15] M. Sipser. *Introduction to the theory of computation*. Computer Science Series. Thomson Course Technology, 2006.

APPENDIX – Nomenclature

C_i	Component i
C^{num}	Number of components
H	Horizon time
M	Model
P	Planning tool
V	Set of all ValueChoices for all components
$V_{i,j}$	ValueChoice j of C_i
$V_{i,j}^{max(x)}$	Max number of times $V_{i,j}$ can be executed
V_i^{num}	Number of ValueChoices in C_i
$V_{i,j}^{R_k}$	Average quantity of R_k consumed by $V_{i,j}$
$V_{i,j}^{sync}$	Number of ValueChoices syncronised with $V_{i,j}$
$V_{i,j}^{x}$	Number of times $V_{i,j}$ must be executed
$\bar{V}$	Set of all stable ValueChoices for all components
$\bar{V}_{i,j}$	ValueChoice j of C_i (stable)
$\bar{V}_{i,j}^{avg}$	Average time required to execute $V_{i,j}$
$\bar{V}_i^{num}$	Number of stable ValueChoices in C_i
$\bar{V}_{i,j}^{prop}$	Propagated time required to execute $V_{i,j}$
$\bar{V}_{i,j}^{up}$	Number of times $\bar{V}_{i,j}^{prop}$ can be updated
$\tilde{V}$	Set of all transitional ValueChoices for all components
$\tilde{V}_{i,j}$	ValueChoice j of C_i (transitional)
R_k	Resource k
R_k^{max}	Maximum capacity of R_k
R^{num}	Number of resources
$T(R)^{prob}$	Resource complexity of the generated problem
$T(R)^{user}$	Resource complexity input by the user
$T(t)^{prob}$	Time complexity of the generated problem
$T(t)_i^{prob}$	Time complexity of C_i in the generated problem
$T(t)^{user}$	Time complexity input by the user

Measures for UGV to UGV Collaboration

Michael S. Del Rose
U.S. Army RDECOM-TARDEC
6501 East Eleven Mile Road
Warren, MI 48397-5000
586.282.6242
mike.delrose@us.army.mil

Anthony Finn
University of South Wales
Mawson Creek Campus, W2-46
South Australia, Australia
+61.8.830.25703
anthony.finn@unisa.edu.au

Robert T. Kania
U.S. Army RDECOM-TARDEC
6501 East Eleven Mile Road
Warren, MI 48397-5000
586.282.5696
robert.t.kania.civ@mail.mil

ABSTRACT
Ground robotic vehicles are continuing to improve in intelligence, mobility, and reliability. Today, more than 3000 ground robotic vehicles are being used by the U. S. Army in the field. These vehicles' duties range from vehicle security to IED detection and neutralization. However, the current operation of ground robotic vehicles are remote control and tele-operational. Underdeveloped adaptive and contextual reasoning algorithms and testing methodologies are limiting their abilities to operate more autonomously. An evaluation framework and a set of metrics need to be developed to enable the research results and value in algorithms to be assessed. In this paper, a simple measure of collaboration between Unmanned Ground Vehicles (UGV) is introduced. The measure is designed to be simple enough to test most all UGVs against. Case studies are used as examples.

Keywords
Collaboration metrics, unmanned ground vehicles, autonomy metrics

1. INTRODUCTION
The capabilities of an Unmanned Ground Vehicles (UGV) are increasing in terms of intelligent mission execution and mobility, thus reducing the operator's need for operational intervention. However, the current U.S. Army-fielded UGVs are limited by their reliance upon either remote control or tele-operation. The limited progress in this area is, in part, due to an inability to accurately characterize – through comparative measurement and test – aspects of the robotic vehicles' intelligence.

This paper discusses a measure of collaboration between UGVs. Measures for robotic vehicles are discussed in the section 2 and the proposed collaboration measures defined in section 3. Section 4 describes the MAGIC 2010 competition collaboration levels of each of the competing teams. Section 5 concludes this paper.

(c) 2012 Association for Computing Machinery. ACM acknowledges that this contribution was authored or co-authored by a contractor or affiliate of the U.S. Government. As such, the Government retains a nonexclusive, royalty-free right to publish or reproduce this article, or to allow others to do so, for Government purposes only. PerMIS'12, March 20-22, 2012, College Park, MD, USA. Copyright © 2012 ACM 978-1-4503-1126-7-3/22/12...$10.00

2. BACKGROUND
A robotic vehicle consists of two control loops at the highest level: A supervisory control loop and a sensor-actuator control loop. The supervisory control loop receives the overarching mission and directs the robotic vehicle to move, lift, deploy, or some other function. This loop is mainly controlled by a human operator and a Ground Control Station (GCS). The sensor-actuator control loops main function is to take the commands coming from the supervisor control loop and make the vehicle move, lift, deploy or perform some action. Feedback is necessary in both control loops to better understand whether or not directions are being carried out. At a lower level, there are many other control loops; for example, actuator control, rate control, trajectory control, mission control, and others. For vehicles to be autonomous, it must use these controls to gather data about its environment based on sensors than act according to mission requirements. At present, however, the UGV generally combines both the human input (e.g. from a mission plan) and that from its own sensors before acting on the world, providing a natural decomposition for measuring the performance of robotic vehicles against levels of autonomy. This is captured using the well-known levels of supervisor-UGV collaboration that vary from one to ten against the Sheridan-Verplank scale [[1]].

UGV to UGV cooperation is described using five levels from Cummings [2]. As a result, when humans (supervisors) collaborate with multiple UGVs, the nature of each human-UGV interaction is similar to the single robotic vehicle case above except the interaction with the robotic vehicle is via another UGV and not directly between the GCS and the UGV. When the robotic vehicle does not have any capacity to collaborate (levels 1-4 as defined in section 3 of this paper), the levels of collaboration can vary from one to ten against the Sheridan-Verplank scale. Alternatively, when there is UGV to UGV collaboration (levels 5-7 in this paper), the human-UGV interactions must exist only at the higher levels of the Sheridan-Verplank scale.

As one of the main issues for the interaction of the UGVs and the human supervisor is the impact of the human decision-making process on the system performance, this duality in the levels of automation presents a problem for the UGV designer. In single vehicles, there are multiple discrete levels of autonomy which can theoretically allow direct comparisons of the system's overall performance to be made against one another. However, when there are networks of vehicles, the problem space becomes significantly larger and more complex. Consequently, when designing a support system that allows the humans to collaborate with multiple vehicles it is necessary to assess the impact of the levels of human-UGV collaboration, the effects of various levels of collaboration between the UGVs and the indirect influences of

interaction between the automation schemes. Predominantly, this is because if the UGV mission is complex or the automation is not highly reliable, the cooperative (or even individual) UGVs may perform poorer than one with no automated assistance.

In addition to the degree of individual and cooperative autonomy, there is another axis commonly used to measure autonomy: mission complexity. Mission complexity is essentially represented by the number of mission-level activities that can be undertaken by the UGV, regardless of whether they are undertaken by UGV, humans, or some combination thereof. It recognizes that the human-UGV enterprise is the system and measures its functional capability holistically. This axis is considerably less well-defined than those pertaining to single and multi-UGV collaboration as it is itself a function of many complex and interdependent variables, such as the degree of environmental difficulty and the complexity of the mission within this environment. Nevertheless, we may characterize the degree of environmental difficulty against metrics that include static and dynamic elements. Static elements include a terrain perspective of traverse-ability, soil type, occupancy, etc. Dynamic elements may include items such as the number, density and type of objects in the environment and the frequency or rate at which they change or move. The framework might also characterize the environment in terms of luminescence/visibility and the electromagnetic spectrum, as well as operational considerations such as the presence of threats or decoys and whether the environment is rural, urban, semi-urban, etc. It should also take into account the weather effects.

Due to a historical absence of universally agreed and quantifiable metrics for UGV performance evaluation, most of the research results associated with artificial intelligence and robotics have been in the form of specific missions and demonstrations rather than experiments with data that is quantitative or fiducially referenced. Moreover, as the previous section points out, complex concepts often require multiple measures to provide valid information as no single measure or methodology exists to satisfactorily assess the overall effectiveness of UGV technology. As a result, to link the performance of a system as a whole to the performance of its components, any metrics must correspond to critical tasks. Collaboration is one such task.

3. COLLABORATION MEASURES

The Collaboration measures are formed to better define and understand the current state of the art in UGV to UGV collaboration as well as give insight into where future efforts should be focused. Such measures need to be well-defined, testable and relevant. For such measures to be testable, they should also be discrete so that dependencies on other factors are not coupled. Also, the less subjective these measures, the better for the community as a whole. In this context, the goal of collaborating UGVs is to have them work together to solve problems and complete the defied mission in adequate time without the need of operator intervention. Figure 1 shows the potential flow of messages between the GCS and other robots.

This flow is what we are measuring for collaboration in known and unknown situations for a given mission.

The following definitions describe increasing levels of collaboration between UGVs while simultaneously trying to reduce the complexity involved if testing were to take place. The higher the level of collaboration, the greater the communication between robots and the lesser the need for human involvement in the decision-making.

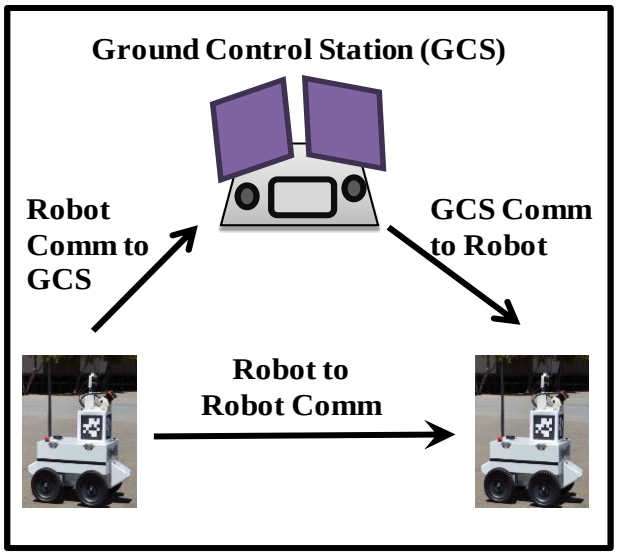


Figure 1. Layout of communications for collaborative robotic vehicles.

- **LEVEL 1:** All information is thru sight of human. No communication is thru a GCS other than direct control commands for the vehicles. The human controls analysis of information based on direct sight. Position and tasks are thru the human.
- **LEVEL 2:** Some information is passed between the GCS and the vehicle. Other information is directly thru the sight of the human. Data captured and sent to the GCS is generally UGV position. The human provides the analysis and decision of robot position and robot tasks. The human controls updates to the vehicles.
- **LEVEL 3:** Data is completely transmitted to the GCS and held there. Vehicle position, tasks, and/or capabilities are held in the GCS. The GCS presents and helps analyze the data for the human. Decision come from the human, then transmits to vehicles to carry out.
- **LEVEL 4:** Data is completely transmitted to the GCS and held there. Transmitted data includes position, tasks, and/or capabilities. More analysis and some decision making is on the GCS, but the human operator oversees. The GCS or the human transmits re-tasking or re-positioning to vehicles to carry out.
- **LEVEL 5:** Most of the vehicle information is passed to the GCS, but some is vehicle to vehicle; this most likely includes position data. The GCS has knowledge of position, tasks, and capabilities of all vehicles. Vehicles may hold some or all of the same information. Analysis is performed on the vehicle with final decision coming from the GCS. Communication of re-tasking and re-positioning is from the GCS or from the vehicle.
- **LEVEL 6:** Most of the information is passed vehicle to vehicle with the GCS also getting the information. Vehicles keep track of others info through their own interpretations of

positions and tasks. Analysis and decisions are between vehicles with the GCS as over watch (giving the human operator the ability to change); thus a global perspective makes adjustments as needed. Vehicles communicate changes in tasks and plans between each other and to the GCS.

- **LEVEL 7:** All information is passed between vehicles. Data of self and others are held in each vehicle with their own interpretation of positions and tasks of others. Vehicles request and decide course of actions between themselves, without the need for a GCS. Communication of tasks is sorted out between vehicles.

The collaboration levels can be summarized in matrix form, shown in Table 1. This matrix is divided into the core activities in collaboration between vehicles.
There are certain assumptions that should be made to reduce the amount of complexity in these measures.

Table 1. Collaboration level related to core activities in UGV to UGV combined work.

	COMM data	Hold data/world model	Analize data	Decisions	COMM decisions	Overwatch
1	none	Human	Human	Human	Human	Human
2	Human with some SMI	Human with some SMI	Human	Human	Human	Human
3	Vehicle to SMI	SMI	SMI/Human	Human	Human	Human
4	Vehicle to SMI	SMI	SMI/Human	SMI/Human	SMI/Human	Human
5	Vehicle to SMI with some to vehicle	Vehicle/SMI	Vehicle/SMI	SMI	SMI	Human
6	Vehicle to Vehicle and SMI	Vehicle/SMI	Vehicle	Vehicle	Vehicle	SMI/Human
7	Vehicle to Vehicle	Vehicle	Vehicle	Vehicle	Vehicle	None

Assumption 1: Mission complexity is separated from the collaboration measures. To accurately define collaboration levels, it must be separated from other core functions. In this assumption, mission complexity is the heart of what we are trying to break down with the core mission capabilities discussed in section 2.

Assumption 2: Communications is separated from collaboration level. Although collaboration between UGVs is heavily reliant on communication and communication paths, the physical communication levels as described in section 2 should be separate measures.

4. MAGIC 2010 CASE STUDY

The MAGIC 2010 competition was a co-sponsored event between U.S. Army Research Development and Engineering Command (RDECOM) and the Australian Defence Science and Technology Organization (DSTO) with the main challenges of:

- Reducing the number of operators to robots,
- Imbedding individual and group behaviours in teams of heterogeneous mobile platforms,
- Demonstrating dynamic allocation and re-planning of robot resources,
- And coordinating all assets in a bandwidth-limited urban environment.

The course consisted of a half km square indoor and outdoor environment to include buildings, animal pens and stalls on the Adelaide Show Grounds in Adelaide Australia. Both stationary and moving Objects Of Interest (OOI) were deployed within the environment. The goal was for a team of robots to accurately and completely map the course and identify and neutralize OOI's, using a laser pointer. No more than two operators were allowed and there had to be at least three robots on the course at any time to avoid penalties. For further details, please refer to Finn et al [4].

The following collaboration measures were taken from the five competing teams reports, presentations, individual discussions with team members, and observations from both the down selection process and the competition.

4.1 Team MAGICian

This team was lead by University of Western Australia out of Perth Australia. Originally they had planned to send local map data to each vehicle directly and to the GCS as well as path planning updates based on frontier exploring regions. However, they did not accomplish this. They had no hand-off of position between OOI and other vehicles without human involvement. In the end, the operator controls all the information. The GCS presents the information in a basic level to help analyze. The GCS holds position data, but not tasks. See Figure 2. They were assessed a collaboration level of 2/3.

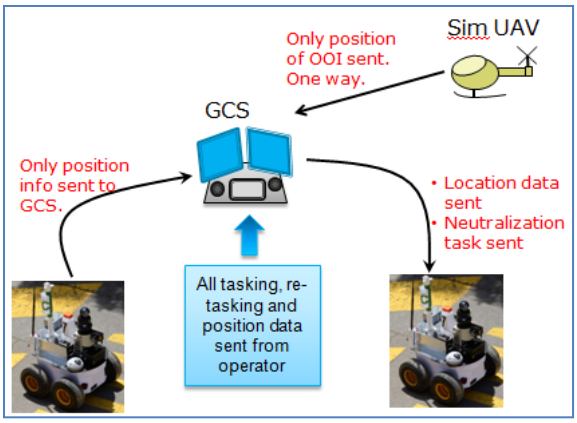


Figure 3. Collaborative communication layout of Team MAGICian

4.2 Team Cappadocia

Team Cappadocia was lead by Aselsan Corporation out of Ankara Turkey. They automatically sent global position updates of robotics vehicles to the GCS. Their path planning was computed by the GCS, but required the operator to confirm and send. There was no hand-off of position data between vehicles except through the GCS. The operator controlled all tasking. The GCS processed the data to a high level to help the operators analyze it. The GCS held the position data of all vehicles and transmitted them at regular intervals to the vehicles. From this, each vehicle held some of the position data of other vehicles. See Figure 3. The collaboration level for their work was assessed at 3/4.

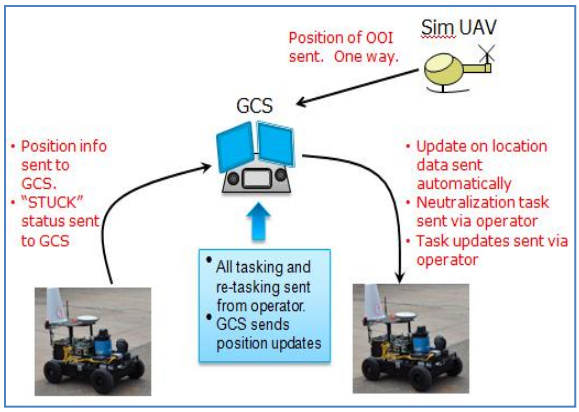


Figure 4. Collaboration communication layout of Team Cappadocia

4.3 Team UPenn

Team UPenn was lead by the University of Pennsylvania. They had vehicle information sent to the GCS automatically. Path planning was computed by the vehicle and the GCS. Verification of OOIs were sent to the GCS automatically. Pause commands during neutralization were sent to the GCS and then to the vehicles without operator intervention. The operator controlled the tasking. The GCS processed the information to a level to help understand and make decisions. The GCS held position information. See Figure 4. With this, Team UPENN had a collaboration level of 4.

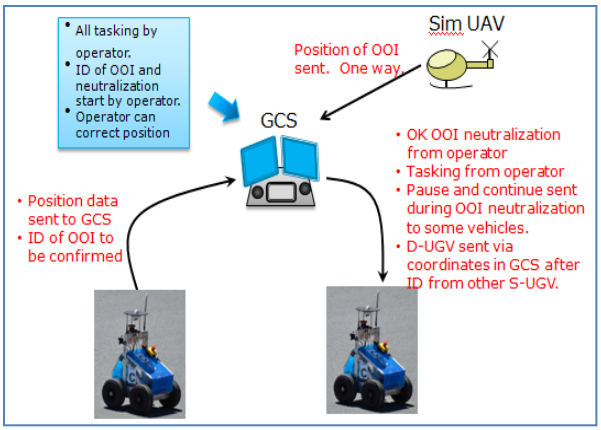


Figure 2. Collaboration communication layout for Team UPenn.

4.4 Team RASR

Team RASR, lead by Robotic Research LLC out of Gaithersburg MD, had automatic updates of vehicle position sent to and from the GCS. Path planning was computed by vehicles and sent to the GCS for further analysis and tasking. Each vehicle holds information of other vehicles and at times coordinates help from others, via "contracts", directly. Hand-off of mobile OOIs were coordinated between vehicles. The operator controls most tasking. The GCS processes to a level to help analyze information and makes some decisions. See Figure 5. Team RASR was assessed a collaboration level of 5.

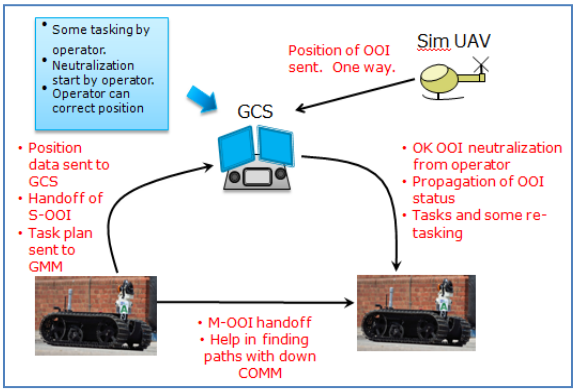


Figure 5. Collaboration communication layout for Team RASR.

4.5 Team Michigan

Team Michigan was lead by the University of Michigan. They sent automatic updates to and from the GCS and vehicles. When vehicles were with a certain distance from each other, they would sync up, updating their position and closing loops. Path planning was computed by the vehicles. The operator did some tasking of the robots. The GCS processed to a level to analyze information

and make many of the decisions. Vehicles held position and tasks of other vehicles. See Figure 6. Team Michigan's collaboration level was assessed at a level 5/6.

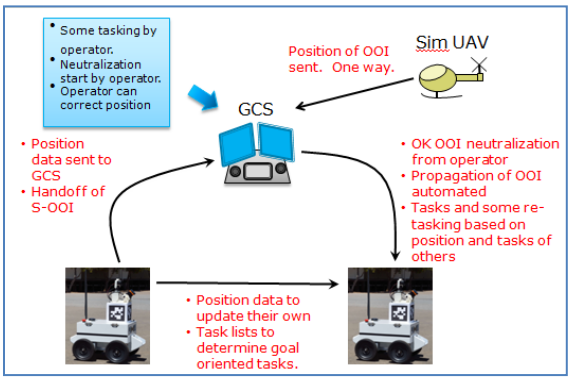


Figure 6. Collaboration communication layout of Team Michigan.

5. ARMY ROBOTIC PROGRAM COLLABORATIVE ASSESSMENT

The U.S. Army has large investment in research activities for unmanned robotic vehicles. However, with the difficulty in developing reliable autonomous ground vehicles given all the unusual environments that the vehicles may be deployed in, previous investments in collaborative UGV missions were minimal. Most Army research dealt with the single vehicle autonomous problem like safety around people, mobility in many environments, communication with command centers, and object recognition. This section identifies two programs that have had collaboration between vehicles as a deliverable: Near Autonomous Unmanned Systems (NAUS) Advanced Technology Objective (ATO) and the Convoy Active Safety Technology (CAST) program.

5.1 Collaboration associated with the NAUS ATO

The purpose of the NAUS ATO was to develop and demonstrate key robotics technologies to reduce robotic risks and increase the utility of future unmanned systems. A key robotic technology that was demonstrated was UGV tactical formation control. This lends itself well with UGV to UGV collaboration.

The NAUS ATO developed vehicle to vehicle communication of each vehicles main position in a tactical formation, without the need of a human operator. This was also sent to the command station. Vehicles would adjust speed and position based on main vehicle with the offset desired to keep formation position. All vehicle speed and position data was passed between vehicles autonomously however control of position and formation was from the main vehicle. No other information was passed nor was there any other tasks requested or performed other than changing speed and position. The system required human intervention when the mission formation changed, as an over-watch to the vehicles, or when there was an error. The main vehicle acted like the control station, so there were no vehicle to vehicle collaboration other than what the main vehicle required. This type of collaboration was assessed as level 4/5.

5.2 Collaboration associated with the CAST project

The CAST project was designed to create an autonomous vehicle convoy, where the lead vehicle is placed anywhere in the convoy order. This lead vehicle directs the group of vehicles in the convoy. The most recent spinout was to increase safety around people, reduce road fatigue of soldiers, improve on situational awareness, and help detect IEDs or other threats. It has been tested in multiple weather conditions to include rain, fog, dust and snow. The collaboration between vehicles is paramount in the successful completion of its mission

There are two modes for the lead vehicle: autonomous and manned. If the lead vehicle is manned by a human driver then the vehicles display vehicle to vehicle communication of position autonomously. The following vehicles are dependent on this and other vehicles communication of position to identify the correct position and speed they must be at. All the decisions are vehicle to vehicle. If the lead vehicle is autonomous then the lead vehicle gets a prior information of the path to follow as well as position and speed of the following vehicles. This information is used to direct the following vehicles when new positions are determined, like in the case of a tighter convoy requirement in certain sections of the terrain. All decisions are vehicle to vehicle. No other information is passed to vehicles; capabilities of each vehicle are assumed the same. The assessed level of collaboration is level 6.

6. SUMMARY AND FUTURE WORK

The goal of this paper was to provide an easy and sufficient method for identifying levels of UGV to UGV collaboration. These levels of collaboration are easily measured for testing, simple enough to exclude multiple dependence on other core functions in robotics (like mobility, mission complexity, environment, or reporting), and sufficiently useable for comparing with other vehicles. These measures are not to be compared along the core robotic functions, but should be tested within the core functions as a constant. The user of these measures should identify the test, perform using multiple robots and evaluate on their collaboration. When comparing with other collaboration level test, one must be careful to fully understand the details that make up the test (like complexity, weather, terrain, etc.). This could skew the results and show one robotic platform outperforms another when in may not.

Future work on these collaboration levels will be on relating them to other core functions and combining multiple core function levels to describe mission complexity. An autonomous mobility metric is being developed that performs relatively the same way as this collaboration metric; simple tests to show the level of autonomy. Environmental and mission complexity are also important measures when displaying the level of a robotic vehicle, thus should be incorporated with the levels outlined above. They should not be incorporated into the levels. In the end, several metrics to determine core functions of a robotic vehicle can be combined with relation to a mission. This will give the decision makers a way to assess if a particular platform can achieve the desired mission.

7. REFERENCES

[1] Sheridan T., Verplank W., *Human & Computer Control of Undersea Teleoperators*, Cambridge, MA, Man-Machine Systems Laboratory, Department of Mechanical Engineering, MIT, 1978

[2] Cummings, M.L., *Human Supervisory Control of Swarming Networks*, Proc. 2nd Conference Autonomous Intelligent Networked Systems, Arlington, VA, 2004

[3] Nourbakhsh I.R., Sycara K., Koes M., Young M., Lewis M., Burion S.; *Human-Robot Teaming for Search & Rescue*, Pervasive Computing, IEEE Computer Society, January-March, 2005

[4] Finn A., Jacoff A., Del Rose M., Kania R., Silva U., Bornstein J.; *Evaluating Autonomous Ground-Robotics*, submitted to Journal of Field Robotics

[5] Finn, A., Scheding S., *Developments & Challenges for Autonomous Unmanned Vehicles*, Springer, ISBN-978-3-642-10703-0, March 2010

[6] Huang H-M., *Autonomy Levels for unmanned System (ALFUS) framework*, Proceedings of the 2007 Workshop on Performance Metrics for Intelligent Systems (PerMIS) 2007

2011 Solutions in Perception Challenge Performance Metrics and Results

Jeremy A. Marvel
National Institute of Standards and Technology
100 Bureau Drive, MS 8230
Gaithersburg, MD 20899
jeremy.marvel@nist.gov

Tsai-Hong Hong
National Institute of Standards and Technology
100 Bureau Drive, MS 8230
Gaithersburg, MD 20899
tsai.hong@nist.gov

Elena Messina
National Institute of Standards and Technology
100 Bureau Drive, MS 8230
Gaithersburg, MD 20899
elena.messina@nist.gov

ABSTRACT
The 2011 Solutions in Perception Challenge presented an international collection of teams with the opportunity to develop algorithms that could accurately detect, recognize, and locate in space an arbitrary collection of artifacts. Researchers at the National Institute of Standards and Technology (NIST) generated a series of artifacts synonymous with parts found in industrial settings, a modular fixturing system capable of accurately and precisely positioning the artifacts within a work volume, and a relative pose scoring metric to quantify an algorithm's performance. Teams were presented with training and validation data sets consisting of red-green-blue color images and 3D point cloud data of the artifacts, and the top performers achieved over 70 % accuracy in translation and pose estimation. In this paper we discuss the design of NIST's contributions, and present the teams' results from the Challenge.

Categories and Subject Descriptors
C.4 [**Performance of Systems**]: Performance Attributes; B.8.2 [**Performance and Reliability**]: Performance Analysis and Design Aids; G.1.6 [**Optimization**]: I.5.4 [**Applications**]: Computer Vision

General Terms
Measurement, Documentation, Performance, Experimentation, Verification

Keywords
Ground Truth, 6DOF Metrology, Laser Tracker, Fixtures

1. INTRODUCTION
In late 2010, researchers from Willow Garage proposed a competition, the Solutions in Perception Challenge (SPC), that would help establish what perception problems have been "solved" and to help advance the state of perception algorithms to enable the next generation of robotics applications [1]. The purpose of establishing the SPC was to determine the current state of maturity for robotic perception algorithms. There is a myriad of algorithms that currently exist world-wide for identifying objects and determining their pose (location and orientation with respect to a coordinate frame), yet it is difficult to ascertain whether an algorithm could be applied to a given task or to know with confidence how robust an algorithm actually is. In addition, efforts to develop these algorithms are being duplicated, but there is no convenient way of readily knowing what algorithms have already solved a particular aspect of the perception problem. The SPC seeks to identify the best available perception algorithms that will be documented in the form of open source software to prevent duplication of development efforts and, in turn, accelerate the development of the next generation of perception algorithms.

Robust perception is a key expertise for attaining technological readiness for next generation robotics [2]. For a wide gamut of application tasks, robots will need to reliably identify objects in their environment and determine their location. The vision for this set of challenges is to define desired capabilities for robotic perception that will enable competences for robots in various domains. It is hoped that these challenges will foster consensus and promote innovation in the research community concerning the state of perception technologies (e.g., as described in [3]). Specifically, a given perception problem either has a definitive solution and, as such, continued research presents minimal payback, or said problem remains unresolved, and still requires innovation and improvements.

In this paper we present efforts of the National Institute of Standards and Technology (NIST) in support of the 2011 Challenge. Section 2 outlines the format and stated goals of the 2011 SPC. The artifacts and fixture developed by NIST are described in detail in Section 3, while the evaluation metrics are discussed in Section 4. Section 5 outlines the teams' performances and the results of the SPC.

2. The 2011 SPC
The inaugural SPC event was held during the 2011 IEEE International Conference on Robotics and Automation (ICRA), 9-12 May, 2011, in Shanghai China. The topic of the 2011 Challenge was single and multiple rigid object identification and 6 degree of freedom (6DOF) pose estimation in structured scenes. Competing teams were required to develop algorithms that could "learn" an arbitrary number of objects from the provided 3D point cloud data that had been augmented with the corresponding red-green-blue point color, and then to correctly identify and locate the same objects in a presented scene.

The initial call for participation in January, 2011 attracted over 30 universities, colleges, businesses, and independent researchers to join the competition. Twelve teams officially registered for the competition by submitting team contact information and

This paper is authored by employees of the United States Government and is in the public domain. PerMIS'12, March 20-22, 2012, College Park, MD, USA. ACM 978-1-4503-1126-7-3/22/12

Figure 1. The sixteen machined NIST artifacts used in the 2011 SPC, arranged randomly.

Figure 2. The 6DOF ground truth fixture with an attached sensor and artifact on the rotation plate.

algorithm proposals. Of these, seven teams representing three different nations qualified for participation in the SPC.

Preceding the SPC in Shanghai, competitors had to successfully pass a number of milestone requirements in order to qualify to participate. Milestone 1 was the basis for entry into the Challenge, and consisted of a team's submission of registration information and a brief overview of their algorithm(s) for object training, recognition, and pose estimation. This first threshold was accepted on a rolling deadline basis up until 15 April, 2011. Twelve teams passed the Milestone 1 requirements. For Milestone 2, teams had to submit initial code bases via online repositories for system testing by 15 April, 2011. The competitors' repositories were required to be populated with their relevant source code, which was required to compile and run without errors. The actual performance of the code submissions was not evaluated at this time. Of the initial twelve teams, seven successfully passed Milestone 2. Milestone 3 marked the final submission of source code and documentation on 1 May, 2011. All programs were expected to compile without error and function properly. These programs were then extensively tested by the competitors (each evaluating their own code bases) and NIST researchers using the NIST data set. All seven qualifying teams successfully passed Milestone 3.

Throughout the qualifying period, teams were given sample training and evaluation data sets with corresponding ground truths to test and validate their algorithms. Each team presented a unique solution to the 2011 Challenge, but several trends in technology were evident. For instance, many teams utilized common algorithms to detect and describe image features such as Scaled Invariant Feature Transform (SIFT) and Speeded Up Robust Features (SURF). Some employed machine learning like K-Means clustering and Support Vector Machines (SVM) for artifact recognition, while others utilized more common pattern and feature matching techniques.

3. ARTIFACTS AND DATA SETS

The data sets composing the training and evaluation sets were assembled using 16 machined aluminum artifacts representing commonly-encountered features of manufactured parts (Figure 1). Each artifact was created from a unique computer-aided design (CAD) model, and was augmented with semi-Lambertian, optically-textured decals. The artifacts were first categorized into three classification groups based on their perceived physical features (specifically, height and surface levelness). The artifacts were designed to be congruent with industrial assembly parts like automotive or aircraft components, and could be rigidly fixtured to a low-cost ground truth system (Figure 2).

The ground truth fixture consists of modular, interlocking aluminum components. The base is used to hold the sensor under evaluation with adjustments to vary the sensor's horizontal and vertical offsets relative to the rotation plate mounted via a slew bearing to the aluminum base. The rotation plate contains one set of alignment holes coincident with the plate's center axis (used only for generating the training data set), and four sets of two offset alignment holes positioned at known distances from the center of the plate. Each set of alignment holes accepts both the NIST artifacts and set of mechanical offsets. Each mechanical offset provides an angular offset relative to the surface of the plate as a machined surface for attaching an artifact, and four sets of alignment holes in nominal differential increments of 49.7°, 41.1°, and 33.6° for attaching to the alignment holes in the rotation plate via steel dowel pins. The plate can also be rotated at 10° increments using a ball plunger quick lock mechanism, to produce over 1,400 6DOF positions for every pairing of artifact and mechanical offset. Up to four artifacts can be simultaneously accommodated on the rotation plate. Relative positioning errors of the ground truth fixture can be attributed to the machining process at NIST—which is typically accurate to within about ±0.02 mm per alignment hole—and inaccuracy associated with the slew bearing tolerances.

Evaluation of the submitted algorithms was broken into two distinct rounds. Round 1 consisted of image frames featuring only one artifact at a time, while the frames in Round 2 contained three artifacts each. Both rounds were composed of several sub-runs based on variations in object translation and rotation. Run 1 consisted of only object translations; Run 2 had only object rotations using the four fixture-based alignment holes; and Run 3 had a combination of translations and rotations. A summary of the runs and types of transformations for each Round-Run combination is provided in Table 1. The teams were not aware of the composition of the data sets prior to the competition.

For each run in Round 1, a sample artifact was randomly selected from the three classification groups. Poses compliant with the Run-based transformation restrictions previously discussed were then randomly selected for each run, and the chosen artifacts were each applied to the same subset of transformations. For Round 2, one object from each classification group was randomly selected to form the test group. Random poses compliant were generated

Table 1. Simplified Data Set Schedule

Round	Run	Objects	Trans.	Rot.	Frames
1	1	1	✓		15
1	2	1		✓	24
1	3	1	✓	✓	144
2	1-1	3	✓		24
2	2-1	3		✓	30
2	3-1	3	✓	✓	30
2	1-2	3	✓		24
2	2-2	3		✓	30
2	3-2	3	✓	✓	30
2-1	1-1	3	✓		24
2-1	1-2	3	✓		24

for each run, with each artifact being assigned a different location on the base plate.

Round 1 was composed of one repetition of Runs 1 to 3, while Round 2 had two repetitions of Runs 1 to 3. Two additional repetitions of Run 1 featuring three artifacts randomly selected from only one classification group were also created. In all, there were 399 frames for the contestants to assess. All of the algorithms were evaluated using the same data set, and the scores were tallied in the days preceding the beginning of ICRA 2011.

4. EVALUATION METRICS

Central to NIST's efforts for the 2011 SPC was the development of a metrological basis of determining algorithmic effectiveness for accurate 6DOF pose estimation. Spatial accuracy can be assessed by analyzing the effects of changing artifact poses; by addressing only the relative transformation, the necessity of registering coordinate frames between the ground truth and the sensor under test is eliminated provided the scales are congruent.

Given a 4x4 3D homogeneous transformation matrix, $\boldsymbol{H}$,

$$\boldsymbol{H} = \begin{bmatrix} & \boldsymbol{R} & & \boldsymbol{T} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

the change in orientation is given by $\boldsymbol{R}$, a 3x3 3D rotation matrix, and $\boldsymbol{T}$ is a 3x1 3D translation matrix. Pose estimations can be scored based on the translational and rotational accuracies separately. Assume that $\boldsymbol{H}_{GT}\left(\boldsymbol{R}_{GT,i,j}, \boldsymbol{T}_{GT,i,j}\right)$ is the ground truth pose for object j in frame I from pose $\boldsymbol{P}$ to $\boldsymbol{P}_{GT}$, and that $\hat{\boldsymbol{H}}\left(\hat{\boldsymbol{R}}_{i,j}, \hat{\boldsymbol{T}}_{i,j}\right)$ is the estimated pose for object j in frame i to $\hat{\boldsymbol{P}}$.

The *translation error*, the length difference between the relative translation vectors of the ground truth and system under test for artifact j in frame i, is calculated as

$$\varepsilon_{T,i,j} = \left\| \boldsymbol{T}_{GT,i,j} - \hat{\boldsymbol{T}}_{i,j} \right\|_2, \tag{1}$$

where $\left\| \cdots \right\|_2$ is the 2-norm length. The *translation score* for each object j is normalized based on predefined offset tolerances,

$$t_{i,j} = \begin{cases} 0.0 & \text{if } \varepsilon_{T,i,j} > \varepsilon_{T,\max} \\ 1.0 & \text{if } \varepsilon_{T,i,j} \le \varepsilon_{T,\min} \\ 1.0 - \dfrac{\varepsilon_{T,i,j} - \varepsilon_{T,\min}}{\varepsilon_{T,\max} - \varepsilon_{T,\min}} & \text{otherwise} \end{cases} \tag{2}$$

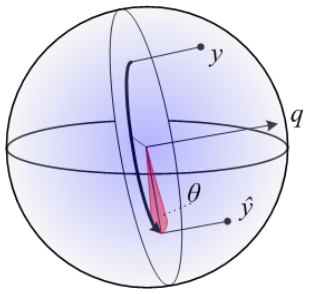


Figure 3. Error magnitude threshold (in red) for the relative axis-angle rotation error, θ.

where $\varepsilon_{T,\min}$ and $\varepsilon_{T,\max}$ are minimum and maximum thresholds for errors in translation. These values were set to 1 cm and 3 cm, respectively, for the 2011 SPC. For all M objects in a given frame, the total *translational frame score* is calculated as

$$s_{T,i} = \frac{1}{M} \sum_{j=1}^{M} t_{i,j} \ . \tag{3}$$

Although the magnitude of translational errors can be summarized by a single normalized scalar, traditional Cartesian rotation representations require separate errors for each axis of rotation. In order to achieve a single rotational score, one can instead use an angle-axis rotation representation, as illustrated in Figure 3 for the rotation from y to $\hat{y}$ about axis q. The scalar angle-axis *rotational error*, $\theta_{i,j}$ for each object j in frame i is computed as

$$\left\| \varepsilon_{R,i,j} \right\|_F^2 = \left\| \boldsymbol{R}_{GT,i,j} - \hat{\boldsymbol{R}}_{i,j} \right\|_F^2 = 6 - 2\left(1 + 2\cos\theta_{i,j}\right) \ge 8 \tag{4}$$

where $\left\| \cdots \right\|_F$ is the Frobenius norm. The *rotation score* for each object in the ground truth is thus equal to

$$r_{i,j} = \begin{cases} 0.0 & \text{if } \theta_{i,j} > \theta_{\max} \\ 1.0 & \text{if } \theta_{i,j} \le \theta_{\min} \\ 1.0 - \dfrac{\theta_{i,j} - \theta_{\min}}{\theta_{\max} - \theta_{\min}} & \text{otherwise} \end{cases} \tag{5}$$

where $\theta_{\min}$ and $\theta_{\max}$ are minimum and maximum thresholds for rotational errors. These were 2º and 20 º, respectively. For all M objects in a frame, the total rotational frame score is thus computed as

$$s_{R,i} = \frac{1}{M} \sum_{j=1}^{M} r_{i,j} \ . \tag{6}$$

5. 2011 CHALLENGE RESULTS

For each frame, a given team's algorithm was scored based on its capacity for correctly recognizing which objects were in a presented scene and accurately locating said objects within the aforementioned tolerances. The recognition scores are presented in Section 5.1, and the pose estimation scores are given in Section 5.2. These scores are then combined in Section 5.3 to establish a team's cumulative performance.

Early testing of the NIST data sets and evaluation metrics was performed on a baseline test system provided by Willow Garage. This system, Textured Object Detection (TOD), is built into

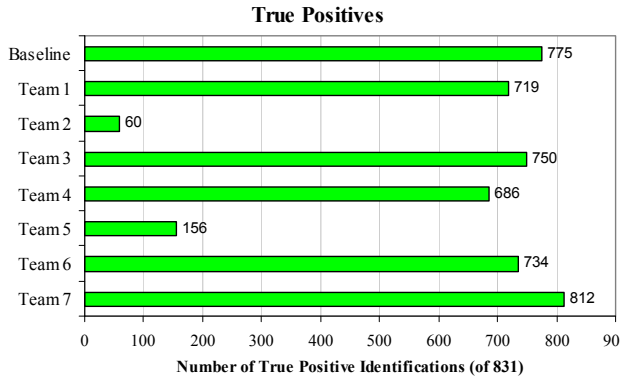


Figure 4. Tabulation of correct artifact identifications.

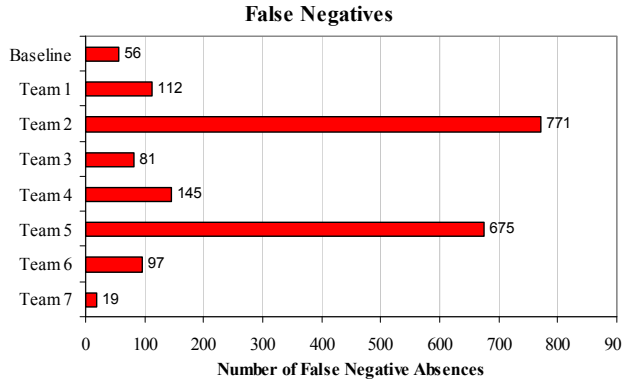


Figure 5. Tabulation of missed artifact identifications.

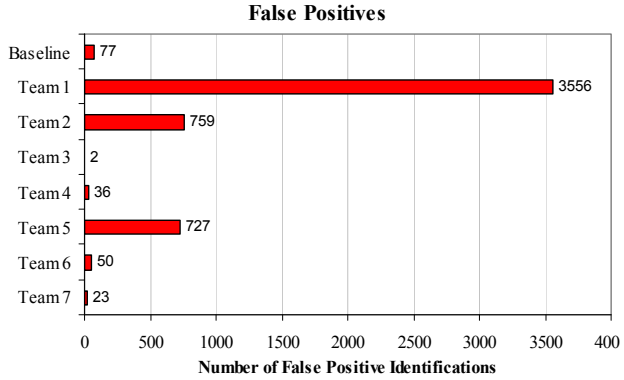


Figure 6. Tabulation of incorrect artifact identifications.

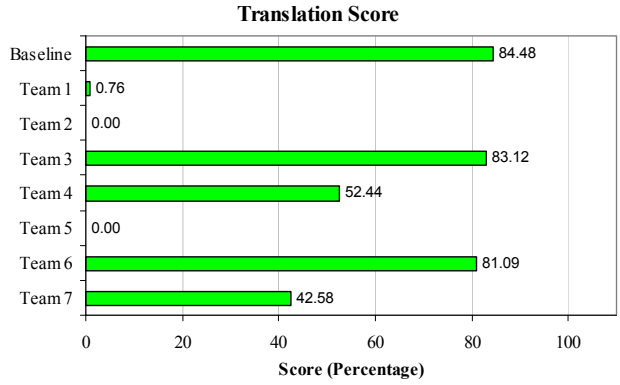


Figure 7. Final translation scores over all 399 frames.

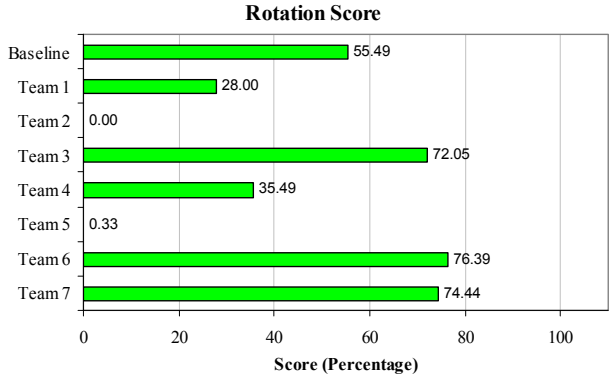


Figure 8. Final rotation scores over all 399 frames.

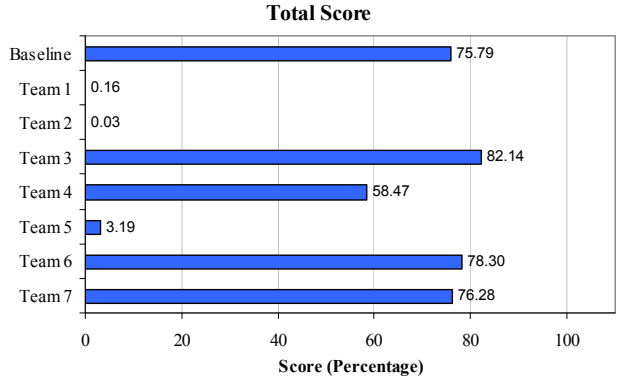


Figure 9. Combined recognition and pose estimation scores.

Willow Garage's open-source Robot Operating System (ROS). Though it is charted alongside the seven competing teams, this baseline system did not compete in the 2011 SPC.

5.1 Recognition Scores

For each frame of data, the ground truth consisted of a set of one or more objects. A true positive count (*hits*, c_h) reflects an algorithm's ability to correctly identify when an object is in the scene. A non-zero false positive count (*noise*, c_n) indicates that an algorithm identified objects that were not actually present in the ground truth, and a non-zero false negative count (*misses*, c_m) implies that the algorithm could not correctly identify objects that were in the scene.

The true positive counts were tallied over all 399 frames, and are illustrated in Figure 4. Five of the seven competing teams correctly identified over 80 % of all objects over all frames (665 artifacts or more of the 831 present over all 399 frames), while the remaining two found fewer than 20 %. Similarly, the false negative (Figure 5) and false positive (Figure 6) counts were left in tabulated format. From these, one can see the total cumulative number of objects missed, and the amount of noise inserted by the algorithms, both of which have a negative affect on the total performance score, as will be discussed in Section 5-3.

5.2 Pose Estimation Scores

Using the scoring metrics discussed in Section 4, the translation and rotation scores were calculated and normalized for all frames to produce a final score for each. These scores are illustrated in Figures 7 and 8, respectively.

Of the seven competing teams, only two teams achieved scores above 80 %, i.e., the estimated translation was within tolerance greater than 80 % of the time. Three of the remaining five had translation scores less than 1 %. In contrast, many teams performed significantly better with orientation estimations, though none of the seven competing teams achieved greater than 77 % accuracy. The algorithm provided by Team #2 did not generate position or orientation estimations, and thus had a score of 0 for each.

5.3 Summary

The recognition and pose estimation scores were combined on a per-frame basis to achieve the total scores. The equation used to compute the final score for each frame i is

$$score = \frac{1}{2}\sum_{i=1}^{N} \max\left(\frac{\left(c_{h,i} - 0.5c_{m,i} - c_{n,i}\right)}{M_i} + \frac{s_{R,i} + s_{T,i}}{2}, 0 \right) \quad (7)$$

where N is the total number of frames and M_i is the number of objects in frame i.

For the recognition component, $c_{h,i}$ is the hit count for frame i, $c_{m,i}$ is the miss count, and $c_{n,i}$ is the noise count. Note that this scoring metric severely punishes misses and noise. This is to stress high detection accuracy (assuming no false positive detections, an algorithm that detects only two of four artifacts in a given frame will score only 25 %), and to dissuade the trivial solution that all artifacts are present in all frames.

The minimum score for any given frame's pose estimation effort is 0; this is to prevent an algorithm's poor performance for one frame from affecting its performance on another. Thus for any summation of recognition and pose estimation scores equaling a value less than 0, the minimum value of 0 will be used instead. For the pose estimation component, $s_{R,i}$ is the rotational score from (6), and $s_{T,i}$ is the translational score from (3). The frame scores were tallied and normalized across all frames to produce a total performance score, illustrated in Figure 9. The algorithm provided by Team #3 performed the highest on NIST's data set, with 82.14 % total accuracy.

6. DISCUSSION

During the week of 1-7 May, 2011, seven teams representing three different countries submitted final code bases for evaluation in the inaugural Solutions in Perception Challenge, held at ICRA 2011 in Shanghai, China. The results of these evaluations were presented during the first day of the Competitions Track, Tuesday, 10 May, 2011. Because all teams were presented with identical data sets taken from a single sensor, team performance was a function of their algorithms' abilities to learn, detect, identify, and estimate the poses of objects. Rank ordering of the teams' algorithms was based on a combinatorial scoring function that took into account both object recognition and pose determination over a collection of independent frames containing image and depth data of 16 machined artifacts. Object recognition scoring was based on enumerations of *hits* (or true positives, in which the teams' algorithms were able to successfully identity objects within a given scene), *misses* (or false negatives), and *noise* (or false positives). Pose determination scoring was determined by a combination of translational and rotational error scores.

Post-competition analysis provided insight into the trends in pose and object identification algorithm performance. The top-performing algorithms utilized image feature description and detection algorithms and probabilistic modeling and classification methods to either detect or estimate the poses of objects. Although some teams clearly performed better than others in estimating poses, the distinguishing factor among the top-performing teams was in their ability to correctly identify which objects were (and were not) present in a given scene. Algorithms with lower scores had high false-positive counts (an indication of poor filtrations of candidate artifacts from scenes, and, in the case of the 2011 SPC data sets, assuming multiple objects could occupy the same coordinates in space).

7. REFERENCES

[1] Newman, M.E. 2011. NIST Contests in China Put Next-Gen Robot Technologies to the Test. In *NIST Tech Beat*. http://www.nist.gov/el/isd/robots-060711.cfm 7 June, 2011.

[2] Georgia Institute of Technology, *et al*. 2009. A Roadmap for US Robotics: From Internet to Robotics. http://www.us-robotics.us/reports/CCC Report.pdf 21 May, 2009.

[3] Anderson, M., Jenkins, O.C., and Osentoski, S. 2011. Recasting Robotics Challenges as Experiments. In *IEEE Robotics and Automation Magazine*. 10-11. June, 2011.

Shape-based Pose Estimation Evaluation using Expectivity Index Artifacts

Chad English
Neptec Design Group
302 Legget Drive
Ottawa, Ontario, Canada
(613) 599-7603
cenglish@neptec.com

Galina Okouneva
Ryerson University
350 Victoria Street
Toronto, Ontario, Canada
(416) 979-5000
gokounev@gmail.com

Aradhana Choudhuri
Canadian Space Agency
6767 Route de l'Aéroport
Saint-Hubert, Quebec, Canada
(450) 926-4800
aradhana.choudhuri@asc-csa.gc.ca

ABSTRACT
This paper recommends the use of three distinct shape artifacts to evaluate shape-based pose estimation systems, and provide their rationale. These artifacts are the Reduced Pose Ambiguity Cuboctohedron (RPAC), a cube, and an 80-triangle tessellated sphere. The rationale for these shapes derives from the range of Expectivity Index (EI) values and ambiguity intervals. The EI varies inversely with expected pose estimation error for a given shape and view, and the ambiguity interval describes a distance between symmetries where a shape fits with the incorrect pose as precisely as with the correct one. These concepts are discussed in detail and used to define the proposed shapes as good for covering a range of circumstances for performance evaluation of shape-based post estimation systems, and are proposed for inclusion in the ASTM E57.02 standards for pose estimation evaluation.

Categories and Subject Descriptors
I.4.8 [**Imagine Processing and Computer Vision**]: Scene Analysis – *motion*, *tracking*

General Terms
Algorithms, Measurement, Performance, Standardization.

Keywords
Pose estimation, shape, Expectivity Index, ambiguity, artifacts.

1. INTRODUCTION
Pose estimation techniques vary widely but generally fall into two categories: feature-based and shape-based. Feature-based pose estimation uses distinct features of objects as seen from a sensing system, matched against corresponding features of a reference object, to get the pose estimate. These features might be edges, corners, texture details, or specially designed targets. The reference object is some database of feature properties and their locations on the object. Once matched, the position and orientation that aligns the reference object to features seen from the sensor system provides the pose estimate.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage, and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. PerMIS'12, March 20-22, 2012, College Park, MD, USA. Copyright © 2012 ACM 978-1-4503-1126-7-3/22/12...$10.00

Shape-based pose estimation does not identify particular features of the object. It generates a pose estimate by aligning the reference object to the sensor data such that the deviation between the reference object and data are minimized. In shape-based techniques, the reference object represents the whole shape, such as a CAD model. The model is approximately fit to the data and the deviation is calculated. The pose is adjusted iteratively following a set mathematical routine until a measure-of-fit (e.g., root mean square of the deviations, maximum deviation, average deviation) is minimized, and the position and orientation corresponding to this minimum value is the estimated pose.

The deviation (e.g., distance, area, volume) between the sensor data and the references shape is defined as the misclosure. Variations of shape-based techniques measure the misclosure differently and make pose adjustments differently, but follow the same general iterative process. A common shape-based method using 3D data to get 6 DOF pose estimates is the Iterative Closest Point (ICP), which has many variants [1].

Shape-based methods have the advantage that they are neither limited to specific features available on objects nor require the addition of fiducial targets as in many feature-based methods. This frees up shape-based methods to work on essentially any object with a definable, rigid surface. The disadvantages of shape-based methods are that they typically require an approximate alignment to start from, require multiple iterations (as opposed to a single calculation), and their performance is highly dependent on the strength of the object's shape, even leading to degenerate cases. (Another disadvantage may be that they require an a priori model of the object shape; however, all pose estimation approaches require some a priori definition of the object such as the location of features or targets on the object.)

As alluded to in the previous paragraph, some shapes are better than others. A common example of a weak shape is a cylinder. The 3D sensor data for a cylinder can align perfectly with the reference cylinder object for an infinite number of angles around the cylinder's circumference. The misclosure will be zero, and therefore only five of the six DOFs can be uniquely determined. The sixth requires a feature to define the orientation around the circumference. Typically in such cases the pose estimation will be augmented with a feature-based technique using some distinct features of the cylinder that define the angle around the cylinder.

If truly a cylinder, it is possible the rotation angle is irrelevant and can be ignored. But even when there is a distinct feature to define the 6th DOF, performance depends on whether the sensor can see

it. A common example is the coffee mug. The handle defines a distinct orientation for the mug, but if the sensor is viewing the opposite side and cannot see the handle then the rotation angle cannot be determined without augmentation via a feature-based technique using textural features on the mug surface such as lettering.

For shape-based methods there is a continuum of shape quality from absolutely definable to undefinable DOFs through which shapes can be strong or weak depending on the view. The following sections describe an index-based method for defining how strong or weak an object is from a given view for shape-based pose estimation. This index, the Expectivity Index, provides a means for de-coupling the measured pose estimation performance as a function of the object shape (and viewpoint) from the performance of the pose estimation system itself.

2. EXPECTIVITY INDEX

The general concept of the Expectivity Index relates to the rate of change of misclosure for a movement in each DOF from a given viewpoint. The EI is derived based on the total misclosure error across all DOFs. The basis for the EI is described visually in Figure 1 below. Figure 1 is a conceptual visual explanation that does not exactly match the mathematics of the EI, but it explains the general principle. It also uses a 2D square example with 3 DOFs instead of 3D with 6 DOFs, but the principle can be generalized. For detailed mathematical derivation, see [2].

In the figure, a square is viewed by a sensor at the top, with coordinate axes defined in x and z, and a rotation angle into the plane of the page. The top row shows a corner view from the sensor. The sensor data are represented as dotted lines. Since the sensor can only see the edges facing it, there are no data for the opposite side.

In (a), the reference object (the square) is offset in z. The misclosure between the sensor data (dotted) and the square can be defined by the space between them, which is highlighted as the area in gray. (In 3D this would be a volume misclosure.) In (b), the offset is in x and in (c) in the rotation angle. The general case of all 3 DOF offsets is shown in (d). It is important to note that, for this view of this shape, there is no small offset in any combination of DOFs from the true pose that produces zero misclosure.

The bottom row of the figure shows the same circumstances but where the sensor is viewing a face of the square. In (e), the offset in z shows a large misclosure. In (f), the offset in x shows no misclosure area (or volume) at all. (Some shape-based alignment algorithms would measure zero misclosure error here while others would note that the left edge of the data doesn't fit to any surface of the square and measure a distance from each data point to the closest edge. Although this is a non-zero misclosure, it is very small compared to (e) as only the few points off the edge would contribute to it.) The rotational and combination offsets in (g) and (h) show their related misclosures.

Now consider that for every offset displacement (in any combination of the 3 DOFs), there is a rate at which the misclosure increases with the size of the displacement. In the corner view (top row), the misclosure increases quickly for any displacement. In the face view (bottom row), a displacement in the z-direction (e) increases the misclosure quickly, but a displacement in the x-direction (f) does not increase it at all (or very slowly if using the points off the edge). If you multiply these rates across all DOF directions from the given viewpoint, the corner-view (top) will have a very large number but the face-view (bottom) will be zero (or relatively small).

This combination of misclosure rates approximately describes the concept of Expectivity Index for a given view of a given shape. (The mathematical derivation of EI found in [2] is more closely described as the harmonic mean of the eigenvectors of the misclosure matrix. For the conceptual purposes in this paper, the

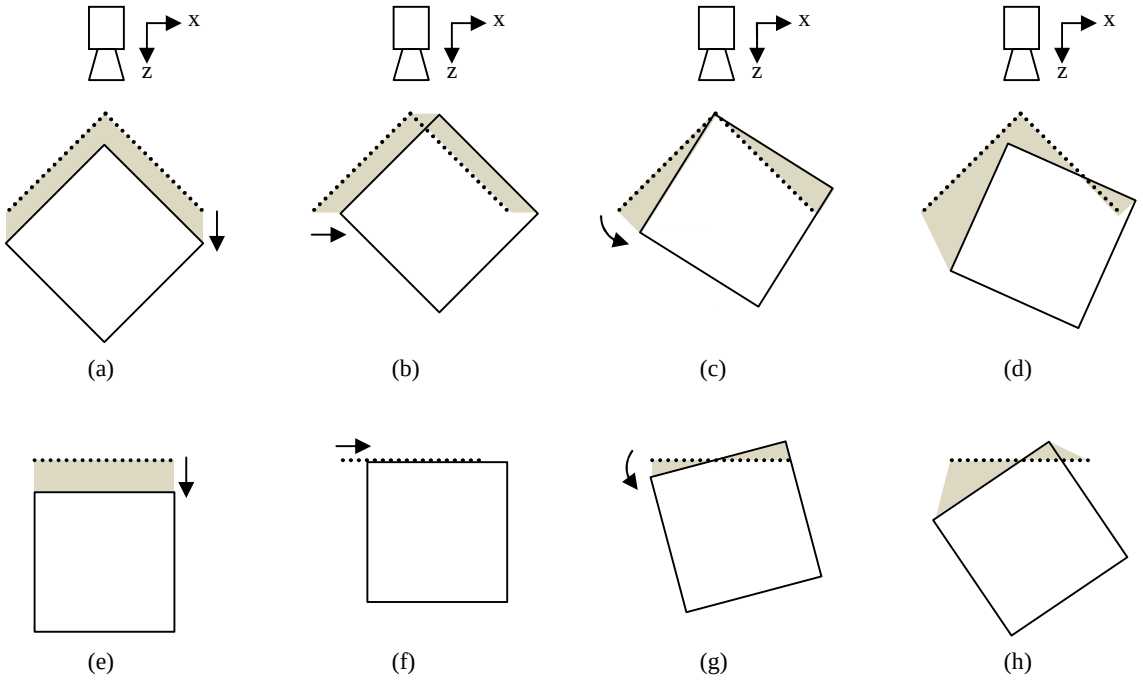


Figure 1: Shape-based misclosures for a square from cornerview (a-d) and face view (e-h).

approximate description here is sufficient.)

Views that have at least one weak DOF where the misclosure is small will have a small EI. This corresponds to views where there is some combined displacement direction in which the sensor data still fits the reference object shape with little misclosure. Views where any displacement causes a large misclosure will have a large EI. The process is not invertible since the EI doesn't tell you which DOFs are weak, but it does tell you the overall alignment strength or weakness across all DOFs for a given view.

By moving the sensor around an object in a circle (sphere in 3D) to get all viewpoints, one can get an Expectivity Index map for the object.

3. POSE AMBIGUITY

The previous section described how shape-based techniques rely on fitting sensor data to the surface of the reference object, where a minimum misclosure should correspond to the correct pose. The EI indicates how quickly the misclosure increases with small displacements from this correct pose. Larger displacements can result in a different problem. The corner-view on the top row of Figure 1 has a high EI and hence produces very good pose estimation near the corner. However, if one rotates the reference square 90°, the data also fits just as well on the next corner even though the pose is now in error by 90°. This problem is due to symmetries in the shape. Every 90 degrees the square will fit the data perfectly and the EI will be the same. There is no measurement information that indicates that there is a large error, leading to an ambiguity of which corner is the true pose of the object.

Ambiguity can be measured using the misclosures as well. For a given view, the reference object can be rotated 360° (spherically in 3D) and the misclosure measured. A close or exact fit for a wrong pose indicates potential ambiguity, and the relative pose difference between them indicates the ambiguity interval. If surface features are repeated near to each other, the ambiguity interval will be small and the object might easily give an incorrect pose estimated by fitting the sensor data to the shape offset by one or more ambiguity intervals (e.g., 90° for a square, 60° for a hexagon). (The ambiguity interval is defined here specifically for objects with regularly repeating shapes. General shapes may not have symmetries that produce exact ambiguities or ambiguity intervals. This principle generalizes as local minima of the misclosure, but is presented here in purest form as exact ambiguities for the purposes of testing the robustness of a shape-based pose estimation system.)

4. RPAC DESIGN

A good shape for pose estimation has a high EI from all views and either no ambiguities or a large ambiguity interval. The high EI means that all views have strong alignment in all DOFs and the large ambiguity interval means it is unlikely that a slight misalignment will result in a good fit to the wrong pose.

The Reduced Pose Ambiguity Cuboctohedron was designed to be an optimized shape from all views based on the EI and ambiguity analyses [3]. A variety of shapes were analyzed and the best EI results corresponded to a regular cuboctohedron, a 3D shape with eight triangular faces and 6 square faces. However, this shape results in twelve identical vertices, so there is great ambiguity around the shape. Angles between surfaces were adjusted

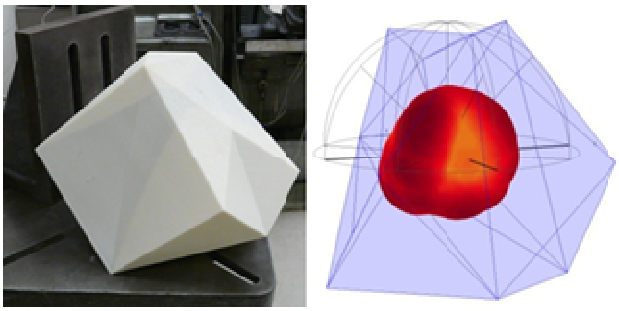


Figure 2: Reduced Pose Ambiguity Cuboctohedron (RPAC): (Left) 3D shape, (Right) EI values plotted as radius and color.

numerically and tested for both EI and ambiguity. The resulting shape, the RPAC, has all unique vertices, angles between surfaces, and surface shapes while maintaining a nearly equal EI around the entire shape only slightly lower than the regular cuboctohedron.

The RPAC therefore represents an optimized object shape for shape-based pose estimation in terms of EI and ambiguity, and the range of objects tested. This optimization corresponds to low pose estimation error from all views and low likelihood of getting into an incorrect view with a good fit, creating a pose estimation bias. Figure 2 shows the RPAC shape and a 3D plot of EI around the shape.

5. EI-BASED ARTIFACTS

While the RPAC describes an optimized shape for good pose estimation, a worst-case shape will have degenerate DOFs where the misclosure is zero or near zero for all values of the DOF such Figure 1(f). A cylinder represents a degenerate case in the rotational DOF as well as a weakness in the translational DOF along the cylinder (especially if the top and bottom are not in the FOV of the sensor). A sphere represents degeneracies in all three rotational DOFs. For position, a large planar surface represents degeneracy in two positional DOFs and the rotational DOF about the planar surface. Defining a weak shape therefore depends on which DOFs are to be made weak.

Some shapes have a mix of pose strengths. A cube is like the square in Figure 1. It has strong corner views (vertex of three faces), weak (planar) face views, and medium edge views (between two faces). A tetrahedron similarly has good vertex views, weak face views, and medium edge views, where EI changes from high to low.

It is also possible to produce shapes that are medium EI and ambiguity around all views. A shape with many faces approaches the shape of a sphere. An icosahedron, for instance, has 20 faces. The more faces the weaker the views (due to smaller angles between surfaces giving smaller misclosure for rotations) and the lower the ambiguity angle.

Figure 3 shows this pattern. From left to right the shapes increase from 3-sided (triangle) to 4, 5, and 6-sided, finally ending in an infinite-sided shape, a circle. The top row shows the vertex view towards the sensor and the bottom row shows the face view. The top row shows that the rotation DOF gets weaker as more sides are added. For a 15° rotational misalignment, the 3-sided object in (a) has a large misclosure. For the same rotational misalignment, the misclosure area becomes smaller and smaller the more sides

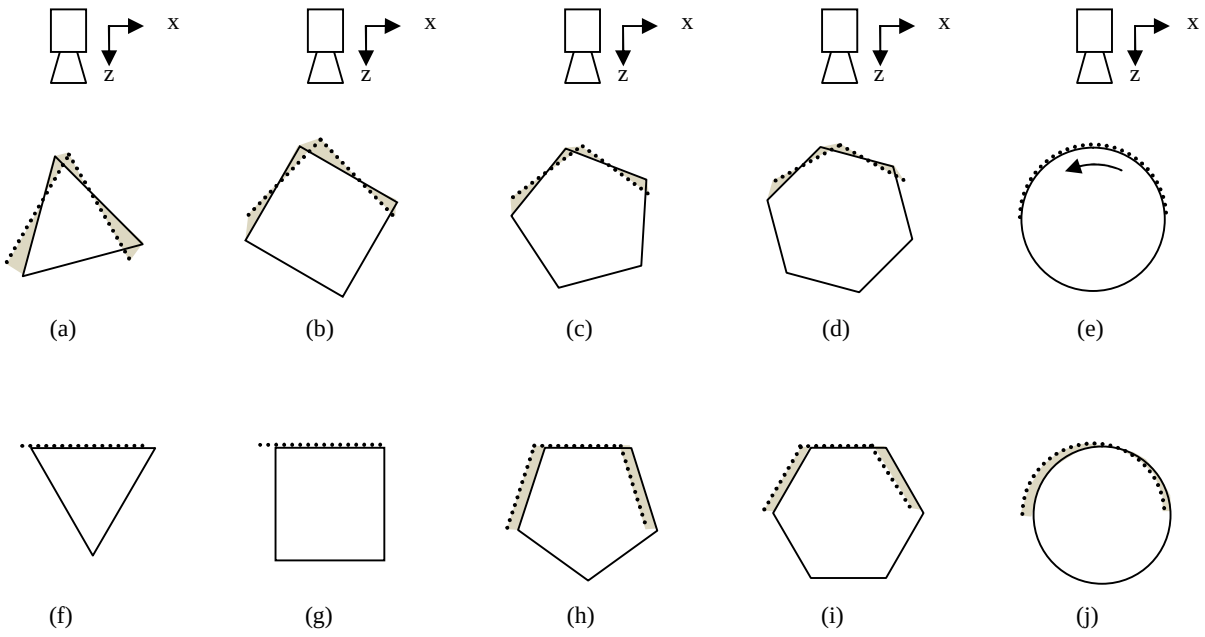


Figure 3: Misclosures and ambiguity for n-sided shapes in their weak DOFs.

that are added through (b), (c), and (d), ending with zero misclosure in (e). Hence the rotational contribution to the EI gets weaker.

At the same time, the ambiguity interval gets smaller from left to right. For the 3-sided object in (a) the shape must be rotated 120° to get a perfect fit but at the incorrect angle. It is 90° in (b), 70° in (c), 60° in (d), and finally 0° in (e) as any rotational misalignment fits perfectly.

The implication then is that the 3-sided shape (triangle) on the left is the strongest and has the largest ambiguity angle. This is true from a vertex view, but not true of the shape in general, as shown by the bottom row. In face views, the left-most figures are very weak in the x-direction and hence have a low EI. Both the 3-sided and 4-sided shapes have a zero EI in face view because they only see a flat plane. Only once other sides of the shape can be seen in the 5-sided view does the x-direction have a sufficient EI. (It is not shown here, but real sensors will not collect much data down the sharp angles of the 5-sided shape. As a general rule of thumb, higher sided shapes will give more data on more faces.)

A resulting trade-off can be seen. Fewer sided shapes have large ambiguity intervals and strong vertex views, but weak face views. Higher sided shapes have stronger face views but weaker vertex views and smaller ambiguity intervals. This is why the RPAC study gave an optimum shape as a mid-level number of faces (cuboctohedron), and distorting the faces to reduce ambiguity improved results.

6. EI ARTEFACTS FOR EVALUATION

The RPAC represents an optimum shape over all views. This makes it work well as a standard for generating baseline performance metrics for a shape-based pose estimation system. The high EI values for all viewpoints means pose errors should be near a minimum achievable for the system.

Since the angles between faces and the sizes and shapes of the faces are unique, the RPAC also offers a means for feature-based pose estimation using a look-up table from the 3D data that can be used as an independent pose estimation check for comparison to the shape-based results. Such a lookup table method has been developed and tested [4].

Baseline performance under optimized conditions represents only a subset of system performance, and hence the RPAC is insufficient to evaluate system performance alone. How well a system performs under degraded conditions is equally as important. The other two proposed artifacts are the cube and the 80-triangle tessellated sphere, shown in Figure 4. These artifacts allow for quantifying the EI and ambiguity shape conditions under which a shape-based pose estimation system breaks down.

The cube provides a range of EI values from high in the corner views to low in the face views. In the face view, only a planar square is visible. The low EI values correspond to small misclosure that occurs for lateral misalignments of the face or rotational misalignment in the plane. There is enough information to provide a pose estimate, but for small displacements the misclosure will be very small as only a few sensor data points will fall off of the surface and vast majority of them will fit perfectly on the surface, as in Figure 1(f). Hence the boundaries of the square become critical to defining the pose. A good pose estimation system may recognize this sensitivity and weigh the boundary points higher, for instance [5].

Since EI correlates with expected pose error (statistically), pose estimation systems can be compared based on how well they do across repeated tests for a given view. The actual EI value does not need to be known. Rather, a system can be evaluated around the whole artifact. A user can then compare performance of several systems at or near the corner views and at or near the face views to determine which system is better, or sufficient, for their application.

The cube also tests system robustness towards high ambiguity intervals. The cube has an ambiguity interval of 90°. A shape-based pose estimation system that consistently gets errors of

multiples of 90° indicates it might have big problems with initiating the alignment process.

The 80-triangle tessellated sphere (Figure 4, right) evaluates performance under a different set of degraded conditions. It is a multi-sided shape approaching a sphere corresponding to shapes toward the right of Figure 3. There is sufficient information to define all 6 DOFs but the rotational information will be weak. It has a medium to low EI because the low angle between the faces means a small misclosure for rotational displacements in any direction (e.g., Figure 3(d)). It also means a low ambiguity interval because of the low rotation angles between triangles.

The combination of low EI and low ambiguity interval, both in rotational dimensions, means small misalignments in the shape-based fitting iterations might cause the fitting algorithm to align it off by one or more ambiguity intervals. An ambiguity failure in the case of the cube above would demonstrate a gross initialization error in the alignment process. An ambiguity failure in the case of the tessellated sphere would demonstrate at what conditions (range, sensor resolution) the pose estimation system reached its limits to make use of small shape details to provide an accurate pose estimate. If it does not provide a statistically consistent pose estimate, but jumps around from test to test, it can't distinguish the tessellated sphere from a true sphere.

The RPAC, cube, and tessellated sphere therefore offer a range of circumstances for shape-based pose estimation systems. The RPAC offers a good target for pose estimation from all views and therefore offers a baseline case for achievable performance metrics. The cube offers a range from poor tracking conditions (face) to excellent conditions (corners), and offers performance evaluation of potential failure modes for planar surfaces. The tessellated sphere offers a borderline case where a high-resolution, low noise sensor could produce good pose performance but a lower resolution or higher noise sensor might have difficulty in estimating the orientation angles. For a given system, it can also provide a measure of what distance a pose estimation system performance breaks down compared to other systems.

The intent is to use these artifacts following the procedure and analysis defined in the ASTM E57.02 standard for pose estimation system performance evaluation. The scale of the target sizes would be chosen relevant to the capabilities of the pose estimation operations.

7. REFERENCES

[1] Rusinkiewicz, S., Levoy, M., Efficient variants of the ICP algorithm, Proc. Third International Conference on 3-D Digital Imaging and Modeling, Quebec City, May 28 – June 1, pp. 145-152, 2001.

[2] Okouneva, G., McTavish, D., Choudhuri, A. 2010. Principal Component Analysis for Pose Estimation Using Range Data. Computers and Simulation in Modern Science, v. IV, pp. 111-124, WSEAS Press

[3] Choudhuri, A., Okouneva, G., McTavish, D., Bouchette, G. 2010. Design of Optimal Shapes for Space Docking using LIDAR-Based Vision Systems, ASTRO 2010 (Toronto, ON, May 4-6).

[4] Saint-Cyr, P., 2010. A Framework for Non-ICP LIDAR-based Pose Estimation Using an Optimally Constrained 3D Target. MASc. Thesis, Ryerson University.

[5] L. H. Mark, Galina Okouneva, P. Saint-Cyr, D. Ignakov, C. English, 2010. Near-Optimal Selection of Views and Surface Regions for ICP Pose Estimation. 6th International Symposium on Advances in Visual Computing (ISVC 2010), Las Vegas, NV, USA, Nov. 29 – Dec. 1, pp. 53-63.

Ground Truth for Evaluating 6 Degrees of Freedom Pose Estimation Systems

Jeremy A. Marvel
National Institute of Standards and Technology
100 Bureau Drive, MS 8230
Gaithersburg, MD 20899
jeremy.marvel@nist.gov

Joe Falco
National Institute of Standards and Technology
100 Bureau Drive, MS 8230
Gaithersburg, MD 20899
joseph.falco@nist.gov

Tsai Hong
National Institute of Standards and Technology
100 Bureau Drive, MS 8230
Gaithersburg, MD 20899
tsai.hong@nist.gov

ABSTRACT
Systems developed to estimate poses of objects in 6 degrees of freedom (6DOF) Cartesian space (X, Y, and Z coordinates plus roll, pitch, and yaw) are reliant on the vendors' own processes to determine performance and measurement accuracy. These practices are not yet standardized, and are rarely reported by the vendors in sufficient detail to enable users and integrators to recreate the process. Efforts must therefore be made to enable the documented and, more importantly, independently repeatable evaluation of such systems using standardized processes, fixtures, and artifacts. In this paper, we describe three 6DOF ground truth systems utilized at the National Institute of Standards and Technology (NIST): a laser-tracker-based system for pose measurement, an aluminum fixture-based system that can be used to set the pose of artifacts, and a modular, medium-density fiberboard (MDF) fixture system. Descriptions, characterizations, and measured accuracies of these systems are provided for reference.

Categories and Subject Descriptors
C.4 [**Performance of Systems**]: Performance Attributes; B.8.2 [**Performance and Reliability**]: Performance Analysis and Design Aids; G.1.6 [**Optimization**]: I.5.4 [**Applications**]: Computer Vision

General Terms
Measurement, Documentation, Performance, Experimentation, Standardization, Verification

Keywords
Ground Truth, 6DOF Metrology, Laser Tracker, Fixtures

1. INTRODUCTION
The usefulness of novel 6DOF pose estimation systems is restricted only by the accuracy with which it can measure objects in the world space. Reporting this is relatively simple, but the initial evaluation and subsequent validation of the reported values are extensive processes requiring the appropriate metrics and either a reference standard system (i.e., an external ground truth) against which the accuracy of a system under test can be measured, or a methodology of computing variances in the data to infer a given system's precision under different operational conditions. The utilization of ground truths is a fundamental aspect of measurement science, and provides a basement of comparison for the estimated quantities measured independently and simultaneously by the system under test.

A fundamental limitation of ground truth utilization, however, lies in the difficulty in obtaining the ground truth, itself. Establishing a measurement system as a ground truth requires extensive efforts and measurement tools in validating its accuracy. As a general rule, the ground truth system must be at least an order of magnitude more accurate than the system under test. The tools required to assess the accuracy of potential ground truths are prohibitively expensive, and must conform to set traceability standards, themselves, in their establishment as ground truths.

In this paper we discuss the evolution of NIST-developed ground truth systems in efforts to make 6DOF metrology evaluation more accessible and expandable. Three different systems are presented, and their accuracies and measurement uncertainties are provided. The issues addressed in this report focus on the development and validation of ground truth systems, and are discussed in an effort to provide examples of the establishment of new ground truths.

2. RELATED WORK
Although 3D pose estimation systems may be evaluated sans ground truth [1], the utilization of an external ground truth is typical for measurement systems for the computation of errors in pose estimations. These errors are then evaluated to infer statistical distribution (mean, standard deviation, and error trends) of the bias and variance of the environmental parameter space for the sensor under test [2].

The ground truths may be either sensor- or artifact-based, and are expected to be at least an order of magnitude more accurate than the system under test. Sensor-based ground truths—where the pose of an object is based on the measured outputs of a system with known accuracy and precision—are traditionally flexible and modular in nature, but require a robust calibration system [3] to establish a common coordinate frame between the ground truth and sensor under test. While many ground truth systems employ some form of fiducial attached to the surface of an evaluated target in order to enable precision metrology (e.g., laser-tracked active targets [4] and camera-tracked active [5] and passive [6]

This paper is authored by employees of the United States Government and is in the public domain. PerMIS'12, March 20-22, 2012, College Park, MD, USA. ACM 978-1-4503-1126-7-3/22/12

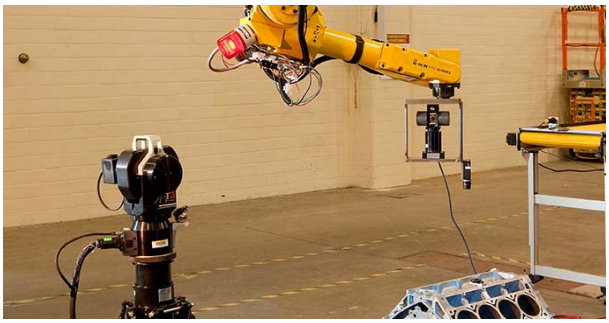


Figure 1. The laser tracker ground truth system (left) and active target attached to an industrial robot arm (right).

Table 1. Uncertainties (Standard Deviations) of Position (X, Y, and Z) and Rotation (Roll, Pitch, and Yaw) Measurements of the Laser Tracker System

	X	Y	Z
Uncertainty (mm)	0.0018	0.0014	0.0021
	Roll	**Pitch**	**Yaw**
Uncertainty (degrees)	0.0007	0.0001	0.0012

targets), not all evaluation systems are compatible with them. Such artifacts may inadvertently change the surface properties of the target, and thus interfere with the performance of certain shape- and feature-based pose estimation systems.

Artifact-based ground truths are based on either fixtured components with associated *a priori* knowledge of transformations and pose uncertainty, or known distributions of features on a specific truthing object. In contrast with the sensor-based ground truths, artifact-based ground truths are typically easier to use in evaluations, are generally more readily repeatable, and are more affordable and accessible to a variety of researchers. For instance, in [7] a simplified cardboard artifact was rigidly affixed to a rotational base for a single DOF in pose variance. The rotational base had position sensor to read orientation angle around a pivot point. Further, in [8], rigid automotive engine components were used for validation of their 3D pose estimation system using feature-based tracking of various component assembly points relative to one another.

Artifact-based ground truths involving physical objects, however, are subject to measurement uncertainties in pose and adherence to construction tolerances, both of which necessarily introduce some error in establishing the ground truth. As such, an alternative artifact-based approach utilizes synthetic data for test and evaluation of pose estimation systems. For instance, [9] utilized computer-generated images of geometric primitives with associated CAD models to evaluate a proposed single-camera 3D pose estimation system, while [10] used simulated 3D point cloud data and robot pose information to validate a 6DOF localization methodology using polygonal indoor maps.

3. LASER TRACKER

The laser tracker system, shown with its active target in a testing configuration in Figure 1, has been utilized as a high-precision ground truth for 3D measurements at NIST since 2008 (e.g., [1, 3, 11]). It has been used to truth component positions of manufacturing and construction systems when tolerance accuracies are unknown or unreliable. The laser tracker boasts high measurement accuracy, but at the expense of monetary cost. The full cost of the system utilized at NIST is approximately $150 000.

3.1 Measurement Configuration

The laser tracker configuration utilized for 6DOF pose measurement has two physical components: a portable active target that measures its own orientation using a motorized receiver and a level sensor, and a base laser unit that measures the position of the active target [4]. Together, they provide the complete 6DOF pose of the active target. The active target can only be used for measuring static 6DOF poses with a precision of ±3 arc-seconds in angle (±0.0008 degrees), and a combined positional accuracy of 15 µm average error with uncertainty of 10 µm at 2.0 m. The active target, which is either attached to or substituted in lieu of the object to be truthed, requires a direct line of sight with the laser unit's beam, and thus only one object can be measured at a time.

3.2 Measured Accuracy

The measurement accuracy of the laser tracker system mentioned earlier was specified by the manufacturer. These accuracy specifications were validated according to the process laid out in the ASME B89.4.19-2006 standard (*Performance Evaluation of Laser-Based Spherical Coordinate Measurement Systems*), and the computed measurement errors were within the manufacturer's specified tolerances. During the validation process, we collected 30 data points per sample position with the measurement sensor mounted on a rigid mount.. The standard deviation of each measurement value (i.e., X, Y, Z, roll, pitch and yaw) was calculated, and is shown in Table 1. These deviations illustrate that the uncertainties of the laser tracker for measuring the ground truth object are also within the specified tolerances, and justify the utilization of the laser tracker system as a ground truth for evaluating 6DOF pose estimation systems with purported accuracy tolerances of ≥0.15 mm.

4. ALUMINUM MECHANICAL FIXTURE

To compensate for the single-target limitation and setup complexity of the Laser Tracker system, we developed a portable aluminum mechanical fixture ground truth system (*GT2011*) capable of supporting several NIST manufacturing part artifacts simultaneously. These artifacts were designed to represent a quorum of features found in typical manufacturing environments. Each artifact is a modular block with machined features found in real-world manufactured parts. GT2011, shown in Figure 2, was designed to generate repeatable ground truth artifact poses, and then provide this pose data in the form of known homogeneous transformation matrices to researchers for algorithm evaluation. The aluminum construction provides stiff transformations, and limit wear of the fixture over time. A limitation of this fixture is that it requires precision machining capabilities to produce; as such, the cost to produce this ground truth in-house was approximately $4,000.

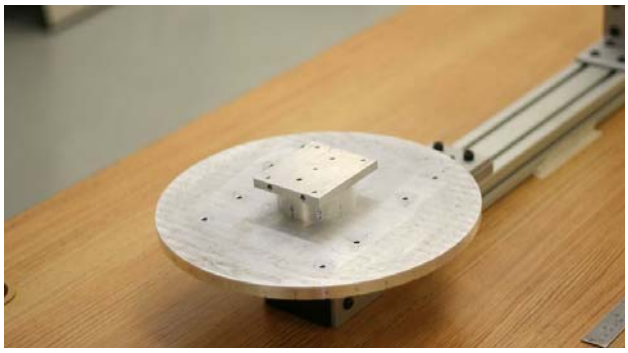


Figure 2. The GT2011 rotation base plate with a mounting plate affixed to the Fixture 0 mechanical offset.

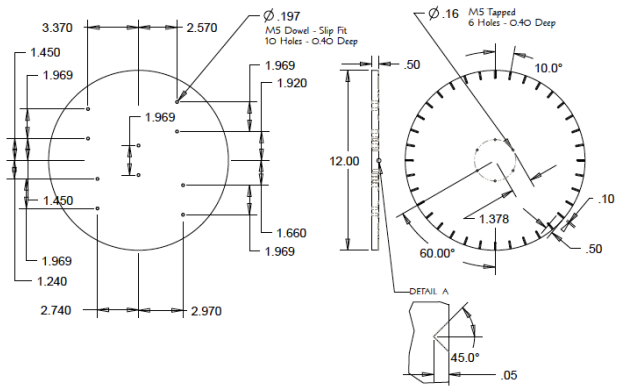


Figure 3. The base plate design of GT2011 featuring five mounting position (left) and 36 rotation presets (right).

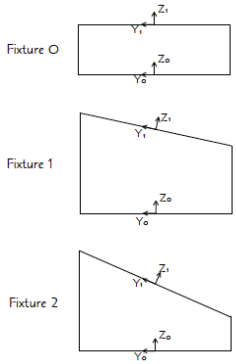


Figure 4. Illustration of the angular tilt offsets generated by the three mechanical offsets.

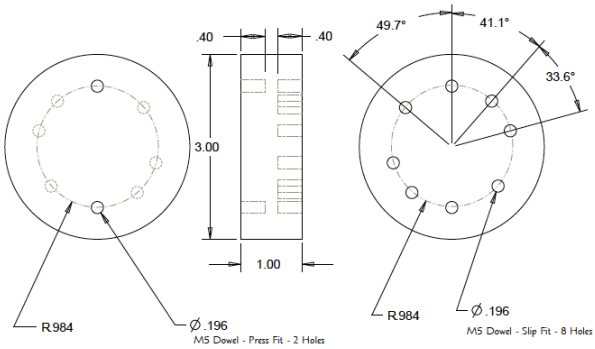


Figure 5. Illustration of the angular rotation offsets generated by each of the three mechanical offset fixtures.

4.1 Design

The design of GT2011 consists of a frame constructed from 8020, a modular engineering system of interlocking aluminum components. The base is used to hold the sensor under evaluation on a vertical arm, with adjustments to vary the sensor horizontal and vertical offsets relative to the rotation plate mounted via a slew bearing to the 80/20 (a modular aluminum framing system) base. The rotation plate (Figure 3), machined at NIST, contains four sets of two alignment holes that accept mechanical offset fixtures (Figure 4) and NIST modular manufacturing part artifacts. Each mechanical offset fixture provides an angular offset as a machined surface for attaching an artifact, and four sets of alignment holes (Figure 5) for attaching to the alignment holes in the rotation plate via dowel pins. The alignment holes enable each offset fixture to be rotated 49.7º, 105.3º, 138.9º and 180.0º. From Figure 4, offset Fixture 0 has a nominal 0º tilt and a vertical (Z axis) offset of 25.4 mm. Offset Fixture 1 has a nominal 12.3º tilt, and a vertical offset of 42.49 mm. And offset Fixture 2 has a 23.8º tilt, and a vertical offset of 34 mm. The plate can also be rotated at 10° increments using a ball plunger quick lock mechanism to produce over 2,300 6DOF positions per artifact.

Up to four artifacts can be placed on the rotation plate at a time for producing artifact occlusions. The fixture's design provides comparatively high accuracy, but has limited range. Relative positioning errors of the ground truth can be attributed to the machining process which is typically accurate to within approximately ±0.02 mm per alignment hole, and inaccuracy associated with the slew bearing tolerances.

4.2 Measured Accuracy

The laser tracker's active target was rigidly affixed to one of the NIST manufacturing part artifacts such that the target was co-centric with the fixture's alignment holes. This artifact was, in turn, mounted on the GT2011 fixture via these integrated alignment holes. For reference, with regard to Figure 3, the center of the base plate is henceforth referred to as TP0, the upper left is noted as TP1, the upper right as TP2, lower left is TP3, and the lower right is TP4. Because TP0 is co-located with the center of the base plate's rotational axis, it is typically utilized at NIST as a reference point for training purposes. It is therefore not evaluated in this study, but instead provides the basis for relative transformation analyses. Additionally, only the position uncertainties of the remaining four TP locations are investigated.

We measured the X, Y and Z coordinates for the laser tracker's active target in each of the four evaluation TP positions (i.e., TP1-TP4) oriented in the zero-rotation configuration. The relative distances between each measurement and the measurement made at TP0 was then computed and compared with the nominal distances based on the original CAD design. In all, 32 data points were taken and averaged at each location to compute the measurement error and uncertainty; the results of these computations are shown in Table 2. Over all four TP locations, the GT2011 fixture

Table 2. Relative Translation Measurement Errors

TP Location	Translation Magnitude Error Mean (mm)	
TP1	0.5479	
TP2	0.3376	
TP3	0.4484	
TP4	0.6281	
	Mean: 0.4905	**Variance:** 0.1257

Table 3. Relative Rotation Measurement Errors

Nominal Angle	Rotation Magnitude Error Mean (degrees)	
33.6º	-0.0099	
55.6º	-0.0083	
49.7º	-0.0179	
	Mean: -0.0120	**Variance:** 0.0051

Table 4. Fixture Tilt Measurement Errors (degrees)

Fixture Rotation	Mean Error	Error Variance
12.3º	-0.0862	0.0229
23.8º	0.6387	0.0656

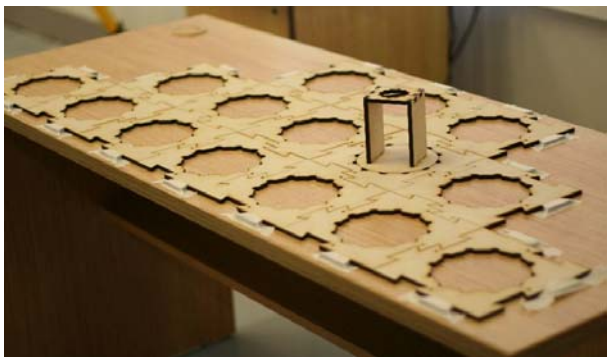


Figure 6. The GT2012 fixture configured for measurement accuracy testing. The base platform can be expaded by attaching additional plates via side interlocks.

has an average position uncertainty of 0.4905 mm, with a variance of 0.1257 mm.

Similarly, we took 18 measurements of the laser tracker's active target at each TP position of the laser tracker sensor for half of the eight angular rotation offsets created by the mechanical offset fixtures (because the alignment holes enforce 180º rotational symmetry, only four of the eight nominal rotations need to be evaluated). The relative angle between each adjacent nominal rotation measurement is computed and averaged to compute the measurement error and uncertainty. The results are shown in Table 3. In all, the GT2011 fixture has an average relative Z axis rotation measurement error of -0.0120º, with a variance of 0.0051º. Simultaneous with this evaluation, the tilt errors of the two non-zero fixtures were also measured. The results of these measurements are given in Table 4.

The magnitude of the aforementioned measurement errors has been attributed to mechanical complications from the construction of the aluminum fixture. Because of the strict tolerances insisted upon during the construction of the GT2011 fixture, the fit for the dowel pins is quite tight and can result in extemporaneous angular and vertical position offsets from the nominal value. Care should be taken to insure that the artifacts are seated properly when placed on the base plate to minimize this error. We also found significant play in the slew bearings which will require design modifications to minimize table movement when loaded with artifacts.

5. MDF MECHANICAL FIXTURE

The aluminum mechanical fixture design suffered from a few key limitations, foremost of which was the limit in range and modularity. Specifically, because of its design and construction, the range and values of position offsets was limited to the rotational base, the construction of which constituted the bulk of the cost of manufacturing. In contrast, the MDF mechanical fixture system (*GT2012*) was designed to be an even lower cost ground truth system. The GT2012 design, shown in Figure 3, was driven by the desire to have a broad user base of researchers capable of affording a medium-resolution ground truth system to use for future work in algorithm development and tuning.

5.1 Design

The design criteria used was based on the need for a modular and reconfigurable set of fixturing to support 6DOF positioning of objects similar to the artifact set used with the GT2011 fixture. The GT2012 fixture is designed to be constructed using a light weight, low cost material, and produced using third-party manufacturing services.

The GT2012 system, shown in Figure 6, is constructed from 6.4 mm MDF using a laser cutting process through a web based manufacturing service. It is modular in design such that a base platform is assembled similar to a puzzle, allowing scalability from simple to complex artifact groupings. Each base puzzle piece (Figure 7) accepts a fixture assembly containing two rotational keys, each containing twelve rotational increments, and an angular offset for adjusting Z offset, roll, pitch, and yaw of a mounted artifact (Figure 8). X and Y offsets are adjusted via puzzle piece placement. Additional base pieces are designed for mounting fuducials for calibration of the competitor's measurement systems. The ground truth system made available to researchers for initial testing is comprised of predictable linear and angular offsets. An evaluation ground truth design would be designed using a slightly modified dimensioning scheme using unpredictable offsets.

The cost to produce this ground truth fixture system with the configuration shown in Figure 6 is approximately $400. The fixture's material design and construction does not support the accuracy of the GT2011 fixture, but its modularity compensates for the range limitations of its aluminum counterpart. Relative positioning errors of the ground truth can be attributed to the laser cutting process which produces a kerf of approximately 0.2 mm.

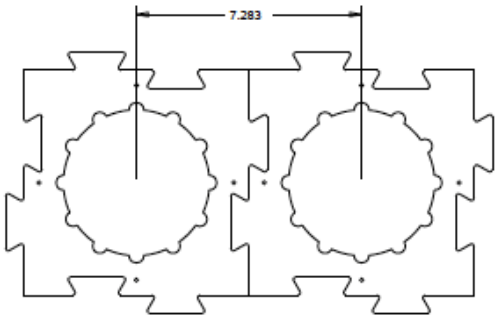


Figure 7. CAD drawing of the GT2012 modular expansion component mounting boards.

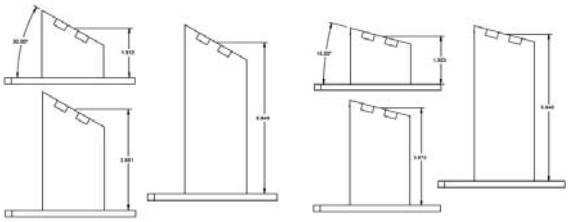


Figure 8. CAD drawing of the GT2012 high-tilt (left) and low-tilt (right) mechanical offsets for three different heights specifications. Not shown is a no-tilt offset option.

Table 5. Relative Translation Measurement Errors (mm) Compared with the Nominal Distance Between Adjacent Mounting Boards

Mean Error	-0.5794
Error Variance	0.3854

Table 6. Relative Rotation Measurement Errors and Uncertainties by Tilt Module (degrees)

Tilt Module	Mean	Variance
No Tilt	-0.0351	0.1147
Low Tilt	-0.0356	0.3203
High Tilt	-0.0498	0.3972
Avg.	0.0402	0.2774

Table 7. Fixture Tilt Measurement Errors (degrees)

Nominal Rotation	Mean Error	Error Variance
15 degrees	0.5901	0.3121
30 degrees	0.1326	0.4176

Table 8. Measurement Accuracy Magnitudes of the Three Evaluated Ground Truth Systems

	Laser Tracker	GT2011	GT2012
Mean Translation Error (mm)	0.015	0.4905	0.5794
Translation Error Variance (mm)	0.0053	0.1257	0.3854
Mean Rotation Error (degrees)	0.03 (active target)	0.1522	0.1686
Rotation Error Variance (degrees)	0.0007	0.0312	0.3124

5.2 Measured Accuracy

To evaluate the measurement accuracy of the GT2012 fixture, we arranged 15 of the modular expansion components described earlier in the configuration shown in Figure 6. The laser tracker was rigidly affixed to the mid-height, no-tilt mechanical offset (shown inserted into one of the expansion boards) and moved to each of the fifteen mounting positions in the zero-Rotation configuration (co-linear with the principle axis of the laboratory table). We measured the X, Y and Z axis coordinates of the laser tracker sensor in each position, and calculated the relative distances between each pose measurement. These distances were then compared against the linear criteria distance of 184.988 mm between the centers of adjacent mounting holes. The results of these comparisons are given in of these calculations are shown in Table 5.

Every mechanical offset fixture integrates two rotation keys—a rotation key base plate for integrating with the modular expansion components, and a smaller key hole to accommodate individual artifact mounting and rotation—each containing twelve rotation increments of 30º, and a preset angular tilt angle. For this study, only the key base plate rotations were assessed for measurement accuracy. To evaluate the rotational accuracy, the relative angular distances between adjacent rotational increments were evaluated and compared with the nominal 30º criteria angle. For each rotational measurement, 30 samples were taken and averaged to calculate the measurement error mean and variance. The results of these calculations are shown in Table 6.

As with GT2011, the mechanical offsets for GT2012 introduce both translational (Z axis) and rotational transformations for a given artifact. For each nominal Z offset (50.0126 mm and 99.9998 mm), three different angular values are introduced: a nominal 0º angular offset (“no tilt”), a nominal 15º offset (“low tilt”), and a 30º offset (“high tilt”). The low and high tilt offsets are illustrated in Figure 8. The six non-zero angular values introduced by the mechanical offsets were measured and compared to the nominal 0º tilt offset. For each measurement, 18 sample data points were taken, and the measurement errors and variances were then calculated. The results of these calculations are show in Table 7,

In contrast with the GT2011 design, the tolerances of GT2012 are far less rigid, and the material properties of MDF allow for faster wear as a function of use and time when compared with the aluminum and steel construction of GT2011. As a result, the measurement uncertainty of the GT2012 fixture increases with use. The low cost of the system, however, permits ready replacement of component parts as they wear.

Table 9. Utility of the Three Ground Truth Systems

	Laser Tracker	GT2011	GT2012
Max number of objects per scene	1	4	Unlimited*
Range (depth)	0 m – 80 m	0.6 m – 2.0 m	Unlimited*
Range (XY)	± 320º azimuth -60º – 77º elevation	0 m – 0.25 m	Unlimited*
Cost (US$)	150 000	4 000	400

* - Theoretical; though, due to the modular design of the fixture, the larger the area spanned by the objects over the fixture, the greater the pose uncertainties.

6. CONCLUSIONS

In this paper we presented three ground truth measurement systems actively utilized at NIST for the evaluation of 6DOF pose estimation systems: a laser-tracker based system; GT2011, a low-cost machined aluminum fixture system; and, most recently, GT2012, a laser-cut, MDF fixture. The laser-tracker ground truth system is used to evaluate the 6DOF pose of a fiducial in Cartesian space, while the two fixture-based systems are intended to provide *a priori* pose data based on known transformations from a reference position via mechanical offsets relative to a given sensor under test. A comparative matrix of measurement errors and variances is given in Table 8.

The evolution of the ground truth systems demonstrate a growing trend in modularity, and an emphasis in lowering cost to make the solutions more accessible to researchers. These are in-line with ongoing standards efforts at NIST, and are being integrated by the ASTM E57.02 standards committee for 6DOF static pose estimation system evaluation. The cost-to-modularity ratios inherent with these efforts are illustrated in Table 9. As was seen, however, a consequence of emphasizing lower cost and modular design is an increase in measurement error and uncertainty.

7. ACKNOWLEDGMENTS

We thank Gerry Cheok from NIST for her technical support in this study. We would also like to thank Tommy Ji of the University of Maryland, College Park for his work in generating conceptual designs of the MDF Mechanical Fixture.

8. REFERENCES

[1] Chang, T., Hong, T., Falco, J., Shneier, M., Shah, M., and Eastman R. 2010. Methodology for evaluating static six-degree-of-freedom (6DoF) perception systems. In *PerMIS '10: Proceedings of the Performance Metrics for Intelligent Systems Workshop*. 2010.

[2] Devore, J. L. *Probability and Statistics for Engineering and the Sciences*. Brooks/Cole Publishing Co., Monterey, CA. 93940. 1987.

[3] Shah, M., Chang, T., and Hong, T. 2009. Mathematical Metrology for Evaluation of 6DOF Visual Servoing System. In *PerMIS '09: Proceedings of the 9th Workshop on Performance Metrics for Intelligent Systems*. 2009.

[4] SMARTTRACK SENSOR. 2007. Automated Precision Inc. http://www.apisensor.com/PDF/SpartTrackeuDE.pdf. 2007.

[5] Nikon Coordinate Measuring Machines. 2011. Nikon Metrology. http://www.nikonmetrology.com/optical_cmm/. 2011.

[6] Vicon Motion Systems. 2011. Oxford Metrics Group PLC. http://www.vicon.com/. 2011.

[7] Madsen, C. B. 1997. A Comparative Study of the Robustness of two Pose Estimation Techniques. *Machine Vision and Applications*. 9, 5-6 (1997), 291-303.

[8] Yoon, Y., DeSouza, G.N., and Kak, A.C. 2003. Real-Time Tracking and Pose Estimation for Industrial Objects Using Geometric Features. In *Proceedings of the IEEE International Conference on Robotics and Automation*. 3473-3478. 2003.

[9] Shakunaga, Takeshi. 1992. An Object Pose Estimation System Using a Single Camera. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*. 1053-1060. 1992.

[10] Wülfing, J., Hertzberg, J., Lingemann, K., Nüchter, A., Wiemann, T., and Stiene, S. 2010. Towards Real Time Robot 6D Localization in a Polygonal Indoor Map Based on 3D ToF Camera Data. In *Proceedings of the 7th IFAC Symposium on Intelligent Autonomous Vehicles*. 2010.

[11] Chang, T., Hong, T., Shneier, M., Holguin, G., Park, J., and Eastman, R.D. 2008. Dynamic 6DOF Metrology for Evaluating a Visual Servoing System. In *PerMIS '08. Proceedings of the 8th Workshop on Performance Metrics for Intelligent Systems*. 2008.

A Proxemic-Based HRI Testbed

Zachary Henkel
Texas A&M Univeristy
College Station, Texas
zmhenkel@cse.tamu.edu

Robin Murphy
Texas A&M Univeristy
College Station, Texas
murphy@cse.tamu.edu

Vasant Srinivasan
Texas A&M Univeristy
College Station, Texas
vasants@cse.tamu.edu

Cindy L. Bethel
Mississippi State University
Starkville, MS USA
cbethel@cse.msstate.edu

ABSTRACT

This paper describes a novel, low cost HRI testbed for the evaluation of robot movement, gaze, audio style, and media content as a function of proximity. Numerous human-robot interaction studies have established the importance of proxemics in establishing trust and social consonance, but each has used a robot capable of only some component, for example gaze but not audio style. The Survivor Buddy proxemics testbed is expected to serve as blueprint for duplication or inspire the creation of other robots, enabling researchers to rapidly develop and test new schemes of proxemic based control. It is a small, four-degree of freedom, multi-media "head" costing approximately $2,000 USD to build and can be mounted on other robots or used independently. To enable proxemics support, Survivor Buddy can be coupled with either a dedicated range sensor or distance can be extracted from the embedded camera using computer vision.The paper presents a sample demonstration of proxemic competence for Survivor Buddy mounted on a search and rescue robot following the victim management scenario developed by Bethel and Murphy.

Categories and Subject Descriptors

I.2.9 [**Artificial Intelligence**]: Robotics

General Terms

Human Factors

Keywords

Human-Robot Proxemics, Robot Scaling Functions, Robot Approach Behavior, Social Robots

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
PerMIS'12 March 20-22, 2012, College Park, MD, USA
Copyright 2012 ACM 978-1-4503-1126-7-3/22/12 ...$10.00.

1. INTRODUCTION

A major topic in human-robot interaction (HRI) is how robots can provide socially consistent responses to the humans operating in close proximity, such as a help-desk or receptionist robot or a robot medic. As predicted by Reeves and Nass [19], human-robot proxemics appears to be a direct transfer of human-human proxemics. Hall [6] asserted that humans interact with each other differently based on their proximity to one another and divided interaction space into four zones: Intimate space, Personal space, Social space and Public space. Each zone manifests its own characteristics. For example, an interaction in the Intimate zone usually involves a voice kept to a whisper, while interactions in the Public zone present a loud voice. Argyle [2] later refined each zone into their currently accepted ranges (see Fig. 1), while Bethel et al [3] synthesized the social literature into a comprehensive model of proxemics for robots. Over 22 studies to date have confirmed some aspect of human-human proxemic behavior occurring in human-robot interactions [1, 3, 5, 7–13, 16–18, 20, 21, 23–29], though no study has been able to address proximity in its entirety. Indeed, the literature suggests that researchers have only begun to skim the surface of this important topic.

The lack of a proxemics oriented robotics testbed is a barrier to further research, which is addressed by this paper through the presentation of Survivor Buddy 2.0. The paper first reviews the literature to postulate a super-set of the six known proxemic dependent attributes that should be supported by the testbed: *affective movement, proxemic readings, voice interactions, audio style manipulation, gaze control,* and *media content delivery.* The paper then presents Survivor Buddy 2.0, a low-cost robot "head" which was developed to meet these demands. Survivor Buddy 2.0 also encapsulates the proxemic functionality into a tactical behavior [14], which isolates proxemic adaptation from the nominal behavior, allowing experimentation with different scaling functions (e.g., linear, exponential) or proxemic strategies without changing any other behaviors of the robot. In addition, Survivor Buddy 2.0 can be mounted to another robot for mobile interactions or remain stationary.

2. RELATED WORK

There are over 22 human-robot interaction studies that have explored some aspect of proxemics [1, 3, 5, 7–13, 16–18, 20, 21, 23–29]. These studies have used more than eight dif-

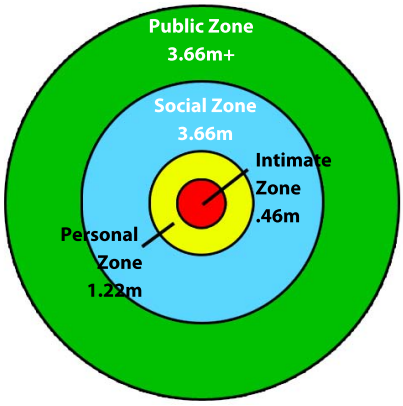


Figure 1: Argyle's Proxmity Zones.

ferent robots to examine at least five facets of proxemics (robot-human distance, social gaze, acceptable motions, audio response, and angle of approach), making it difficult to replicate results or extend systems. Thus the literature shows that understanding proxemics is an accepted and critical component of social robots, yet there is very little collaboration, reuse or standardization of the platforms used to evaluate proxemics in social robots. This absence coupled with the criticality of proxemics highlights the need for a proxemic-based HRI testbed that can be either stationary or mobile.

2.1 Types of Proxemic Research in HRI

There have been five research foci in proxemics, showing the richness of the topic in HRI. One foci seeks to answer what distance from a robot is considered comfortable for the human. Work by Mitsunage et al [11], Oosterhout [16], Syrdal et al [23], Takayama et al [24], and Walters [29] all explore the appropriate distance a robot should maintain while interacting with a human. Results have varied from proposed frameworks which automatically determine the appropriate distance [29], to somewhat confounding results that indicate a complex personality-based nature to the problem [16]. Others conducting these studies have found attributes like pet ownership and gender to be of importance [24]. Some have attempted learning user preference [11] or using feedback as a reward to a learning system to better shape proxemic behavior [10]. Significant technical work has also been completed on maintaining the appropriate distance throughout an interaction including human movement [17].

A second foci is social gaze as a function of proxemics, where the duration and type of gaze depends on proximity. Young-Min et al [9] created a fuzzy logic system to obtain and maintain gaze with a human based on proxemic conditions. Mumm et al [13] recently illustrated that humans increase their distance when a robot they do not like applies an ample amount of gaze behavior toward them.

A third foci is movement of the robot and joints or effectors, with the consensus that motor or actuation speeds should be altered based upon proximity. Mizoguchi et al [12] noted that motor behavior and speed created different responses from users and later it was indicated that motor speeds and timing should be adjusted based on proximity [10]. Bethel et al [3] formalized a scheme which prescribed modifications to movements for each proximity level.

The fourth foci is audio manipulation related to proximity. Partala et al [18] found that altering audio from neutral, sad, and happy levels could successfully manipulate valence. Shiomi et al [21] illustrated that a whispering voice was more effective in convincing users to complete a task, as it encouraged a lower proximity level and a higher bond, while Walters [26] found that humans keep a further distance from a synthetic voice.

A fifth area of investigation is the angle of approach and proxemic behavior. Two studies by Walters et al [27, 28] conclude that humans much prefer the robot to approach from the left or right, rather than the front or back. These studies also indicate regardless of approach angle, humans still expect appropriate proxemic behavior from robots.

2.2 Existing Proxemic Testbeds

The review of existing studies and systems highlights the lack of a standardized or even shared proxemics platform among HRI researchers. Such a system would be of general benefit, as it would provide an easy boilerplate to use toward specific experimental needs.

Fifteen of the examined studies utilized mobile robots [1, 3, 8–12, 16, 17, 20, 23, 24, 27–29], while seven others worked with stationary robots which observed the environment from a set location [5, 7, 13, 18, 21, 25, 26]. In studying social gaze as a function of proxemics, Mumm [13] utilized a stationary robot capable of autonomously maintaining gaze. Similarly, studies investigating using a whispering voice robot [21], as well as those evaluating voice styles of a mechanical looking robot [26] also used stationary platforms. One experiment interested in voice manipulation [18] coupled with proxemics created a simulation based method for creating proxemic situations, limiting the need for any robotic platform at all. Most studies interested in a robot's approach to a human [27, 28] or maintaining an appropriate proxemic distance [17] have all operated mobile robots, capable of traversing their environment. Of the studies utilizing mobile robots, most have been of a form factor which requires ample space to move about. Many studies have used systems similar to the PeopleBot configuration [8] [28] [23] [27] [11] [16]. Only Adalgeirsson [1] and Bethel [3] have employed portable or non-anthropomorphic robots. Adalgeirsson's MeBot is an attachment for a mobile phone, while Bethel's work involved search and rescue robots, capable of navigating inside rubble.

Sixteen of the 22 studies [1, 3, 9–13, 16, 17, 20, 24–29] surveyed have performed experiments which rely on actual autonomous systems, with the remainder using Wizard of Oz control of the robot or simulation [5, 7, 8, 18, 21, 23]. Researchers are often interested in a specific attribute and use simulated approaches to other aspects of their experiments. For example, Mumm et al [13] utilized a pre-recorded human voice in order to establish the likeability of their robot in social gaze testing. Walters [26] powered up a robot's fans, servos, and sensors into an idle state in order to give a more "live" feel to experiments focusing on voice. Additionally, most have performed studies in controlled lab environments, while a few have experimented in more everyday situations. For example, Ooserhout et al performed experiments regarding robotic proximity during a three day arts and technology festival [16]. Many rely on the laboratory setting or at least a controlled condition as it allows for more reliable capture

of sensory data. Additionally, Wizard of Oz approaches are sometimes used in order to ensure repeatability or to enable a task the robot would otherwise find difficult. For example, Sydral et al used a Wizard of Oz approach in their study of the relationship between individual differences and proxemic behaviors [23]. Huettenrauch et al also used a Wizard of Oz approach to their study of spatial distancing in a co-presence situation [8]. Finally, Partala et al [18] have looked to novel methods like changing image sizes on a screen in order to simulate proximity, rather than using a physical robot at all.

3. APPROACH

The approach taken to create a proxemic competent HRI testbed is twofold: design and build a *physically competent robot* and create *a software architecture* that allows proxemics to be inserted in a quantifiable, reproducible way avoiding ad hoc implementations.

3.1 Necessary Physical Capabilities

Taken together, the HRI literature suggests that a proxemically competent robot should have at least six capabilities:

- **Affective expressiveness**, where the testbed has sufficient degrees of freedom and resolution of joint control to produce a scalable set of motions [1] [3] [7] [10] [12] [20] [25].
- **Audio Control**, where the audio volume has sufficient range to extend across all zones [5] [12] [18] [21] [25] [26].
- **Gaze Control**, where the testbed has the sensors and processing power to support gaze control algorithms such as eye and face tracking [10] [13] [9].
- **Media Content Control**, where the testbed can present and control a variety of media (web, video, voice, videoconferencing, etc.) [1].
- **Approach Control**, where the testbed degrees of freedom allow it change approach angles yet still accomplish the mission, particularly approaching from the side (non-threatening) while maintaining eye contact (showing engagement) [1] [3] [8] [10] [11] [16] [17] [23] [24] [28] [27] [29].
- **Proxemic Awareness**, where the testbed either has proxemic sensing or can access range data from a host robot or external sensor [1] [3] [8] [10] [11] [12] [13] [16] [17] [18] [23] [24] [25] [28] [27] [29] [26] [9].

3.2 Software Architecture

The approach to software is to encapsulate proxemics into a tactical behavior following [14], compatible with behavioral and hybrid deliberative/reactive architectures. This architecture is captured in Fig. 2. The nominal behaviors are a mapping of $B(s) = r$, where strategic behaviors (or operator commands) b produce motor and media output or responses, r, from sensor input s. These outputs are then filtered by the proxemic behavior, which acts to apply a gain or bias function based on sensed distance, s_{range}, leading to a new behavioral mapping: $B_{proxemic}(s_{range}, B) = r'$. Note that the range sensor can be either dedicated to the proxemic behavior or shared with the nominal behaviors, as is consistent with behavioral robotics.

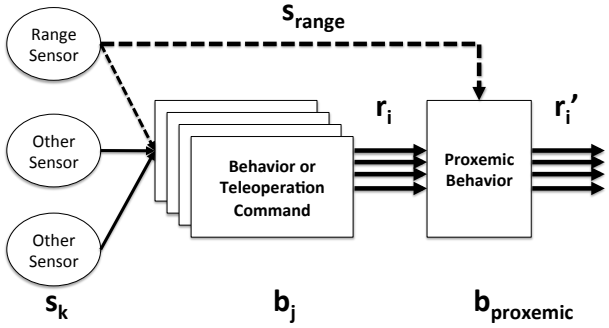


Figure 2: The proxemic behavior as a tactical behavior, where $B_{proxemic}(s_{range}, B) = r'$.

The advantage of this approach is that it isolates proxemic adaptation from the nominal behavior. This means an agent (human or computer) can compute r without having to be aware of s_{range} proxemics. The proxemic behavior can apply different scaling functions (e.g., linear, exponential) or proxemic strategies without changing any other behaviors of the robot, thus allowing direct comparison of proxemic interactions.

A proxemic behavior consists of a minimum of five components, each assigned a scaling function: *joint movement range, joint movement speed, approach speed, gaze control,* and *audio control*. More components can be added as warranted by the research. The scaling function is a collection of scaling functions for each zone: Intimate, Personal, Social, and Public.

4. IMPLEMENTATION

The implementation of the novel HRI testbed capturing the six capabilities described above is called Survivor Buddy. The implementation consists of two major systems: the *platform itself* and the *proxemic sensing system*. Proxemic sensing is discussed separately because it can be done in a number of ways (through the Survivor Buddy video camera, through a dedicated range finder, or through sharing range data from the base robot). For this version of Survivor Buddy, 2.0, proxemic sensing is done with a dedicated range finder, though in the future it is expected that the testbed will have onboard algorithms for using optical flow with the video camera.

Survivor Buddy enables proxemic behavior through allowing the scaling of several attributes, including: joint movement range, joint movement speed, approach speed, gaze control, and audio control. By default, each attribute uses a linear drop-off function for scaling, though it is possible to apply non-linear functions as well. In the base implementation, the maximum value of a feature (joint movement range, joing movement speed, approach speed, gaze control, audio control) is used when the human is at a distance greater than approximately three meters. From approximately three meters to half of a meter the value of each feature is scaled linearly, with its acceptable minimum being placed at the half-meter meter distance. These ranges and functions are

configurable to meet other requirements as well. These parameters were chosen as defaults because they provide a simple starting point for exploration.

4.1 The Survivor Buddy Platform

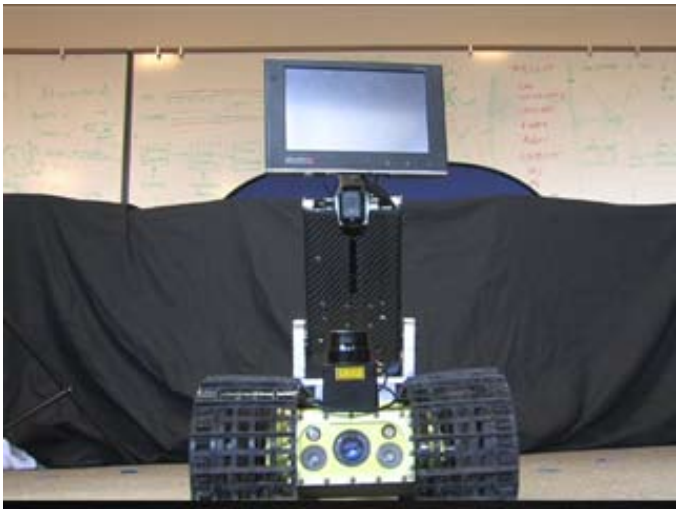


Figure 3: Survivor Buddy 2.0 with a dedicated Hokuyo proxemics sensor mounted to a ASR/Inuktun Extreme urban search and rescue robot.

The Survivor Buddy 2.0 platform consists of the effectors and multi-media monitor, control software, and a Voice Toolkit. The Survivor Buddy 2.0 effector design is described in [15] and costs approximately $2,000 USD to build. The robot "head," seen in Fig 3, has four degrees of freedom, a footprint of 14cm X 9.5cm and weighs only 1.72 kg. It is easily attachable to other robot bases, but also has the ability to be mounted in a standalone fashion. Survivor Buddy's "head" is a small, 7-inch MIMO 740 touchscreen monitor, which also contains a webcam and microphone with a speaker system mounted in the "neck". Actuation is through Dynamixel motors, allowing high velocities at each joint in order to enable fluid motion.

Survivor Buddy is controlled through a tether to a netbook and power supply, using a software framework for position control, affective behaviors, shared autonomy, resource control and Windows extended desktop to control media content. The software framework is written in C#, communicates directly with each of the four motors. The software also includes a Voice Toolkit that enables communication through text to speech packages, voice recognition engines, and live streaming of voice. The system can be used by both local and remote users, allowing communication from across the internet.

Survivor Buddy was designed to be mounted to most robot bases, as well as mounted on a stationary object for general HRI interaction studies. A stationary mounting is ideal for allowing humans to approach the robot, as well as interacting with multiple humans within a space; thus it could be used for experiments such as [5, 7, 13, 18, 21, 25, 26].

4.2 Six Capabilities

The Survivor Buddy 2.0 design meets the six necessary capabilities, as described below:

- **Affective expressiveness**: Survivor Buddy 2.0 has four degrees of freedom, a head tilt, pan, and roll and a neck that raises and lowers.
- **Audio Control**: Survivor Buddy 2.0 has onboard speakers and microphone.
- **Gaze Control**: Survivor Buddy 2.0 has sufficient degrees of freedom to establish agency, communicate social attention (e.g., maintain eye contact), regulating the interaction process, projecting mental state, and manifesting interaction content [22].
- **Media Content Control**: the Windows extended desktop allows an operator or agent to display any application running on the netbook or through an internet connection on the Survivor Buddy 2.0 monitor.
- **Approach Control**: Survivor Buddy 2.0 has sufficient degrees of freedom to "turn sideways" to "soften" the angle of approach, even if the base robot approaches the human straight on.
- **Proxemic Awareness**: Survivor Buddy 2.0 can accept range readings from an external sensor or use the built-in webcam with a stereo range, optical flow, or other depth algorithm.

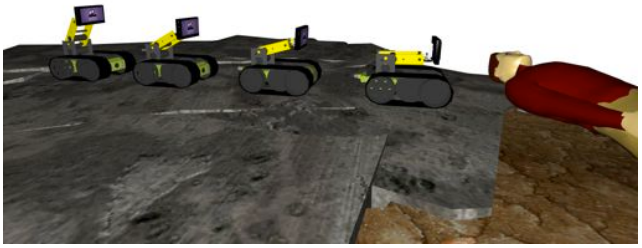
4.3 Proxemics Sensing

For the initial demonstration of proxemic competence, the proxemics sensing platform was a dedicated Hokuyo URG-04LX Laser Range Finder mounted on the robot base. It is capable of providing 240 degrees of range data with a resolution of .36 degs/step. Power and communications are through a separate tether using a USB connection.

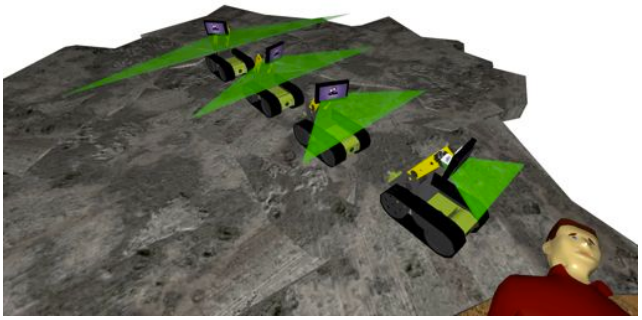
The sensing platform is controlled by a custom C# program developed to provide fast and meaningful proxemics data to active behaviors. The program, layered in style, is first responsible for the management of the laser hardware and the delivery of updated reading information on a continuous basis. The readings are delivered through a subscriber service, where all subscribers are pushed new information as it becomes available. The service provides angle, distance, and timestamp. Additionally, a higher level program, a subscriber to the lower level program, provides tracking data of human objects within the current environment. This program likewise allows subscribers, and delivers information such as human size, location, velocity, and time.

5. SEARCH AND RESCUE VICTIM MANAGEMENT EXAMPLE

The scenario from [3,4] was used to demonstrate the proxemic competence of the Survivor Buddy HRI testbed. In the scenario a robot searches through rubble for survivors, then interacts with the survivor. Bethel [3,4] showed in 128 human-subject trials that there was strong evidence manifested in difference of respiration rates which indicated stress increased significantly for a robot that did not obey proxemic "rules" compared to one that did. In this demonstration which duplications the path and position of the robot to a victim, the Survivor Buddy head is mounted on an ASR/Inuktun Extreme. Survivor Buddy remains in a folded



(a)



(b)

Figure 4: Simulation of Survivor Buddy as it moves from the personal space to intimate space of the victim from the side (a) and from above (b).

up, low-profile position until a survivor is located. At that point, the Survivor Buddy raises up and begins an interaction session with the user, modifying the control based on proxemics.

Fig. 4 shows how Survivor Buddy would visibly change pose and the range of motion of a "no" (manifesting interaction content) as it moves from the victim's *personal space* to their *intimate space*. Fig. 5 provides a filmstrip of the actual robot responding to the decreasing distance by decreasing the volume (shown on the Survivor Buddy monitor using a graphics equalizer type of display), decreasing the range and velocity of motion (shown by the bars marking the change in extent of motion), and change in pose (illustrated by a side view).

6. CONCLUSIONS AND FUTURE WORK

Though there are at least 22 studies focused on proxemics-based attributes of HRI, there has yet to be a common or comprehensive proxemics-based testbed. Survivor Buddy 2.0 is a complete proxemics-based HRI testbed. It consists of a physical component capable of proxemic awareness, audio, gaze, media content, and approach control as well as affective expressiveness. The software architecture component treats the proxemic capabilities as an independent tactical behavior, allowing the application of of different scaling functions or proxemic strategies without changing any other behaviors of the robot. As demonstrated with a canonical victim management scenario, Survivor Buddy 2.0 is small enough to be mounted on another robot for mobile interactions where a robot approaches a person as well as used for the more common situations of a person approaching the robot.

Future work includes adding support for additional sensing hardware, offering more software frameworks for behavior design and scaling complex behaviors as a function of proximity as well using Survivor Buddy 2.0 for human-subject tests of autonomous gaze control. Survivor Buddy 2.0 design drawings and software are available upon request.

7. REFERENCES

[1] S. Adalgeirsson and C. Breazeal. Mebot: A robotic platform for socially embodied telepresence. In *Human-Robot Interaction (HRI), 2010 5th ACM/IEEE International Conference on*, pages 15 –22, 2010.

[2] M. Argyle. *Bodily communication*. Routledge, second edition, 1975.

[3] C. L. Bethel and R. R. Murphy. Non-facial/non-verbal methods of affective expression as applied to robot-assisted victim assessment. In *Proceedings of the ACM/IEEE international conference on Human-robot interaction*, HRI '07, pages 287–294, New York, NY, USA, 2007. ACM.

[4] C. L. Bethel and R. R. Murphy. Non-facial and non-verbal affective expression for appearance-constrained robots used in victim management. *International Journal of Behavioral Robotics*, in print 2011.

[5] H. Bouraoui, A. Khamis, and F. Krray. A testbed platform for assessing human-robot verbal interaction. In *Autonomous and Intelligent Systems (AIS), 2010 International Conference on*, pages 1 –6, 2010.

[6] E. T. Hall. *The Hidden Dimension*. Anchor, Oct. 1966.

[7] J. Harris and E. Sharlin. Exploring emotive actuation and its role in human-robot interaction. In *Human-Robot Interaction (HRI), 2010 5th ACM/IEEE International Conference on*, pages 95 –96, 2010.

[8] H. Huettenrauch, K. Eklundh, A. Green, and E. Topp. Investigating spatial relationships in human-robot interaction. In *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, pages 5052 –5059, 2006.

[9] Y.-M. Kim and D.-S. Kwon. A fuzzy intimacy space model to develop human-robot affective relationship. In *World Automation Congress (WAC), 2010*, pages 1 –6, 2010.

[10] N. Mitsunaga, C. Smith, T. Kanda, H. Ishiguro, and N. Hagita. Robot behavior adaptation for human-robot interaction based on policy gradient reinforcement learning. In *Intelligent Robots and Systems, 2005. (IROS 2005). 2005 IEEE/RSJ International Conference on*, pages 218 – 225, 2005.

[11] N. Mitsunaga, C. Smith, T. Kanda, H. Ishiguro, and N. Hagita. Adapting robot behavior for human–robot interaction. *Robotics, IEEE Transactions on*, 24(4):911 –916, 2008.

[12] H. Mizoguchi, T. Sato, K. Takagi, M. Nakao, and Y. Hatamura. Realization of expressive mobile robot. In *Robotics and Automation, 1997. Proceedings., 1997 IEEE International Conference on*, volume 1, pages 581 –586 vol.1, Apr. 1997.

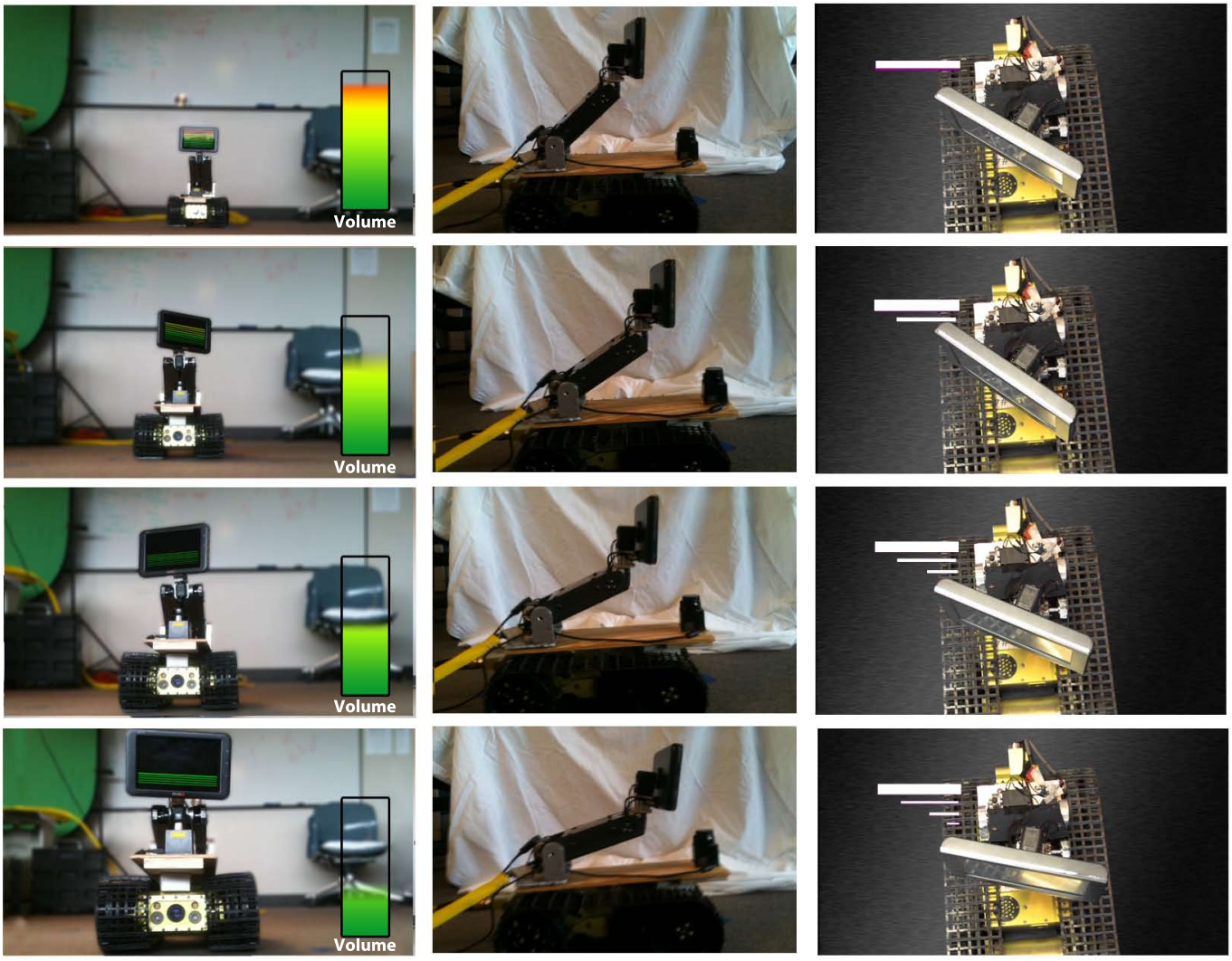


Figure 5: Views Survivor Buddy as it moves from the personal space to intimate space of the victim. Audio volume is shown on the screen in the left set of images, lowering on approach in center, and decreasing magnitude of the range of head motion of "no" is indicated by bars in the third set of images.

[13] J. Mumm and M. B. Human-robot proxemics: Physical and psychological distancing in human-robot interaction. In *Proceedings of the 6th ACM/IEEE Conference on Human-Robot Interaction (HRI'11)*, volume 1, Apr. 2011.

[14] R. Murphy. *Introduction to AI Robotics*. MIT Press, Cambridge, MA, 2001.

[15] R. Murphy, A. Rice, N. Rashidi, Z. Henkel, and V. Srinivasan. A multi-disciplinary design process for affective robots: Case study of survivor buddy 2.0. In *ICRA 2011 to appear*, 2011.

[16] T. v. Oosterhout. A visual method for robot proxemics measurement. In *Proceedings of Metrics for Human-Robot Interaction: A workshop at the Third ACM/IEEE International Conference on Human-Robot Interaction (HRI08)*, volume 1, pages 66–68, Apr. 2008.

[17] A. Oskoei, W. M., and D. M.L. An autonomous proxemic system for a mobile companion robot. In *Artificial Intelligence and Simulation of Behaviour (AISB 10): Second International Symposium on New Frontiers in Human-Robot Interaction.*, volume 1, pages 66–68, Apr. 2010.

[18] T. Partala, V. Surakka, and J. Lahti. Affective effects of agent proximity in conversational systems. In *Proceedings of the third Nordic conference on Human-computer interaction*, NordiCHI '04, pages 353–356, New York, NY, USA, 2004. ACM.

[19] B. Reeves and C. Nass. *The media equation: how people treat computers, television, and new media like real people and places*. Cambridge University Press New York, NY, USA, 1996.

[20] M. Saerbeck and C. Bartneck. Perception of affect elicited by robot motion. In *Human-Robot Interaction (HRI), 2010 5th ACM/IEEE International Conference on*, pages 53 –60, 2010.

[21] M. Shiomi, K. Nakagawa, R. Matsumura, K. Shinozawa, H. Ishiguro, and N. Hagita. Could i have a word?effects of robot's whisper. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pages 3899 –3904, 2010.

[22] V. Srinivasan and R. Murphy. A survey of social gaze. In *Proceedings of the 6th international conference on Human-robot interaction*, HRI '11, pages 253–254, New York, NY, USA, 2011. ACM.

[23] D. Syrdal, K. L. Koay, M. Walters, and K. Dautenhahn. A personalized robot companion? - the role of individual differences on spatial preferences in hri scenarios. In *Robot and Human interactive Communication, 2007. RO-MAN 2007. The 16th IEEE International Symposium on*, pages 1143 –1148, 2007.

[24] L. Takayama and C. Pantofaru. Influences on proxemic behaviors in human-robot interaction. In *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, pages 5495 –5502, 2009.

[25] T. Tasaki, S. Matsumoto, H. Ohba, M. Toda, K. Komatani, T. Ogata, and H. Okuno. Dynamic communication of humanoid robot with multiple people based on interaction distance. In *Robot and Human Interactive Communication, 2004. ROMAN 2004. 13th IEEE International Workshop on*, pages 71 – 76, 2004.

[26] M. Walters, D. Syrdal, K. Koay, K. Dautenhahn, and R. te Boekhorst. Human approach distances to a mechanical-looking robot with different robot voice styles. In *Robot and Human Interactive Communication, 2008. RO-MAN 2008. The 17th IEEE International Symposium on*, pages 707 –712, 2008.

[27] M. L. Walters, K. Dautenhahn, and Woods. Robotic etiquette: results from user studies involving a fetch and carry task. In *Proceedings of the ACM/IEEE international conference on Human-robot interaction*, HRI '07, pages 317–324, New York, NY, USA, 2007. ACM.

[28] M. L. Walters, K. Dautenhahn, and S. N. Woods. Exploratory studies on social spaces between humans and a mechanical-looking robot. *Connect. Sci.*, 18(4):429–439, 2006.

[29] M. L. Walters and R. Dautenhahn, Te Boekhorst. An empirical framework for human-robot proxemics. In *Proceedings New Frontiers in Human-Robot Interaction, K. Dautenhahn (Ed.), symposium at the AISB09 convention*, pages 144–149. SSAISB, 2009.

Synergistic Methods for using Language in Robotics

Ching L. Teo
University of Maryland
Dept of Computer Science
College Park, Maryland 20742
+01 3014051762
cteo@cs.umd.edu

Yezhou Yang
University of Maryland
Dept of Computer Science
College Park, Maryland 20742
+01 3014051762
yzyang@cs.umd.edu

Cornelia Fermüller
University of Maryland
Institute for Advanced
Computer Studies
College Park, Maryland 20742
+01 3014051743
fer@umiacs.umd.edu

Yiannis Aloimonos
University of Maryland
Dept of Computer Science
College Park, Maryland 20742
+01 3014051768
yiannis@cs.umd.edu

ABSTRACT

This paper presents an overview of our work on integrating language with vision to endow robots with the ability of complex scene understanding. We propose and motivate the Vision-Action-Language loop as a form of cognitive dialogue that enables us to integrate current tools in linguistics, vision and AI. We present several experimental results of preliminary implementation and discuss future research directions that we view as crucial for developing the cognitive robots of the future.

Categories and Subject Descriptors

I.2.9 [**Artificial Intelligence**]: Robotics; I.2.10 [**Artificial Intelligence**]: Vision and Scene Understanding—*perceptual reasoning*

General Terms

Theory, Algorithms

Keywords

Cognitive Robotics, Computer Vision, Computational Linguistics

1. INTRODUCTION

A *cognitive robot* is a robot capable of simulating cognitive processes that mimic human intelligent behavior requiring capabilities such as visual perception, sensorimotor activation and high-level reasoning. In this paper, we argue that *Language* is an important, and till now an overlooked component that is crucial for developing cognitive robots. As we will show in sec. 3, language, when processed appropriately, can be leveraged to bridge the so-called *semantic* gap between low-level sensory signals (visual, auditory, haptics etc.) and high-level concepts (words, ideas etc.). In this work, we focus on visual signals, and show how we can use the Vision-Action-Language loop depicted by Fig. 1 as a form of cognitive dialogue to facilitate several important vision tasks: 1) Object recognition, 2) Action recognition and 3) Scene description. We first motivate why language is useful for cognitive robots followed by an overview of the cognitive dialogue framework.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage, and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
PerMIS'12 , March 20-22, 2012, College Park, MD, USA.
Copyright © 2012 ACM 978-1-4503-1126-7-3/22/12 ...$10.00.

1.1 Why Language for Robotics?

Let us examine in some more detail what is really going on when a human (a cognitive system with vision and language) is interpreting a visual scene. When we fixate at an object and recognize it, then this means an immediate entry to the linguistic system. Indeed, if we recognize a "street", the word street lights up in the linguistic system, with a number of consequences. The word "street" has many "friends". These are other words that tend to co-occur with "street", such as "human", "car", "house", etc. Modern computational linguistics has created, using a large corpus, resources where this information can be obtained, e.g. probability distributions for the co-occurrence of any two words, lists of the friends of any word, and so on. Thus, recognizing a noun in the scene creates expectations for the existence of other words in the scene that vision can check for. In this case, **language acts as a contextual system** hat aids perception. There is however much more than this. Let's say you are in a kitchen. Because you have prior knowledge about kitchens, their structure and the actions taking place in them and a large part of this knowledge is expressed in language, we can utilize this information during visual inspection. A knife in the kitchen will most probably be used for "cutting" a food item, so the vision can look for it. In this case, **language acts as a high level prior knowledge system** that aids perception. There is still more. Let's say you observe someone pick up an object, put it in the trunk of a car, then get into the car and drive away. Given this, you know that the object is gone, it is inside that car. In this case, **language acts as part of a reasoning process.**

When we visually inspect a scene, it appears that our linguistic system is working in the background together with visual perception to achieve meaning and understanding. This is an aspect of perception that has not been studied systematically. There has been a lot of work on what could be called "parallel vision", i.e. given an image or an image sequence, how do we find edges, contours, motions and other features, how do we segment the scene and group the features into objects, etc. On the other hand, "sequential vi-

sion" has not received as much attention. As you interpret a visual scene, you fixate at some location and you recognize nouns, verbs, adjectives, adverbs and prepositions. Because the linguistic system is highly structured, these recognitions produce a large number of inferences about what could be happening in the scene. This leads you to fixate at a new location, and the same process is repeated. In this case, **language acts as part of an attention mechanism.**

Thus, language is beneficial not so much for communication, but for facilitating the shaping of different cognitive spaces. Finally, it should be clear that instead of language one could use a formal system with properties like the ones of language. The symbols of the system would be labels of the different concepts that the system possesses and they would have to obey a number of constraints. Language gives us this for free. In the next section, we describe how the Vision-Action-Language loop integrates language to realize some of the uses that was described here.

1.2 The Vision-Action-Language Loop

The Vision-Action-Language loop is depicted in Fig. 1.

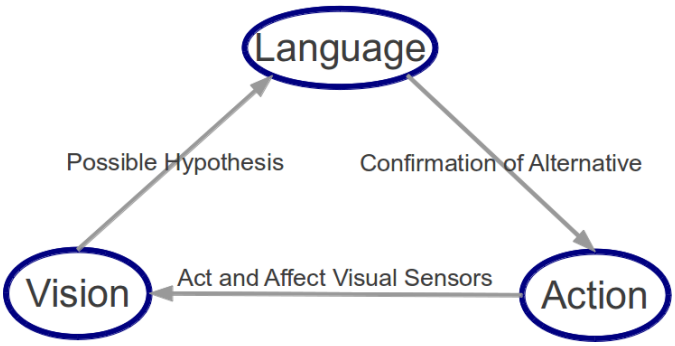


Figure 1: The Vision-Action-Language Loop.

Each of the three nodes can be seen as a distinct process (an *executive*) in the Robot's operating system . The *Visual* executive takes care of low-level visual processing associated with the task at hand: e.g, segmenting an object, or extracting certain visual features. The output of the visual executive are a set of possible hypothesis on the task which is then passed on to the *Language* executive. The Language executive will then act as a reasoner, using high-level knowledge embedded in language to decide which, if any, of the hypothesis makes sense; and provide reasonable alternatives. The output of the Language executive is therefore a set of potentially modified hypothesis which can be acted upon by the *Action* executive. Based on the set of modified hypothesis, the Action executive will then decide the most appropriate next course of action that will affect the visual sensor: e.g. to move to a new location or to change the sensors' pan-tilt-zoom (PTZ) unit. This Vision-Action-Language loop continues until the Action executive decides that a certain end goal or objective had been realized which is then relayed to the rest of the robot's operating system. We call it a cognitive "dialogue" as the three executives are constantly working in a synergistic manner to update each others prior beliefs, so as to achieve a shared goal or objective together.

2. RELATED WORKS

The use of language in robotics has been pursued recently in the fields of Computer Vision, AI and Robotics. We highlight a few prominent related studies in these areas.

In the field of computer vision, the classical view of Marr and others [17] considered language to be part of high-level vision, diametrically opposed to the low-level visual processes that processes the signals directly. As a result, language was only used "at the end" of the visual processing pipeline. With advances on textual processing and detection, several works recently focused on using sources of data readily available "in the wild" to analyze static images. The seminal work of [4] showed how nouns can provide constraints that improve image segmentation. [9] (and references herein) added prepositions to enforce spatial constraints in recognizing objects from segmented images. [1] processed news captions to discover names associated with faces in images, and [11] extended this work to associate poses detected from images with the verbs in the captions. Some studies also considered dynamic scenes. [2] studied the aligning of screen plays and videos, [15] learned and recognized simple human movement actions in movies, and [10] studied how to automatically label videos using a compositional model based on AND-OR-graphs that was trained on the highly structured domain of baseball videos The work of [5] attempts to "generate" sentences by first learning from a set of human annotated examples, and producing the *same* sentence if both images and sentence share common properties in terms of their triplets: (Nouns-Verbs-Scenes). No attempt was made to generate *novel* sentences from images beyond what has been annotated by humans.

In AI, the use of language had been largely confined to classical problems in computational linguistics: 1) speech recognition 2) language modeling (e.g. machine translation) and 3) text generation. In speech recognition, current approaches include automatic speech recognition and understanding, both need language information as prior knowledge. For language modeling, the work of IBM models uses large parallel text corpus to build HMM style language models [12], and then apply it into several applications, such as machine translation. In terms of text generation, classic approaches [25] are based on three steps: selection, planning and realization. A common challenge in generation problems is the question of: what is the input? Recently, approaches for generation have focused on formal specification inputs, such as the output of theorem provers [20] or databases [6]. Most of the effort in those approaches has focused on selection and realization.

State of the art robotics uses language as a communication system; conversational robots of the new millennium have more or less sophisticated mechanisms to map words to related sensorimotor experiences so that they engage into more natural human robot interaction (e.g. [18], cf. also [22] for an extensive review). Language has been used to trigger action-sensory state associations ([26]) or predesigned control programs ([16]); mappings from natural language to symbolic logic or temporal logic and then to basic control primitives of the robot ([13]) have also been developed for controlling robots with high level task descriptions. The system of [19] describes model that enables the agent to ground evidences from multiple modalities: language, vision, etc. However, none of these approaches takes advantage of language as a contextual system and as part of a reasoning system. With the exception of a few notable approaches on understanding of gestures by robot platforms (cf. for example [14]) or using visual scenes to prime speech understanding ([23]) there has not been much work on scene interpretation by robotic agents. There are many reasons for this, but basically computer vision solutions developed in the image/video databases arena that use language as a contextual system do not transfer to robots.

3. INTEGRATING LANGUAGE

In this section we present preliminary implementations of the Vision-Action-Language cognitive dialogue on three tasks: 1) Object recognition, 2) Action recognition and 3) Scene description.

For each task, we highlight how each implementation is related to the cognitive dialogue and summarize the results from experiments performed on a robot that is endowed with the presented algorithms.

3.1 Attributes-Based Object Recognition

The key goal of any object recognition task is to provide distinct labels to objects within the image. For language to be integrated into this task, we propose to use *attributes* that link visually extracted information to textual descriptions that humans would use to describe these objects. An attribute can be defined as a property that is *innate* to the object, and as a result is *invariant* under most circumstances. In addition, the use of attributes has strong links to human perception [3]. Such properties makes attribute detection an important capability for cognitive robots. Our approach first segments the image into foreground regions and background, and then computes on the foreground object attribute properties. In this study we focused on shape properties. Since our application was the description of kitchen tools (3.4 we have identified the following five computable attributes:

Is elongated: An ellipse was fitted to the mask provided by segmentation, and the ratio of major to minor axis was used to set a threshold (Fig. 2a).

Is round: If the ratio of major to minor axis is about the same, the object was considered round.

Has a handle: If the error from fitting two separate ellipses was lower than from fitting a single ellipse, the object was considered having a handle (Fig. 2b).

Is a container: Depth discontinuities were found in the depth mask. If the object could be segmented into parts, with one part of a mostly concave depth map and the other part of a mostly convex depth map, the object was considered a container (Fig. 2c).

Has a flat part: If an object was classified as consisting of two parts by the 2D shape attribute method, a plane was fitted to the larger part.

(a) (b) (c)

Figure 2: Examples of shape attributes (a) *Elongated* (b) *Has handle* (c) *Is a container*.

The Language Executive is a simple language model that uses the attributes extracted by the Visual executive to perform a classification of the object's identity as shown in Fig. 3 using a decision tree based classifier.

3.2 Action Recognition

For this task, we are interested in recognizing actions associated with certain hand-tools. The basic intuition is to exploit the close semantic relationship between actions and tools in a large text corpus to improve the recognition of actions and tools in the visual space. The basic framework is summarized in Fig. 4

The Visual Executive extracts visual features related to the action (trajectories of hand) and tools. It then performs a classification of these features to produce initial hypothesis of their labels, which is expected to be noisy. The Language Executive first creates a language model that gives the conditional probability of how likely an action has occurred given the tool. This was done by mining a large text corpus [8] for correlated tools and actions. We then combined the probabilities to determine the final labels of tool and associated action. This step can be repeated in a few iterations, where at each iteration, we retain only the top N hypothesis of actions and tools until we do not see any significant updates or only a single pair of tool and action exists.

3.3 Scene Description

The goal of this task is to produce a textual description of an image or video sequence based on a triplet $\mathcal{T}$ of objects, actions and environments (locations) that co-occur in the scene. The full details of the implementation are described in [27], and we link it to the Vision-Action-Language loop described here. The key component of the approach in [27] is the dynamic programming optimization of an HMM that integrates language and visual input as shown in Fig. 5.

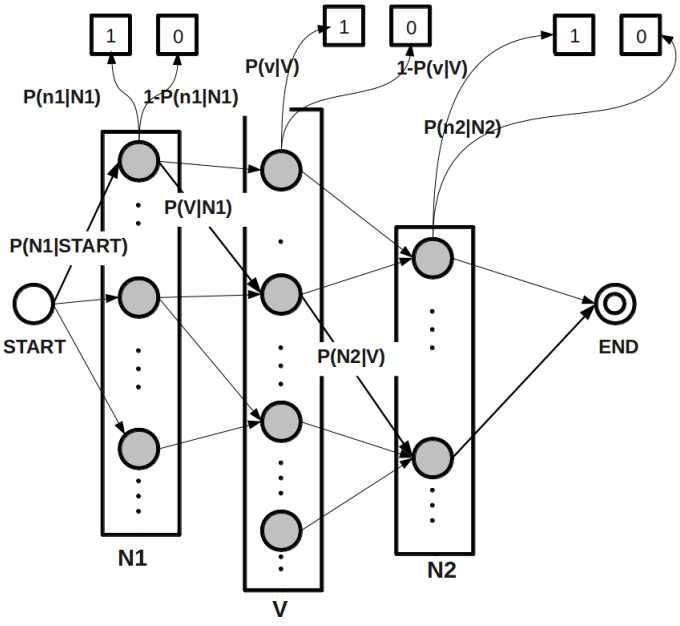


Figure 5: The HMM used to predict the optimal triplet $\mathcal{T}$: N_1, N_2 corresponds to objects and tools, and V corresponds to verbs (actions). The relevant transition and emission probabilities are also shown. See text for more details.

The key idea to this approach is to model the detection scores from visual object and scene detections as *emissions* (observations) in the HMM. This is the Visual Executive in the framework. The transition probabilities, learned from the same large text corpus [8], describe how the different components of $\mathcal{T}$ relate to each other. This forms the Language Executive. Optimizing over the HMM essentially finds the most likely $\mathcal{T}$ that supports both visual observations and linguistic correctness, which simulates the cognitive dialogue between the processes. A template based method of generating sentences is then used to generate a descriptive sentence from $\mathcal{T}$.

3.4 The Telluride Experiments

The algorithms described in the preceding sections are implemented on a mobile robot whose goal is to observe a human perform certain actions with kitchen tools and to ultimately generate a

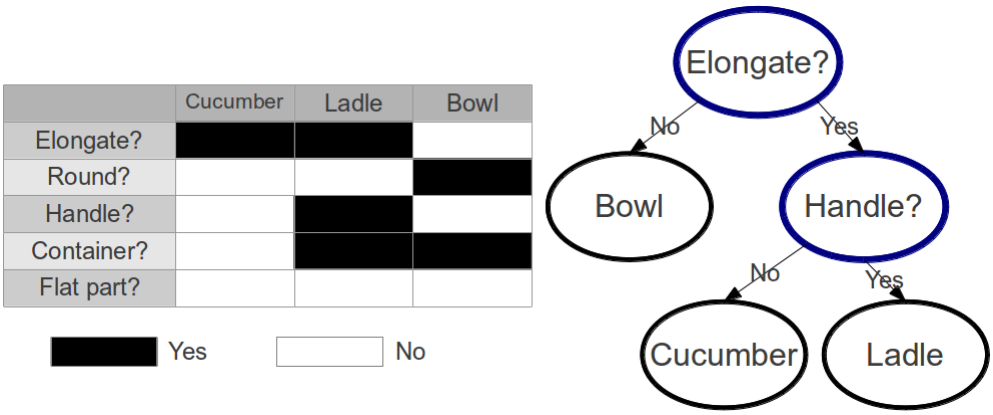


Figure 3: Example of using attributes for object recognition.

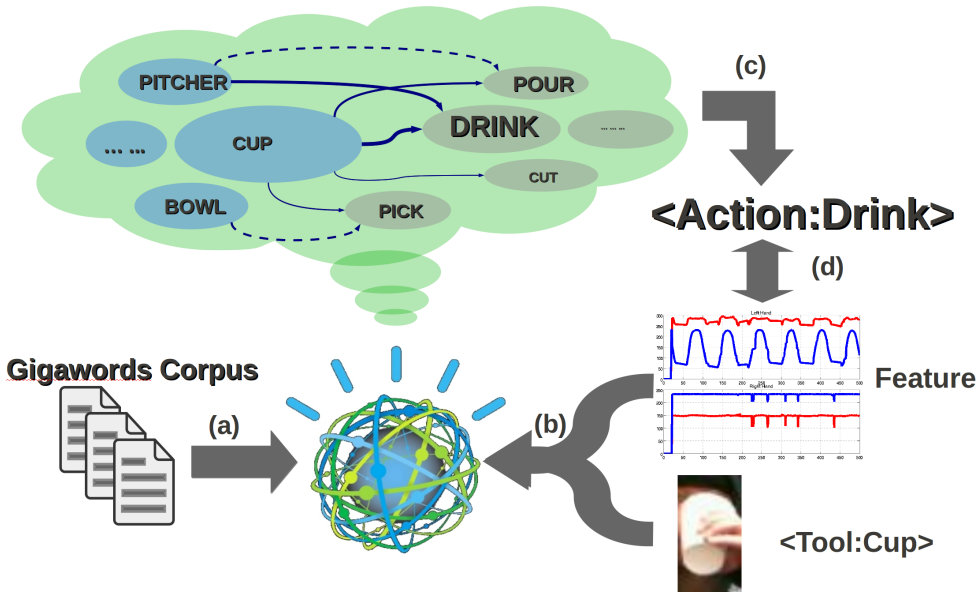


Figure 4: Key components of the approach.: (a) Training the language model from a large text corpus. (b) Detected tools are queried into the language model. (c) Language model returns prediction of action. (d) Action features are compared and beliefs updated.

sentence that describes the actions. All the experiments were conducted during the 2011 Telluride Neuromorphic Workshop[1], and we first describe the experimental setup and procedure and report accuracy results.

3.4.1 Experimental Setup

The robot (Fig. 6), is looking at the table where humans perform tasks using a number of tools and objects. The session begins with a number of objects, $o \in O$ and tools $t \in T$ on the table which the robot observes. Then a person approaches and begins an action $a \in A$, out of set of $|A|$ actions.

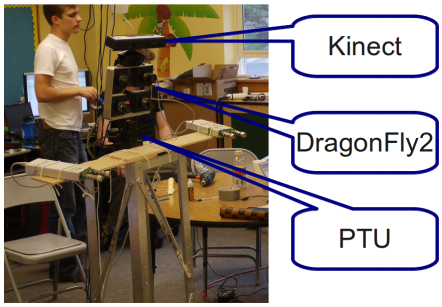


Figure 6: The Telluride Robot used in the experiments with its sensory hardware.

The robot first extracts visual features of objects and tools from the table and attempts to label them using attributes as described in sec. 3.1. This yields a set of scores over all objects o and tools t. When the action starts, it tracks the hand and elbow locations of the human (using the on-board kinect sensor) to extract action features (velocity and Fourier coefficients). Together with the labels of the tools, we use the approach described in sec. 3.2 to compute a detection score for each action a. With these initial detection scores, we use the algorithm described in sec. 3.3 to generate the final triplet $\mathcal{T}$ of object, tools and actions in order to generate a reasonable sentence that describes the scene. The overview of the processing pipeline is shown in Fig. 7.

The experimental test dataset consists of 9 actions: A={slice, mash, peel, chop, pour, stir, toss, sprinkle, pour} performed by 2 different human actors using 9 common tools: T={knife, masher, peeler, pitcher, ladle, fork/spoon, shaker, mug, bowl} and 7 other objects: O={bowl, mug, tomato, cucumber, coffee, soup, salt}. In total, there are 18 video clips, each with 9 actions performed by the 2 actors.

3.4.2 Results

The output of the initial visual processes is the triplet of $\mathcal{T} = \{a, o, t\}$ of action, objects, tools associated with the video observed. The initial output triplet $\mathcal{T}_1, \mathcal{T}_2$ (one for each actor) is then passed on to scene description algorithm (sec. 3.3) which then modifies the triplet if necessary to form the final output triplet $\mathcal{T}_1^*, \mathcal{T}_2^*$. We evaluate the effectiveness of our approach by comparing the overall recognition accuracy Acc, computed as the weighted average from the recognition of the three components in $\mathcal{T}_{1,2}$ and $\mathcal{T}_{1,2}^*$ with the ground truth. The results over the 18 videos are summarized in Table 1.

These results show that on average, we are able to improve upon the recognition accuracies of objects, tools and actions from pure visual processes with the help of the Language Executive. Mistakes still occur and this is because we have not exploited the "Action" Executive of the cognitive dialogue. We address this issue (along with others) in the next section.

[1]http://ine-web.org/telluride-conference-2011/telluride-2011/index.html

Test Video (Truth)	$\mathcal{T}_1, \mathcal{T}_2 (Acc)$	$\mathcal{T}_1^*, \mathcal{T}_2^* (Acc)$
{slice,tomato,knife}	{slice,tomato,knife} {slice,tomato,knife}(1.0)	{slice,tomato,knife} {slice,tomato,knife}(1.0)
{mash,bowl,masher}	{mash,bowl,mug} {sprinkle,bowl,mug}(0.5)	{mash,bowl,masher} {sprinkle,bowl,shaker}(0.67)
{peel,cucumber,peeler}	{toss,cucumber,peeler} {peel,cucumber,peeler}(0.83)	{peel,cucumber,peeler} {peel,cucumber,peeler}(1.0)
{chop,cucumber,knife}	{chop,cucumber,knife} {mash,cucumber,knife}(0.83)	{chop,cucumber,knife} {chop,cucumber,knife}(1.0)
{toss,bowl,fork/spoon}	{toss,bowl,fork} {toss,bowl,spoon}(1.0)	{toss,bowl,fork} {toss,bowl,spoon}(1.0)
{sprinkle,bowl,shaker}	{sprinkle,cucumber,bowl} {sprinkle,bowl,shaker}(0.83)	{sprinkle,bowl,shaker} {sprinkle,bowl,shaker}(1.0)
{stir,bowl,fork/spoon}	{pour,bowl,spoon} {pour,bowl,spoon}(0.67)	{pour,bowl,spoon} {pour,bowl,spoon}(0.67)
{pour,mug,pitcher}	{stir,mug,pitcher} {stir,mug,pitcher}(0.67)	{pour,mug,pitcher} {pour,mug,pitcher}(1.0)
{pour,bowl,ladle}	{pour,bowl,ladle} {pour,bowl,ladle}(1.0)	{pour,bowl,ladle} {pour,bowl,ladle}(1.0)
Overall	0.81	**0.93**

Table 1: Triplet accuracy: Initial predictions and final predictions

4. FUTURE WORK

In this section, we discuss possible future research directions that we believe are important for integrating language into vision and AI for solving problems of scene recognition.

4.1 Adding Action

As we have noted in sec. 3.4.2, the mistakes observed in the Telluride Experiments are due to the fact that the robot is stationary and is passively observing the scene. If the robot becomes an active mobile agent, endowed with an Action Executive, problems that had limited the visual processing performance could be mitigated via several strategies:

- Fixation based tracking: As the scene is dynamically changing, with the human actor moving from one part of the scene to another, tracking where the humans are moving the PTZ unit to focus on them will improve the recognition accuracy of the visual processing by reducing false alarms (limited search space)
- Moving to a new location: Objects and tools that are manipulated will change position throughout the process, and may become occluded from time to time. By moving closer or changing its location, the robot could actively aid the recognition by re-tracking the occluded objects or bringing them closer, aiding visual processing.
- Reacting in a reasonable manner: Adding a robotic arm would allow the robot to directly manipulate objects, which would bring the Vision-Action-Language loop to a deeper level. For example, if an object is determined to be occluded by another in front of it, the language reasoner will hint at the robot to attempt to move the occluder so that recognition can be enhanced, by an action called "move", which is then mapped to the robot's motor system to perform the required action.

4.2 Multi-level recognition of actions

Actions are compositional in nature. Starting from simple actions occurring on a part of the body, we can compose actions from

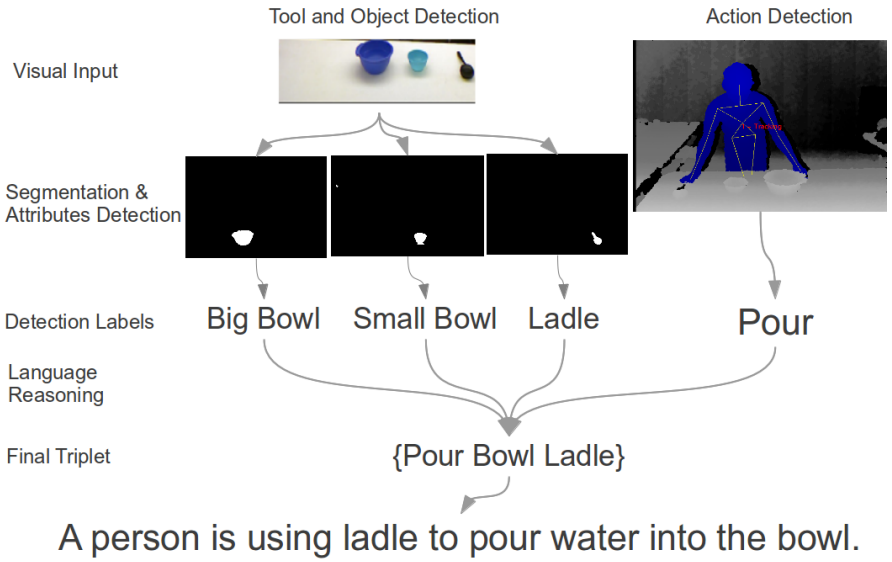


Figure 7: Processing pipeline: from visual features to sentence generation.

several limbs to create more complex actions, and we can further combine a sequence of simple actions with tools together to form an activity. Language can be used to enhance the action recognition at the higher levels and its composition from lower-levels onwards. The key idea is that Language provides a structure that enforces certain constraints on how actions can be composed. For example, focusing on hand-tools alone, there are sets of reasonable actions associated with tools (sec. 3.2). Yet, these actions together are often used to accomplish a global purpose, such as baking a cake. We are in the process of creating several datasets based on cooking recipes so that Language can be used to enforce temporal and logical constraints on how actions can be chained together. The Language executive will work across all levels, from bi-grams of actions to inferring the most likely activity from the sequence of such bi-grams, with a corpus learned from digital cooking recipes.

4.3 Mining from Text and Corpus

We have till now considered the Language Executive to be derived from static sources of corpora. However, for an active agent to be able to accommodate to changes in its surroundings, it is more practical to construct such models "on the fly". Methods such as [7] that perform approximate search through large databases are most promising. In addition, more sophisticated methods that utilize algorithms for relational database mining can be used to extract indirect correlations between objects and their attributes. One interesting way is to exploit relevant questions that humans pose for such objects, and use them to infer possible attributes: e.g. "Is X round? Is Y sharp?". Additionally, one can use various bootstrapping algorithm e.g. [24] using seeds derived from various semantic databases: ImageNet, WordNet etc [21] to extract adjectives where such objects occur.

5. CONCLUSIONS

In this paper, we have argued for the importance of exploiting language in the context of endowing artificial agents with cognitive capabilities. We have demonstrated how the Vision-Action-Language loop can be viewed as a cognitive dialogue between various processes, and we have implemented this dialogue on three tasks, namely object action, and scene recognition. Experiments on our data collected at Telluride confirm that language is a powerful tool which improved object, tool and action recognition. We also discussed potential directions for future work needed to complete the Vision-Action-Language framework in more general settings and for active mobile agents.

6. ACKNOWLEDGMENTS

The support of the European Union under the Cognitive Systems program (project POETICON) and the National Science Foundation under the Cyberphysical Systems Program and the Institute for Neuromorphic Engineering, is gratefully acknowledged. Ching Teo and Yezhou Yang are supported in part by the Qualcomm Innovation Fellowship.

7. REFERENCES

[1] T. L. Berg, A. C. Berg, J. Edwards, and D. A. Forsyth. Who's in the picture? In *NIPS*, 2004.

[2] T. Cour, C. Jordan, E. Miltsakaki, and B. Taskar. Movie/script: Alignment and parsing of video and text transcription. In *ECCV*. 2008.

[3] A. Desolneux, L. Moisan, and J. M. Morel. *From Gestalt Theory to Image Analysis*, volume 34. 2008.

[4] P. Duygulu, K. Barnard, J. F. G. de Freitas, and D. A. Forsyth. Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. In A. Heyden, G. Sparr, M. Nielsen, and P. Johansen, editors, *ECCV (4)*, volume 2353 of *Lecture Notes in Computer Science*, pages 97–112. Springer, 2002.

[5] A. Farhadi, S. M. M. Hejrati, M. A. Sadeghi, P. Young, C. Rashtchian, J. Hockenmaier, and D. A. Forsyth. Every

picture tells a story: Generating sentences from images. In K. Daniilidis, P. Maragos, and N. Paragios, editors, *ECCV (4)*, volume 6314 of *Lecture Notes in Computer Science*, pages 15–29. Springer, 2010.

[6] D. Golland, P. Liang, and D. Klein. A game-theoretic approach to generating spatial descriptions. In *Proceedings of EMNLP*, 2010.

[7] A. Goyal and H. Daumé III. Approximate scalable bounded space sketch for large data NLP. In *Empirical Methods in Natural Language Processing (EMNLP)*, Edinburgh, Scotland, 2011.

[8] D. Graff. English gigaword. In *Linguistic Data Consortium, Philadelphia, PA*, 2003.

[9] A. Gupta and L. S. Davis. Beyond nouns: Exploiting prepositions and comparative adjectives for learning visual classifiers. In D. A. Forsyth, P. H. S. Torr, and A. Zisserman, editors, *ECCV (1)*, volume 5302 of *Lecture Notes in Computer Science*, pages 16–29. Springer, 2008.

[10] A. Gupta, A. Kembhavi, and L. S. Davis. Observing human-object interactions: Using spatial and functional compatibility for recognition. *IEEE Trans on PAMI*, 31(10):1775–1789, 2009.

[11] L. Jie, B. Caputo, and V. Ferrari. Who's doing what: Joint modeling of names and verbs for simultaneous face and pose annotation. In NIPS, editor, *Advances in Neural Information Processing Systems*, NIPS. NIPS, December 2009.

[12] D. Jurafsky and J. H. Martin. *Speech and Language Processing (2nd Edition) (Prentice Hall Series in Artificial Intelligence)*. Prentice Hall, 2 edition, 2008.

[13] H. Kress-Gazit, G. Fainekos, and G. Pappas. From structured english to robot motion. In *IEEE/RSJ Conference on Intelligent Robots and Systems*, San Diego, CA, 2007.

[14] V. Krüger, D. Herzog, Sanmohan, A. Ude, and D. Kragic. Learning actions from observations' robotics and automation magazine. *Robotics and Automation Magazine*, 17(2):30–43, 2010.

[15] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld. Learning realistic human actions from movies. In *CVPR*, 2008.

[16] C. Madden, M. Hoen, and P. Dominey. A cognitive neuroscience perspective on embodied language for human-robot cooperation. *Brain and Language*, 112:180–188, 2010.

[17] D. Marr. *Vision*. W.H. Freeman, San Francisco, CA, 1982.

[18] C. Matuszek, D. Fox, and K. Koscher. Following directions using statistical machine translation. In *Proceeding of the 5th ACM/IEEE International Conference on Human-Robot Interaction*, 2010.

[19] N. Mavridis and D. Roy. Grounded situation models for robots: Bridging language, perception and action. In *In Proceedings of the AAAI-O5 workshop*, pages 32–39, 2005.

[20] K. McKeown. Query-focused summarization using text-to-text generation: When information comes from multilingual sources. In *Proceedings of the 2009 Workshop on Language Generation and Summarisation (UCNLG+Sum 2009)*, page 3, Suntec, Singapore, August 2009. Association for Computational Linguistics.

[21] G. A. Miller. Wordnet: A lexical database for english. *Communications of the ACM*, 38:39–41, 1995.

[22] K. Pastra. *Vision-Language Integration: a Double-Grounding Case*. PhD thesis, Department of Computer Science, University of Sheffield, 2005.

[23] D. Roy and N. Mukherjee. Towards situated speech understanding: visual context priming of language models. *Computer Speech & Language*, 19(2):227–248, Apr. 2005.

[24] M. Thelen and E. Riloff. A bootstrapping method for learning semantic lexicons using extraction pattern contexts. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing - Volume 10*, EMNLP '02, pages 214–221, Stroudsburg, PA, USA, 2002. Association for Computational Linguistics.

[25] D. Traum, M. Fleischman, and E. Hovy. Nl generation for virtual humans in a complex social environment. In *In Proceedings of he AAAI Spring Symposium on Natural Language Generation in Spoken and Written Dialogue*, pages 151–158, 2003.

[26] J. Weng. Developmental robotics: Theory and experiments. *International Journal of Humanoid Robotics*, 1:199–236, 2004.

[27] Y. Yang, C. Teo, H. Daume, and Y. Aloimonos. Corpus-guided sentence generation for natural images. *EMNLP*, 2011.

Reusable Semantic Differential Scales for Measuring Social Response to Robots

Lilia Moshkina
College of Computing
Georgia Institute of Technology
Atlanta, GA, USA
+1-916-872-0557
lilia@gatech.edu

ABSTRACT
This paper presents eight novel reusable semantic differential scales measuring a variety of concepts relevant to the field of social HRI: Understandability, Persuasiveness, Naturalness, Appropriateness, Welcome, Appeal, Unobtrusiveness and Ease. These scales were successfully used in two HRI experiments, and were found to have acceptable (> 0.7) or higher levels of internal reliability. These scales are reusable and were designed to simplify comparison between HRI studies, especially in the area of social robotics, where measuring the quality of interaction and social response to robots is of paramount importance.

Categories and Subject Descriptors
H.1.2 [**Models and Principles**]: User/Machine Systems – *human factors.*

General Terms
Measurement, Human Factors

Keywords
Social robotics, Measurement, Semantic Differential Scales

1. INTRODUCTION
As robots move more and more from highly specialized domains into everyday use, it becomes important to assess the social response that they invoke in people they interact with. System and task performance measures, though objective, reflect only one side of the story – how well the participants or robots could perform a task, rather than how satisfactory, easy, persuasive or pleasant the interaction with a robot was. Moreover, for many aspects of social robotics strictly performance measures may not even be applicable, and assessing the subjective quality of interaction becomes crucial. For example, in the area of affective HRI it would be useful to know whether people find certain affective behaviors in robots more persuasive, natural and welcoming than others, or whether robotic personality makes some collaborative human-robot tasks seem more appealing and less arduous. The answers to these questions would inform future robot design, thus enhancing the quality of human-robot interaction.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage, and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
PerMIS'12, March 20-22, 2012, College Park, MD, USA. Copyright © 2012 ACM 978-1-4503-1126-7-3/22/12...$10.00

Currently, self-assessments are among the most commonly used methods of evaluation in HRI studies; they allow querying people's perceptions of their interaction through self-reports. Unfortunately, given the early stages of HRI research, such questionnaires are often put together in an ad hoc manner to suit a particular study, making replication of results and comparison between different studies extremely difficult. Reusable psychometric scales measuring concepts of common applicability to HRI would partially ameliorate this problem and are, therefore, in great demand.

This paper presents eight novel reusable semantic differential scales measuring a variety of concepts relevant to the field of social HRI: Understandability, Persuasiveness, Naturalness, Appropriateness, Welcome, Appeal, Unobtrusiveness and Ease. It also describes the use of these scales in two HRI experiments, and reports their internal consistency reliability.

2. RELATED WORK
Social robotics is a very young field, and only few reusable self-assessment tests are currently in existence. One of the most widely used ones is the Negative Attitudes towards Robots Scale (NARS), developed and tested by Nomura and Kanda [1-3]. This scale measures general negative attitudes towards robots via three subscales: Situations and Interactions with Robots, Social Influence of Robots, and Emotions and Interaction with Robots, with each subscale item given as a Likert-style question.

Bartneck et al. [4] present an overview of other existing scales which have been successfully used in HRI experiments and have acceptable internal reliability. These scales (most of them translated by Bartneck et al. [4] into semantic differential scales from Likert scales) measure the concepts of Anthropomorphism, Animacy, Likeability, Perceived Intelligence, and Perceived Safety of Robots.

Although not originally designed for use in robotics, NASA Task-Load Index Scale (NASA-TLX) developed by Hart et al. [5] has been employed successfully for measuring task demand on human participants involved in joint human-robot tasks (primarily via teleoperation).

Finally, there exist a number of measures of attitude and usability which were developed for non-robotics domains, such as marketing, consumer research, product/software evaluation, etc. (e.g., attitude towards the ad (Aad) measure by Burner [6], usability evaluation models TAM (Technology Acceptance Model) by Davis [7] and SUMI (Software Usability Measurement Inventory) by Kirakowski [8], and others). The use of these measures in social robotics is problematic for two reasons: 1) they

are more or less domain-specific in both wording and intent, and therefore not readily applicable for HRI; 2) even if they were applicable with minor adjustments, they may not possess the same reliability and validity when applied to robots. For example, Mutlu et al. [9] noted that some of the scales used previously to evaluate humans were not reliable in evaluating a humanoid robot in interactive studies.

Regrettably, the available self-assessment measurement tools cover but a fraction of concepts of interest to social robotics, and in order to partly fill this gap, this paper significantly expands the repository of such measures, providing a set of eight constructs successfully tested in live HRI studies, with acceptable internal reliability.

3. CONSTRUCTION OF THE SEMANTIC DIFFERENTIAL SCALES

3.1 Design considerations

The semantic differential scale, devised originally by Osgood et al. [10], is a self-assessment (rating) tool, and has been used frequently for measuring social attitudes and perceptions. Bartneck et al. [4] advocate its use for HRI evaluation over Likert scales due to consistency of presentation and reduction of acquiescence bias (common to Likert scales, which force a respondent to either agree/disagree with or report their like/dislike of a statement). In addition, once developed, these scales can be reused in other studies, thus allowing inter-study comparison.

Typically, semantic differential scale is a 5 to 9 point bipolar rating (sub)scale, with opposites at each end, and respondents are required to select the point that most closely reflects their opinion; this provides both extreme options as well as more neutral ones. By combining 3 to 10 (or sometimes even more) such subscales together, a composite scale expressing an overarching concept can be designed.

As with any evaluation measure, there are certain considerations that need to be taken into account in the development of semantic differential scales; in the design of the scales presented in this paper, special consideration was paid to a number of points brought up by Al-Hindawe [11]. In particular, the following design decisions were made:

- Both complementary opposites (e.g., sincere – insincere, conscious – unconscious) and more subtle, gradable antonyms (e.g., entertaining – boring, distracting – easy to tune out) were used, as deemed appropriate. Complementary opposites are not always available, and simple negation may project an unintended meaning; for example, using a direct opposite of quiet, "loud", would not quite relate the idea of "distracting" as opposed to simply "loud".
- 5 items (adjective pairs) per scale were chosen to provide enough information about the chosen concepts, yet not be overly tedious for the subjects to go through.
- In all the scales, negatively valenced adjectives were placed on the left, and positively – on the right. This was done for consistency, to reduce any errors due to unexpected (from the subjects' point of view) reversal of polarity.
- Five-point scales (as opposed to 7- or 9-point), although course-grained, were chosen to reduce the burden on the respondents and make grading less tedious.

As a result all eight scales followed the same format: each concept was measured by a 5-item scale, with each subscale presented as a 5-point semantic differential scale. To promote greater flexibility and adaptability, each such concept scale can be grounded by a more specific question, rather than a concept name; this allows the scales to be flexible enough to be used in a variety of scenarios and robot tasks. For example, *Persuasiveness* scale can be applied to a robot's request, message, speech, actions, etc., and would be useful in any scenario in which a robot attempts to convince participants to perform a certain task (e.g., evacuate from a dangerous zone, or perform proscribed rehabilitative exercises). Figure 1 gives an example of the Persuasiveness scale as given to experiment participants; the presentation of the scale itself was preceded by a task-specific question.

It should also be noted that, although these scales were developed with robots in mind, they could be easily applicable to a wider domain, e.g., virtual or other embodied agents.

3.2 Individual Construct Description

The scale development was subdivided into two sets, each set used in a different HRI experiment. Understandability, Persuasiveness and Naturalness constructs comprised the first set, and Appropriateness, Welcome, Appeal, Unobtrusiveness and Ease the second set; Naturalness scale was used in both studies.

Persuasiveness scale measures to what extent a robot was found to be persuasive, and can be applied to: a robot's request, message, speech, actions, etc. This construct is tied closely to task compliance; the expectation is that the more persuasive the subjects find the robot they interact with, the more willing they would be to perform the requested task. In addition to persuasiveness per se, it also incorporates the notions of sincerity, appropriateness, and convincingness.

Understandability scale measures the extent to which a robot is perceived as understandable, and can refer to: a robot's behavior, actions, speech, expressions, "state of mind", intentions, and other attributes. This construct can help explain good or poor task performance; e.g., if the robot's behavior or request is not very clear, this ambiguity could lead to confusion and performance degradation on the human's part.

Finally, *Naturalness* scale measures to what extent a robot is judged as natural/naturalistic, and can refer to either a robot as a whole, or its appearance, speech or behavior separately. This scale combines a number of subscales of two existing overlapping constructs, *Anthropomorphism* and *Animacy*, presented in Bartneck et al.[4], and eliminates redundancy. This construct contrasts a machine with a living being, and when used in conjunction with other scales (e.g., Persuasiveness), would help understand how perception of a robot as "natural" relates to other subjective impressions. This scale can also be administered to indirectly address the issue of "uncanny valley".

Table 1 shows the adjectival opposites comprising the items for each construct in the first set, where the first adjective of each pair is positioned on the left-hand side, anchored at "1", and the second – on the right-hand side, anchored at "5".

In the second set, *Appropriateness* scale measures the extent to which a robot is perceived as appropriate for a particular type of task or process; it can be used in regards to a robot as a whole, or its appearance, behavior, capabilities, and other attributes individually. To increase acceptance of new technology, it is important to determine how well people think a robot matches a particular task or situation: the better suited the robot is, the higher acceptance rates could be expected.

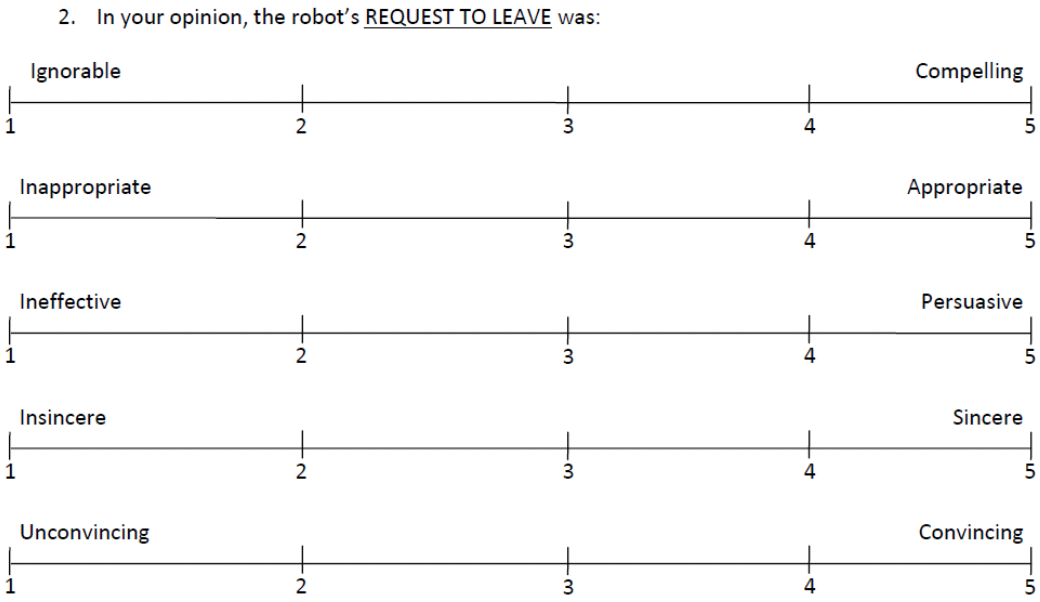


Figure 1. Persuasiveness Scale

Table 1. Adjectival Pairs comprising Understandability, Persuasiveness, and Naturalness Scales

Understandability	Persuasiveness	Naturalness
Confusing – Clear	Ignorable – Compelling	Fake – Natural
Unreadable – Easy to Read	Inappropriate – Appropriate	Machinelike – Humanlike
Inconsistent – Consistent	Ineffective – Persuasive	Unconscious – Conscious
Hard to Understand – Easy to Understand	Insincere – Sincere	Artificial – Lifelike
Inexpressive – Expressive	Unconvincing – Convincing	Inert – Interactive

Table 2: Adjectival Pairs Comprising *Appropriateness*, *Welcome*, *Appeal*, *Unobtrusiveness*, and *Ease* Scales

Appropriateness	Welcome	Appeal	Unobtrusiveness	Ease
Inappropriate – Appropriate	Unwelcome – Welcome	Boring – Interesting	Distracting – Easy to Tune Out	Hard – Easy
Wrong for Task – Right for Task	Undesired – Desirable	Not Fun – A lot of Fun	Interfering – Minding its Own Business	Complicated – Simple
Ill- Suited – Well-Suited	Disliked – Liked	Useless – Useful	Annoying – Inoffensive	Demanding – Undemanding
Improper – Proper	Tolerated – Encouraged	Dull – Exciting	Irritating – Undemanding	Long – Short
Mismatched – Matched to Task	Unwanted – Wanted	Tedious – Entertaining	Bothersome – Quiet	Complex – Basic

Table 3. Internal Consistency Reliability for *Understandability*, *Persuasiveness* and *Naturalness* scales, by condition and overall. Overall, all the scales had acceptable reliability.

Scale	Condition	Control	Mood	Combined	Overall
Understandability (5 items)	Cronbach's Alpha	.625	.810	.450	.654
	N	14	14	15	43
Persuasiveness (5 items)	Cronbach's Alpha	.825	.408	.830	.799
	N	14	14	.15	43
Naturalness (5 items)	Cronbach's Alpha	.828	.824	.632	.779
	N	14	14	15	43
Understandability (expressive excluded, 4 items)	Cronbach's Alpha	.716	.880	.254	.714
	N	14	14	15	43

Welcome scale measures to what extent a robot makes participants feel welcome, and can be applied to, for instance, their participation in a joint task, their presence, offer of assistance, etc. For example, the presentation of Welcome scale to experimental subjects described in this paper was preceded by "In your opinion, YOUR PRESENCE during the interaction with the robot was", followed by the corresponding rating subscales, but it could be adjusted quite easily to a different type of task by changing the wording of the question.

Appeal scale measures the extent to which participants find an activity involving a robot appealing; it can refer to facts or a presentation given by a robot, a meeting, a joint task, etc. This construct would be especially useful in entertainment or interactive learning domains.

To measure the extent to which a robot is perceived as distracting during a task, a meeting, or any other joint activity, a scale of *Unobtrusiveness* was developed; the lower the score, the higher the distraction due to the robot, as the negatively valenced adjectives are anchored at "1".

Ease scale measures the perceived ease of a task, a problem, or a joint project. Although not as detailed as the NASA-TLX scale [5], it provides similar overall information with much less overhead.

Finally, in this set the same *Naturalness* scale was used as before, but due to a poor intra-scale correlation result, the "inert – interactive" pair was replaced with a different activity-related pair, "inanimate – animate", which resulted in better intra-scale correlation and internal consistency. Table 2 shows the adjectival opposites for the scales in the second set.

4. USE OF THE SCALES IN HRI EXPERIMENTS

The presented semantic differential scales were originally developed for use in a set of HRI experiments assessing the effect of affective robotic behavior on participants' task performance, request compliance and subjective impressions of the robot they interacted with. The first of these, employing the scales of Understandability, Persuasiveness and Naturalness, was performed in the context of a Search-and-Rescue scenario, and the second one, employing the remainder of the aforementioned scales (plus the Naturalness scale) was set up as a Robot as a Museum Guide scenario. Two types of statistical analysis were performed to evaluate these scales based on the results of the studies: 1) factor analysis (principal components) to determine whether all the subscales within a scale refer to the same construct; and 2) internal consistency reliability test (measured by Cronbach's Alpha) which reflects the homogeneity of the scale.

4.1 Search-and-Rescue Experiment

This study was designed to evaluate the effect of robotic expressions of Negative Mood and Fear on human participants, and followed a 1-factor between-subject design with three conditions: Control (no affect was expressed by the robot), Negative Mood (the robot displayed signs of Negative Affect in response to changes in the environment), and Combined (the robot exhibited both Negative Mood and Fear when appropriate); see Moshkina [12] and Park et al. [13] for more details. A biped humanoid robot Nao by Aldebaran Robotics served as a guide at a mock-up search-and-rescue site (Figure 2 shows Nao expressing Negative Mood/Anxiety, Left and Fear, Right), and the participants played the role of a site inspector. During the site tour, when the robot perceived that the conditions became dangerous, it requested the participants to evacuate the premises. A total of 48 people participated in the experiment, out of which 43 participants had valid questionnaire data, 14 each in control and negative mood conditions, and 15 in the combined condition.

Figure 2. Nao's Expressions of Negative Affect (Left) and Fear (Right) [13].

The scales of *Understandability* (to assess how well the subjects understood the robot's behavior), *Persuasiveness* (to assess how persuasive the subjects found the robot's request to evacuate) and

Naturalness (to assess how natural the robot as a whole appeared) were presented upon the completion of the interaction portion of the experiment (after the participants "evacuated" or a certain time elapsed since the evacuation request).

As a result of factor analysis, two factors (dimensions) were extracted for both *Understandability* and *Naturalness* scales, and one for *Persuasiveness*. Further intra-scale correlations analysis showed that the "expressive – inexpressive" pair did not correlate with any other subscales within the *Understandability* scale, and removing this item resulted in a single dimension returned by a subsequent factor analysis. Similarly, removal of the "interactive – inert" pair (which was correlated with only one other subscale) from the *Naturalness* scale resulted in a single dimension, based on a subsequent factor analysis. This adjectival pair was replaced for the *Robot as a Guide* experiment with an "inanimate – animate" subscale.

To determine internal consistency reliability, Cronbach's Alpha was computed for each scale, both for each experimental condition and the experiment overall (see Table 3 for internal consistency reliability results). Overall, the scales have acceptable internal consistency, above 0.7, as recommended by Nunnally [14], with Cronbach's Alpha values ranging from 0.714 for the 4-item *Understandability* scale to 0.799 for *Persuasiveness* (after the "inexpressive – expressive" pair was removed from *Understandability* due to its poor intra-scale correlations rating). Although in some conditions the reliability was lower, it could be due a small number of respondents (14 or 15 per condition), given that the overall results reflecting a larger number of participants are better.

4.2 Robot as a Museum Guide Experiment

The goal of this experiment was to identify the effect of *Extraverted* and *Introverted personality* display by a humanoid robot on participants' task performance (to establish whether some traits are task-appropriate) and their perception of robot's appropriateness, friendliness, intrusiveness and naturalness in the context of a mock-up building demolition exhibit setting. The study followed a 1-factor between-subject design with two conditions: *Extraverted* and *Introverted*, where the display of Extraversion or Introversion served as the independent variable. Two experimental tasks were performed by participants in both conditions, with one task hypothesized to be better suited for an *Extraverted* robot, and the other for an *Introverted* robot.

In this experiment, the same humanoid robot Nao served as a guide at an explosive building demolition exhibit (please refer to Moshkina [12] for details). After a brief introduction by the robot, the subjects participated in two tasks, counterbalanced for order: a *quiz* following a presentation on building demolition by Nao, and a *math* problem solving task for which the robot served as a proctor. There were a total of 30 participants in this study, 15 per condition; the data of 15 participants in each condition were available for analysis for the *quiz* task, and the data of 14 participants in each condition for the *math* task.

Appropriateness scale, designed to measure how well the robot's behavior matched the task it was performing was used for each experimental task: *quiz* and *math*. *Welcome* scale, designed to determine how welcome the robot made the participants feel, and *Appeal* scale, designed to identify how appealing the participants found the facts presented by the robot, were given to the subjects upon the completion of the *quiz* task. *Unobtrusiveness* scale, designed to measure the level of perceived distraction due to the robot, was given after the *math* task; in this scale, the **higher** the score, the **less** distracting (or more unobtrusive) was the robot. *Ease* scale, designed to identify how easy the math problem was perceived to be was used following the math task, in conjunction with a finer-grained, but more time- and effort-consuming NASA-TLX scale [5]. Finally, the modified *Naturalness* scale was used at the conclusion of the experiment.

Similar to the scales employed for the Search-and-Rescue experiment, the same two types of statistical analysis were performed to evaluate the scales used in the Robot-as-a-Guide study. To identify whether any scales should be reduced further, Factor Analysis (principal components) was performed; each scale was found to be comprised of a single factor, reflecting the same concept. In order to determine the internal consistency reliability, Cronbach's Alpha was computed for each scale, both for each experimental condition and the experiment overall (Table 4). Overall, the alpha values showed moderate to high internal consistency for all scales, and the results per condition were all at the acceptable level as well. The internal consistency of the *Naturalness* scale was improved from 0.779 to 0.827 with replacement of the "interactive" item with "animate"; and only one factor was extracted by factor analysis for the modified scale, indicating that it reflects the measured construct better than the original one.

Additionally, Pearson's Correlations test revealed a strong negative correlation at the 0.01 level (R=-.518) between the ratings of NASA-TLX [5] and *Ease* scales: the easier the subjects found the problem, the less demanding it appeared. The results of the TLX ratings, however, provided a greater differentiation between the conditions, therefore our recommendation would be to use the *Ease* scale where reducing the effort of taking a questionnaire is important, and the effect size is expected to be large.

5. CONCLUSION

Eight novel semantic differential scales measuring a variety of concepts were presented in this paper. These scales were tested in two live HRI experiments with 48 and 30 subjects, respectively, and were found to have at least acceptable (over 0.7), but in most cases much higher (up to 0.942 for *Appropriateness*) internal consistency reliability, and therefore can be recommended for use in other HRI experiments to promote repeatability. These scales cover a variety of concepts relevant to the HRI domain, and are flexible enough to be used in a variety of scenarios and robot tasks. For example, *Persuasiveness* scale can be applied to a robot's request, message, speech, actions, etc., and would be useful in any scenario in which a robot attempts to convince participants to perform a certain task (e.g., evacuate from a dangerous zone, or perform proscribed rehabilitative exercises). Future work would include further testing of these scales in a variety of HRI studies with larger numbers of participants, and development of additional measurement tools covering a wider range of domains.

Table 4. : Internal Consistency Reliability for *Appropriateness, Welcome, Appeal, Unobtrusiveness, Ease* and *Naturalness* scales, by condition and overall

Scale	Condition	Introverted	Extraverted	Overall
Quiz Appropriateness (5 items)	Cronbach's Alpha	.923	.902	.918
	N	15	15	30
Math Appropriateness (5 items)	Cronbach's Alpha	.885	.970	.966
	N	14	14	28
Welcome (5 items)	Cronbach's Alpha	.881	.896	.914
	N	15	15	30
Appeal (5 items)	Cronbach's Alpha	.796	.837	.848
	N	15	15	30
Unobtrusiveness (5 items)	Cronbach's Alpha	.847	.970	.927
	N	14	14	28
Ease (5 items)	Cronbach's Alpha	.777	..878	.865
	N	14	14	28
Naturalness (5 items)	Cronbach's Alpha	.724	.813	.827
	N	15	14	29

6. ACKNOWLEDGMENTS

The author is grateful to Professor Ronald C. Arkin for his guidance and sponsorship of this research, and to Sunghyun Park for his indispensable help in programming the robot and preparation of the experiments. All the research presented in this paper was performed at Georgia Institute of Technology; however, the author currently holds a position as a National Research Council Post-Doctoral Research Associate.

7. REFERENCES

[1] Nomura, T., Kanda, T. 2003. On proposing the concept of robot anxiety and considering measurement of it, in *Proc. IEEE International Workshop on Robot and Human Interactive Communication*.

[2] Nomura, T., Kanda, T., Suzuki, T. 2006. Experimental investigation into influence of Negative Attitudes towards Robots on Human-Robot Interaction, in *AI&Society*, vol. 20.

[3] Nomura, T., Kanda, T., Suzuki, T., Kato, K. 2008. Prediction of human behavior in Human-Robot Interaction Using psychological scales for Anxiety and Negative Attitudes towards Robots, in *IEEE Transactions on Robotics*, vol. 24, p. 442.

[4] Bartneck C., K.D., Croft E., Zoghbi S. 2009. Measurement instruments for the anthropomorphism, animacy, likeability, perceived intelligence, and perceived safety of robots. *Int Journal of Social Robotics*, vol. 1: pp. 71-81.

[5] Hart, S.G., Staveland, L.E.1988. Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research. In *Human Mental Workload*, Hancock, P.A., and Meshkati, N., Eds, North Holland Press: Amsterdam. p. 239-250.

[6] Burner, G.C. 1998. Standardization and justification: do Aad scales measure up? *Journal of Current Issues in Research in Advertising*, 20(1), pp. 1-18.

[7] Davis, F.D. 1989. Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS Quarterly*, vol. 13, pp. 319-339.

[8] Kirakowski, J., and Corbett, M. 1993. SUMI: the Software Usability Measurement Inventory. *British Journal of Educational Technology*, 24(3), pp. 210-212.

[9] Mutlu, B., Osman, S., Forlizzi, J., Hodgins, J., and Kiesler, S. 2006. Perceptions of ASIMO: An exploration on co-operation and competition with humans and humanoid robots. In Extended Abstracts of the *Human-Robot Interaction Conference* (HRI06).

[10] Osgood, C.E., Suci, G.J., and Tannenbaum, P.H. 1957. *The Measurements of Meaning*, Champaign: University of Illinois Press.

[11] Al-Hindawe, J. 1996. Considerations when constructing a semantic differential scale. In *La Trobe Papers in Linguistics*, vol. 9.

[12] Moshkina, L. 2011. *An integrative framework of time-varying affective robotic behavior*. Ph.D. Dissertation, Georgia Institute of Technology.

[13] Park, S., Moshkina, L., and Arkin, R.C. 2010. Recognizing Nonverbal Affective Behavior in Humanoid Robots. In *Proc. of 11th Intelligent Autonomous Systems Conference*, Ottawa, Canada.

[14] Nunnally, J.C. 1978. *Psychometric theory*, New York: McGraw-Hill.

Levels of Human and Robot Collaboration for Automotive Manufacturing

Jane Shi
GM Global R&D Center
30500 Mound Road
Warren, MI 48090-9055
248-807-4212
Jane.Shi@gm.com

Glenn Jimmerson
Consultant for USCAR
1000 Town Center Drive
Southfield, MI 48075
Gjimmerson@wideopenwest.com

Tom Pearson
(Retired)
Ford Motor Company
Dearborn, MI 48126
tompearson@wideopenwest.com

Roland Menassa
GM Global R&D Center
30500 Mound Road
Warren, MI 48090-9055
586-907-1853
Roland.Menassa@gm.com

ABSTRACT
United States Consortium for Automotive Research (USCAR) conducted a concept feasibility study in 2010-2011 to investigate critical requirements to implement fenceless (the long term goal) or minimally fenced (the short term goal) robotics work cells for automotive applications. One output of the study defines the levels of human and robot collaboration and addresses the levels of complexity that drive the probabilities of successful implementation. The development of these definitions was accomplished through interviews with technology providers, observation of current robot system installations, and discussions with automotive manufacturing engineers and robotic technical experts. In this paper, we attempt to categorize robotic systems for low, medium and high levels of human and robot collaboration with current state application examples in automotive body shop, automotive powertrain manufacturing and assembly, as well as in automotive general assembly. We propose potential human and robot collaboration applications in future state where sensors, when closely integrated with robotic systems with greater dynamic response and related new technology advancements, could enable a closer and more dynamic human and robot collaboration. Finally we highlight the assessment of the successful implementation probabilities for the low, medium, and high levels of human and robot collaborative applications.

Categories and Subject Descriptors
C.3 [**Special-Purpose and Application-Based Systems**]: Process control systems; robotics, flexible automation

General Terms
Human Robot Collaboration, Flexible Robotic Assembly, Factory Automation, Automotive Manufacturing Processes

Keywords
Human Robot Collaboration, Industrial Robots, Automotive Manufacturing Processes

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage, and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. PerMIS'12, March 20-22, 2012, College Park, MD, USA. Copyright © 2012 ACM 978-1-4503-1126-7-3/22/12...$10.00

1. INTRODUCTION
Today's industrial robots are used inside heavy fence guarding and safety peripheral equipment that are costly, inflexible and bulky (configured fixed infrastructure and extra floor space). This trend is restrictive, very expensive, and inefficient with higher fences, more E-Stops, more lockouts, increased clearance distances, more set down fixtures/stations, more floor space required, more system complexity, and more ways to keep humans isolated from the robotic automation. Figure 1 is one of examples of a robotic workcell where human and robot will work on the same part in close proximity.

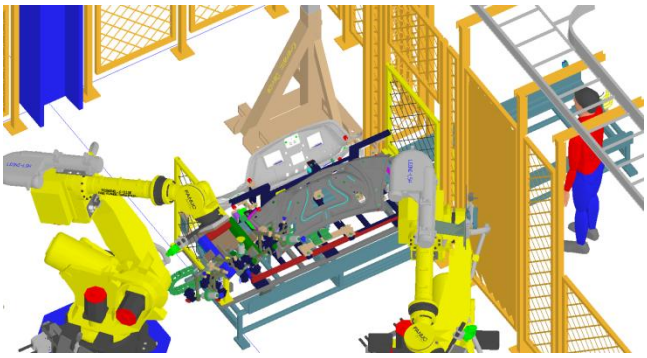


Figure 1 An Example of Automotive Workcell where Human and Robot Work in Close Proximity

To explore how fenceless or minimally fenced robotic systems can increase manufacturing flexibility, improve the efficiency, and reduce the system complexity, the United States Consortium for Automotive Research (USCAR) conducted a study in 2010-11 to investigate both social and technical feasibilities for the fenceless (the long term goal) or minimally fenced (the short term goal) robotics work cells for automotive applications.

The USCAR study covered the following areas to assess the success factors to implementing fenceless or minimally fenced robotic systems from the societal and economic perspective:

- Stakeholders who could become partners in implementing fenceless robotic work cells for automotive applications.
- Safety policy and perception that are barriers to implement safe fenceless robotic systems on the plant floor.

- Valid business cases that demonstrate the potential long term viability and economic foundation for fenceless robotic systems.

From the technical perspective, the USCAR study identified critical technology elements in establishing long term robotic sensing and control performance and standards in the following areas:

- Sensors, their capabilities, their performance, and their technical gaps for detecting human as well as object position and speed.
- Robotic control architectures that enable intrinsic robotic safety and highly dynamic adaptability to human proximity.
- Current as well as future application scenarios and their system functional requirements.

In this paper, we present one of the USCAR project outputs on the levels of human and robot collaboration. We first outline the development of the definitions, highlight three levels of human and robot collaboration applications in the current state, propose future human robot collaboration (HRC) application scenarios with required new technology capabilities with its assessment of successful implementation probabilities for the low, medium, and high levels of human and robot collaborative applications.

2. Development of Definitions

This section defines the various levels of human and robot collaboration and addresses the levels of complexity that drive the probabilities of successful implementation. The purpose of the definition is to provide consistent descriptions of the collaboration levels and align them with the manufacturing processes used to support decisions to fund future research and development of fenceless robotics systems.

The development of these definitions was accomplished through interviews of a variety of stakeholders including robot manufactures, system integrators, technology providers, safety professionals in occupational health and safety, manufacturing engineering, robotics technical specialists from automotive companies, ANSI/RIA/ISO standards committee. The observation of current robot system installations was conducted by visiting a number of manufacturing plants in both body shops and powertrain. The categorization of robotic systems for low, medium and high levels of human and robot collaboration and for probability of successful implementation has been developed as the following scenarios with specific characteristics:

- **Low**

  Three characteristics of low level human and robot collaboration are:

  - The human does not interact directly with the robot or the robot end-of-arm-tooling (EOAT).
  - When loading parts, operators load to a fixture, rotary device, or other transfer device.
  - Humans do not enter into the working range of the robot, of the end of arm tooling, or of parts being manipulated by the robot.

- **Medium**

  One or multiple operators load directly to the robot end-of-arm-tooling with following four characteristics:

  - Robot is in automatic mode.
  - Robot servo drives are de-energized.
  - Robot is extended to full extension.
  - No robot motion or EOAT motion occurs until the human exits the robot working range AND initiates a secondary input.

- **High**

  One or multiple operators and the robot perform simultaneous actions within the working range of the robot with the following four characteristics:

  - Robot is in full automatic mode.
  - Robot servo drives are energized.
  - Robot motions occur while a human is within any part of the robot full working range.
  - Robot speeds and/or motions may be modified, by the robot controller, based upon sensor inputs or communication between the robot and the human.

3. Application Examples in the Current State

This section provides several application examples in automotive body shop and powertrains in the current state of human and robot collaboration for all three levels with fences and safety peripheral equipment. Their characteristics are identified and presented.

3.1 Low Level in the Current State

Figure 2 illustrates a body shop workcell where an operator loads multiple body components into a fixture on a rotary table. The operator exits the area protected by light curtains and presses a push button to initiate the welding cycle. The rotary table rotates into the welding position and welding robots spot weld the components.

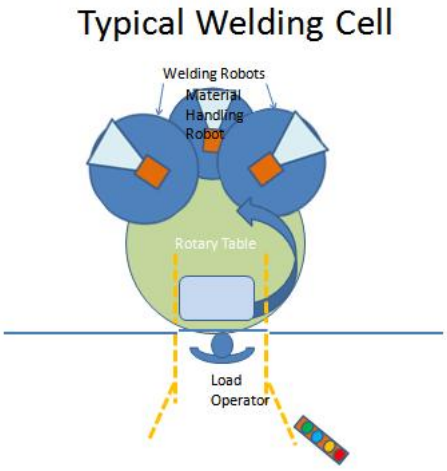


Figure 2 An Automotive Body Shop Example of Low Level Human and Robot Collaboration

While the robots are welding, the operator is loading an additional fixture on the rotary table.

When the welding cycle is complete, a material handling robot removes the welded sub-assembly for transfer to the next operation.

Once the welded sub-assembly has been removed by the material handling robot, the operator has exited the protected area, and the operator has pushed the button to signify the protected area is clear, the rotary table rotates and the cycle repeats.

Figure 3 illustrates a powertrain application where an operator obtains a transmission fluid pump from a container and places it onto a conveyor. The pump is transferred into a robotic material handling cell where it is removed from the input conveyor and loaded onto a washer pallet. The pump is then transferred into the washer system. This cell is located adjacent to an aisle that is shared between pedestrian and powered material handling vehicle (PMHV) traffic.

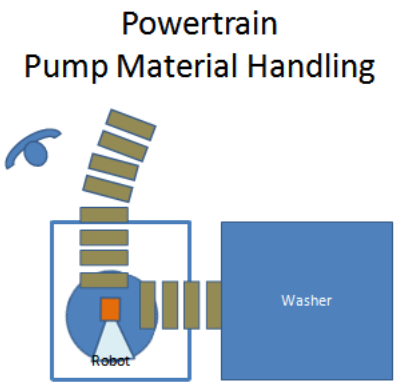


Figure 3 An Automotive Powertrain Example of Low Level Human and Robot Collaboration

The key characteristics of this low level human and robot collaboration is the use of intermediate hardware, such as the rotary table or the input conveyor, to buffer and transfer the parts between the operator and the robots. The hardware plays a role to prevent the direct interaction between the human and the robot by (1) enabling the human work outside the robot working envelop and (2) enable the human to perform his or her own tasks asynchronously with the robot. However, the hardware transfer device, such as the rotary table or the input conveyor, adds cost, takes additional space, and makes the workcell less flexible for new product changes.

3.2 Medium Level in the Current State

Figure 4 illustrates a body shop workcell where the operator loads the outer fender skin directly into the end-of-arm-tooling of the robot. Next the operator loads additional components of the fender assembly directly into the EOAT. During this loading sequence, the robot is extended to the limit of its working range and the servo drives are de-energized.

After the completion of all loading, the operator moves outside the light curtain protected area and pushes a button to initiate the EOAT clamping. The robot servo drives are energized and the robot moves into the welding position. After welding has been completed, the robot hands off the fender assembly to the next cell for additional processing.

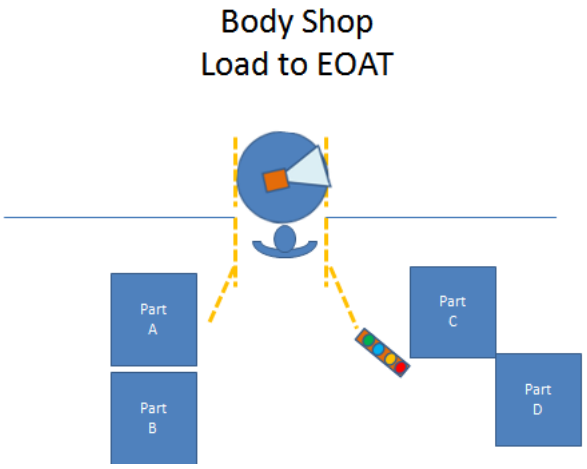


Figure 4 An Automotive Body Shop Example of Medium Level Human and Robot Collaboration

Figure 5 illustrates a powertrain RTV (room temperature vulcanizing) application where the operator opens a sliding door in the robot enclosure and loads a front engine cover into a fixture. The fixture is located within the operating area of the robot. The operator loads the part into the fixture while the robot servo drives remain de-energized and the robot is at the full extension of its operating range.

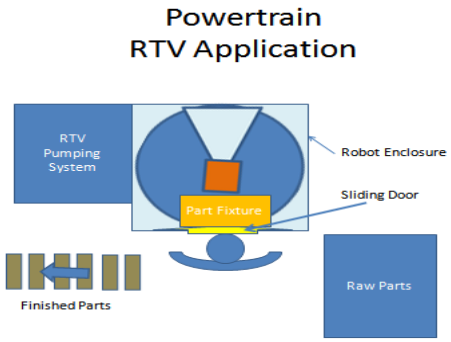


Figure 5 An Automotive Powertrain Example of Medium Level Human and Robot Collaboration

The key characteristics of this medium level human and robot collaboration is the robot operating state when the human operator is in its working envelop. The intermediate hardware, such as a fixture to hold a part for the robot to pick up from, may or may not be used. Without the hardware, the operator will directly interact with the robot within its working envelop. The operator performs his or her own tasks synchronously with the robot. This means that the robot will not continue to its next task elements until the operator initiates the robot's motion via a secondary input such as a palm button outside the robot working envelop. In

this case, the operator is pacing the workcell's throughput rate and controlling robot's productivity.

3.3 High Level in the Current State

Currently, there are no applications installed in automotive body shop or powertrain production that allow the high level human and robot collaboration with traditional industrial robots.

The only application is the use of "Intelligent Lift Assist" in the automotive general assembly as shown in Figure 6.

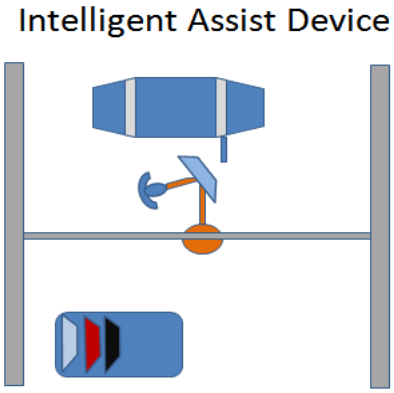


Figure 6 An Automotive General Assembly Example of High Level Human and Robot Collaboration

In this example an operator is coupled to, and in direct control of, a robotic arm (an intelligent assist device - IAD). The interface between the operator and the robotic arm functions in much the same manner as a robot teach pendant. The operator must maintain pressure on the control device, the robot speed is limited, and sensors detect rapid motions of the operator and/or the robotic arm. These devices provide the strength and accuracy of a robotic manipulator, while allowing the flexibility of path and decision making capabilities of the human operator. The only motions of the robotic arm, not under the direct control of the human operator, are a return to start function that allows the robot to return to a designated location, at slow speed with a clear signal from a laser scanner in the return area, after the human has released the controlling device.

The productive portion of the robot in the above example is when the human controls and directs the robot (IAD) and human and robot is acting as a single entity on the production line. The portion of robot automatic return to its home position is executed simultaneously while the operator is working on his or her tasks asynchronously.

4. Future State Application Examples and their Assessment of the Successful Implementation Probabilities

This section proposes a few application examples in automotive manufacturing in the future state of human and robot collaboration for all three levels in three areas of opportunity: (1) future capabilities that are better than current practices; (2) future capabilities that are currently prohibited per specifications and technology capabilities; (3) being able to envision future applications we have never considered. For each level of human and robot collaboration applications, its new or desired characteristics are compared with the current state.

The levels of human to robot collaboration, combined with the analysis of the current safety perceptions, and interviews with the technology implementers and users, provide us with a framework for estimating the potential for successful implementation of the various fenceless robotics systems. This section also presents the assessment of the probability of successful implementation of fenceless robotic systems and increased human to robot collaboration.

4.1 Low Level in the Future State

For the low level human and robot collaboration applications, all three characteristics in the future state are unchanged compared with the current state:

- The human does not interact directly with the robot or the robot end-of-arm-tooling.
- When loading parts, operators load to a fixture, rotary device, or other transfer device.
- Humans do not enter into any part of the full working range of the robot, the end-of-arm-tooling, or of parts being manipulated by the robot.

The only change in the future state is to partially or totally eliminate the physical barriers and enclosures as illustrated in Figure 7 below. In its place, safety reliable sensor systems are installed. Intrusion into the robot working space is detected by the sensor systems and robot motions are automatically e-stopped to prevent collision or contact that could result in injury or damage.

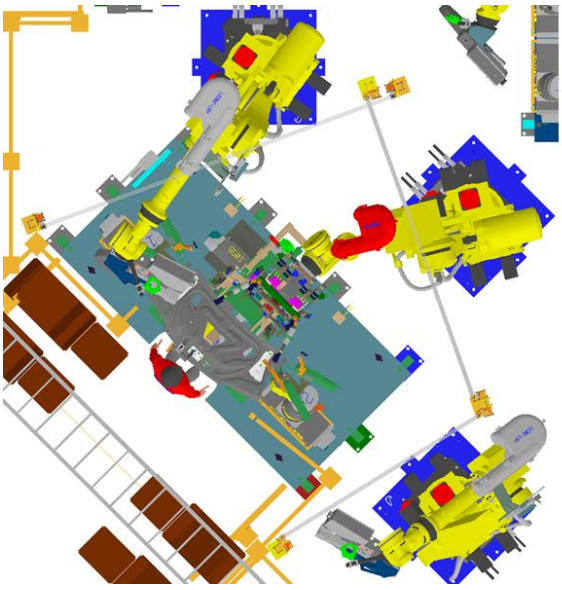


Figure 7 Proposed Future State Low Level Human Robot Collaboration Application with Sensor-based Peripheral Guarding and Safety-Rated Soft Space Limiting Technology

The key new technology capability in these future state low level applications is the detection of any un-expected intrusion in the robot full working range. The future state sensor-based peripheral guarding function is same as the current light screens and/or safety mat at the workstation cell. The performance requirement of sensor-based restricted zone should be equal or better than the current light screen guarding or safety mat monitoring in the current state.

The low level of human to robot collaboration is typically those applications that fit the "things we are doing today" opportunity. Currently, humans and robots work in close proximity, separated by physical barriers. Safety procedures identify how humans are to be protected from accidental contact with the moving robot system elements. These procedures, however, require humans to follow the procedures as written and therefore rely on human compliance in the current state. The new technology capabilities should make it not only feasible to eliminate physical barriers, but also to eliminate the reliance on human compliance to procedures in the future state.

For these reasons, the probability of successful implementation for the future "Low" level categories of human and robot collaboration and elimination of physical barriers in certain situations (with no robot carried parts) is considered to be high in next 5 years.

4.2 Medium Level in the Future State

For the medium level human and robot collaboration applications, three of four characteristics in the future state are changed and one additional significant characteristic compared with the current state:

- Robot is in automatic mode.
- Robot servo drives are energized.
- Robot may be anywhere within its operating range.
- Full robot automatic cycle resumes without the operator(s) initiating a secondary input.
- Robot motion and/or EOAT motion may occur while the human is within the robot full working range but outside restricted zone.

Current fences and enclosures can be totally or partially, eliminated and replaced with safety reliable sensor systems. Intrusion into the robot restricted space is detected by the sensor system and robot motions are automatically stopped, slowed or modified to prevent collision or contact that could result in injury or damage.

The new technology capabilities required in these medium level future state applications are (1) the safety reliable sensor systems; (2) the automatic detection of human's position relative to the robot's position; (3) the tighter and responsive integration of the sensor signals with robot's motion control systems.

Compared with the future state low level application scenario, the operator can directly interact with the robot without hardware transfer devices within the robot full working range in the proposed future state medium level applications. Without the hardware, the operator will directly interact with the robot within its working envelop. The operator performs his or her own tasks synchronously with the robot for a small portion of time when part is being loaded or unloaded as shown in Figure 8. We defined this type of human robot interaction as of "transitional" nature [5]. In the future state, the robot can continue to its next task elements without explicit secondary input from the operator and the sensory systems are responsible to detect the relative position between human and robot to resume the automatic cycle. In this case, the workcell's throughput rate and robot's productivity can be improved compared with the current state medium level applications.

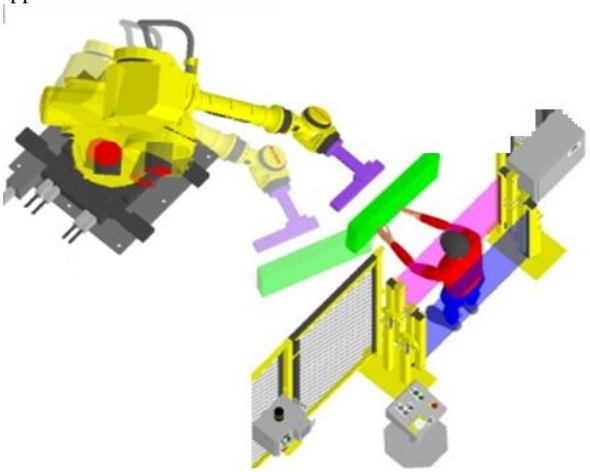


Figure 8 Proposed Future State Medium Level Human and Robot Collaboration Application with the Automatic Detection of Human's Position Relative to the Robot's Position

ANSI/RIA/ISO 10218-1-2007 5.12.3 Safety-rated soft axis and space limiting defines the safety rated space limiting function implemented with software. Commercially several robot vendors have already implemented this type of safety rated space limiting technology such as SafeMove from ABB [6], Cartesian Position Check from FANUC [7], Space monitoring functions from Kuka [8].

From the user's application perspective, the robot restricted space/zone should have the following function characteristics for the proposed future state medium level applications:

1. Multiple zones can be created.
2. Zone geometry is flexible and easily defined.
3. Zones representing end-of-arm tooling can be created.
4. Each Zone has the capability to be dynamically enabled or disabled by software during operation.
5. Robot response to intrusion into a defined zone is selectable and can be triggered by software using signals from other sensors to modify the robot speed, stop/pause the robot, or e-stop the robot.
6. Requirements for "safety reliable" are satisfied
   a. Redundancy
   b. Diversity
   c. Monitored
7. Verification of functionality.
8. Verification of supporting functions.

The "Medium" level of human to robot collaboration applications introduces one key technology capability requirement of human

position detection. The approved sensor technologies that would enable removal of physical barriers in "low" level applications typically operate on a line-of-sight requirement. In other words, the human must not be occluded form the line of vision of the applied sensor. When multiple operators and/or multiple robots are introduced into the workspace the potential for the creation of blind spots for the safety sensors is increased. In this more complex case, the probability of successful implementation for the "medium" level categories of human to robot collaboration and elimination of physical barriers is considered to be likely in the next 5 years with maturing technologies as well as the wide acceptance of the robot and human close work in close proximity concept [12].

4.3 High Level in the Future State

For the high level human and robot collaboration applications, last one of four characteristics in the future state are changed compared with the current state:

- Robot is in automatic mode.
- Robot servo drives are energized.
- Robot motions occur while a human is within any part of the robot working range.
- Robot moves in automatic programmed mode and its speeds and/or motions have to be synchronized with the human motion in a partnership fashion for the same task goal.

The ideal scenario of the high level human and robot collaboration in the proposed future state is for robots to *work as human's co-worker and partners* [5] in automatic programmed mode with motion synchronization and communication with the humans. In this regard, progress has been made through ongoing research to enable humans and robots collaboration successfully in a manufacturing assembly environment in Europe [9] as reported by Kruger et al [1] and Schraft et al [3], in Japan by Wojtara et al [2] and Tan et al [4].

In addition to the new technology capabilities in the medium level, the new technology capabilities required in the high level future state applications are (1) robots have to be situation aware [11]; (2) Mutual understanding of current task contexts and anticipation of next steps by both human and robot [10]; (3) Robots have to be able to be trained to adapt to dynamic situations using human natural communication mechanisms [13].

There are many automotive applications that could potentially benefit from this high level of human and robot collaboration technologies. Examples of this level of collaboration may include assembly operations, such as, windshield installation, headliner installation, seat installation, instrument panel installation, or any operation in which the robot manipulates and holds components in place while a human worker secures, connects, adjusts, or otherwise configures the component. This level of human and robot collaboration is believed to be the most technologically challenging and very difficult to accomplish in next 5 years.

5. Summary

In this paper, we presented one of the USCAR project outputs on the levels of human and robot collaboration. We outlined the development of the definitions, highlighted three levels of human and robot collaboration applications in the current state, proposed human and robot collaboration application scenarios with required new technology capabilities in the future state, and finally provided assessment of successful implementation probabilities for the low, medium, and high levels of human and robot collaborative applications.

6. References

[1] Kruger, J., Lien, T.K., and Verl, A. 2009. *Cooperation of Humand and Machines in Assembly Lines*. CIRP Annals - Manufacturing Technology Vol. 58, Issue 2, 2009, Pages 628-646

[2] Wojtara, T., Uchihara, M., Murayama, H., Shimod, S., Sakai, S., Fujimoto, H. and Kimura, H. 2009. *Human–robot collaboration in precise positioning of a three-dimensional object* Automatica Volume 45, Issue 2, February 2009, Pages 333-342

[3] Schraft, R.D., Meyer, C., Parlitz, C., Helms, E., *PowerMate – A safe and intuitive robot assistant for handling and assembly tasks*, Proceedings of the 2005 IEEE International Conference on Robotics and Automation, 2005, pp. 4074-4079

[4] Tan, JTC., Duan, F., Zhang, Y., Watanabe, K. , Kato, R., Arai, T. *Human-Robot Collaboration in Cellular Manufacturing: Design and Development*, Proceedings of the 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2009, pp. 29-34

[5] Shi, J., Menassa, R. *Transitional or Partnership Human and Robot Collaboration for Automotive Assembly*, Proceedings of PerMIS 2010

[6] ABB Robotics Web accessed Jan. 30, 2012 http://www.abb.com/product/us/9AAC910011.aspx

[7] FANUC Robotics Web http://www.fanucrobotics.com accessed Jan. 30, 2012

[8] KUKA Robotics http://www.kuka-robotics.com/en/ accessed Jan. 30, 2012

[9] International Federation of Robotics (IFR) "*Standardization Activities Prepare for Future Safe Human-Robot-Collaboration Revision of ISO 10218*", Sept. 2011 http://www.ifr.org/news/ifr-press-release/standardisation-activities-prepare-for-future-safe-human-robot-collaboration-290/ accessed Jan. 30, 2012

[10] Hoffman, G., and Breazeal, C., *Effects of Anticipatory Action on Human-Robot Teamwork*, Proceedings of HRI 2007

[11] Endsley, M. R., Bolt´e, B., and Jones, D. G., *Designing for Situation Awareness: An Approach to User-Centered Design*. New York: Taylor and Francis, 2003.

[12] Goodrich, M. A. and Schultz, A. C. *Human–Robot Interaction: A Survey* Foundations and Trends in Human–Computer Interaction Vol. 1, No. 3 (2007) 203–275

[13] Green, Billinghurst, Chen and Chase: *Human-Robot Collaboration: A Literature Review and Augmented Reality Approach in Design* International Journal of Advanced Robotic Systems, Vol. 5, No. 1 200

Towards Measuring the Quality of Interaction: Communication through Telepresence Robots

Katherine M. Tsui
Dept. of Computer Science
University of Massachusetts Lowell
1 University Ave., Lowell MA
+1 978 934 3385
ktsui@cs.uml.edu

Munjal Desai
Dept. of Computer Science
University of Massachusetts Lowell
1 University Ave., Lowell MA
+1 978 934 3385
mdesai@cs.uml.edu

Holly A. Yanco
Dept. of Computer Science
University of Massachusetts Lowell
1 University Ave., Lowell MA
+1 978 934 3642
holly@cs.uml.edu

ABSTRACT

Personal video conferencing is now a common occurrence in long distance interpersonal relationships. Telepresence robots additionally provide mobility to video conferencing, and people can converse without being restricted to a single vantage point. The metrics to explicitly quantify person to person interaction through a telepresence robot do not yet exist. In this paper, we discuss technical requirements needed to support such a communication. We also look at the fields of human-computer interaction (HCI), computer supported cooperative work (CSCW), communications, and psychology for quantitative and qualitative performance measures which are independent of interpersonal relationships and communication task.

Categories and Subject Descriptors

I.2 [**Robotics**]; D.2.8 [**Software Engineering**]: Metrics—*complexity measures, performance measures*

General Terms

Measurement

Keywords

Human-robot interaction, human-computer interaction, embodied video-mediated communication

1. INTRODUCTION

Both video conferencing and telepresence robots are recent technologies. Friends and family who are located across continents keep in touch with each other through their web cameras and streaming video chat applications such as iChat and Skype launched in 2003 and 2006 respectively [2,55]. As of December 2010, there were 145 million connected Skype users, and in the fourth quarter of 2010, video calls were 42% of the Skype-to-Skype minutes [56]. A number of telepresence robot platforms have emerged in the last decade: InTouch Health's RP-7 in 2003, RoboDynamics' TiLR in 2005, HeadThere's Giraffe (now Giraff Technologies AB) in 2006, Willow Garage's Texai (now Suitable Technologies) in 2009, Anybots' QB and VGo Communications' VGo in 2010, and Gostai's Jazz and 9th Sense's TELO in 2011. This mobile video conferencing technology is currently out of the price range for many personal consumers as the platforms range from $6,000 USD for a VGo robot [69] to $5,000 monthly rental fees for an RP-7 [31]. However, we anticipate that in the near future the telepresence robot will become a common household electronic device, like the personal computer [65].

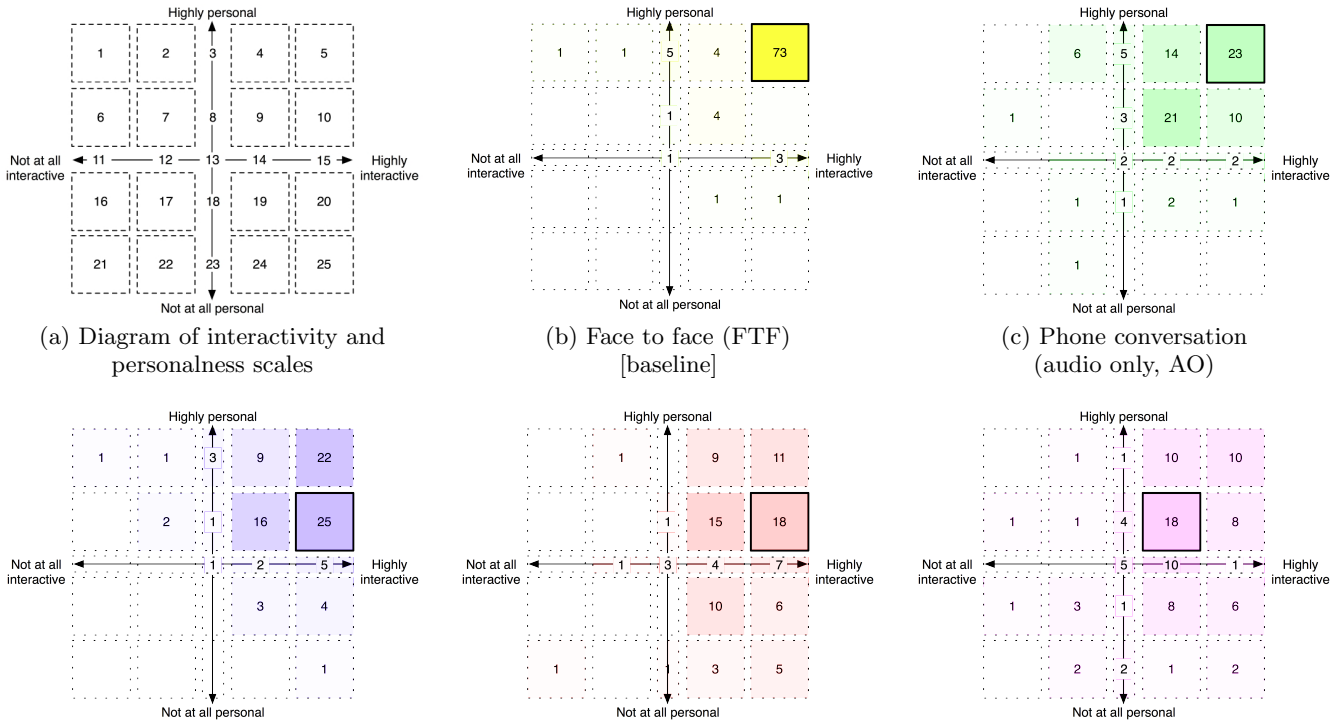
Figure 1: Hugo (an augmented VGo Communication's VGo telepresence robot) is being driven remotely and being used to walk alongside a colleague, actively participating in a mobile conversation. The driver can be seen on Hugo's screen.

We believe that telepresence robots can be used to recreate the closeness a remote person would have if he or she were physically present with his or her family and friends better than a telephone or video chat conversation. Hassenzahl provides insight as to why:

> We have all experienced the awkward silence when we have run out of stories to tell while not wanting to hang up on our loved one. This is the result of a misfit between the conversational model embodied by a telephone and the psychological requirements of a relatedness experience. [21]

Telepresence robots provide a remote person with a physical

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
PerMIS'12 March 20-22, 2012, College Park, MD, USA
Copyright 2012 ACM 978-1-4503-1126-7-3/22/12 ...$10.00.



(a) Diagram of interactivity and personalness scales

(b) Face to face (FTF) [baseline]

(c) Phone conversation (audio only, AO)

(d) Video conferencing (video mediated communication, VMC)

(e) Telepresence robot (TPR)

(f) Instant messaging/chat (IM)

Figure 2: (a) Participants were asked to categorize communication technologies. Original diagram by Jake Knapp of Google; modified to include region enumeration. (b-f) Frequency counts are shown inside each category and the mode is marked by a solid black outline (n=96).

avatar in addition to two-way video and audio (Figure 1). For some people, the robot may still be used exclusively as a conversation tool. Other people may want to use telepresence robots to check on their family, while still others may simply want to be present in a space to feel more included in an activity.

Researchers have investigated the efficacy in which people can use telepresence robots to navigate in remote locations (e.g., [40,59,60,62]), the interfaces to do so (e.g., [40,58,61]), and how the robots should be designed (e.g., [8, 11, 12]). Telepresence robots have great potential to provide utility in workplaces (e.g., [37,62]), in schools (e.g., [53]), in homes (e.g., [9]), and for excursions to museums, sporting events, and the theater (e.g., [5]), for example. However, the quality of a person to person interaction through a telepresence robot has not yet been explicitly quantified. In this paper, we discuss the performance measures needed to assess a communication by leveraging work from the fields of human-computer interaction (HCI), computer supported cooperative work (CSCW), communications, and psychology.

1.1 Comparison of Interaction Mediums

We conducted a survey to investigate how people would categorize several communication technologies with respect to interactivity and personalness. The baseline was "face to face" (FTF) interaction; the technologies included video conferencing, telephone call, telepresence robot, and instant messaging/chat. Each technology has at least one layer of indirection. For example, a phone conversation can be misinterpreted given the lack of facial expression. Text-based instant messaging additionally lacks vocal intonation but includes some level of emotion through emoticons and meta-actions (e.g., smiley face :), *hug*). Video conferencing has audio and facial expressions and gestures seen through a webcam; however, the webcam provides a single vantage point and is subject to adjustment (or lack thereof) by the video conferencing recipient. Telepresence robots also have two-way audio and video, and additionally provide a mobile embodiment to the remote party which allows for independent movement.

The survey was conducted using Amazon's Mechanical Turk (MTurk). For each means of communication, MTurk Workers were asked where they would place it in Figure 2a with respect to the communication's personalness and interactivity. That is for example, a highly personal and highly interactive communication method would be placed in the top-right quadrant in category 5. Because telepresence robots are an emerging commercial technology, we showed MTurk Workers photos of five examples: VGo, RP-7, QB, Texai, and TiLR. We also provided the following definition: "A telepresence robot can be thought of embodied as video conferencing on wheels: the robot is a representation of you. You can see what is around the robot through its camera and hear through its microphones. People with the robot can hear and see you too." Ninety-six people participated in the survey and were each paid $1.00.

Figures 2b-f show the category frequency for each communication method. Face to face interaction was chosen en masse as both highly personal and highly interactive; 76% of the participants (73 of 96) selected category 5 in Figure

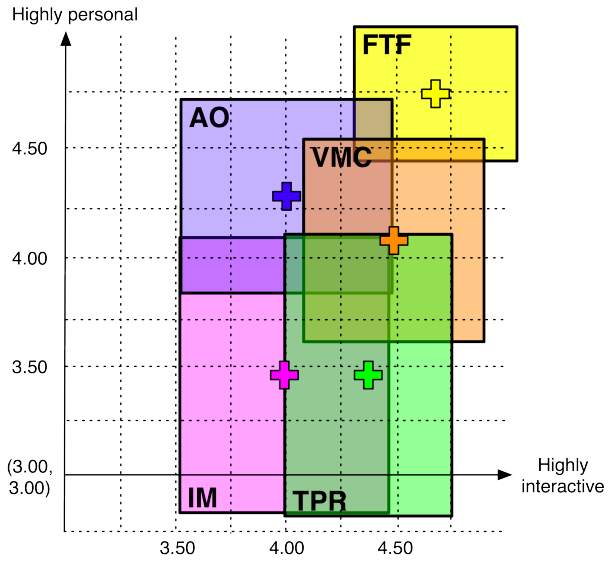


Figure 3: Averages and standard deviations for face to face (FTF), phone call (audio only, AO), video conferencing (video mediated communication, VMC), telepresence robot (TPR), and instant messaging/chat (IM). Plus signs denote averages in the form (interactivity $\bar{I}$, personalness $\bar{P}$), and rectangles denote $\pm$ 1 SD.

2a. The communication technologies however had less of a consensus. Participants selected categories in the top right quadrant (categories 4, 5, 9, and 10 in Figure 2a) for phone conversations (71%), video conferencing (75%), telepresence robot (55%), and instant messaging (48%). The communication technologies were rated all as personal and interactive but to varying degrees given that 25 or fewer participants' votes comprised the modes.

We then transformed each communication method's categorical data into continuous data by separating each axis and assigning values. For the interactivity axis, a value of one was assigned to the left-most category (not at all interactive) and five to the right-most (highly interactive). The frequency count for each column was summed and divided by the number of participants (n=96), thus yielding the weight of the value. We multiplied each category value by its calculated weight. Summing these results provided the average value in rational form, which provided insight if a communication method split two categories on a single axis. We similarly calculated the average value along the personalness axis where a value of one was assigned to bottom-most category (not at all personal) and five to the top-most (highly personal).

Figure 3 shows the averages and standard deviations for the communication methods. We conducted unpaired t-tests for all of the communication method permutations with respect to personalness and also with interactivity. The significance value is α=0.005 as we divided the goal 95% confidence value by the ten test permutations. Face to face interaction rated as the most personal and the most interactive form of communication ($\bar{P}_{FTF}$=4.75 (0.61), $\bar{I}_{FTF}$=4.64 (0.74)) We found that the face to face interaction was significantly more personal than all of the communication technologies ($p_{personal}$<0.002). It was significantly more interactive compared to a phone call and instant messaging ($p_{interactive}$<0.001), but not so when compared to video conferencing (p<0.158, t(190)=1.419) or telepresence robots (p<0.010, t(190)=2.586).

Phone calls were also highly personal but less interactive than face to face interactions ($\bar{P}_{AO}$=4.34 (0.88), $\bar{I}_{AO}$=3.94 (0.96)). We found that phone calls were significantly more personal than instant messaging and telepresence robots ($p_{personal}$<0.001), but significantly less interactive than video conferencing (p<0.001, t(190)=3.570) and also telepresence robots (p<0.007, t(190)=2.720) though not significantly. On the other hand, video conferencing was highly interactive but less personal than face to face interactions ($\bar{P}_{VMC}$=4.11 (0.92), $\bar{I}_{VMC}$=4.48 (0.82)). We found that video conferencing was both significantly more personal and more than interactive instant messaging (p<0.001). When compared to telepresence robots, video conferencing was significantly more personal ($p_{personal}$<0.001) but was not significantly different with respect to interactivity (p<0.295, t(190)=1.052).

As shown in Figure 2e, 92% of the participants rated telepresence robots as interactive despite being given only pictures of telepresence robots and a brief description as to their capabilities. However, there was a lack of consensus as to how personal an interaction using a telepresence robot could be. We hypothesize that this result is because telepresence robots are a new commercial product and while people may know of their existence, they are not yet familiar with them. Therefore, we must look at performance measures that assess the quality of interaction through telepresence robots in pieces: the quality of a communication from a technical standpoint (audio and video), and the quality of a human-human communication through a telepresence robot.

2. AUDIO SIGNAL MEASURES

The most important component of communicating through a telepresence robot is the conversation itself. Rosenberg notes that audio quality can be measured in terms of being able to understand speech and the fidelity of the speech itself [50]. In terms of the speech fidelity, the audio quality must be comparable at least to that of a landline phone [12]. The ITU-T G.711 Recommendation was initially designed for the Public Switched Telephone Network with 64kbps bandwidth in 1972 [30]. G.711's digital counterpart, the ITU-T G.729 Recommendation, was established in 1996 and is popular for voice-over-IP telecommunication given its low bandwidth requirements (8kbps), although at the cost of high compression [29]. Rosenberg notes that as the audio fidelity increases, the length of a conversation also increases [50]. In a study of Skype's SILK codec versus G.729, he reports that users spent 40% longer in calls with the SILK super-wide bandwidth (24kHz) codec.

A codec's speech fidelity is measured by its Mean Opinion Score (MOS), which is one item of a series of subjective rating questions measuring the quality of speech listed in ITU-T Recommendation P.805 (see Table 1). Telecommunication users may be explicitly asked to rate the quality of their connection on a 5-point semantic differential scale where 1=bad and 5=excellent. MOS can be determined using controlled user studies in which the sound origin, sound destination, and background noise are manipulated [27]. MOS can also be derived from simulation tests such as the Perceptual Evaluation of Speech Quality (PESQ) [25].

Speech intelligibility is measured on a 5-point scale the like

Table 1: Subjective evaluation of conversational quality from ITU-T Recommendation P.805 [27]

Question	Scale
What is your opinion of the connection you have just been using? [Mean Opinion Score (MOS)]	1=bad quality; 5=excellent quality
How would you assess the sound quality of the other person's voice?	1=severe distortion; 5=no distortion at all, natural
How well did you understand what the other person was telling you?	1=severe loss of understanding; 5=no loss of understanding
What level of effort did you need to understand what the other person was telling you?	1=severe effort required; 5=no special effort required
How would you assess your level of effort to converse back and forth during the conversation?	1=severe effort required; 5=no special effort required
Did you detect (insert distortion of interest here)? If yes, how annoying was it?	yes/no 1=severe annoyance; 5=no annoyance

MOS scale [57]. Steeneken notes that speech intelligibility can be predicted using several methods. The Speech Interference Level (SIL) subtracts the average noise level within the 500-4000Hz range from the estimated speech level [7]. The expected SIL result is a decibel level where values less than 3 are bad, between 3 and 10 are poor, between 10 and 15 are fair, between 15 and 21 are good, and above 21 are excellent [57]. The Speech Transmission Index (STI) predicts nonsensical speech accounting for the speech and noise range, bandwidth, and physical characteristics of the environment [23]. The STI value ranges between 0 and 1 where values less than 0.30 are bad, between 0.30 and 0.45 are poor, between 0.45 and 0.60 are fair, between 0.60 and 0.75 are good, and above 0.75 are excellent [57]. Barnett and Knight proposed a common intelligibility scale where CIS = 1 + log(STI) [4]. The Speech Intelligibility Index (SII) is similar to STI and also predicts syllabic phonemes in speech [1]. The SII value also ranges between 0 and 1 where values less than 0.45 are poor and above 0.75 are good [57].

Speech intelligibility can also be quantified in terms of the number of echoes, feedback occurrences, and cutouts (e.g., [20, 41]). We designed a study, detailed in [12], to investigate the use of telepresence robots in ad-hoc scenarios, specifically moving down a hallway while simultaneously having a conversation. We noted each run in which echo, feedback, and cutout occurred through analysis of the robot driver's screen captured video which included audio. It is also possible to obtain a speech intelligibility measure qualitatively as telecommunications users may explicitly be asked in post-experience surveys; ITU-T Recommendation P.805 contains four questions relating to intelligibility (Table 1).

3. VIDEO SIGNAL MEASURES

Audio is critical for carrying the content of a communication between two parties. Video can communicate emotion through facial expression and gestures, mutual gaze, and conversational attention [67]. Video information is also critical for telepresence robots in navigating a remote location. Due to the mobility afforded by these robots, the information must be transferred wirelessly. Video streams constitute a significant portion of the data transferred and can be adversely affected by the network connection. The quality of a wireless connection is influenced by several factors including bandwidth, latency, and packet loss.

We designed one study, detailed in [12], to compare the video streams from the QB and VGo telepresence robots against a Sprint EVO Android phone. We placed an eye chart four feet in front of the robot and asked the participants to read the letters from both the phone and the robot's video display. We asked the participants to follow a person (an experimenter) through an area with a hallway, cubicles, and a cafeteria. Following each run, the participants rated the video from the robot and EVO phone with respect to field of view, ability to perceive scale, pauses in video, latency, contrast, resolution, color depth, and quality of degradation on a 7-point semantic differential scale (see Table 2).

Table 2: Video characteristics rating questions for comparing QB and VGo telepresence robots and EVO phone used in Desai et al. [12].

Item	Scale
Overall quality	1=poor, 7=good
Field of view	1=too narrow, 7=too wide
Scale perception	1=could not gauge scale, 7=could gauge scale
Contrast/white balance	1=poor, 7=high
Resolution	1=too low, 7=too high
Color depth	1=low/grayscale, 7=high/true color
Degradation in quality	1=very noticeable, 7=not at all noticeable
Pauses in video	1=few, 7=many
Latency	1=low, 7=high

Based on the results and our observations, the guiding principle for video streams for telepresence robots is to have two video profiles: one while the robot is mobile (dynamic video profile), and another profile for when the robot is not moving (stationary video profile) [12]. Two profiles are needed because the required video characteristics are mutually exclusive at times. Video is the most important sensor information while controlling a telepresence robot. A dynamic video profile should contain characteristics including low latency, few pauses, graceful video degradation, and scale perception. While the robot is stationary, the video profile should contain characteristics including sharp contrast/white balance, increased resolution, and 8-bit color depth or higher.

ITU-T Recommendation P.910 provides a protocol by which multimedia content can be subjectively tested, including sample questions regarding an image's color, contrast, bor-

Table 3: Quantitative communication performance measures surveyed from HCI, CSCW, communications, and psychology. Communication modes included face to face (FTF), audio only (AO), video-mediated communication (VMC), and embodied VMC (eVMC) including telepresence robots.

Measurement	Study Examples			
	FTF	AO	VMC	eVMC
Frequency of communication over time	[16]		[16]	
Number of words				
• in total	[45]		[19, 45]	
• per participant	[45]	[42]	[42, 45]	
Rate of words over time / percentage dialogue			[19]	[54]
Duration of conversation	[16]		[16, 19]	[54, 66]
Number and/or duration of silences	[32, 52, 64]	[32, 42, 52]	[17, 32, 42, 52, 64]	
Number of overlaps		[42]	[42]	
• simultaneous starts	[45, 52, 64]	[52]	[17, 45, 52, 64]	
• floor holding/disfluencies (e.g., "um," "er")	[45, 52]	[52]	[45, 52]	
• sentence completion	[45, 52]	[52]	[45, 52]	
• interruptions	[45, 52]	[52, 64]	[17, 19, 45, 52, 64]	
Number of explicit handovers (e.g., question, name of next speaker)	[32, 45, 52]	[32, 52]	[32, 45, 52]	
Number of turns (attempts to gain the floor to speak)	[32, 33, 45, 52]	[32, 42, 52, 64]	[19, 32, 42, 45, 52, 64, 68]	[54]
Duration of turn / words per turn	[32, 45, 52, 64]	[32, 52]	[19, 32, 45, 52, 64]	
Distribution of turns	[45, 52]	[52]	[45, 52]	
Number of backchannels				
• verbal (e.g., "mm," "uh huh," "okay")	[32, 45]	[32]	[19, 32, 45]	
• head nod	[32]	[32]	[32]	
• gaze	[33, 48]		[68]	[54, 66]
Number of gestures (i.e., kinetic, spatial, point, other)	[6, 32]	[32]	[32]	

ders, movement continuity between frames, flicker, and smearing/blurring [24]. Questions are rated on a modified MOS n-point scale where 1=bad and n=excellent. ITU-R Recommendation BT.500 provides a protocol for subjective testing of the quality of television pictures [26]. Questions are rated on either a 5-point MOS scale, a 5-point impairment scale (1=very annoying, 2=annoying, 3=slightly annoying, 4=perceptible but not annoying, and 5=imperceptible), or a 7-point comparison scale (-3=much worse, 0=same, +3=much better). Video signal quality can be measured objectively using simulation tests such as the Perceptual Evaluation of Video Quality (PEVQ) [28].

4. HUMAN-HUMAN COMMUNICATION MEASURES

A high fidelity video and audio channel given sufficient bandwidth provides the foundation for a human-human communication. 1 common evaluation technique used by companies investing in new telecommuting or virtual team collaboration technologies is to ask a group of sample users to solve a task collectively. The outcome is measured based on the quality of the solution and the time it took to converge (e.g., [64]). Another evaluation technique is to insert the new technology into an existing workflow. Organizational behavior is measured prior to and after the intervention. We used this technique in one of our remote worker studies, detailed in [62]. We selected six remote participants who had recurring meetings with teammates in Mountain View, CA; the remote participants, located across the United States and Europe, used either a QB or VGo telepresence robot to attend their meetings in place of their normal video conferencing setup. Our pre- and post-experiment questionnaires included 5-point Likert scale team cohesion statements [39]. These statements, however, would not be appropriate for investigating how telepresence robots affect familial relationships. Our goal is to investigate quantitative and qualitative communication performance measures which are independent of interpersonal relationships and communication task.

Quantitative Measures. Table 3 summarizes quantitative communication performance measures and provides examples of studies utilizing them. These studies have been drawn from HCI, CSCW, communications, and psychology and look at different communication methods (i.e., face to face (FTF), audio only (AO), video mediated communication (VMC), and embodied video mediated communication (eVMC)). The frequency counts (e.g., number of words, silences, overlaps, handovers, turns, backchannels, gestures) and lengths (e.g., duration of conversation, silences, turns) may be calculated from a recording into speech patterns and speaker segmentation post-hoc coding. Researchers are also investigating real time methods of processing audio signals (e.g., [46]). Fels et al. [15] counted the number of successful, partially successful, and failed communications in the PEBBLES (Providing Education By Bringing Learning Environments to Students) telepresence robot project. Kiesler et al. [34] included a count for correctly recalling information facts after interacting with a robot or robot-like agent.

Qualitative Measures. Open and axial coding from grounded theory [18] can be used to enumerate qualitative data such as observer notes (e.g., [66]) and interviews about the participants' experiences (e.g., [13, 35, 37]). Fish et al. [16] looked at the conversational content from face to face and video-mediated interactions. In the PEBBLES project, Fels et al. [14] counted behavioral instances, specifically the

communication interaction, concentration, and initiative of the remote participant.

Self report scales can provide a means to measure subjective qualitative data. A human-human communication without a medium (or face to face, FTF) is difficult to directly measure given the inherent involvement of interpersonal relationships, and there are a number of scales that investigate different types of relationships and situations (see [51] for an overview). Witmer and Singer developed the Presence Questionnaire (PQ) to measure personal and social presence in virtual environments [49, 70]. The PQ items are rated on a 7-point semantic differential scale. Four subscales have been derived using factor analysis: involvement (α=0.89),[1] sensory fidelity (α=0.84), adaptation/immersion (α=0.84), and interface quality (α=0.57). The involvement and sensory fidelity subscales contain seven items relating to auditory and visual communication which can be applied to telepresence robots shown in Table 4.

Yarosh and Markopoulos developed the Affective Benefits and Cost of Communication Technologies (ABCCT) to study communication technologies for personal use [71]. They created a simple language version for native English speakers ages 8-10. The ABCCT-child was derived from interviews of parent-child conversations, discussion with social connectedness experts, and an examination of the adult ABC-Q (Affective Benefits and Costs in Communication Questionnaire [22, 63]). The ABCCT-child investigates the benefits (α=0.88) and costs (α=0.80) of using a communication technology. The questionnaire has 22 items which are rated on a 5-point scale {never, rarely, sometimes, usually, always} [71]. There are four benefits subscales: emotional expressiveness, engagement and playfulness, presence in absence, and opportunity for social support. Three subscales comprise the costs scale: feeling obligated, unmet expectations, and threat to privacy. Unlike the Presence Questionnaire, the ABCCT questionnaire does not explicitly discuss the quality of auditory and visual communication. Instead, it focuses on connectedness between two parties, the engagement and expressiveness supported by a communication technology, and potential unmet expectations relating to the response time and attention levels using a communication technology. The ABCCT-child questionnaire items are fully detailed in Yarosh and Markopoulos 2010 [71].

5. APPLICATION OF COMMUNICATION MEASURES

We will conduct a pilot study (n=3) in which people with special needs will operate an augmented VGo telepresence robot Hugo in their families' homes [61]. These participants are students and clients of the Crotched Mountain Rehabilitation Center (CMRC) community; for clarity, we will refer to them as "the participants at CMRC." Our goal is to establish if our target population finds benefit from socially engaging with their families through the telepresence robot as compared to video conferencing. We anticipate that the initial sessions may be subject to a novelty effect from the technologies; in our previous research, we have observed this novelty effect cease within 15 minutes of using a telepresence robot. The person being visited by the participant at CMRC (herein known as "the remote person") will interact with the telepresence robot for two sessions, and the VGo video conferencing software on a laptop for two sessions.

Neither video nor audio of the communication transmitted or received through our telepresence robot will be recorded during our studies. It is important for our participants to understand that our telepresence robot will not record audio or video and thereby ensuring their privacy. The lack of audio and video recording prevents analysis of many of the quantitative communication measures in Table 3. However, we will note the duration of the conversation and the level of conversational success as in Fels et al. 2001 [14]. We will ask both the participant at CMRC and the remote person to recall topics of conversation immediately following the end of the communication as in Kiesler et al. 2008 [34].

After the second use of each technology, we will administer the quality of speech rating questions listed in ITU-T Recommendation P.805 (Table 1), the Presence Questionnaire [70], and the ABCCT questionnaire [71] both to the participant at CMRC and his/her family. Following the completion of all four sessions (two with the robot and two with the laptop), we will conduct interviews based on the events that occurred during the sessions to gauge if the participant at CMRC and his/her family found the telepresence robot and the video conferencing software to be useful.

We will then conduct a longitudinal follow-on study in which participants at CMRC will be loaned our telepresence robot for up to one month each. They will be able to use the telepresence robot whenever they want. Like the pilot study, no audio or video will be recorded given the nature of this study. We will additionally note the frequency of the telepresence robot's use, the duration of the conversations, and the audio and video statistics of each session.

6. CONCLUSIONS AND FUTURE WORK

We have discussed potential quantitative and qualitative performance measures needed to assess the communication portion of the interaction, which are independent of interpersonal relationships and communication task. Further, we have described how the questions from the ITU-T Recommendation P.805, the Presence Questionnaire, and the ABCCT questionnaire will be use in studying the differences between telepresence robots and video conferencing. Interaction through a telepresence robot also includes the concept of presence inherent to the ability of independently moving about a remote space. Researchers have investigated the Temple Presence Inventory [38] and the 1ness Questionnaire [3] to measure presence achieved through robotic telepresence interactions [36, 43, 47]. We believe that items from these scales and the Presence Questionnaire can be added to explicit communication measurements to provide a means to assess the quality of a person to person interaction through a telepresence robot.

7. ACKNOWLEDGMENTS

This research has been funded in part by NSF (IIS-1111125, IIS-0905228, IIS-0546309). We would like to thank Jake Knapp of Google and Elizabeth Craig of North Carolina State University. We also thank Anybots and VGo Communications for loaning us prototype robots. Figure 1 photo by John Fertitta of UMass Lowell.

8. REFERENCES

[1] Cronbach's alpha measures the internal consistency of related questions and α>0.7 is considered reliable [10, 44].

Table 4: Select items from Witmer and Singer's Presence Questionnaire [70]

Question	Scale
How much did the visual aspects of the environment involve you?	not at all / somewhat / completely
How much did the auditory aspects of the environment involve you?	not at all / somewhat / completely
How completely were you able to actively survey or search the environment using vision?	not at all / somewhat / completely
How well could you identify sounds?	not at all / somewhat / completely
How well could you localize sounds?	not at all / somewhat / completely
How closely were you able to examine objects?	not at all / pretty closely / very closely
How well could you examine objects from multiple viewpoints?	not at all / somewhat / extensively

[1] Amer. Natl. Standards Institute. S3. 5-1997, Methods for the Calculation of the Speech Intelligibility Index, 1997.

[2] Apple. Apple Introduces iChat AV and iSight. Press release, June 2003. `http://www.apple.com/pr/library/2003/jun/23ichat.html`.

[3] J. Bailenson and N. Yee. A Longitudinal Study of Task Performance, Head Movements, Subjective Report, Simulator Sickness, and Transformed Social Interaction in Collaborative Virtual Environments. *Presence: Teleoperators and Virtual Environments*, 15(6):699–716, 2006.

[4] P. Barnett and R. Knight. The Common Intelligibility Scale. *Inst. of Acoustics*, 17(7):201–206, 1996.

[5] J. Beer and L. Takayama. Mobile Remote Presence Systems for Older Adults: Acceptance, Benefits, and Concerns. In *Proc. of Intl. Conf. on Human-Robot Interaction*, pp. 19–26. ACM, 2011.

[6] M. Bekker, J. Olson, and G. Olson. Analysis of Gestures in Face-to-Face Design Teams Provides Guidance for How to Use Groupware in Design. In *Proc. of 1st Conf. on Designing Interactive Systems: Processes, Practices, Methods, & Techniques*, pp. 157–166. ACM, 1995.

[7] L. Beranek. Airplane Quieting II: Specification of Acceptable Noise Levels. *Trans. Amer. Soc. Mech. Engrs*, 69:97–100, 1947.

[8] B. Cohen, J. Lanir, R. Stone, and P. Gurevich. Requirements and Design Considerations for a Fully Immersive Robotic Telepresence System. In *Proc. of Human-Robot Interaction Wksp. on Social Robotic Telepresence*, 2011.

[9] S. Coradeschi, A. Loutfi, A. Kristoffersson, S. Von Rump, A. Cesta, and G. Cortellessa. Towards a Methodology for Longitudinal Evaluation of Social Robotic Telepresence for Elderly. In *Proc. of Human-Robot Interaction Wksp. on Social Robotic Telepresence*, 2011.

[10] L. Cronbach. Coefficient Alpha and the Internal Structure of Tests. *Psychometrika*, 16(3):297–334, 1951.

[11] B. Deml. Human Factors Issues on the Design of Telepresence Systems. *Presence: Teleoperators and Virtual Environments*, 16(5):471–487, 2007.

[12] M. Desai, K. M. Tsui, H. A. Yanco, and C. Uhlik. Essential Features of Telepresence Robots. In *Proc. of Intl. Conf. on Technologies for Practical Robot Applications*. IEEE, 2011.

[13] X. Ding, T. Erickson, W. Kellogg, S. Levy, J. Christensen, J. Sussman, T. Wolf, and W. Bennett. An Empirical Study of the Use of Visually Enhanced VoIP Audio Conferencing: The Case of IEAC. In *Proc. of SIGCHI Conf. on Human Factors in Computing Systems*, pp. 1019–1028. ACM, 2007.

[14] D. Fels, J. Waalen, S. Zhai, and P. Weiss. Telepresence Under Exceptional Circumstances: Enriching the Connection to School for Sick Children. *Proc. of IFIP INTERACT01: Human-Computer Interaction*, pp. 617–624, 2001.

[15] D. Fels, L. Williams, G. Smith, J. Treviranus, and R. Eagleson. Developing a Video-mediated Communication System for Hospitalized Children. *Telemedicine J.*, 5(2):193–208, 1999.

[16] R. Fish, R. Kraut, R. Root, and R. Rice. Evaluating Video as a Technology for Informal Communication. In *Proc. of SIGCHI Conf. on Human Factors in Computing Systems*, pp. 37–48. ACM, 1992.

[17] E. Geelhoed, A. Parker, D. Williams, and M. Groen. Effects of Latency on Telepresence. Technical Report HPL-2009-120, Hewlett-Packard Labs, 2009.

[18] B. Glaser and A. Strauss. *The Discovery of Grounded Theory: Strategies for Qualitative Research*. Aldine Publ., 1977.

[19] D. Grayson and L. Coventry. The Effects of Visual Proxemic Information in Video mediated Communication. *ACM SIGCHI Bulletin*, 30(3):30–39, 1998.

[20] H. Haas. The Influence of a Single Echo on the Audibility of Speech. *J. Audio Eng. Soc.*, 20(2):146–159, 1972.

[21] M. Hassenzahl. Encyclopedia Chapter on User Experience and Experience Design. Webpage, 2011. `http://www.interaction-design.org/encyclopedia/user_experience_and_experience_design.html`, accessed April 2011.

[22] W. IJsselsteijn, J. Baren, P. Markopoulos, N. Romero, and B. Ruyter. Measuring Affective Benefits and Costs of Mediated Awareness: Development and Validation of the ABC-Questionnaire. *Awareness Systems*, pp. 473–488, 2009.

[23] Intl. Electrotechnical Commission. 60268-16. Objective Rating of Speech Intelligibility by the Speech Transmission Index, Mar. 1998.

[24] Intl. Telecommunication Union. Rec. P.910, Subjective Video Quality Assessment Methods for Multimedia Applications, Sep. 1999.

[25] Intl. Telecommunication Union. Rec. P.805, Perceptual Evaluation of Speech Quality (PESQ): An Objective Method for End-to-End Speech Quality Assessment of Narrow-Band Telephone Networks and Speech Codecs, Feb. 2001.

[26] Intl. Telecommunication Union. Rec. BT.500-11, Methodology for the Subjective Assessment of the Quality of Television Pictures, 2002.

[27] Intl. Telecommunication Union. Rec. P.805, Subjective Evaluation of Conversational Quality, Apr. 2007.

[28] Intl. Telecommunication Union. Rec. J.247, Objective Perceptual Multimedia Video Quality Measurement in the Presence of a Full Reference, Aug. 2008.

[29] Intl. Telecommunication Union. G.711: Pulse Code Modulation (PCM) of Voice Frequencies, Nov. 2009.

[30] Intl. Telecommunication Union. G.729: Coding of Speech at 8 kbit/s Using Conjugate-Structure Algebraic-Code-Excited Linear Prediction (CS-ACELP), Mar. 2011.

[31] InTouch Health. FAQ, 2011. `http://www.intouchhealth.com/ITHFAQs.pdf`, accessed Nov. 2011.

[32] E. Isaacs and J. Tang. What Video Can and Cannot Do for Collaboration: A Case Study. *Multimedia Systems*, 2(2):63–73, 1994.

[33] K. Jokinen, M. Nishida, and S. Yamamoto. Eye-gaze Experiments for Conversation Monitoring. In *Proc. of 3rd*

Intl. Universal Communication Symp., pp. 303–308. ACM, 2009.

[34] S. Kiesler, A. Powers, S. Fussell, and C. Torrey. Anthropomorphic Interactions with a Robot and Robot-like Agent. *Social Cognition*, 26(2):169–181, 2008.

[35] D. Kirk, A. Sellen, and X. Cao. Home Video Communication: Mediating "Closeness". In *Proc. of Conf. on Computer Supported Cooperative Work*, pp. 135–144. ACM, 2010.

[36] A. Kristoffersson, S. Coradeschi, K. Severinson Eklundh, and A. Loutfi. Sense of Presence in a Robotic Telepresence Domain. *Universal Access in Human-Computer Interaction. Users Diversity*, pp. 479–487, 2011

[37] M. Lee and L. Takayama. "Now, I Have a Body": Uses and Social Norms for Mobile Remote Presence in the Workplace. In *Proc. of SIGCHI Conf. on Human Factors in Computing Systems*, pp. 33–42. ACM, 2011.

[38] M. Lombard and T. Weinstein. Measuring Presence: The Temple Presence Inventory. In *Proc. of Intl. Wksp. on Presence*, 2009.

[39] M. Michalisin, S. Karau, and C. Tangpong. The Effects of Performance and Team Cohesion on Attribution: A Longitudinal Simulation. *J. of Business Research*, 57(10):1108–1115, 2004.

[40] F. Michaud, P. Boissy, D. Labonté, S. Brière, K. Perreault, H. Corriveau, A. Grant, M. Lauria, R. Cloutier, M. Roux, et al. Exploratory Design and Evaluation of a Homecare Teleassistive Mobile Robotic System. *Mechatronics*, 20(7):751–766, 2010.

[41] G. Miller and J. Licklider. The Intelligibility of Interrupted Speech. *J. Acoustical Soc. of Amer.*, 1950.

[42] A. Monk and C. Gale. A Look Is Worth a Thousand Words: Full Gaze Awareness in Video-mediated Conversation. *Discourse Processes*, 33(3):257–278, 2002.

[43] D. Nguyen and J. Canny. More than Face-to-Face: Empathy Effects of Video Framing. In *Proc. of SIGCHI Conf. on Human Factors in Computing Systems*, pp. 423–432. ACM, 2009.

[44] J. Nunnaly. *Psychometric Theory*. McGraw-Hill, New York, 1978.

[45] B. O'Conaill, S. Whittaker, and S. Wilbur. Conversations Over Video Conf.s: An Evaluation of the Spoken Aspects of Video-Mediated Communication. *Human-Computer Interaction*, 8(4):389–428, 1993.

[46] L. O'Gorman. Latency in Speech Feature Analysis for Telepresence Event Coding. In *Intl. Conf. on Pattern Recognition*, pp. 4464–4467. IEEE, 2010.

[47] B. Okdie, R. Guadagno, F. Bernieri, A. Geers, and A. Mclarney-Vesotski. Getting to Know You: Face-to-Face Versus Online Interactions. *Computers in Human Behavior*, 27(1):153–159, 2011.

[48] K. Otsuka, J. Yamato, Y. Takemae, and H. Murase. Quantifying Interpersonal Influence in Face-to-Face Conversations Based on Visual Attention Patterns. In *Proc. of SIGCHI Conf. on Human Factors in Computing Systems*, pp. 1175–1180. ACM, 2006.

[49] E. Perse. *Presence Questionnaire*, pp. 276–283. Routledge, Taylor & Francis, 2009.

[50] J. Rosenberg. Quality Matters, Aug. 2010. `http://www.tmcnet.com/ucmag/columns/articles/99344-quality-matters.htm`.

[51] R. Rubin, A. Rubin, E. Graham, E. Perse, and D. Seibold. *Communication Research Measures II: A Sourcebook*. Routledge, Taylor & Francis, 2009.

[52] A. Sellen. Remote Conversations: The Effects of Mediating Talk with Technology. *Human-Computer Interaction*, 10(4):401–444, 1995.

[53] K. Sheehy and A. Green. Beaming Children Where They Cannot Go: Telepresence Robots and Inclusive Education: An Exploratory Study. *Ubiquitous Learning*, 3(1):135–146, 2011.

[54] D. Sirkin, G. Venolia, J. Tang, G. Robertson, T. Kim, K. Inkpen, M. Sedlins, B. Lee, and M. Sinclair. Motion and Attention in a Kinetic Videoconferencing Proxy. *Human-Computer Interaction*, pp. 162–180, 2011.

[55] Skype. Skype Introduces Video Calling for Macintosh Users. Press release, Sept. 2006. `http://about.skype.com/2006/09/skype_introduces_video_calling.html`.

[56] Skype S.A. Amendment No. 2 to Form S-1 Registration Statement. Prospectus, Mar. 2011. `http://www.sec.gov/Archives/edgar/data/1498209/000119312511056174/ds1a.htm#rom83085_3a`.

[57] H. Steeneken. Standardisation of Performance Criteria and Assessments Methods for Speech Communication. In *European Conf. on Speech Communication and Technology*, vol. 1, pp. 255–258, 2006.

[58] L. Takayama, E. Marder-Eppstein, H. Harris, and J. Beer. Assisted Driving of a Mobile Remote Presence System: System Design and Controlled User Evaluation. In *IEEE Intl. Conf. on Robotics and Automation*, pp. 1883–1889, 2011.

[59] T. Tsai, Y. Hsu, A. Ma, T. King, and C. Wu. Developing a Telepresence Robot for Interpersonal Communication with the Elderly in a Home Environment. *Telemedicine and e-Health*, 13(4):407–424, 2007.

[60] K. Tsui, M. Desai, H. Yanco, and C. Uhlik. Telepresence Robots Roam the Halls of My Office Building. In *Proc. of Human-Robot Interaction Wksp. on Social Robotic Telepresence*, 2011.

[61] K. Tsui, A. Norton, D. Brooks, H. Yanco, and D. Kontak. Designing Telepresence Robot Systems for Use by People with Special Needs. In *Proc. of Intl. Symp. on Quality of Life Technologies 2011: Intelligent Systems for Better Living, held in conjunction with RESNA 2011 as part of FICCDAT*, 2011.

[62] K. M. Tsui, M. Desai, H. A. Yanco, and C. Uhlik. Exploring Use Cases for Telepresence Robots. In *Proc. of Intl. Conf. on Human-Robot Interaction*. ACM, 2011.

[63] J. Van Baren, W. IJsselsteijn, P. Markopoulos, N. Romero, and B. de Ruyter. Measuring Affective Benefits and Costs of Awareness Systems Supporting Intimate Social Networks. In *CTIT Wksp. Proc. Series*, vol. 2, pp. 13–19, 2004.

[64] R. van der Kleij, J. Maarten Schraagen, P. Werkhoven, and C. De Dreu. How Conversations Change over Time in Face-to-Face and Video-mediated Communication. *Small Group Research*, 40(4):355–381, 2009.

[65] V. Venkatesh and S. Brown. A Longitudinal Investigation of Personal Computers in Homes: Adoption Determinants and Emerging Challenges. *MIS Quarterly*, pp. 71–102, 2001.

[66] G. Venolia, J. Tang, R. Cervantes, S. Bly, G. Robertson, B. Lee, and K. Inkpen. Embodied Social Proxy: Mediating Interpersonal Connection in Hub-and-Satellite Teams. In *Proc. of SIGCHI Conf. on Human Factors in Computing Systems*, pp. 1049–1058. ACM, 2010.

[67] R. Vertegaal, R. Slagter, G. van der Veer, and A. Nijholt. Eye Gaze Patterns in Conversations: There Is More to Conversational Agents than Meets the Eyes. In *Proc. of SIGCHI Conf. on Human Factors in Computing Systems*, pp. 301–308. ACM, 2001.

[68] R. Vertegaal, G. van der Veer, and H. Vons. Effects of Gaze on Multiparty Mediated Communication. In *Graphics Interface*, pp. 95–102, 2000.

[69] VGo Communications. VGo Communications. Webpage, 2010. `http://vgocom.com`, accessed Oct. 2010.

[70] B. Witmer, C. Jerome, and M. Singer. The Factor Structure of the Presence Questionnaire. *Presence: Teleoperators & Virtual Environments*, 14(3):298–312, 2005.

[71] S. Yarosh and P. Markopoulos. Design of an Instrument for the Evaluation of Communication Technologies with Children. In *Proc. of Intl. Conf. on Interaction Design and Children*, pp. 266–269. ACM, 2010.

Technology Readiness Levels for Randomized Bin Picking

Jeremy A. Marvel
National Institute of Standards and Technology
100 Bureau Drive, MS 8230
Gaithersburg, MD 20899
+1 (301) 975-4592
jeremy.marvel@nist.gov

Roger Eastman
Loyola University, Maryland
Department of Computer Science
4501 North Charles Street
Baltimore, MD 21210
+1 (410) 617-2281
reastman@loyola.edu

Geraldine Cheok
National Institute of Standards and Technology
100 Bureau Drive, MS 8230
Gaithersburg, MD 20899
+1 (301) 975-6074
cheok@nist.gov

Kamel Saidi
National Institute of Standards and Technology
100 Bureau Drive, MS 8230
Gaithersburg, MD 20899
+1 (301) 975-6069
kamel.saidi@nist.gov

Tsai Hong
National Institute of Standards and Technology
100 Bureau Drive, MS 8230
Gaithersburg, MD 20899
+1 (301) 975-3444
hongt@nist.gov

Elena Messina
National Institute of Standards and Technology
100 Bureau Drive, MS 8230
Gaithersburg, MD 20899
+1 (301) 975-2661
elena.messina@nist.gov

ABSTRACT

A proposal for the utilization of Technology Readiness Levels to the application of unstructured bin picking is discussed. A special session was held during the 2012 Performance Metrics for Intelligent Systems workshop to discuss the challenges and opportunities associated with the bin picking problem, and to identify the potentials for applying an industry-wide standardized assessment and reporting framework such as Technology Readiness Levels to bin picking. Representative experts from government, academia, and industry were assembled to form a special panel to share their insights into the challenge.

Categories and Subject Descriptors

C.4 [**Performance of Systems**]: Performance Attributes; I.5.4 [**Applications**]: Computer Vision

General Terms

Measurement, Documentation, Performance, Experimentation, Verification

Keywords

Bin Picking, Technology Readiness

1. INTRODUCTION

Manufacturing technologies have witnessed a veritable boom in robot integration and improved sensing modalities for safety and task automation. Worldwide manufacturing initiatives stress the integration of robot technologies in modernized manufacturing facilities, and push the boundaries of both productivity and innovation in an ever-increasingly competitive market.

Despite years of considerable progress in 3D pose estimation systems and vision-guided robotics, one of the greatest challenges to manufacturing automation is the task of component acquisition from a randomized bin of parts. A special session was held at the 2012 Performance Metrics for Intelligent Systems workshop that focused on the state of the art and metrics of technology readiness levels (TRLs) for bin picking solutions that are robust against random pose and part variations. We addressed the indicators of maturity of approaches for overcoming shape variation, pose and orientation uncertainty, weak or no distinguishing image features, and limited grasping options. Presenters discussed both the TRL development process and the needs and challenges from the perspectives of both users and vendors regarding bin picking for manufacturing automation.

The principal goal of the special session was to establish a common understanding of how to match the robotic bin picking perception requirements of manufacturers against the current capabilities of vendor systems. Further, we intended to determine the best mechanisms for advancing the capabilities and greater deployment of robotic bin picking. This could be through an advanced perception TRL framework or other common set of metrics and evaluation criteria that can be developed by the user, vendor, research, and government communities through a consensus standardization process.

We discussed the requirements and processes involved with the grading of different levels of bin picking difficulty, and the feasibility of establishing a set of standardized artifacts for bin picking solution validation. Additional topics of discussion included the challenges inhibiting solution integration, and opportunities for advancement in next-generation manufacturing environments.

This report provides an account of the proceedings of the 2012 Performance Metrics for Intelligent Systems (PerMIS) workshop special session, and outlines preliminary action items for the development of a process for evaluating and documenting the maturity of technologies for bin picking. Section 2 presents an overview of the bin picking problem, and discusses the challenges

(c) 2012 Association for Computing Machinery. ACM acknowledges that this contribution was authored or co-authored by a contractor or affiliate of the U.S. Government. As such, the Government retains a nonexclusive, royalty-free right to publish or reproduce this article, or to allow others to do so for Government purposes only. PerMIS'12, March 20-22, 2012, College Park, MD, USA. Copyright © 2012 3ACM 978-1-4503-1126-7-3/22/12…$10.00

Disclaimer: Certain trade names and company products are mentioned in the text or identified in certain illustrations. In no case does such an identification imply recommendation or endorsement by the National Institute of Standards and Technology, nor does it imply that the products are necessarily the best available for the purpose.

and measurable properties of its use. In Section 3, we provide a summary of TRLs and discuss their applicability (and that of other technological maturity assessment scales) to bin picking. And Section 4 outlines the discussion topics from the special session regarding the use of TRLs within the bin picking problem domain.

2. BIN PICKING

The application of bin picking in manufacturing is an interesting problem in that the concepts are fairly ubiquitous and practically everyone understands the underlying concepts of acquiring objects from a bin of parts, but also that there are no consistent definitions of what the process actually entails. A major contributing factor to this ambiguity is the observation that there is no single bin picking application, but rather a spectrum of specific instantiations based on any number of unique constraints that are user-, process-, and product-specific.

In the classic literature, the process of bin picking has been frequently reduced to a three-step process consisting of isolating a specific object from the background, determining the pose of that object, and then creating a path trajectory to move a robot in and grasp the object (e.g., [1-3]). The base definition is necessarily vague given the broad spectrum of bin picking applications, and thus includes parts acquisition processes ranging from picking objects off a conveyor belt (though many would argue that a defining aspect of bin picking is extracting parts within a box or bin) to taking parts out of a randomized bin of multiple part shapes. The acquisition of a single part from a collection of parts is often considered an integral component of manufacturing, and can be considered a superset of many common industrial processes including kitting, palletizing, packaging, and assembly. Successfully integrating bin picking into a product line brings a number of inherent benefits, including higher throughput, reliability, and flexibility, but first the challenges of each particular picking scenario must be overcome.

When attempting to assess how difficult a particular picking problem is, it is helpful to have a common comparative scale. Just as there is no single bin picking problem, however, so, too, is there no single comparative metric for determining the complexity of a given problem. A common—and arguably over-simplistic—method comes from the Electrical Engineering Handbook, and utilizes a relative three-tier difficulty rating that assigns a complexity value based on the controllability of part position and appearance [4]. More recently, however, researchers have begun assessing problem difficulty based on the maturity of the component technologies required to address a particular bin picking application [5].

There are a number of categorical parameter spaces by which the difficulty of a bin picking application can be scaled. Commonly this spectrum is scaled according to the degrees of freedom of the parts to be acquired, i.e., ranging from X and Y axes variation ("2D"), to X and Y axes variation plus Z axis rotation ("2.5D"), to full X, Y, and Z position and rotation variation ("6D"). However, more recently, trends in describing the bin picking application have separated solutions for the problem domain according to image segmentation difficulty properties such as image feature strength [6]. There are a number of categorical parameter spaces by which the difficulty of a bin picking application can be scaled. These spaces include scenario complexity, part location or orientation, part or shape variation, image feature strength, part rigidity, and part overlap and interlock (i.e., when two or more parts become connected and require separation before they can be used).

Once a solution to a bin picking problem has been developed, there are three principal performance metrics by which that solution can be evaluated: speed, efficiency, and accuracy. Speed refers both to the time required to acquire an individual part from a bin (*picking time*), and the number of picks per given period of time (*bandwidth*). Efficiency is measured in terms of time utility (e.g., the time spent searching for parts to acquire versus the time actually spent picking them up), grasping quality and acquisition success, and robot trajectory optimization (i.e., how efficiently does the manipulator move into the bin and avoid collisions with parts or the bin?). Accuracy is the measurement error in object recognition and part pose estimation.

There are three primary challenge domains that may complicate the integration of a bin picking solution into the manufacturing process. The first domain, sensing, includes the inherent difficulties in sensor and algorithm development, but also includes components of process and workcell optimization. The types of challenges that an integrator must overcome include object identification issues due to lighting variations caused by surface reflectivity, shadows, and material transparency. Pose estimation algorithms may be further misled by shape and surface variations incurred during the manufacturing process, or by weak, inconsistent, or non-existent image features. Each effectively prevents an adequate fit of the detected part to a known model. Moreover, variations in the bin itself—such as position uncertainty and bin damage—may present additional challenges if the system does not know exactly where it should be searching for parts.

The second challenge domain reflects issues with the hardware involved, including the robot, the gripper, and the parts being acquired. Specifically, the robot's dexterity and reach may limit the number of parts that can actually be acquired. Challenges with the gripper's dexterity and design may restrict the number of possible grasp points as well as limit the grasp efficiency and quality. Similarly, the weight, durability, and separation of the parts may further restrict how they can be handled.

The third (and arguably most difficult) challenge domain to overcome includes the pragmatic issues of bin picking solution integration. This includes considerations such as cost, which is defined in terms of both financial burden and the times required to bring a system online, to train and tune the system for new parts, and to support the repurposing of an existing system for a new process. Further, issues concerning the bin picking problem application's uniqueness are often considerable. For example, when introducing a new part to the production line, what solution components can be recycled, and how well does the new solution actually fit the specific bin picking need? Conversely, can the new process be changed to be more congruent with the old bin picking system? Many times the old system must be shelved, and a new system built up from initial concepts. These considerations ultimately tie in to the understanding of the bin picking problem, itself: does the integrator know and understand the process well enough to be able to identify reusable components? Moreover, what level of understanding and awareness does the user have about bin picking in general? This final element is frequently characterized by users either not knowing what solutions are available, or having unrealistic expectations of the capabilities of robotic bin picking systems.

Table 1. Example TRL Description Summaries Based on the NASA [8], DOD [9], and DOE [10] Guidelines

TRL	Summary and Description
1	Basic principles observed and reported. Research begins to be translated into applied research and development (R&D)
2	Technology concept or application formulated. Practical (albeit speculative) applications can be invented after basic principles are observed.
3	Characteristic proof of concept. Active R&D is initiated, and includes analytical and lab studies for physical validation of analytical predictions of individual elements of the technology.
4	Laboratory validation of components. Basic technological components are integrated to verify they work together.
5	Target environment validation of components. Higher fidelity of component integration testing in a reasonably supporting environment to allow for simulated environment testing.
6	System/subsystem model in target environment. Models and prototypes demonstrating a significant technological readiness improvement are tested in a relevant environment.
7	System prototype in operational environment. Functional prototypes demonstrating the completed system in its approximate expected configuration are evaluated in an operational environment.
8	Final system qualified through demonstration. Technologies are proven to work in their final form and under expected conditions through test and demonstration.
9	Final system proven through vetting. Applications of technologies in their final form are proven through successful operations under mission conditions.

3. TECHNOLOGY READINESS LEVELS

Originally proposed by the National Aeronautics and Space Administration (NASA) [7, 8], the TRL structure describes a process for evaluating the maturity of technologies prior to their incorporation into deployable systems. The primary users of TRL scales are agencies and organizations, both domestic and international, with aeronautical and aerospace interests, but many users modify the language of NASA's TRL model to better suit differences in the user's production patterns, technologies, or management structures (e.g., the U.S. Department of Defense (DOD, [9]), and U.S. Department of Energy (DOE, [10]).

TRLs are used to measure maturity of technologies when determining the risks associated with inserting them into a mission (or mission component), and are critical to communication with partners, suppliers, and customers. The TRL structure is frequently implemented as a nine-stage hierarchy, as illustrated in Table 1. Generally speaking, TRL-6 is a desirable stage prior to any technology being integrated into a mission, and is considered the "go/no go" point. TRLs, however, are only one of several tools for the decision process. Key Decision Points (KDPs), for example, determine the readiness of a program/project to advance to the next phase, and are outlined in NASA's Procedural Requirements [11].

Despite its wide utilization in aerospace and aeronautics both within the U.S. and internationally, there is no standardized TRL structure or implementation. As a result, the TRL for a specified technology may not be identical for all missions or applications. Specifically, the readiness level for a given technology may be different depending on the considering agency, environmental factors, intended use, or even who within a given agency is assessing the technology. Similarly, there is a significant lack of clear exit criteria (i.e., conditions for moving from one level to another) for higher TRLs, and the guidelines for assessing TRLs are frequently vague or even conflicting.

Applying TRLs to new problem domains such as manufacturing is complicated by the TRL structure's inability to handle certain factors that are important to these domains. For instance, though the TRL structure can readily be applied to the manufacturing domain, it does not address the requisite factors of throughput, profit, market needs, or the ease of labor and implementation issues. Because TRLs are typically applied to one-off or otherwise relatively small-scale production, applications requiring large-scale production or distribution are often incompatible. Instead, focus is placed on technological maturity, and consideration of factors such as the capabilities of processes or technologies would not be addressed using the current TRL structure.

As an alternative, the U.S. Department of Defense (DOD) introduced Manufacturing Readiness Levels (MRLs) [12], a 10-level administrative process focused on the actual production process. MRLs are used to quantitatively assess the maturity of technology components from a manufacturing perspective, and are used to determine the risks involved with bringing products to the production phase. This process involves an initial assessment of the basic needs for manufacturing products, and is used to document and demonstrate that given technologies are ready for wide scale manufacturing.

These deficiencies have thus prompted efforts to reassess the TRL structure. NASA, for instance, is reevaluating its TRL definitions and exit criteria, and efforts are being considered to create standards for assessing and reporting TRLs. Beyond NASA, the International Organization for Standardization (ISO) is coordinating space agencies and other stakeholders to develop an international TRL standard (ISO TRL work group, 14N665, *Definition of Technology Readiness Levels and their criteria of assessment*). Through this effort, ISO is also discussing the necessary steps to broaden the scope of the standard beyond aerospace, eventually encompassing other topics such as manufacturing.

4. PANEL DISCUSSION

Following presentations on TRLs and opportunities in bin picking by Karen McNamara from NASA, and Jeremy Marvel from the National Institute of Standards and Technology (NIST), respectively, a special panel of experts from government, industry, and academia was assembled to address the challenge of assessing and reporting technologies for addressing the bin picking problem domain. Alphabetically, these panel members were:

- Bob Bollinger, Procter & Gamble (P&G)

- Paul Evans, Southwest Research Institute (SWRI)
- Joyce Guthrie, United States Postal Service (USPS)
- Eric Hershberger, Cognex
- Carlos Martinez, ASEA Brown Boveri (ABB)
- Karen McNamara, NASA
- James Wells, General Motors (GM)

Roger Eastman from Loyola University, Maryland, moderated the discussion, and prompted dialogs based on topics relating to the development, utilization, and assessment of bin picking solutions.

The discussion began with an effort to expand the categorical classification of the user's perspective of bin picking. From a manufacturing perspective, there are three distinct and readily identifiable phases for which bin picking will be employed based on the stage of production in which the objects are being picked. As the manufacturing process nears a finished product, the level of care required to prevent damage increases. Early stages, for example, typically require the acquisition of raw (unfinished) materials frequently presented in randomized bins. In contrast, in-process and finished components require increasing levels of fixturing to prevent damage that would affect the functionality or aesthetics of the parts. The bin picking process varies accordingly based on the shipping or presentation method.

Improved inter-process component transfer is an impetus for production optimization, and the ability to handle material in a lean fashion is what is driving bin picking. One of the panelists described the production process as a series of transformations in which the components are transferred between robots, hoppers, bins, conveyor belts, dunnage, and so on. Intermediate transformation steps, e.g., moving parts from a hopper to a conveyor belt to be acquired by delta robots, add cost and complexity to the manufacturing process. The capacity for handling parts as they would naturally be presented in an unstructured form—particularly if the gripper does not have to be changed or the robot reprogrammed to handle the part changes—would thus improve process efficiency.

The distinction between structured and unstructured (i.e., random) presentation of parts within a bin plays a vital role in determining the complexity of the problem. As the strictness of fixturing decreases, the difficulty inherent in developing a bin picking solution increases. Structured bin picking (i.e., parts presented in known, repeatable positions and orientations) is largely considered to be a solved problem, and is addressed by simple matrix handling. In contrast, no general approach for addressing unstructured bin picking (where the locations, shapes, and identities of parts may not be known *a priori*) has been produced.

The degree of randomization of the parts within the bin thus contributes to complexity. For example, a bin full of cast parts is considered to be an easier problem than a bin of irregularly shaped mail. Solutions to such problems have not been forthcoming, and some solution providers have enacted policies to decline requests for unstructured bin picking. Despite years of research in algorithms, robotics, and sensor systems, no unstructured bin picking solution has been developed that is reliable, small, cost effective, or widely applicable. Even within classes of parts (e.g., plastic container caps), the required flexibility of bin picking solutions has not materialized, and the capacity to compensate for product line changes requires hard automation (i.e., large, highly-fixtured, part-specific feeder and handler systems). The issue is further complicated by cases where such hard automation is impossible due to large variances in part shape and size.

In contrast with the hard automation solutions, the cost for robot bin picking solutions is not driven by the cost of the robot. Rather, it is the cost of integrating the robot into the manufacturing process that presents the largest hurdle. Specifically, handling safety and process-specific ancillary assembly line system requirements contribute the most to the price of the system, and thus hinder cost efficiency and flexibility. Specialized fixturing and dunnage to ease the burden on perception add additional cost to the system, and must be redesigned or repurposed as the products and processes change. The actual cost of the robot is comparatively small, as is the impact of the robot on the complexity of the bin picking solution. Though different bin picking classifications may require different robots, the control, repeatability, and reliability of robots in general are considered largely solved. Similarly, the gripping of the objects for process utilization, though considered a specialized component given the parts being acquired and subsequent utilization, is also considered solved.

If the physical aspects of the bin picking problem are considered solved, then what is the greatest hindrance? The panel agreed that perception (and associated sensing technologies) of the various components in the manufacturing setup is the limiting factor in the improvement of bin picking. For example, the USPS already has the technology to handle packages once they have been acquired, but reiterated that perceiving the locality of the materials as they come in presents the greatest barrier to full automation.

Similarly, the bin itself provides a challenge in a number of ways. Identifying variations of the bin in terms of placement, shape, and condition (e.g., due to incurred damage to the bin) add complexity to both the part location process and to collision-free trajectory generation. Recognizing when the bin is empty is a common challenge, as missed parts at the bottom of the bin lead to waste, and, in terms of mail delivery, loss of business functionality and reliability. Once a part has been acquired, if the robot needs to control or attach the part to a fixture or another part, the system will need to know exactly how the part is being held, which requires additional perception capabilities for process validation.

Another common theme expressed by the panel members was the desire to have robots and humans working collaboratively on the production line. This functionality requires an extension of the perception capabilities of the workcell to include robot safety, for which the panel discussed improved situational awareness of the workcell integrating multiple sensors and algorithms from multiple vendors. An additional consideration included a fundamental reconsideration of the requirements of the workcell, and a redesign of process components (e.g., the bins containing the parts) such that they are robot friendly rather than requiring robots to work within the confines of human accessibility.

The second part of the panel discussion focused on whether the development of an evaluative maturity measurement process like TRLs would aid in the advancement of bin picking technologies. Most of the larger manufacturers have internal processes similar to TRLs that they use to measure the maturity of technologies prior to integration. One company, for instance, has a management-integrated process for evaluating required technologies (e.g., technologies necessary for the design of a new car) and technologies that improve existing processes. Ultimately, the

technology evaluation process is merely an input into the decision process and is not a goal in and of itself.

These internal processes, however, are typically proprietary, or otherwise unavailable for other users to utilize as either an example or as a means of benefiting from the larger manufacturers' experiences. The question was thus raised of the panel: how can small companies learn from the experiences of larger companies; what reporting processes other than TRLs are available? As an alternative, it was suggested that a new standardized test method or generalized competition format could be used in lieu of the application- and user-specific technology maturity scales. It was further suggested that trade organizations such as the Robotic Industries Association may be able to provide aggregated abstractions of the technological knowledge for dissemination.

When discussing the metrics by which different technologies could be evaluated, a number of metrics were suggested as being common to users of bin picking solutions. Beyond the expected metrics such as picking speed and throughput discussed in Section 2, the panel also recommended measurement concepts such as agility and repurposing. Agility is the capacity of a robot working with product A to quickly re-task to begin working with product B, and repurposing refers to the amount of time, effort, and skill required to have a robot perform a different task.

5. CONCLUSIONS AND ACTION ITEMS

In the TRL for Randomized Bin Picking special session of the 2012 PerMIS workshop, a panel of experts was organized to discuss the needs and challenges of unstructured bin picking, and to assess whether a TRL structure would help facilitate the documentation and advancement of bin picking technologies. The panel agreed that structured bin picking—situations in which objects are presented in a regular matrix such that parts acquisition requires little to no actual perception to locate a particular object—has been largely solved with a comparatively high level of maturity. In contrast, unstructured bin picking—situations in which presented objects have inconsistent or unknown pose or shape—is considered an immature technology, and that some form of communication structure is needed to help unite the research community in order to fully address the problem

It was also agreed that the creation of some form of taxonomy for assessing and documenting the technological readiness of core processes and technologies would greatly benefit their integration and application in manufacturing practices. However, it was not certain that the TRL structure is necessarily the best approach for describing maturities of application-targeted manufacturing technologies. It was recommended that future efforts attempt to identify the full spectrum of alternatives in order to discover the one that is best for capturing the problem domain.

Particular to the domain of manufacturing, the panel decided the logical next step in addressing the challenges of unstructured bin picking was to first assess the current state of the art in picking technologies. Two action items were thus discussed. The first was to form a task group to identify, create, and document metrics and test methods for evaluating bin picking solutions. This process would include, but is not limited to, the development of standardized artifacts and data sets, performance evaluation frameworks, and a standardized lexicon of bin picking metrics. The second action item involves the documentation of available technologies (including sensing, perception, trajectory creation, and grasping) and categorically assessing their capabilities as applied to the bin picking problem domain.

REFERENCES

[1] Ikeuchi, K., Horn, B.K.P., Nagata, S., Callahan, T., and Fein, O. Picking Up an Object from a Pile of Objects. In *MIT AI Memos*. 1983. Pp. 139-166.

[2] Fuchs, S., Haddadin, S., Keller, M., Parusel, S., Kolb, A., and Suppa, M. Cooperative Bin-Picking with Time-of-Flight Camera and Impedance Controlled DLR Lightweight Robot III. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*. 2010. Pp. 4862-4867.

[3] Skotheim, Ø., Thielemann, J.T., Berge, A., and Sommerfelt, A. Robust 3D Object Localization and Pose Estimation for Random Bin Picking with the 3DMaMa Algorithm. In *Proceedings of the SPIE Three-Dimensional Image Processing and Applications*. Vol. 7526. 2010. Pp. 75260E-75260E-11.

[4] Dorf, R.C., Ed. *The Electrical Engineering Handbook*. 2nd Edition. Boca Raton: CRC Press LLC. 1997.

[5] Shafi, A. Bin Picking: Definitions, Success Rates and New Solutions with Random Positions, Object Shape Variation and Weak Imaging Features. In *Proceedings of the International Conference for Vision Guided Robotics*. 2011.

[6] Oh, J.-K., Baek, K.K., Kim, D., and Lee, S. Development of Structured Light based Bin Picking System Using Primitive Models. In *Proceedings of the IEEE International Symposium on Assembly and Manufacturing*. 2009. Pp. 46-52.

[7] Sadin, S.R., Pvinelli, F.P., and Rosen, R. The NASA Technology Push Towards Future Space Mission Systems. In *Acta Astronautica*. Vol. 20. 1989. Pp. 73-77.

[8] Mankins, J.C. Technology Readiness Levels: A White Paper. NASA, Office of Space Access and Technology. 1995.

[9] U.S. Department of Defense. *2012 Defense Acquisition Guidebook*. http://dap.dau.mil.

[10] U.S. Department of Energy. *DOE G 413.3-4, U.S. Department of Energy Technology Readiness Assessment Guide*. 2009.

[11] U.S. National Aeronautics and Space Administration. *NASA Procedural Requirements NPR7120.5D: NASA Space Flight Program and Project Management Requirements*. 2010.

[12] U.S. Department of Defense. *DoD Manufacturing Readiness Level Deskbook*, V2.01. July, 2011. http://www.dodmrl.com.

Characterizing Performance Guarantees for Multiagent, Real-Time Systems Operating in Noisy and Uncertain Environments

Damian Lyons
Computer & Information Science
Fordham University
Bronx, NY 10458
718-817-4480
dlyons@cis.fordham.edu

Ronald Arkin
School of Interactive Computing
Georgia Institute of Technology
Atlanta, GA 30332
404-894-9311
arkin@cc.gatech.edu

Stephen Fox
Computer & Information Science
Fordham University
Bronx, NY 10458
718-817-4480
stfox88@gmail.com

Shu Jiang
School of Interactive Computing
Georgia Institute of Technology
Atlanta, GA 30332
404-894-9311
sjiang@gatech.edu

Prem Nirmal
Computer & Information Science
Fordham University
Bronx, NY 10458
718-817-4480
prem.nirmal88@gmail.com

Munzir Zafar
School of Interactive Computing
Georgia Institute of Technology
Atlanta, GA 30332
404-894-9311
mzafar7@gatech.edu

ABSTRACT

Autonomous robots offer the potential to conduct Counter-Weapons of Mass Destruction (C-WMD) missions in an efficient and robust manner. However, to leverage this potential, a mission designer needs to be able to determine how well a robot system will operate in the noisy and uncertain environments that a C-WMD mission may require. We are developing a software framework for verification of performance guarantees for C-WMD missions based on the *MissionLab* software system and a novel process algebra approach to representing robot programs and operating environments.

In this paper, we report on our initial research for the Defense Threat Reduction Agency (DTRA) in understanding what is required from a performance guarantee to give a mission designer the information necessary to understand how well a robot program will perform in a specific environment. We link this to prior work on metrics for robot performance. Using a simple mission scenario, we explore the implications of uncertainty in the four components of the problem: the robot program, and the sensors, actuators and environment with which the program is executed.

Categories and Subject Descriptors

I.2.9 Robotics; D.2.4 Software/Program Verification; D.2.6 Programming Environments

General Terms

Performance, Languages, Verification, Robotics.

Keywords

Performance guarantees, probabilistic and emergent robotic systems.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage, and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. PerMIS'12, March 20-22, 2012, College Park, MD, USA. Copyright © 2012 ACM 978-1-4503-11267-3/22/12...$10.00

1. INTRODUCTION

To effectively deploy an autonomous robot or robot team to search and locate weapons of mass destruction, it is important to have performance specifications and guarantees available for the equipment. Because of the severe potential downside in these mission-critical operations, the robot and its software must have the best chance of succeeding given the environmental conditions and other constraints in which it must operate. However, this environment may be uncertain, and the software that operates the robot or robot team may be probabilistic [20], emergent [1], and/or multiagent [3]. Although tremendous strides have been made in software verification (e.g., [9]), this high-impact problem remains extremely challenging.

An important component of the solution is to understand what performance guarantees are useful and possible for Counter-Weapons of Mass Destruction (C-WMD) missions. In this paper we present an overview of the system, which is based on the *MissionLab*[1] mission specification system [17], being developed for integrating the generation and use of performance guarantees as an iterative step in the design of robot software for C-WMD missions. Using examples in this design framework, we analyze what mission performance guarantees are of value to a mission designer from the perspectives of understanding how well the system will function and of understanding how to improve its performance.

In the next section, we review related work in the area of automatic verification of system performance, and in the development of performance measurements and guarantees. Section 3 reviews a selection of performance measurements. In Section 4 we introduce a simplified example scenario to help understand how uncertainty in sensor, actuator and environment

[1] *MissionLab* is freely available for research and educational purposes at: http://www.cc.gatech.edu/ai/robot-lab/research/*MissionLab*/.

models influences the form of the performance guarantee, making it quite different from the form of liveness and safety guarantees typically seen in software verification. Section 5 then introduces the architecture we have developed to integrate verification into the *MissionLab* software system.

2. RELATED WORK

The field of formal specification and verification of software systems (e.g., Hinchey et al. [7], Clark et al. [4]) has made impressive progress. However, leveraging these results to validate software for mobile robot systems has raised challenges. Probabilistic [20] and behavior-based mobile robotics [1] employ assumptions quite different from those used more generally in the formal analysis of software. One key example is a reliance on emergent behavior: even simple behavior-based systems exhibit complex behavior when acting in a complex environment. This means that formal analysis must include the control program and models of the sensory and motor apparatus as well as environment models.

Discrete-Event Control techniques (e.g., Ramadge [19], Kosecka [10]) have been applied to this problem. Most use Finite State Automata (FSA) as a modeling tool. However, FSA models can suffer from state-space explosion when used to model the kind of realistic search environments that occur in C-WMD. While prior work addresses issues of noisy and uncertain applications, it does so for problems at a relatively low sensorimotor level as compared to for example, algorithms from data mining, artificial intelligence, machine learning and complex adaptive systems theory. Also, work in this area is focused on automatically producing a control strategy or controller, whereas our focus is on verifying software produced by some other means (in our case, generated by a human operator using *MissionLab*). More recently the discrete-event and hybrid approach has been extended to robot path planning and motion control (e.g., Kress-Gazit [11]) with the idea that a human provides a high-level, rich constraint description in linear or interval temporal logic, and a controller is automatically synthesized for these constraints. However, the input constraint or constraints in these systems are quite complex and themselves may now need verification.

The metrics for the performance measurement and guarantees of behavior-based and probabilistic software systems have not been standardized so far, although considerable work is proceeding in the characterization of performance metrics for robot performance [8]. This is the case not only with behavior-based systems but with a broader category of systems that are required to carry out specific tasks intelligently by interacting with real world environments. Serious effort is underway towards standardization of these metrics [16] but the challenges are many. Behavior-based system requirements need to cover a wide spectrum of behaviors ranging from simple tasks such as point-to-point locomotion to relatively complex tasks such as human-robot interaction. The expectations are growing regarding reliable and predictable performance as new possibilities in design are being explored and milestones are being achieved.

Urban search and rescue (USAR) is a domain that is being heavily studied in this context. There are two groups of performance metrics for the characterization of USAR systems that can be broadly classified as system characterization and behavior characterization. System characterization seeks accurate specification of specific robot capabilities to facilitate direct comparisons of different robotic platforms, and particular configurations of similar robot models. The National Institute of Standards and Technology (NIST) has taken a leadership role for defining performance standards for USAR robots [8]. These standards are categorized as human-robot interaction, system, safety, mobility, etc., along with documentation for standard reproducible test procedures. For our purposes, these system metrics will primarily serve as specifications of particular capabilities of the robot with the view of providing a guarantee to the user regarding the ranges of behaviors the system provides, before it is deployed in the real world in the context of a C-WMD mission. Behavior characterization deals with the problem of predicting performance guarantees for high-level tasks to be carried out in uncertain, unstructured, and potentially hostile environments such as navigation, localization and mapping, room search, etc. Some related research exists in performance characterization of higher-level algorithms, i.e., [18] [5], that is intended for the comparison of different algorithmic performance. This comparison would traditionally be done by demonstration (empirical evaluation) instead of formal analysis. Such metrics, however, may prove to be useful as they may improve the expressiveness with which the operator can specify required performance.

3. PERFORMANCE CRITERIA

An important requirement for any evaluation is the establishment of the performance criteria which will serve as the basis for specification and evaluation of the system in question. A method is needed for defining performance goals which not only accommodates various ranges of capabilities but in our case also comfortably fits into the process algebra framework we use for verification; this framework is based on that described in [12]. The absence of any published standards in this regard as well as the growing needs for the capabilities of C-WMD/USAR systems makes this an important area of investigation.

Due to the complexity associated with many formal methods, the performance of control algorithms designed for robots has traditionally been guaranteed only through empirical evaluation and demonstration on real systems. Many performance criteria have been devised to compare the performance of such algorithms in this context [8]. Those criteria serve as a reference for defining the mission performance criteria for our verification procedures.

Since we are targeting the USAR/C-WMD applications, a good starting point is to identify the most common requirements in this application area. These include navigation, exploration, localization, mapping, search, and victim identification (among other things). We can then refer to the large body of literature available for the performance evaluation of the algorithms designed for these high-level system goals. In navigation, for example, [18] has proposed a set of useful performance metrics along with their formulae and algorithms that could directly be applicable to our framework. These include safety metrics (e.g. mean obstacle distance), dimensional metrics (e.g. trajectory length, time of completion) and smoothness metrics (e.g. bending energy, smoothness of curvature). Similar propositions are made in [5]. Related work is available for other areas of application as well. Currently there is no universal agreement with regards to these metrics, but it is hoped that the availability of common tools and techniques to verify, validate, and formally prove performance guarantees for high-level mission controllers will lead to standardization of such performance characterization.

The metrics discussed above can be accommodated as part of our framework, allowing the user to specify the mission goals and expectations, i.e., specific mission criteria. In the case of multiple metrics/criteria, the user may then choose to investigate whether a mission is likely to experience a catastrophic failure or whether a graceful degradation is more likely. This is a powerful feature of our approach; we are not just interested in binary yes/no answers regarding performance guarantees as might be typical for more traditional software verification. The information that a mission designer or operator needs to decide whether to deploy a robot mechanism for a C-WMD mission includes not only the standard concepts of mission completion ('liveness') and safety, but also information about how likely overall success might be, given the noisy and uncertain environment for the mission.

4. ROBOT SCENARIOS

Performance criteria need to reflect the missions with which robots will be tasked. In this section we look at several example missions and consider how they impact what must go into a performance criterion. In the first example the robot control strategy is deterministic, where the sensor and actuators operate with no noise and where there is no uncertainty in the environment model.

4.1 Deterministic Scenario

A robot searching an area for a target executes actuator commands to move through the search area, deploying its sensors to search for the target.

- The robot program is deterministic.
- If the actuators always carry out the motion commands exactly, then the robot program can always rely on knowing where it is and hence where it has been.
- If the sensors always report the situation in the environment with certainty, then obstacles, other agents and the target can always be reliably detected.
- Finally, if the environment in which the robot operates has no associated uncertainty, then the robot program will always fulfill its mission requirements or it will always fail.

This deterministic scenario does not reflect many actual operating situations; however, it is necessary to include it as a base case. We introduce a very straightforward example of a search task to drive this and the succeeding scenarios. Consider a robot moving from one location A to a second location B repeatedly as shown in Figure 1.

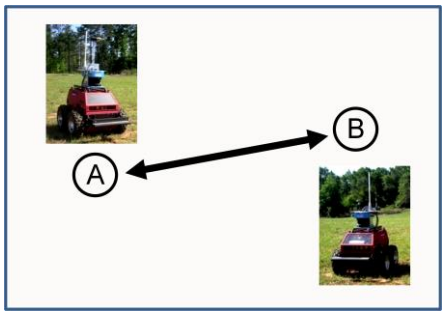


Figure 1: Repeated Traverse Mission

The mission designer is interested in two kinds of guarantees which we can broadly categorize using the traditional Liveness and Safety terms:

1. **Liveness**: Will the robot achieve a mission objective? Examples might include:
   - Will the robot arrive at B? (Note that the complexity of the control strategy or environment model, or the accuracy of the sensors or actuators, may still render this a difficult verification problem.)
   - Will the robot complete *n* traversals from A to B?
   - Will the robot complete *n* traversals from A to B by time *t?*
2. **Safety**: Will the robot be free of error situations while carrying out its mission object? Examples could include:
   - Will the robot avoid any and all obstacles between A and B?
   - Will the robot keep its power consumption within safe levels at all times?
   - Will the robot always read its radiation sensor at a rate of 10Hz or higher.

Because there is no uncertainty in this example scenario, the performance guarantees exhibit a binary nature; the robot program will conform to the performance guarantee or it won't. This is typical of the kind of verification constraints seen in general-purpose software verification.

4.2 Nondeterministic Environment

Consider a modification of the previous example in which the terrain between locations A and B has an element of uncertainty with respect to its traversability. The actuators and sensors remain deterministic in their performance and the search program itself is deterministic.

The environment in which the robot now has to operate is one that can contain patches of terrain that are more difficult to traverse and the robot will make less progress on these patches. Any particular execution of the robot mission will encounter some number of patches and be slowed as a result. Different executions might encounter different numbers of patches, and hence exhibit a range of performance.

This possible range of performance complicates the performance guarantee beyond the binary case we have discussed before. Now consider the liveness condition: Will the robot complete *n* traversals from A to B in time *t?* In the deterministic scenario, the robot would either always or never achieve this. However, in this scenario, there will be some executions in which the robot does achieve this performance and some in which it does not.

4.2.1 Expected performance

If we leverage the probabilistic concept of expected value, then one approach is to ask:

- Is the number of *expected traversals* from location A to location B in time *t* equal to *n?*
- Alternatively we can ask, is the *expected time* for the robot to complete *n* traversals from location A to location B equal to *t*?

Even though the environment is not deterministic, this form of the performance guarantee maintains the easy binary structure of the

deterministic case. This increases the realism of the scenario without complicating the way in which the mission designer has to understand performance.

Nonetheless, this approach does hide the variation in performance behind the concept of expected value. That variation may itself be a useful and sometimes necessary tool for the mission designer.

4.2.2 Performance Confidence

In scenarios where the options are limited and the risks are high, a mission designer may consider it reasonable to deploy a robot for a mission even though the reasons to believe the robot will succeed are somewhat slim. Therefore it is also important to make the information about the variability in performance available to the designer in a performance guarantee.

Returning to the traverse example, a designer can reasonably want to know:

- *how likely it is* that the robot will complete *n* traversals from location A to location B in time *t* given the environment in which it has to carry out the mission.

This additional information is purchased at the cost of complicating the performance guarantee to include a probability that needs to be interpreted by the mission designer. A reasonable interpretation might be: For a very large number of executions in this environment, in what percentage of executions does the robot complete *n* traversals from location A to location B in time *t* or less?

4.3 Noisy Sensors and Actuators

Moving another step towards making our initial, deterministic scenario more realistic, let us now consider a situation where the robot sensors and actuators operate with noise. That is, the motion command communicated to the robot by the robot program may not always produce the same effect on the robot, and a sensor reading taken during the identical environmental conditions may yield different measurements. The robot program remains deterministic.

4.3.1 Expected Performance

The consequence of this uncertainty for the repeated traversal mission is that the robot may not always reach the locations A and B, irrespective of terrain traversability. After some number of traversals, the robot may conceivably have drifted far from A and B. A mission designer might ask:

- After *n* traversals from A to B, will the expected location of the robot be within a distance *r* of location B?

This is an application of the expected value concept again, but in this scenario to a spatial objective rather than a temporal one.

4.3.2 Performance Confidence

In the scenarios in which knowledge of the variation in performance is important, a designer may want to ask:

- After *n* traversals from A to B, how likely is the robot to be within a distance *r* of location B, given the environment in which the program is carried out.

This more complex performance criterion can be interpreted as follows: after a large number of different executions of the program in this environment, in what percentage of them was the robot within a distance *r* of the location B.

Even this more complex form of the performance criterion hides information. If the likelihood of being within *r* of location B is a value *p*, then for the remaining *1-p* cases we can ask, how badly do they each fail to meet this criterion?

4.3.3 Performance Distribution

A description of the performance of the system in the cases in which the robot program does not meet its performance criteria contains valuable information. Let us consider that the sensor and actuator models are now extended to include the case of sensor and actuator failure. For the repeated traversal mission, not only may the robot position drift from the goal locations, it may go catastrophically wrong as the robot becomes stuck at a location.

Consider the graphs shown in Figure 2. The horizontal axis is position and the vertical is the likelihood of attaining that position given the environment in which the program is executed. The location of the point B is indicated as a vertical line intersecting the horizontal axis.

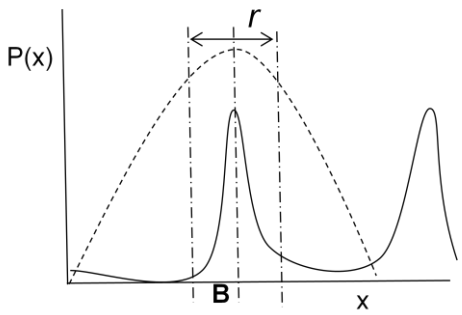


Figure 2: Two examples of spatial distributions

The figure shows examples of two different models for the distribution of the spatial likelihood. The first, shown as a dotted line, is one in which the likelihood falls off smoothly on either side of the location B. If a threshold range *r* around location B is selected, and the performance criterion asks the likelihood of the robot being within *r* of location B, then in both of the example distributions shown here, the likelihood is fairly large. However, in the case of the distribution shown as a dotted line, the failure cases are also locations close to location B. This is a model of a favorable kind of failure.

This is in contrast to the distribution indicated as a solid line in Figure 2. In that case, few of the failure cases, those cases outside of the spatial interval *r* around B, are close to B. The failures in this case are mostly severe failures.

4.4 Probabilistic Robot Program

The final level of complexity that we add to the simple scenario introduced in this section is the inclusion of probabilistic algorithms for control of the robot mechanism. Probabilistic algorithms have been developed for many applications including mapping and for robot localization. Let us consider that we add a probabilistic localization algorithm, such as Monte-Carlo Localization, to the robot program that controls the robot to carry out the repeated traverse mission and explore what this implies for the performance criterion.

The effect of a good probabilistic algorithm should be to improve the performance of the robot in a noisy and uncertain environment, and that of a poor algorithm, to reduce the

performance. The mission designer is only interested in whether the robot can achieve location B, with constraints perhaps on the time, the number of traversals and so forth. We note therefore that although the addition of this probabilistic algorithm complicates the mechanics of verification, it does not change the form of the performance guarantee for the program.

5. INTEGRATING VERIFICATION AND DESIGN

This performance guarantee component is being embedded into the *Missionlab* software package, a comprehensive robot mission development, simulation and execution environment. The robot software designer builds her program within *MissionLab* using the visual software authoring tools provided. *MissionLab* allows the high-level mission that is generated to be tested in simulation first, for verification of the user's intent, and then deployed to one or more robot platforms for execution.

The newest components of *MissionLab*, which are based on the formal modeling described in Lyons and Arkin [12], allow the designer to carry out an additional software verification step to establish performance guarantees for the user-defined mission software. This can be very useful in mission-critical or emergency response situations (including C-WMD missions such as finding, containing, and neutralizing Chemical-Biological-Nuclear (CBN) weapons), where it is not uncommon for robot operators to customize the robot software, and even hardware, for the specific mission; and failure of the mission is not an option in these emergency situations.

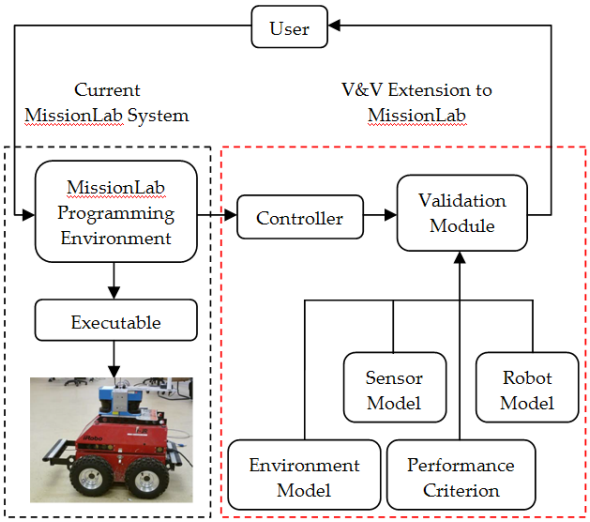


Figure 3: *MissionLab* System with integrated verification module.

Figure 3 depicts the verification extension to the existing *MissionLab* system. The extension provides an operator feedback loop in the robot software design process. The process starts with the designer creating a robot program in the usability-tested *MissionLab* programming environment for a specific mission [6] [14]. Once the high-level mission is specified, the designer may simulate the robot behavior within *MissionLab* to verify correct behavior according to the operator's intent. However, this simulation cannot ever fully capture the interaction between the robotic hardware and the real environment. To further guarantee mission success in the real environment, the robot controller can be validated using the verification module. The verification module provides an output to the user indicating whether the controller will meet the performance criteria specified by the operator. If the controller cannot meet the specified criteria, the designer may modify the robot program and the design loop continues. Once it does satisfy the requisite criteria, the designer may proceed to generate an executable for the robot and then deploy it to undertake the mission.

5.1 Verification Module Inputs

The inputs to the verification module are the robot software controller (specified in an intermediate language referred to as CNL [17]), sensor, robot, and environment models, and the user-specified performance criteria. In *MissionLab*, the robot controller is specified visually by the designer at a very high level of abstraction. An example of using *cfgedit* in *MissionLab* to design a mission is shown in Figure 4. The models of sensors, robots and the environment in which the robot program will execute can simply be selected from existing libraries. These libraries are part of the verification system and are constructed using the modeling approach described in this paper. Figures 5-7 show examples of the model libraries. Due to the limited space here, only a subset of exemplar components of the libraries are shown.

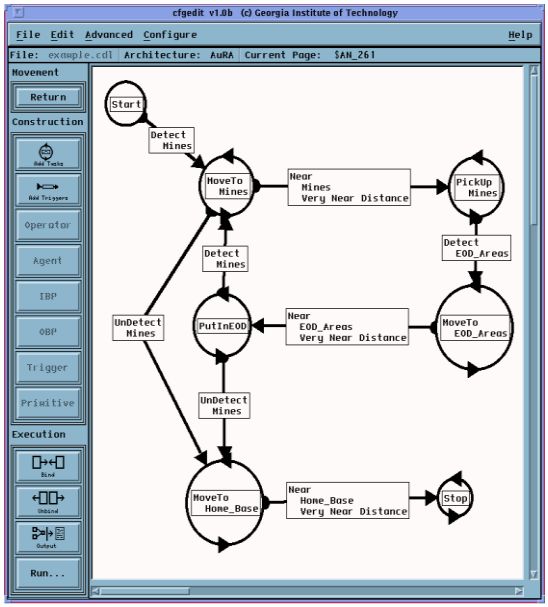


Figure 4: Example of Mission Design in *MissionLab*

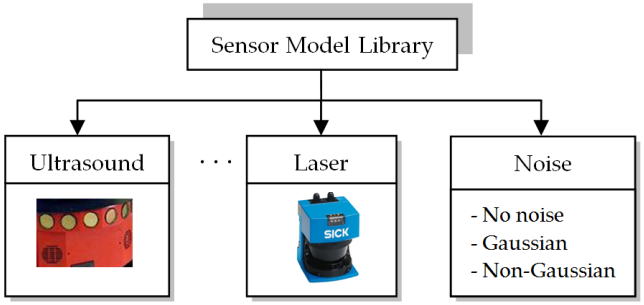


Figure 5: Example of sensor model library

Once the mission has been built, the designer selects from the libraries of sensor and robot models that include a range of noise and uncertainty characteristics (Figures 5 and 6). In a similar fashion the designer composes an environment model by selecting from a library of environments (Figure 7).

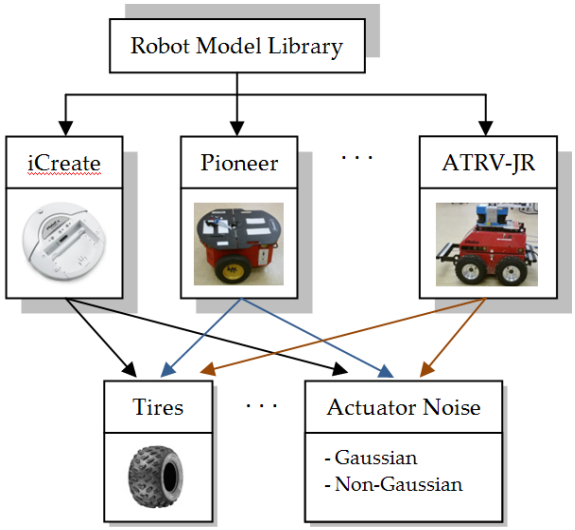


Figure 6: Example of robot model library

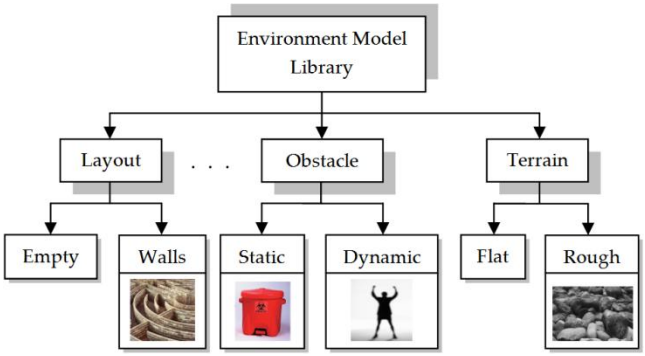


Figure 7: Example of environment model library

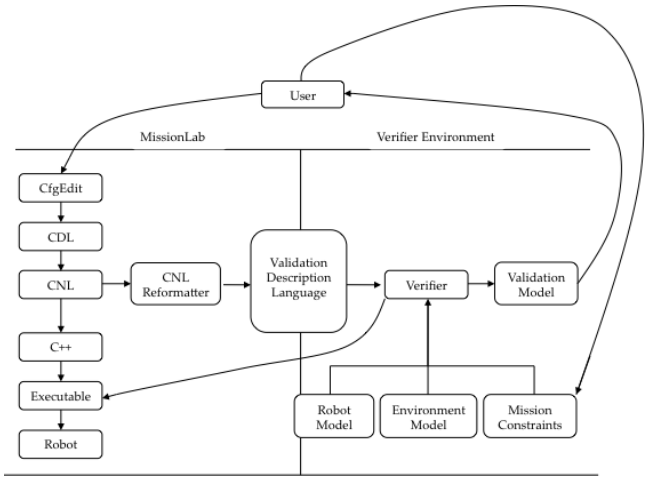


Figure 8: Overall architectural design showing user interaction

Based on the sensor, motor and environment choices made, the designer is offered a selection of customizable verification conditions and constraints. Verification includes the testing of the combination of robot program with the environment model for specific properties of safeness, liveness, and/or efficiency. The result of this testing is the establishment of performance guarantees for the software in the environment represented by that environment model. If the result is unsatisfactory, in terms of design objectives, the designer can use the feedback from the verification to iteratively refine the robot program. In other words, besides telling the designer "yes/no" that the robot program is satisfactory vis-à-vis the mission, the verification module also identifies potential causes of failure in the program and provides the designer with this useful information. This process is illustrated in Figure 8.

Input – Program P in MissionLab,
Environment class E
from library
Performance criterion C(E)

Output – Will P meet C(E)?
In what way will P *not*
meet C(E)

Figure 9: Verification Module Input and Output

5.2 Verification Module

The verification module is based on an approach introduced by Lyons and Arkin [12] to present robot programs and the environment in which they operate as networks of processes. The programs and environments are specified and analyzed using process algebra [13], which is a mathematical framework that takes a compositional approach to describing process networks. The semantics of a process in this framework is a port automaton: an automaton augmented with the ability to send and receive communication messages.

This approach has a number of important advantages:

- The robot program, sensor and actuator models, and environment model can all be specified in one notation.

- The concurrent and communicating composition of program, sensor and actuator models and environment is the object of verification
- Noisy and incomplete information is represented as the interaction of stochastic processes.
- The algebraic foundation supports verification by automated algebraic reasoning rather than by 'simulated execution' or enumerative model checking, both of which have significant computational complexity.

The verification module does not need to carry out a general software verification step**,** e.g., [9]. In general purpose software verification, the verification criterion can include a constraint on any of the variables within the program and their value.

The performance guarantee in our application concerns the robot and its operating environment, not the robot program directly. Variables from the environment, such as the position of the robot, time, and so forth, can be included in the performance guarantee. However, variable values within the robot program are only of interest in so far as they may affect these variables from the environment.

Furthermore, the models for the robot and its environment, selected by the mission designer to validate the program, come from the robot, sensor and environment libraries mentioned earlier. This means significant preprocessing can be carried out on these models to simplify their composition with other models, and their verification with a robot program.

6. CONCLUSION

In this paper, we described a software framework for validating performance guarantees for C-WMD missions based on extensions to the *MissionLab* mission specification system and on a novel process algebra approach to represent robot programs and operating environments. The key focus in the paper is on the problem of what the performance guarantee should look like from an operator's perspective. We reviewed the state of the art in performance measurements for robots and presented candidate measurements for the performance guarantee. Using a simple example scenario, we looked at the implications of uncertainty in sensor and actuators, as well as uncertainty in the environment, on the form of the performance guarantee.

To be useful to a mission designer, the performance guarantee must allow intuitive expression of the variance in performance of the program due to uncertainty, including the use of the expected value of environment variables, the likelihood of an environmental variable being within a specified range, and, to understand the severity of failure, the distribution of values for an environment variable.

The study described in this paper serves as the basis for our on-going work for the Defense Threat Reduction Agency in process algebra verification of robot missions and in the construction of the verification module for *MissionLab*.

7. ACKNOWLEDGMENTS

This work was supported by the Defense Threat Reduction Agency, Basic Research Award # HDTRA1-11-1-0038.

8. REFERENCES

[1] Arkin, R.C., *Behavior-based Robotics*, MIT Press, 1998.

[2] Arkin, R.C., Diaz, J. *Line of Sight Constrained Exploration for Reactive Multiagent Robotic teams*, AMC02, July 2002, pp. 455-461.

[3] Balch, T. and Parker, L., *Robot Teams: From Diversity to Polymorphism*, AK Peters, 2002.

[4] Clark, E., Grumberg, O., Peled, D., *Model Checking*. MIT Press 1999.

[5] Daniele Calisi, Daniele Nardi *Performance evaluation of pure-motion tasks for mobile robots with respect to world models*, *Autonomous Robots* 27(4):465-481,2009.

[6] Endo, Y., MacKenzie, D., and Arkin, R.C., Usability Evaluation of High-level User Assistance for Robot Mission Specification, *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 34, No. 2, pp. 168-180, May 2004.

[7] Hinchey M.G., and J.P. Bowen, *High-Integrity System Specification and Design*, FACIT series, Springer-Verlag, London, 1999.

[8] Jacoff, A., Messina, E., Standard Test Methods For Response Robots, ASTM E54.08.01 Intelligent Systems Division, NIST 2011

[9] Jhala, R., Majumdar, R., Software Model Checking, *ACM Computing Surveys* V41 N4 Oct 2009.

[10] Kosecka, J. (1996). *A Framework for Modeling and Verifying Visually Guided Agents, Analysis and Experiments*, Ph. D. dissertation, Dept of Computer and Information Science, University of Pennsylvania.

[11] Kress-Gazit, H., and G. J. Pappas, Automatic Synthesis of Robot Controllers for Tasks with Locative Prepositions, *IEEE International Conference on Robotics and Automation*, Anchorage, Alaska, May 2010.

[12] Lyons, D., and Arkin, R., Towards Performance Guarantees for Emergent Behavior, *Proc. 2004 IEEE International Conference on Robotics and Automation*, New Orleans, LA, May. 2004.

[13] Lyons, D.M., *Representing and analyzing action plans as networks of concurrent processes*. IEEE Transactions on Robotics and Automation, V9 N3 June 1993 pp.241-256.

[14] MacKenzie, D., and Arkin, R., Evaluating the Usability of Robot Programming Toolsets, *International Journal of Robotics Research*, Vol. 4, No. 7, April 1998, pp. 381-401.

[15] MacKenzie, D., Arkin, R.C., and Cameron, R., *Multiagent Mission Specification and Execution*, Autonomous Robots, Vol. 4, No. 1, Jan. 1997, pp. 29-52.

[16] Madhavan, Raj; Tunstel, Edward; Messina, Elena (Eds.), *Performance Evaluation and Benchmarking of Intelligent Systems*, ISBN 978-1-4419-0491-1, 2009.

[17] MissionLab v7.0 User Manual, available at http://www.cc.gatech.edu/aimosaic/robot-lab/research/MissionLab/mlab_manual-7.0.pdf

[18] Muñoz,N.D., and J. A. Valencia, N. Londoño, *Evaluation of Navigation of an Autonomous Mobile Robot*, 2007.

[19] Ramadge R.J., and W. M. Wonham, 1987. *Supervisory control of a class of discrete event processes*. SIAM J. Control and Optimization, 25(1), pp. 206-230.

[20] Thrun, S., Burgard, W., and Fox, D., *Probabilistic Robotics*, MIT Press 2005.

Design, fabrication and characterization of a single-layer out-of-plane electrothermal actuator for a MEMS XYZ stage

Yong-Sik Kim
National Institute of Standard and Technologies,
Intelligent System Division
100 Bureau Dr. Gaithersburg, MD, 20899, USA
1-301-975-8081
mk37do@gmail.com

Nicholas G. Dagalakis
National Institute of Standard and Technologies,
Intelligent System Division
100 Bureau Dr. Gaithersburg, MD, 20899, USA
1-301-975-5845
nicholas.dagalakis@nist.gov

Satyandra Gupta
University of Maryland,
Department of Mechanical Engineering
University of Maryland, College Park, Maryland 20742, USA
1-301-405-5306
skgupta@umd.edu

ABSTRACT

This paper presents the design, fabrication and characterization of a single-layer out-of-plane electrothermal actuator based on MEMS (Micro-Electro-Mechanical System). The proposed electrothermal actuator is designed to generate motions along the out-of-plane or normal to a wafer by a Joule heating when the current flows through the actuator. This out-of-plane electrothermal actuator is based on a single layer of a SOI (Silicon on Insulator) wafer and two notches near the middle of the actuator beams. Due to these notches, the thermal expansion of the beams in the actuator generates an eccentric loading, which converts into the bending of the beams. This bending of the beam finally generates the out-of-plane motion at the middle of the beam. This behavior is described by the prepared analytic equations and compared by the results from FEM (Finite element Model) analysis. With fabricated samples, a 30 μm displacement is measured along out-of-plane at 5 V driving voltage. The 1st mode of the resonant frequency for the out-of-plane motion is expected to occur at 74.9 kHz from FEA. The proposed actuator is based on the standard SOI-MUMPs (SOI-Multi User Manufacturing Process) , so it has good integration capability with other system employing same fabrication techniques. To test its integration capability, a MEMS XYZ stage is fabricated by embedding the proposed out-of-plane electrothermal actuator onto an existing MEMS XY stage. The range of motion of the fabricated XYZ stage is measured about 35 μm x 35 μm x 30 μm along X, Y and Z axes without any changes on its fabrication process.

Categories and Subject Descriptors

B.7.1: [Integrated Circuits]: Types and Design Styles -- advanced technologies

General Terms

Experimentation, Design

(c) 2012 Association for Computing Machinery. ACM acknowledges that this contribution was authored or co-authored by a contractor or affiliate of the U.S. Government. As such, the Government retains a nonexclusive, royalty-free right to publish or reproduce this article, or to allow others to do so, for Government purposes only. PerMIS'12, March 20-22, 2012, College Park, MD, USA. Copyright © 2012 ACM 978-1-4503-1126-7-3/22/12...$10.00

Keywords

MEMS, out-of-plane, electrothermal actuator, eccentric loading, buckling

1. INTRODUCTION

The MEMS (Micro Electro Mechanical System) based actuators have been widely used at metrology and micro-manufacturing applications such as micro confocal imaging [2, 3, 4], bio-cell manipulation [5] and nano-assembly [6], due to their accurate motion, nanometer level resolution, small footprint, system integration flexibility and tens of micron meters of motion [1].

There are some factors to consider at designing MEMS actuators; range of motion, thrust force, frequency response and integration capability with external system. The system integration capability includes the difficulty of the fabrication process. If the fabrication of the proposed system is too difficult to complete or totally different fabrication techniques are needed, its integration capability is also affected by these factors. If the proposed system doesn't need any special processes and can share with the fabrication process of the target systems, the proposed system has a reasonable integration capability with the target systems. Therefore, the integration capability is also an important factor to consider at design level when the tight interaction with other external system is required.

Among a variety of integration, a combination of in-plane actuation systems with out-of-plane actuation systems is not common. This is because the fabrication processes for the in-plane actuators are considerably different from that of the out-of-plane actuators. For out-of-plane motion, structures in height are needed, so at least two or more layers are needed to be stacked up and combined together for the out-of-plane motion as shown in bi-morph or multi-morph actuators [11, 12]. But, most in-plane actuators don't need multiple layers for their operation [8]. This structural differences cause difficulty in integration. This paper describes the design and fabrication of the out-of-plane actuator with reasonable integration capability with in-plane actuation systems.

Among various MEMS based actuators, an electrothermal actuator can generate a stronger force with a smaller footprint and a simpler design than other actuators including electrostatic actuators [7]. But, many out-of-plane electrothermal actuators reported have multi-layers forms. The bimorph electrothermal

actuator utilizes different thermal expansion coefficients between layers. When one layer is on top of the other and their ends are linked together, different thermal expansion can generate the out of plane motion [12, 13]. But the connection of parts between multi-layers tends to be under significant shear stress during operation, so those tend to fail faster than other parts and reduce the total life-time of the system. To avoid this kind of a problem, a single layer electrothermal actuator was introduced [10]. The single layer actuator is made of a single layer with different height like bridges or steps [9, 10], so there is no stress concentration between layers. With the single layer, the life-time of the proposed actuator can be protected [10]. But this step-bridge shape needs additional processes to build the desired step shapes on the single layer. This can reduce its integration capability with external applications. Another approach is to use a single-layer simple beam operating within its buckling mode. By utilizing a buckling mode, the out-of-plane motion can be achieved without a special fabrication process. But most buckling modes are bi-stable and a little bit unpredictable, so it is not easy to control the actuator at a desired position between bi-stable positions accurately [11].

This paper presents the design, fabrication and testing of an electrothermal actuator for out-of-plane motion. The proposed actuator is also made of a single layer and follows standard Silicon-On-Insulator Multi-User-Manufacturing Processes (SOI-MUMPs). With these features, the proposed actuator is expected to have easy integration with other in-plane actuators, if they are also based on the same fabrication processes (SOI-MUMPs). The proposed electrothermal actuator design consists of a single beam and two notches at both ends of the beam as shown in Figure 1(a). Due to the single layer with notches, Joule's heating of the beam generates the eccentric load, which converts into the bending moment for the out-of-plane motion. To demonstrate its integration capability, one existing XY stage is adopted and the proposed actuator is embedded into it, which follows standard SOI-MUMPs. This paper is organized as follows: An analytic expression was developed for the design of the proposed actuator with appropriate dimensions in section 2. This design was verified by finite-element-analysis (FEA) in section 3. Detail fabrication description was provided in section 4. The range of motion was measured at section 5. Easy integration with existing MEMS systems was demonstrated by building one MEMS XYZ stage with the proposed actuator in section 6. The conclusion and discussion are in section 7.

2. DESIGN AND ANALYSIS

Figure 1 illustrates the CAD design of the single layer electrothermal actuator and a close-up view near its attachment points notches (see Figure 1(b)). These are the flexure type structures necessary for implementing the out-of-plane bending motion of the actuator.

The schematic diagram of the proposed actuator beam can be represented like the one shown in Figure 1(c). When the thermal expansion occurs, the expected deformation is shown in Figure 1(d). It is clear that the thermal expansion generates the out-of-plane motion due to a bending moment. Figure 1(e) shows the free-body diagram of the beam. Due to the beam notches, the external force P can be regarded as an eccentric loading applied to the beam ends at distance e. The eccentric loading P can be replaced with the centric force P and a couple of moment M, as shown in Figure 1(f). Based on the free-body diagram in Figure 1(f), the beam differential equation can be written and solved as [14]:

(1)

— — — — (2)

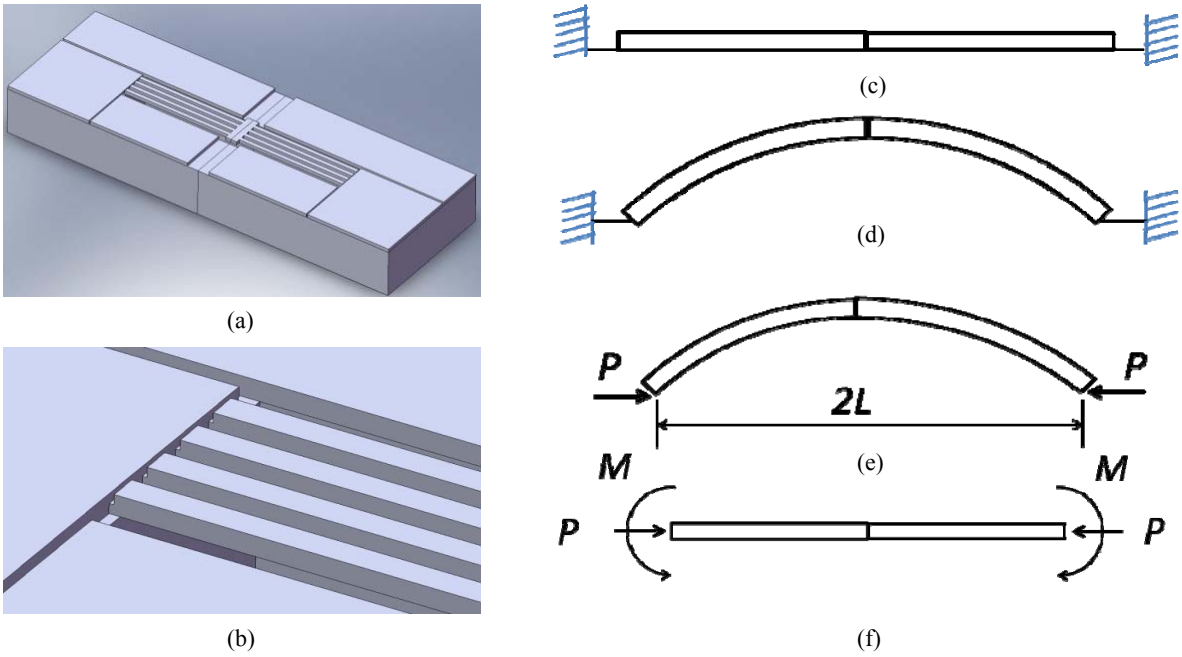
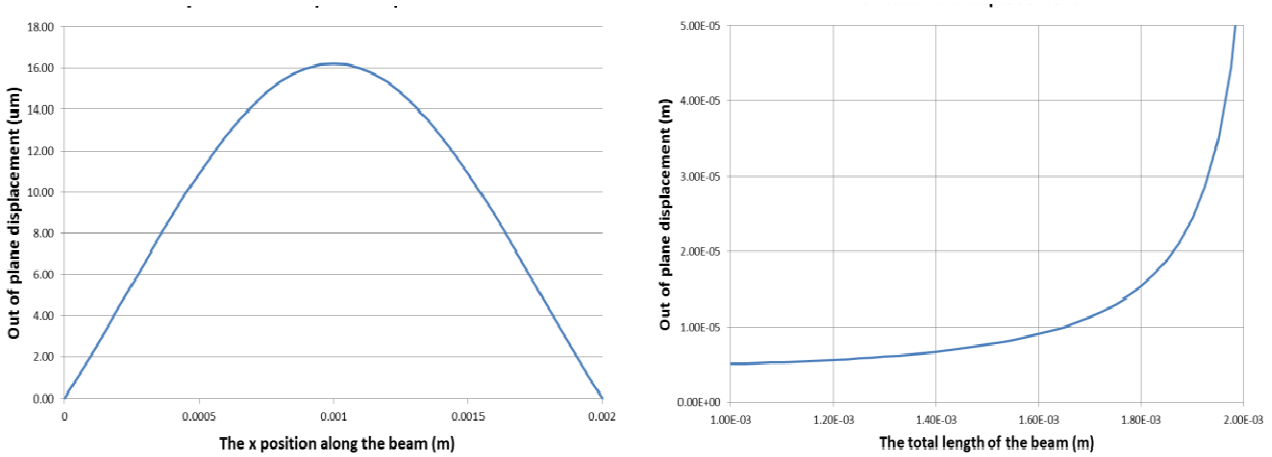


Figure 1. Schematic design of the electrothermal actuator; (a) the basic design of the electrothermal actuator for the out-of-plane motion; (b) the notch on the beam in the actuator; (c) a schematic diagram of the proposed actuator; (d) the expected deformation when the thermal expansion occurs; (e) a free-body diagram of the proposed actuator; (f) the converted free-body diagram



(a) (b)

Figure 2. Electrothermal actuator beam out-of-plane displacements; (a) for selected dimensions; (b) for various beam lengths

Where, E is the Young's modulus of Silicon and I is the moment of inertia. e is the distance from the center of the beam to the eccentric loading. The solution of the equation (2) with its boundary conditions (x0=0, y0=0 and x1=L, y1=0) can be given as:

(3)

The expected deformation profile of the beam for a P = 10 mN force is derived based on equation (3) and shown in Figure 2(a).

(4)

From equation (3), the maximum beam displacement can be calculated and given by equation (4). The maximum force that the electrothermal actuator can generate is expected to be 45 mN for the chosen dimensions [15]. The maximum beam displacement as a function of beam length is plotted in Figure 2(b). Due to the secant function, the output displacement is expected to be infinite with 2 mm beam length. But, this doesn't occur, because buckling occurs prior to this deformation.

Based on the equations described above, four design parameters were selected and listed in the table 1. The beam width is set to be 10 % bigger than the beam thickness to avoid any in-plane buckling. Other factors are obtained from similar research [15].

Symbol	Design parameter	Optimized values
W	Beam width	33 μm
n	Notch depth	15 μm
L	Beam length	1000 μm
T	Beam thickness	30 μm

Table 1. The dimensions of the design parameters

3. FINITE ELEMENT ANALYSIS OF THE XY STAGE

Finite element analysis (FEA) is utilized to verify the mechanical properties of the proposed actuator determined at the previous section. For this purpose, one of commercial FEA tools, ANSYS1 with the coupled field or multiphysics option, is utilized. The material properties needed for the simulation are cited from reference [15].

For more reasonable results, a few boundary conditions and assumptions are used; (1) the electric input voltage is applied at the two ends of the proposed actuator, (2) both ends of the proposed actuator are assumed to be firmly fixed at the substrate and all the other components are set free without any constraints, (3) one end is electrically isolated from the other, (4) to solve the thermal analysis problem both ends of the proposed actuator areconnected to heat sinks, (5) natural convection or radiation heat transfer are not included in this analysis. .

In the FEA process, a total of 12,650 nodes are generated for the proposed model and occupy 195.7 MB memory. The results from this analysis are described in Figure 3. With a driving voltage of 5.4 V a 2.54 μm displacement is expected along the out-of-plane direction. This displacement generates a maximum of 899.4 MPa von Mises stress and 541.34℃, which is still under its failure strength of 7 GPa (see Figure 3(b)) and its maximum allowable temperature limit of 550 ℃ (see Figure 3(c)). The notches experience of the highest stresses based on this analysis. With these features, the frequency response of the proposed actuator was also checked. Figure 3(d) shows its first resonance mode from the modal analysis. This mode is expected to occur along the out-of-plane direction at 70.4 kHz. The second mode is related to in-plane deformation at 97.6 kHz.

4. MICRO-FABRICATION

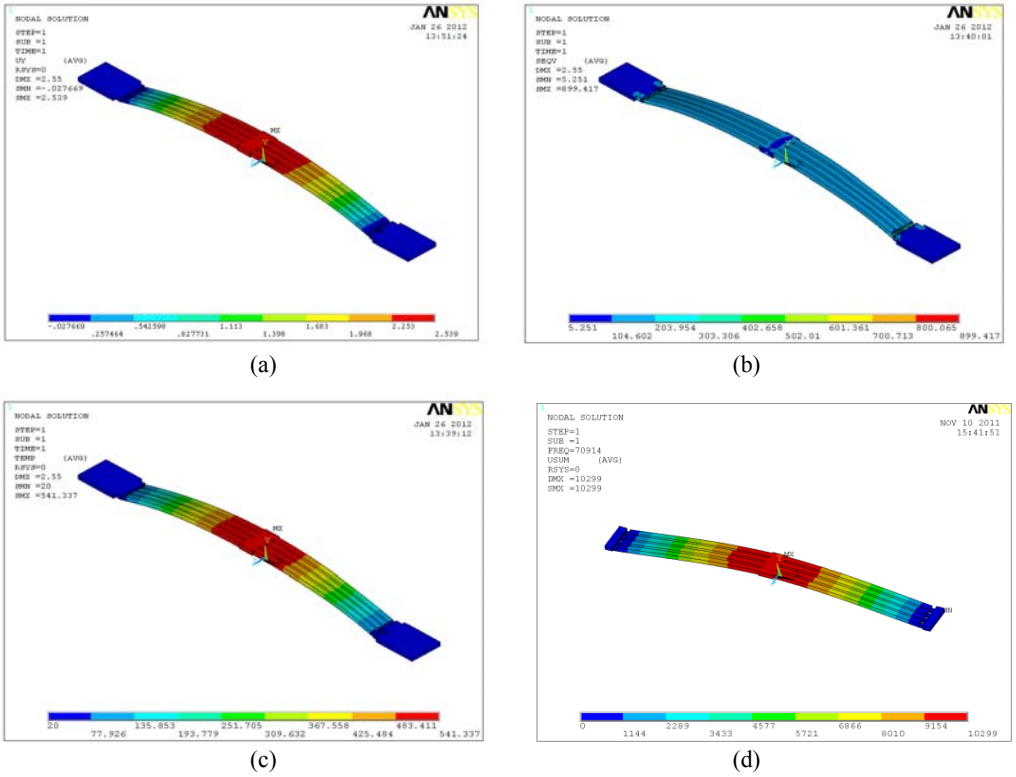


Figure 3. Mechanical properties based on a FEA model; (a) The expected out-of-plane motion; (b) Von Mises stress distribution; (c) temperature distribution; (d) 1st resonant mode

The fabrication process for the proposed out-of-plane actuator is described in Figure 4. The fabrication process is based on the standard SOI MUMPs [15], but additional etching is required for the notches. The starting material is an SOI wafer, with a 30 µm thick device layer, one 2 µm thick buried oxide layer and one 400 µm thick handle layer, as shown in Figure 4(a). The fabrication process consists of four steps; one deposition and three etching steps. The deposition is for electrical connection to the proposed actuators; one layer is a 10 nm of chrome for adhesion and the other layer is 100 nm of gold for wire-bonding and against oxidation. The lithography of this deposition is implemented by a lift-off method and an electron-beam evaporator (Denton Infinity 221). The three etching steps are for the fabrication of the deep Si-trenches and implemented by the Bosch1 process (Deep RIE-Unaxis SHUTTLELINE DSEII1). The first etching is for the notches only (Figure 4(c)) and the second etching is for the main devices including the actuators (Figure 4(d)). Due to the etching depth difference, the notches are separately fabricated from the main devices. Following these steps, the fabrication of the device layer can be completed and the actuator should be released for its operation. For releasing, the handle layer of the SOI wafer was selectively etched away which is the third etching and is shown in Figure 7(e). The final step is to remove the buried oxide layer between the device layer and the handle layer. This layer can be selectively etched away by chemical buffered oxide etchant (B.O.E.) (Figure 7(f)). After the completion of these processes, the proposed actuator can generate the desired motion when a voltage difference is applied to the deposited metal pads.

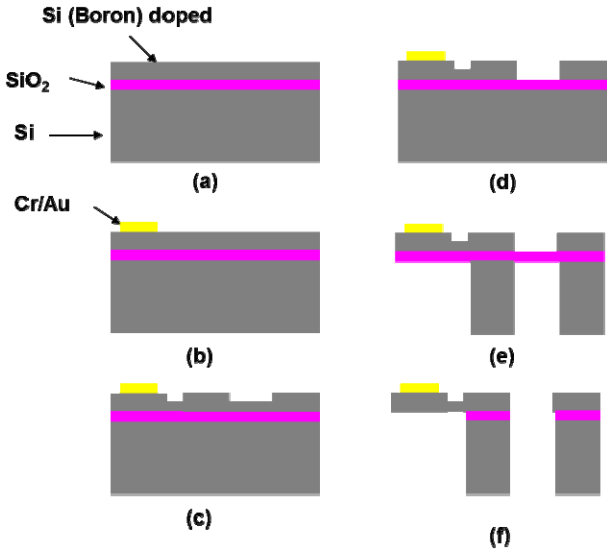


Figure 4. Fabrication sequence: (a) clean SOI wafer as a starting material; (b) metal deposition for electrical connection; (c) etching of the notches on the device layer; (d) etching of the main devices on the device layer; (e) etching of the handle layer to open the bottom side of the devices; (f) removal of the buried oxide layer to release the proposed actuator for its operation

5. TESTING RESULTS

Testing of the proposed actuator, included the measurement of the range of motion and its deformed profiles. An Agilent1 power

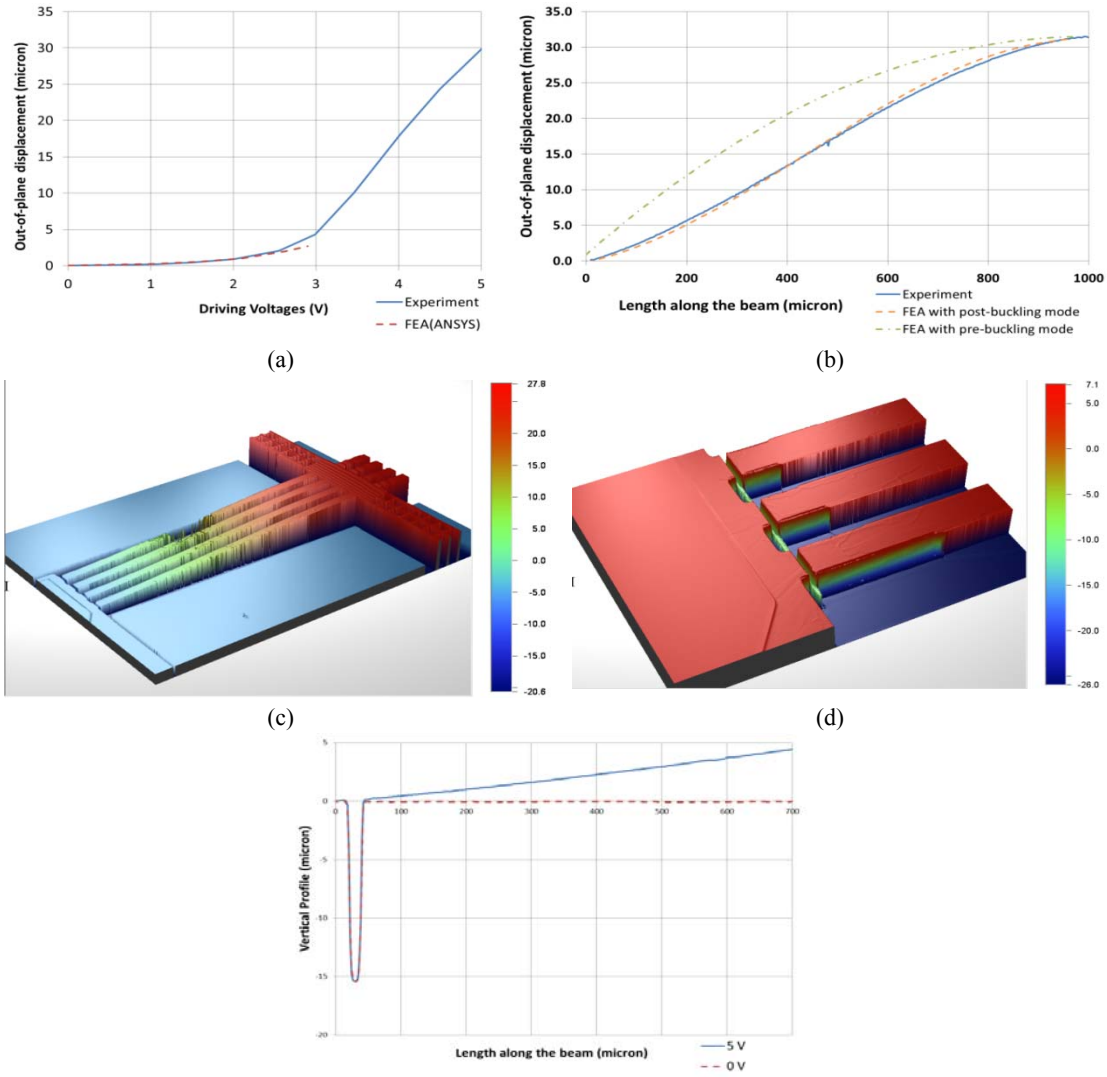
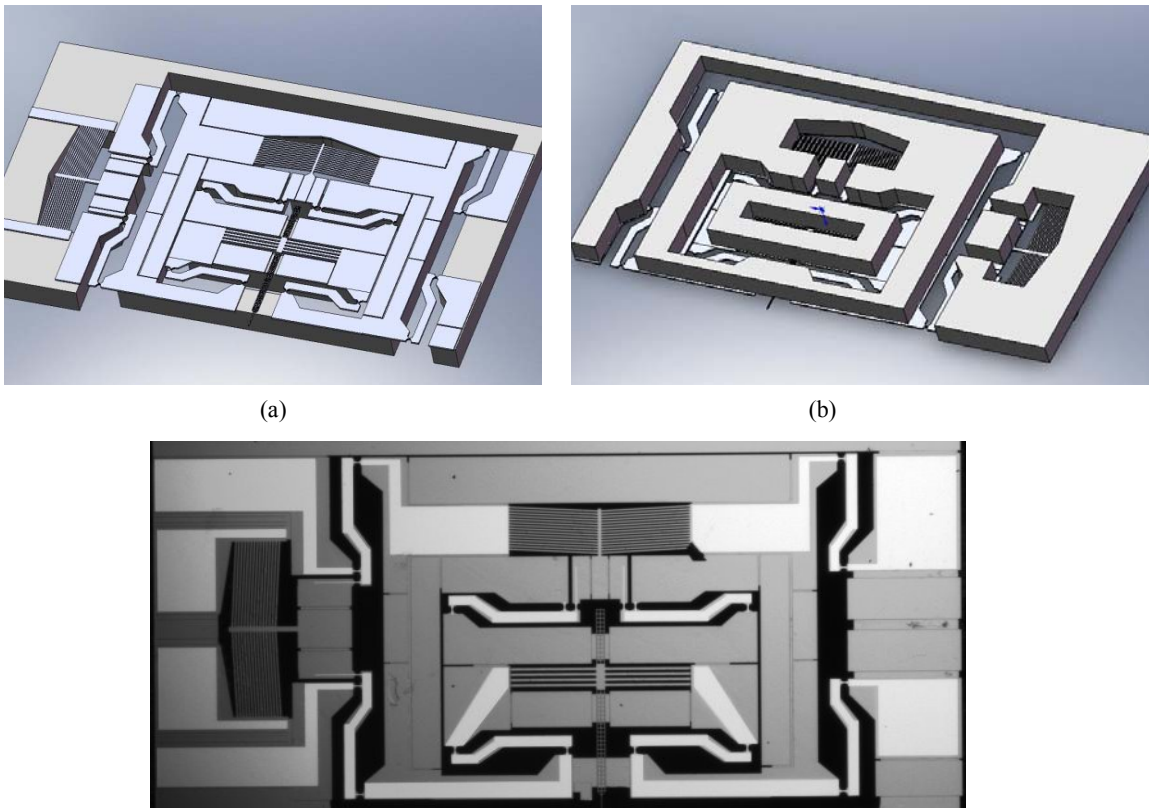


Figure 5. Mechanical behavior of the proposed out-of-plane actuator: (a) relationship between driving voltage and the displacement; (b) measured vertical profile and the expected profile from FEA; (c) 3D scanned image of the half of the actuator (in μm); (d) the 3D scanned imaged on the notch(in μm); (e) the deformation profile near the notch during operation

supply 3322A is utilized to drive the proposed actuator. The mechanical behavior of the proposed actuator is measured with the VEECO1 optical profiler WYKO1 NT1100, which is capable of measuring vertical axis motion with less than 0.1 nm resolution and 1 mm measurement range. The range of motion is measured by controlling the driving current at the power supply from zero to 120 mA for every one voltage difference. After several experiments it was determined that 120 mA is the maximum allowable current that the actuator can endure. The relative position between the two ends of the actuator to the shaft connecting the beams is measured to obtain its displacement, since the maximum displacement is expected to occur at the shaft due to the structure symmetry.

The out-of-plane displacement is measured and plotted in solid line at Figure 5(a). This result is compared with the FEA results plotted with a dotted line. The FEA without the buckling assumption predicts the behavior of the proposed actuator accurately for less than 3 V driving voltage. The relationship between the driving voltage and its corresponding out-of-plane motion in this range is similar with the pattern obtained from the in-plane bent-beam type electrothermal actuator [15]. This is verifying that the same Joule heating and thermal expansion principle apply for their actuations.

For higher than 3 V driving voltage, the actuator starts buckling and its displacement accelerates as shown in Figure 5(a). Due to the buckling, this behavior shows a different pattern of movement from the FEA prediction. Figure 5(b) shows three plots; the experimentally measured deformation profile for 5 driving voltages, the FEA result for pre-buckling mode and the FEA result for post-buckling mode. As buckling occurs near 3 V, the measured profile is similar with the profile for post-buckling mode, not the one for pre-buckling mode. The proposed actuator utilizes both pre-buckling and post-buckling for its operation.



(a) (b)

(c)

Figure 6. A MEMS XYZ stage design with the out-of-plane actuator: (a) angled view of CAD model; (b) a schematic diagram; (c) the fabricated MEMS XYZ stage sample)

Figures 5(c) and 5(d) show the 3-dimensional (3D) images of one-half of the actuator and its notches. Figure 5(e) illustrates the deformation profile near the notch for 5 V driving voltages, when it is in operation and not in operation. The notch gap doesn't change that much during these operations.

6. INTEGRATION WITH THE EXSITING MEMS XY STAGE

To verify the integration capability of the actuator it was embedded into an existing MEMS XY stages whose fabrication process follows the standard SOI-MUMPs, since in this case, significant change in the fabrication process is not needed. The selected MEMS XY stage adopts a serial kinematic mechanism for its operation [15]. By utilizing the serial kinematic mechanism, the proposed single-layer out-of-plane actuator is successfully embedded into the moving platform of the MEMS XY stage. By embedding this out-of-plane actuator, the combined system can generate 3 degrees of freedom (D.O.F.) translational motions along X, Y and Z axes. The detail design of the MEMS XYZ stage is described in Figures 6(a) and 6(b). The in-plane X axis motion is generated by the outer electrothermal actuator and four levers. The moving platform actuated by the outer electrothermal actuator contains the stage aligned along the Y axis. This stage also has one moving platform where the proposed actuator is embedded for the out-of-plane motion or Z axis motion. With this serial kinematic mechanism or dual nested structure, the XYZ stage is implemented without serious coupled motion between them.

Figure 6(c) shows the fabricated sample of the MEMS XYZ stage. The bright white areas indicate the metal deposition for electric connection to the three electrothermal actuators. Greyed area is made of silicon or the device layer of the SOI wafer. Dark black areas are empty area to release the moving platforms and to allow them to move along the designated directions. With a series of experiments, this XYZ stage shows it can generate 40 μm x 40 μm x 30 μm along X, Y and Z axes, respectively.

7. CONCLUSION

This paper describes the design, fabrication and testing of the single layer out-of-plane electrothermal actuator based on MEMS. This actuator is designed to generate out-of-plane motion and the fabricated devices also demonstrate near 32 μm displacements with driving voltage 5 V. Its first resonance mode is expected to occur around 70.4 kHz from FEA. The desired motion is implemented by adopting the notches near its anchors. Joule heating is utilized to generate the thermal expansion of the beam in the actuator, which converts into an eccentric load due to the notches near the beam ends. This eccentric load causes bending of the beams, which transforms into the out-of-plane motion. Due to its simple structure, the fabrication process for the proposed actuator is possible by exploiting the standard SOI-MUMPs. This simple structure also provides good integration capability with other existing MEMS systems which are also based on the SOI-MUMPs. For demonstration, one MEMS XYZ stage is fabricated and tested by embedding the proposed actuator into an existing MEMS XY stage. With this MEMS XYZ stage, 40 μm x 40 μm x 30 μm displacements are measured along each axis without any significant coupled motion error.

8. ACKNOWLEDGMENTS

This research was performed in part in the NIST Center for Nanoscale Science and Technology Nano Fabrication Clean Room. This work was supported by the Measurement Science for Intelligent Manufacturing Robotics and Automation Program of the Intelligent System Division, Engineering Laboratory, National Institute of Standards and Technology, USA.

9. REFERENCES

[1] Du E, Cui H and Zhu Z 2006 Review of nanomanipulators for nanomanufacturing Int. J. Nanomanufacturing 1 (1) 83-104

[2] W. P. Sassen, V.A. Henneken, M. Tichem, P. M. Sarro, "An improved in-plane thermal folded V-beam actuator for optical fibre alignment," J. Micromech. Microeng. 18 (2008) 075033

[3] S. Kwon and L. P. Lee, "Stacked two dimensional micro-lens scanner for micro confocal imaging array," Proc. Fifteenth IEEE International Conference on Micro Electro Mechanical Systems, Las Vegas, NV, USA 20-24 Jan. 2002, pp. 483-486, 2002

[4] Toshiyoshi H, Su G J, LaCosse J and Wu M C 2003 "A surface micromachined optical scanner array using photoresist lenses frabricated by atheraml reflow process" J. Lightwave Technol. 21 1700–8

[5] Kim, K., Liu, X., Zhang, Y., and Sun, Y., 2008, "Nanonewton Force-Controlled Manipulation of Biological Cells Using a Monolithic MEMS Microgripper With Two-Axis Force Feedback," J. Micromech. Microeng., 18 5 ,p. 055013.

[6] Kim C. H., Jeong H M, Jeon J U and Kim Y K 2003 Silicon micro XY-stage with a large area shuttle and no-etching holes for SPM-based data storage J. Microelectromech. Syst. 12 470-8

[7] B. Sahu, C. R. Taylor "Emerging Challenges of Microactuators for Nanoscale Positioning, Assembly, and Manipulation," J. of Manufacturing Science and Engineering, June 2010, vol. 132/030917-1

[8] Bell, D. J., Lu, T. J., Fleck, N. A., and Spearing, S. M., 2005, "MEMS Actuators and Sensors: Observations on Their Performance and Selection for Purpose," J. Micromech. Microeng., 15(7), pp. S153–S164.

[9] J. H. Comtois and V. M. Bright, "Surface micromachined polysilicon thermal actuator arrays and applications," in Proc. Tech. Dig. Solid-State Sens. Actuators Workshop, Hilton Head Island, SC, Jun. 2-6, 1996, pp. 174-176

[10] W. C. Chen, P. I. Yeh, C. F. Hu and W. Fang, "Design and Characterization of Single-Layer Step-Bridge Structure for Out-of-Plane Thermal Actuator," J. of MEMS, Vol. 17, No. 1, Feb. 2008, p70 – p77

[11] M. McCarthy, N. Tiliakos, V. Modi, L. G. Frechette, "Thermal buckling of eccentric microfabricated nickel beams as temperature regulated nonlinear actuators for flow control," Sensors and Actuators A: Physical Volume 134, Issue 1, 28 February 2007, Pages 37-46 International Mechanical Engineering congress and Exposition 2005

[12] J. H. Comtois, V.M. Bright, "Applications for surface-micromachined polysilicon thermal actuators and arrays," Sensor and Actuators A 58, 1997, p19-25

[13] D. M. Burn, V. M. Bright, "Design and performance of a double hot arm polysilicon thermal actuator," proc. Of the SPIE micromachining and microfabrication conference, Austin, TX, Sep. 1997, pp.296-306

[14] F. P. Beer and E. R. Johnston, Jr, "Mechanics of Materials," 2nd edition, McGrawHill, 1992, pp 650 - 653

[15] [Kim YS, Nicholas NG, Gupta SK 2011 A Two Degree of Freedom Nanopositioner with Electrothermal Actuator for Decoupled Motion, ASME/IDETC conference, DC, USA

Intelligent Energy Management: Impact of Demand Response and Plug-in Electric Vehicles in a Smart Grid Environment

Seshadri Srinivasa Raghavan
Power Electronics, Renewable Energies and Energy Harvesting Laboratory
Electrical and Computer Engineering Department
University of Maryland
A.V. Williams Building; College Park, MD 20742
+1(301)-405-3317
sesha@umd.edu

Alireza Khaligh
Power Electronics, Renewable Energies and Energy Harvesting Laboratory
Electrical and Computer Engineering Department
University of Maryland
A.V. Williams Building; College Park, MD 20742
+1(301)-405-8985
khaligh@ece.umd.edu

ABSTRACT

Modernization of the power grid to meet the growing demand requires significant amount of operational, technological, and infrastructural overhaul. The Department of Energy`s "Grid 2030" strategic vision outlines the action plan to alleviate the concerns through the development of a "Smart Grid" (SG). Key emphasis is placed on the role of consumers and their level of interaction with the power grid. Demand response (DR), distributed generation (DG) and distributed energy storage (DES) are some of the key energy management strategies areas within the smart grid paradigm. Majority of the DR programs is currently being supported by commercial and industrial sectors. With the introduction of plug-in hybrid electric vehicles (PHEVs) and advancements in communication, additional avenues for residential consumers to participate in DR programs is expected to open up. This paper first presents the idea behind the SG and the importance of DR. Currently available DR programs and their benefits are quantified across different regions. Specific DR programs suited for PHEV participation are studied. The economic benefits of controlled charging for the PHEV owner is also evaluated.

Categories and Subject Descriptors

J.4 [**Social and Behavioral Sciences**]: Economics, Psychology, and Sociology

General Terms

Economics

Keywords

Controlled charging, demand response, plug in hybrid electric vehicles, smart grid

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage, and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. *PerMIS'12*, March 20-22, 2012, College Park, MD, USA. Copyright © 2012 ACM 978-1-4503-1126-7-3/22/12...$10.00

1. INTRODUCTION

The United States electric power industry in the past decades, has been predominantly functioning and continues to function in a centralized manner. Federal Energy Regulatory Commission (FERC) order no. 888 and 889, paved the way for the first wave of modernization of the electricity industry through restructuring. Restructuring decomposed the electricity industry into individual generation companies (GENCOs), transmission companies (TRANSCOs) and distribution companies (DISTCOs). The independent operation of the three components is guaranteed by the Independent System Operator (ISO).

However restructuring placed an incremental strain on bulk transmission systems during peak or critical period jeopardizing the security and safety of the market operations [1]. According to the Department of Energy (DOE) Office of Transmission and Distribution, electric system in the USA is "*aging, inefficient, congested and incapable of meeting future energy demands without significant capital expenditures and changes"*. Power disturbances and power quality issues alone caused $110-$188 billion of loss to various industries, which is between 13% - 23% of the total asset value of the power industry [2]. Based on the Commission for Environmental Cooperation of North America data, the United States produced approximately 3,858 billion kWh of energy in 2002 emitting a total of 2,253 million metric tons of pollutants (Carbon dioxide, Sulphur dioxide and Nitrous oxides) with coal fired power plants contributing to 50% of the total generation . According to the DOE, if the current grids were 5% more efficient, energy savings would have the effect of removing fuel and harmful pollutant emissions from 53 million cars. The energy industry also forms a key component in trade with oil producing nations, which are historically very volatile and unstable.

Thus, a paradigm shift in the way our electricity supply and delivery system works is urgently needed to address key issues such as: 1) energy and fuel efficiency; 2) reliability; 3) national economy and security; 4) environmental friendliness; and 5) providing consumers

with choices pertaining to buying and selling of electricity. Increased consumer participation, interaction and responsiveness would pave way for more decentralization and hence increased reliability of the electric power system. Distributed generation enhances the opportunity to diversify power generation portfolio by harnessing renewable energies that are found aplenty in the demography under consideration. Increased levels of distributed generation automatically paves for deployment of advanced tools and technologies to monitor and control power flow communication between supply and demand. The culmination of the aforementioned developments constitutes what is collectively known as the Smart Grid (SG), Fig. 1. The overall vision of the SG is to provide decentralized, cleaner, reliable, flexible, intelligent, efficient, affordable, and consumer interactive power [3]-[5].

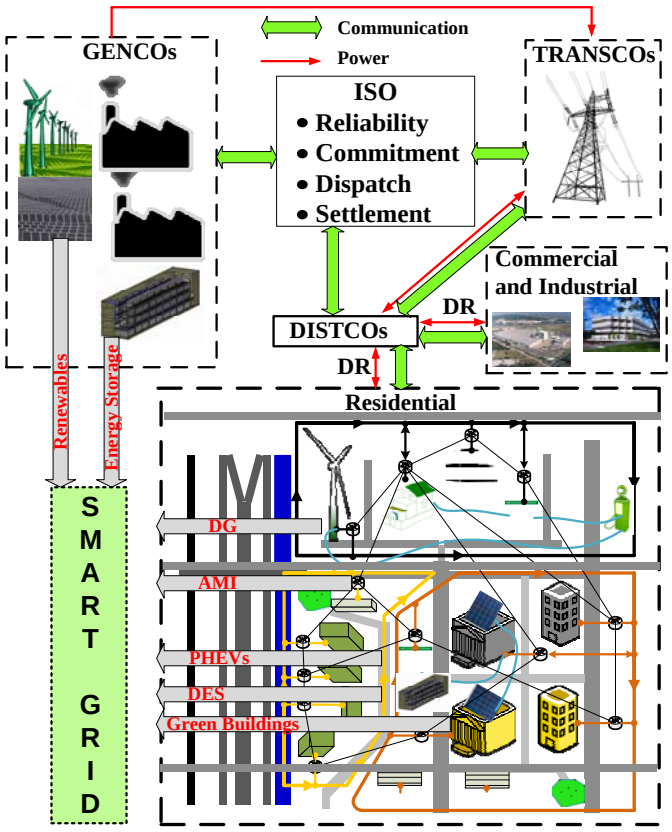


Fig. 1. Smart Grid framework

Some of the key features of the SG are:

- Two way information exchange between the wholesale and retail electricity markets.
- Emphasis on renewable energy integration at utility as well as distribution levels.
- Utilizing large scale energy storage systems that can address the intermittency of nonconventional fuel sources and also provide backup power.
- Enabling integration of PHEVs/EVs.
- Deployment of advanced metering infrastructure (AMI) which facilitates consumer participation and efficient demand management through demand response (DR) programs.
- Virtual power plant development through distributed generation (DG) and distributed energy storage systems (DES).

As one of our nation`s critical infrastructure and key resource, reliability of the power grid is of paramount importance. Considering the fact that almost 90% of outages occur in the distribution system [6] and the average age of a distribution transformer is 2 years beyond its life span [2], the potential for a bottom-up approach to address reliability concerns through the smart grid framework is very promising. In addition, as the reliability of the power system is influenced by the balance between generation and demand in real time, it is economically viable to focus on demand side load management to enhance the reliability of the power system [7]. Thus DR, DG and DES become important ingredients within the smart grid framework, to ensure demand-supply balance is adequately and efficiently maintained.

Majority of DR and DES initiatives have been restricted to large scale consumers in the industrial and commercial sectors. With the advent of PHEVs, a unique opportunity is available for residential consumers to directly participate in market operations. Moreover, the dual operation (vehicle to grid and grid to vehicle) mode of the PHEVs makes them suitable for DR programs, as DG sources and even as dynamic DES resource. However, electrification of the transportation industry imposes additional load on the already fatigued power system. In order to alleviate such concerns, it is imperative to understand the impact of PHEV penetration from a demand side perspective. During the initial stages of PHEV adoption, there is a possibility of geographic clustering of PHEVs in certain localities. DR thus becomes even more crucial in order to mitigate the detrimental effect of incremental demand imposed by the PHEV while simultaneously increasing the overall reliability of the electric power system. Participation in demand side operations (DR, DG and DES) also provides the PHEV owner with an additional revenue stream. This secondary revenue stream potentially increases the value proposition of PHEVs.

Rest of the paper is organized as follows: Section II details the current DR programs and the benefits of DR. Section III specifically focuses on the role of PHEVs in DR programs. Impact of charging rates and times are also studied. Section IV presents a sample case study to illustrate the potential benefit for PHEV owners by participating in DR programs. Section V presents the conclusions and the road ahead for PHEV and DR within the SG framework.

2. DEMAND RESPONSE

FERC defines DR as "*Changes in electric use by demand-side resources from their normal consumption patterns in response to changes in the price of electricity, or to incentive payments designed to induce lower*

electricity use at times of high wholesale market prices or when system reliability is jeopardize" [8]-[9]. In this section, different types of DR and the benefits of DR are presented.

A. Types of DR programs

The different types of DR programs are typically classified on the nature of load and how the load changes are brought about [7], [8]-[10]. If the planned changes in load consumption are a result of consumer action, then that type of DR is called dispatchable DR. Dispatchable DR programs includes direct load control, curtailable/interruptible rates, and other programs offered aimed at improving the reliability of the system. On the other hand, non dispatchable DR refers to programs where the consumer decide when to reduce energy consumption based on dynamic price changes which primarily depends on the system load. DR programs can also be alternatively classified on the manner in which load consumption pattern is altered. Price based DR programs refer to changes in electricity usage patterns in response to price of electricity. Incentive based DR programs are offered by ISOs and load serving entities that provide consumers with load reduction incentives in addition to the retail electricity pricing. Certain programs such as critical peak pricing and peak time rebates can be classified as dispatchable or non-dispatchable DR depending on the ISO and the contractual agreements between the energy service provider and the consumer.

TABLE I
CLASSIFICATION OF DR PROGRAMS [8]-[10]

DR Program	Nature of Load	Types of Options
Direct load control	Dispatchable	Incentive based
Interruptible control	Dispatchable	Incentive based
Critical peak pricing with control	Non-dispatchable	Price based
Load as capacity resource	Dispatchable	Price based
Spinning reserves	Non-Dispatchable	Incentive based
Non-spinning reserves	Non-Dispatchable	Incentive based
Emergency DR	Dispatchable	Incentive based
Regulation services	Non-Dispatchable	Incentive based
Demand bidding and buyback	Non-Dispatchable	Incentive based
Time of use pricing	Dispatchable	Price based
Critical peak pricing	Dispatchable	Price based
Real -time pricing	Dispatchable	Price based
Peak time rebate	Dispatchable	Price based
System peak response transmission tariff	Dispatchable	Price based

Direct load control (DLC) programs typically involves shutting down end user`s equipment remotely. Usually the equipments are heaters, washer/dryers or air conditioners used by residential customers. Interruptible control programs integrate appropriate load curtailment options within the retail tariff. Discounted rate or billing credits are provided for large commercial and industrial consumers for agreeing to reduce load during system contingencies or peak periods. In demand biding or buyback programs, consumers (large consumers, 1 MW or over) offer bids to curtail based on wholesale retail prices. Regulation, spinning and non spinning reserves are part of the ancillary services market programs. Individual consumers bid their respective load curtailments as reserves. Depending on the nature of market clearing mechanism, the consumers are paid the market price. Ancillary services are paid on the basis of capacity and the energy provided. When there is an unforeseen shortage of reserves, under the emergency DR program, suitable payments are made to the consumer for load reductions. Depending on the average cost of generating and delivering power, different time of use (TOU) pricing blocks are defined depending on the time. The prices are high during peak periods when typically costly peaking power plants are committed to meet peak demands. Under the real time pricing (RTP), the price of electricity varies on hourly basis depending on the wholesale price of electricity. RTP prices are notified to the customers day ahead or on a hour ahead basis. As variation of TOU and RTP, critical peak pricing (CPP) rates are applied based on some specific reliability or price based trigger. The detailed classification of the available DR programs is summarized in Table. 1.

B. Benefits of DR

The potential impacts of DR are influenced by the physical structure of the electricity market under consideration and also the extent to which DR programs are implemented. There are potential benefits for all the stakeholders in the electricity market. For the generating companies, load management during critical and peak period ensures de-commitment of expensive units, improved system reliability, and avoiding potential capacity expansions. Efficient demand side management also increases the utilization of existing infrastructures (distribution and transmission), improves overall system reliability and also reduces the ability of the market participants to exercise market power. Consumers can expect more choices to buy electricity from the retail market and reduced electricity bills. By load shifting or curtailing during peak and emergency periods, the volatility of the wholesale market prices can also be reduced. Overall, the cascaded benefits of DR programs benefit the entire generation, supply and consumption chain. FERC in collaboration with The Brattle Group [21] developed the DRIVE [11] (Demand Response Impact and Value Estimation) model as part of its National Assessment and Action Plan on DR [8], [9]. The DRIVE model provides quantitative insights on how different DR programs affect generation (scheduling, planning and expansion, operating costs and emissions) and also load (load duration curve and peak demand). Five DR programs are considered in the DRIVE model: 1) dynamic pricing without enabling technology (AMI); 2) dynamic pricing with enabling technology; 3) direct load control; 4) interruptible tariffs and 5) other DR programs. Consumers are divided into 4 different categories: 1) residential; 2) small commercial and

industrial; 3) medium commercial and industrial and 4) large commercial and industrial. The detailed specifications about the different DR programs, generation commitment and load forecast under each consumer category used in the DRIVE model for each of the 13 FERC defined regions can be found from [11]. Fig. 2a-2c, summarize the results using the DRIVE model.

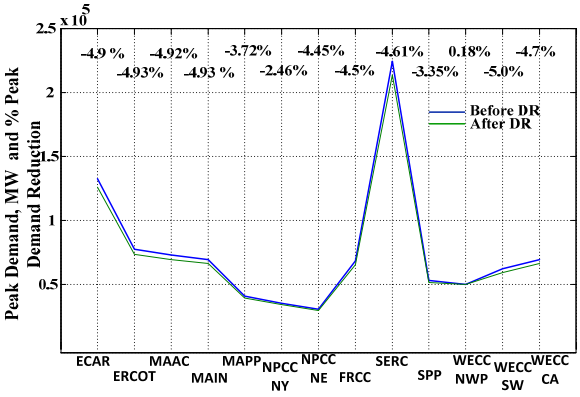


Fig. 2a. Impact of DR on Peak Demand, 2030

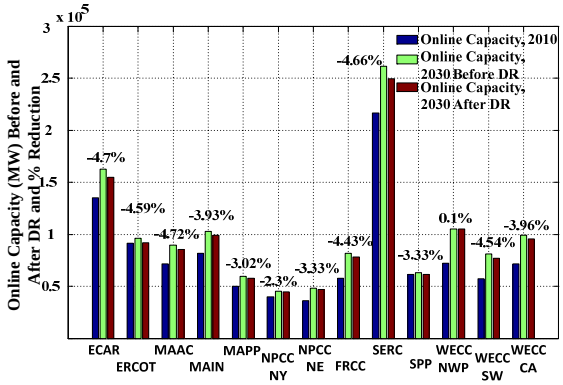


Fig. 2b. Impact of DR on online capacity, 2030

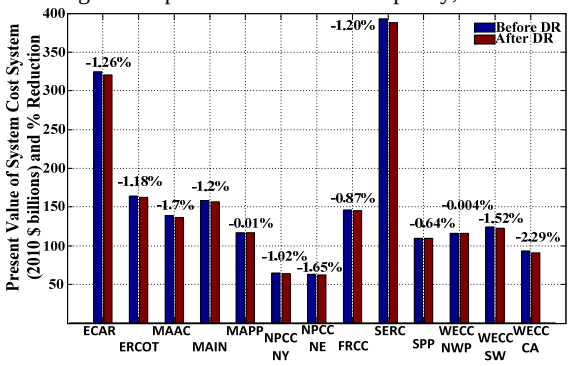


Fig. 2c. Impact of DR on present value of system cost,2030

It can be observed from Fig. 2a that across all regions, the peak demand in 2030 is reduced by an average of 4%. Also, it can be observed that the impact of DR on peak demand varies from region to region based on the regional generational mix, number of consumers in each category, forecasted load and the extent of DR program implementation in that particular region. As a direct result of the different DR programs, the generation resources are better utilized when compared with no DR scenario as shown in Fig. 2b. The average percentage reduction in the online capacity after DR is 3.65%. The average avoided capacity installation is found be to 3,877 MW in 2030. The cumulative effect of better asset utilization directly affects the present value of system cost (cost of energy, capacity and emissions). Fig. 2c shows the impact of DR on the present value of system cost in the year 2030. The present value of system cost is reduced by an average of $1.75 billion (2010) at 8% discount rate

3. PHEVS AND DR

In order to evaluate the role of DR with respect to large scale adoption of PHEVs, we must understand the timescales during which DR programs are activated in the electricity supply chain. Fig. 3 [10] shows the electricity system planning and scheduling based on timelines. The organization of the electricity market structure determines the decisions made during each timescale. Long term planning and expansions decisions are highly capital intensive. Generation and transmission system investments are typically huge and it requires several years to select, build, operate and monitor. Short term operational decisions involve scheduling the available resources to meet forecasted demand. As the overall purpose of the electricity system is to maintain supply-demand balance, system balancing decisions that involve unit commitment and economic dispatch (ED) are done on day ahead or hour ahead basis. Regulation services are needed typically within minutes and reserves (spinning and non spinning) are called around 20 times a year. As it can be seen from Fig. 3, both price based and incentive based DR programs can be offered at all timescales. Price based DR programs like RTP and CPP can typically be integrated into day ahead or hour ahead markets. If the LSE has an accurate sense of understanding of the demand side requirements, TOU rates can be designed days or months ahead. Incentive based DR predominantly involves load curtailment commitment ahead of time and they can also be offered to consumers at all timescales.

Considering the physical characteristics of the electrochemical battery (quick response, high $/kWh, and low $/kW) , the suitable DR programs for PHEVs are highlighted in Fig. 3. Consumers having access to smart meters and smart charging infrastructures can take advantage of low electricity prices during off peak hours. Also the complementing nature of travel behavior and system demand makes them suitable to participate by providing ancillary services through V2G transactions during peak hours. Some of the fundamental drawbacks of PHEVs when it comes to reliability programs is the uncertainty regarding the spatial mobility of the PHEVs and its inability to store large amount of energy which makes them unsuitable for emergency DR and operational capacity planning. Currently not many utility companies offer critical peak pricing and peak time rebates to individual residential consumers simply due to the amount of energy a PHEV battery can hold. However, with the increase in PHEV penetration, aggregated PHEVs can

potentially offer such services. Currently TOU pricing seems to be the DR program that can easily be provided for the PHEVs to take advantage of low cost electricity during off peak times.

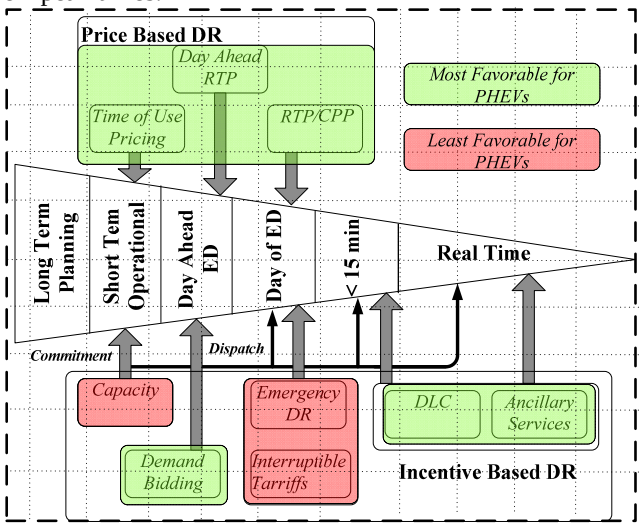


Fig. 3. Commitment and dispatch timescales [17] and the role of PHEVs for DR

In the next subsections, the technicalities of PHEVs that influence its DR potential and the benefits are investigated.

A. PHEV driving requirements

The primary purpose of the PHEV is to satisfy the daily driving requirements of the owner. Driving requirements vary from consumer to consumer and it is very difficult to quantify the daily driving distances and purposes accurately. Suitable field tests and transportation survey data are required to understand PHEV driving requirements. Apart from the driving distance, vehicle class (sedan, small car, SUV etc.), driving cycle (city, highway, congested etc.) and the driver style (aggressive, passive etc.) are the other major factors that determine the battery capacity requirements [11]. Also for any given battery design, the entire capacity is not available for driving in order to extend the battery life time. Typically the battery state of charge window is between 35%-95% of the rated capacity. It is estimated that the specific energy consumption of PHEVs can vary between 0.15 kWh/mile - 0.34 kWh/mile [12] with a median of 0.24 kWh/mile for sedans. Using these values, the battery capacity requirements (kWh) for different driving distances for a mid size sedan when driven all electric, is provided in Table. 2. Early adopters of PHEV would primarily have unidirectional grid to vehicle charging and therefore the driving requirements directly affect the charging requirements of the PHEV.

B. PHEV charging requirements

The incremental power demand imposed by the PHEV depends on the time of charging and the rate of charging. From an electricity supplier point of view, suitable time to charge the PHEVs would be after midnight when the local demand is low and cheap base load coal power plants are running. From the PHEV owner point of view, he/she would prefer to charge the PHEV as soon as they arrive back home. In the absence of any economical incentive, consumers would time the PHEV charge depending on the accessibility to a charging station and the daily driving requirements. The rate at which the PHEV battery is charged determines the power demand from the PHEV. Some of the available charging options are summarized in Table III, assuming a lossless system [14]. Level I Charging uses standard 120 VAC circuit and Level II charging uses 240 VAC circuit. The higher rating of the Level II charging enables the PHEV battery to be charged quicker in comparison to Level I charger. DC charging or Fast Charging is typically meant for large commercial locations where the charging time is about 15-20 minutes to provide 50% recharge.

TABLE II
BATTERY CAPACITY REQUIREMENTS FOR DIFFERENT DRIVING DISTANCES AND SPECIFIC ENERGY CONSUMPTION

Driving Distance	Specific Energy Consumption = 0.15 kWh/mile	Specific Energy Consumption = 0.34 kWh/mile	Specific Energy Consumption = 0.24 kWh/mile
10	1.5	3.4	2.4
15	2.25	5.1	3.6
20	3.0	6.8	4.8
25	3.75	8.5	6.0
30	4.5	10.2	7.2
35	5.25	11.9	8.4
40	6.0	13.6	9.6

TABLE III
SUMMARY OF DIFFERENT PHEV CHARGING LEVELS

Level	Application	Voltage (V)	Amperes(A)	Maximum Power (kW)
I	Residential	120	15	1.8
I	Residential	120	20	2.4
II	Residential	240/208	30	7.2/6.24
II	Commercial	240/208	30	7.2/6.24
	DC Fast Charging	n/a	n/a	Up to 50

Fig. 4a shows the system demand curve from San Diego Gas and Electric (SDGE) [15] superimposed with PHEV demand from 100 PHEVs each requiring 10.4 kWh. The impact of time of charging and the rate of charging is depicted in Fig. 4a- Fig. 4b, assuming 90% energy conversion efficiency (includes grid side losses and charging losses within the PHEV). PHEV charging starts at 19:00 p.m. and it is assumed that after 07:00 a.m. the PHEVs depart for work. As it can be observed from Fig. 4a, if the consumer comes back home and charges it immediately at 19:00 p.m., the peak demand for Level 2 charging with PHEV is 30% more than the actual system demand without PHEV and 22% more than the peak demand imposed by Level 1 charging. Such large spikes can potentially be detrimental to the distribution side transformer.

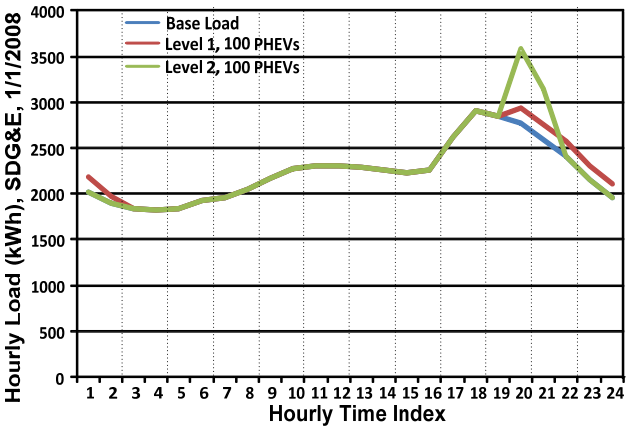


Fig. 4a. Impact of uncontrolled charging on demand curve

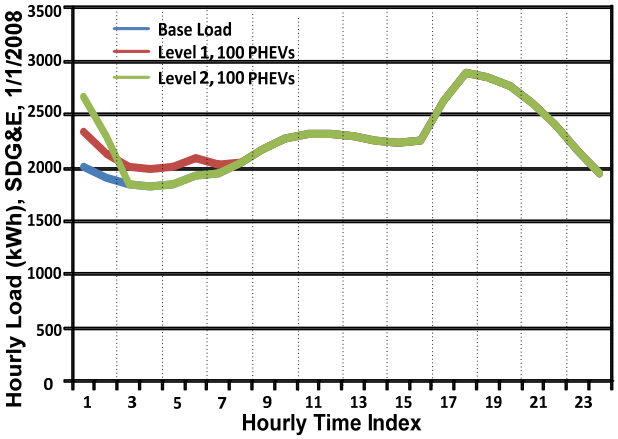


Fig. 4b. Impact of controlled charging on system demand

Fig. 4b shows the same demand curve with the PHEV charging shifted to off peak hours starting from mid night. In this case for both Level 1 and Level 2 charging, the incremental demand is met during the hours when the actual system demand is very low. The peak demand with the PHEVs is less than the actual system demand peak without PHEVs. In addition, it can be inferred that faster the charging rate, the more is the demand imposed by the PHEV on the grid. Through off peak charging, the penetration of PHEVs can also be potentially increased without causing an increase in peak demand while simultaneously improving asset utilization.

4. Benefits for the PHEV Owner

In this section, a case study based on SDGE data, is presented to highlight the benefits of DR pertaining to PHEV charging. SDGE offers tiered as well as TOU rates to all consumers. The classification of the electricity tariffs into tiered and non-tiered rates vary from one energy service provider to another. Super off-peak rates are currently being offered by utilities specifically targeting EV/PHEV consumers by offering them low rates between midnight and 6:00 a.m.. TOU rates can also be viewed as two tier rates. The total electricity cost consists of 3 components: Utility Distribution Company (UDC) costs, Electric Energy Commodity Costs (EECC) and Department of Water Resources Bond Charge. Standard residential home energy consumption is assumed to be 500 kWh/month. Daily driving distance of the PHEV was assumed to be 37 miles and at 0.28 kWh/ mile of specific energy consumption, the usable battery capacity of the PHEV is 10.4 kWh. Incremental demand imposed by the PHEV is therefore 312 kWh/month. The residential load is assumed to be under the tiered structure and different TOU rates are applied to the PHEV charging demand to understand the economic benefits of TOU pricing.

Fig. 5 shows the monthly charging cost of the PHEV under different pricing options. As expected, charging during off peak and super off peak result in reduction of 69 % and 73%, respectively in comparison to the tiered rate. Fig. 7 shows the impact of the time of charging of PHEV on the annual cost reduction in fuel, when compared to HEV with 50 mpg and a conventional internal combustion engine (ICE) with 25 mpg.

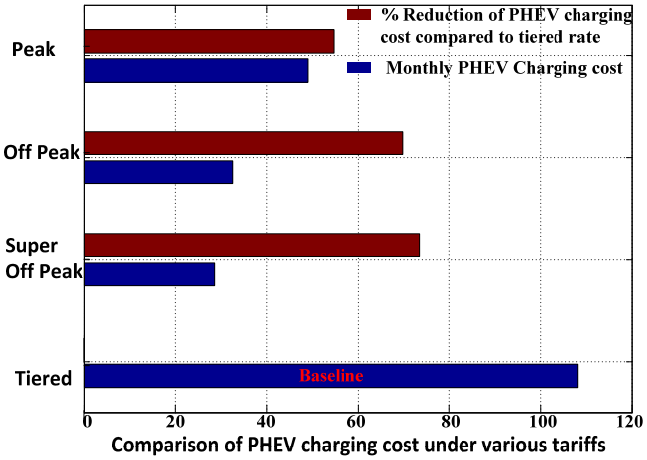


Fig. 5. Comparison of monthly PHEV charging costs for different tiered and different TOU rates

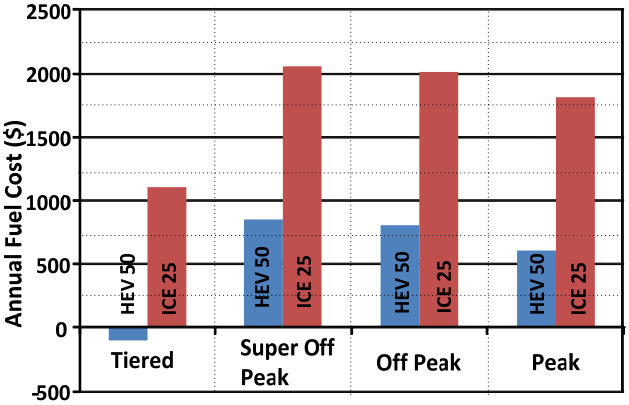


Fig. 6. Annual fuel cost savings and percentage reduction of PHEV under different TOU rates compared with fuel costs of HEV and ICE

Assuming $4.5 per gallon of gasoline and the rates obtained from SDGE, Fig. 6 shows the annual fuel cost savings and the percentage reduction in fuel cost. The time of charging directly affects the annual fuel cost savings which in turn would affect the number of years it takes for

the PHEV owner to recover the incremental premium paid over an HEV or ICE. We can infer that if PHEVs owners chose tiered rate over TOU rates, from annual fuel cost perspective the PHEV is less economically favorable to an HEV 50. It must be understood that as such the battery capacity of 10.4 kWh is minuscule in terms of grid scale operations. However, in the long term future, aggregated PHEVs can potentially provide valuable services in the form of V2G. In a hypothetical scenario, assuming the entire battery is not depleted for driving alone, the PHEV owner can also make additional revenue by offering ancillary services with the help of advanced power electronic interfaces [17].

5. CONCLUSIONS

The transition from centralized functioning of the grid to a smart grid opens up new avenues for residential electricity consumers to participate in grid operation. However, the increasing emphasis on transportation electrification can potentially be detrimental to the functioning of the grid, if suitable demand side management programs are not employed. The role of DR thus becomes more crucial to ensure the adequacy and the security of the power supply is always maintained. This paper presented the role of DR, different types of DR programs and the benefits of DR. Specific DR programs suited for PHEVs was also studied considering the technical issues pertaining to PHEV charging. The system level impact of time and rate of PHEV charging was presented to highlight the need for DR to facilitate controlled charging of the PHEVs. Through controlled charging mechanism, the detrimental impact of PHEV penetration on the peak demand can be avoided. Consumers can potentially save money on charging costs, by shifting the PHEV charging to off peak times.

6. ACKNOWLEDGEMENT

This work was supported by the U.S. Department of Energy (DE-EE0002979: A World-Class University-Industry Consortium for Wind Energy Research, Education, and Workforce Development) which is gratefully acknowledged.

7. REFERENCES

[1] M. Mallette and G. Venkataraman, " The role of plug-in hybrid electric vehicles in demand response and beyond," in *Proc. IEEE Transmission and Distribution Conference and Exposition*, New Orleans, LA, Apr. 2010.

[2] U.S. Department of Energy "Grid 2030" - A National Vision for Electricity`s Second 100 Years. [Online]. Available: http://www.oe.energy.gov/DocumentsandMedia/Electric_Vision_Document.pdf.

[3] U.S. Department of Energy, Title XIII- Smart Grid, Sec. 1301. Statement of Policy on Modernization of Electricity Grid.

[4] F. Rahimi and A. Ipakchi ," Demand Response as Market Resource Under the Smart grid Paradigm, " *IEEE Transactions on Smart Grid*, vol. 1, no. 1, pp. 82-88, June 2010.

[5] C. Wei, " A Conceptual Framework for Smart Grid, " in *Proc. IEEE Asia-Pacific Power and Energy Engineering Conference*, Chengdu, China, Mar. 2010.

[6] Md. Rahat Hossian, A. Maung Than Oo and A. B. M. Shawkat Ali, " Evolution of Smart Grid and Some Pertinent Issues, " in *Proc. Australasian Universities Power Engineering Conference*, Christchurch , New Zealand , Dec. 2010.

[7] M. H. Albadi and E. F. El- Saadany, " A summary of demand response in electricity markets, " *Electric Power System Research Journal*, vol. 78, no. 11, pp. 1989-1996, Nov. 2008.

[8] National Action Plan on Demand Response," *Technical Report, Federal Energy Regulatory Commission*, June 2010.

[9] "National Assessment of Demand Response Potential," *Technical Report, Federal Energy Regulatory Commission*, June 2009.

[10] U. S. Department of Energy, "Benefits of Demand Response in Electricity Markets and Recommendations for Achieving Them,", *Technical Report, Lawrence Berkeley National Laboratory*, Feb. 2006.

[11] Demand Response Value and Impact Estimation (DRIVE) Model. [Online]. Available: www.ferc.gov.

[12] S. S. Raghavan and A. Khaligh, "Deterministic scheduling of a fleet of plug-in hybrid vehicles for distributed generation," *IEEE Power and Energy Magazine*, July 2011, in press

[13] B. Adornato, R. Patil, Z. Filipi, Z. Baraket and T. Gordon, "Characterizing Naturalistic Driving Patterns for Plug-in Hybrid Electric Vehicle Analysis," in *Proc. IEEE 5th Vehicle Power and Propulsion Conference*, Dearborn, MI, Sep. 2009.

[14] S. W. Hadley, "Evaluating the Impact of Plug-in Hybrid Electric Vehicles on Regional Electricity Supplies," in *Proc. iREP Symposium on Bulk Power System Dynamics and Control*, Charleston, SC, Aug. 2007.

[15] Energy Policy Initiative Center, University of San Diego School of Law. [Online].
Available: http://www.sandiego.edu/epic/data_center/electricity.php.

[16] San Diego Gas and Electric, Electric Tariff Book-Residential Rates. [Online].Available:
http://www.sdge.com/regulatory/elec_residential.shtml

[17] S. S. Raghavan, O. C. Onar, and A. Khaligh, "Power electronic interfaces for future plug-in transportation systems," IEEE Power Electronics Society Newsletter, vol. 24, no. 3, pp. 23-26, July 2010.

Characterization of Forward Rectilinear-Gait Performance for a Snake-Inspired Robot

James K. Hopkins
Department of Mechanical Engineering
and Institute for Systems Research
University of Maryland
College Park, Maryland 20742
+1 (301) 342 0873
jhopkin1@umd.edu

Satyandra K. Gupta
Department of Mechanical Engineering
and Institute for Systems Research
University of Maryland
College Park, Maryland 20742
+1 (301) 405 5306
skgupta@umd.edu

ABSTRACT
Snake-inspired locomotion is much more maneuverable compared to conventional locomotion concepts and it enables a robot to navigate through rough terrain. A rectilinear gait is quite flexible and has the following benefits: functionality on a wide variety of terrains, enables a highly stable robot platform, and provides pure undulatory motion without passive wheels. However, historically speed has been a limitation for the locomotion type. In this paper, Fused Deposition Modeling (FDM) is utilized to reduced the weight and thereby increase the speed potential of a snake-inspired robot design based on a rectilinear gait. FDM also provides feasibility for development of complex and capable mechanism designs for executing rectilinear motion. The new design is analyzed, fabrication and evaluated based on various anchoring material velocity experiments.

Categories and Subject Descriptors
B.8.2 [**Performance and Reliability**]: Performance Analysis and Design Aids

General Terms
Performance, Design and Experimentation

Keywords
FDM, Parallel Mechanism, Kinematics and Dynamics

1. INTRODUCTION
Among the various snake-inspired robot gaits, rectilinear-gait based motion has demonstrated very favorable results through many useful features. Motion based on a rectilinear gait is highly stable due to the fact that the majority of the robot mass is always in contact with the terrain and only a small portion of the robot is lifted from the terrain at any given time. It is also this feature that allows rectilinear motion to function on a wide variety of terrains, as the shape of the robot can easily contour to terrain changes. In general, robot platforms which demonstrate serpentine motion have only been successful through the inclusion of passive wheels on each segment (or other methods to impart anisotropic friction) to simulate the snake pushing laterally against small discontinuities in the terrain. These passive wheels may result in a system which is only effective over smooth terrain. Rectilinear motion provides pure undulatory motion without passive wheels.

Although rectilinear gaits are very useful, the current platforms that demonstrate them are relatively slow. Average human walking speed is approximately 2-3 mph [1]. For a robot utilizing rectilinear-gait based locomotion to be used in real world applications such as exploration, rescue operations, and general military reconnaissance, the robot must at least achieve human walking speed in order to keep pace with the human field team that support it. In order to achieve the desired forward velocity, a new design is needed for snake-inspired robots.

Most robots utilizing rectilinear gaits advance by lifting and displacing robot segments forward using friction between the robot and terrain. Examples of these robots include: Kevin Dowling's Snake robot [2], the PolyBot [3], CMU's Modular Snake robots [4], NEC Quake Snake [5], GMD-Snake [6], CONRO [7] and M-TRAN [8]. Note that PolyBot, CONRO and M-TRAN are actually reconfigurable robots that are capable of emulating snake-inspired locomotion. In these gait types, since a significant amount of the robot displacement per cycle is normal to the surface being traversed, forward displacement per cycle is considerably limited. Therefore, to achieve human walking speeds, the segments of the robot would have to be drastically lengthened and larger joint motors would be needed to actuate the longer segments, making the robot unsuitable for use in small, tight spaces.

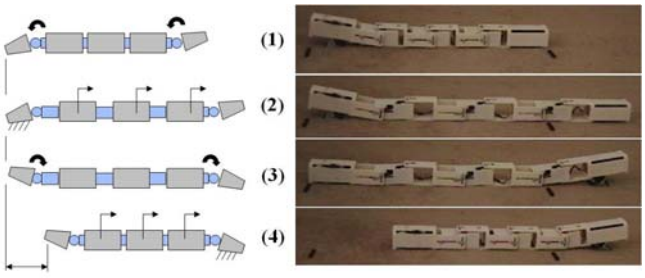


Figure 1. Forward Motion Gait Sequence

In order to address these limitations, a design for a new drive mechanism capable of achieving high speed motion has been developed by the authors. The new design also utilizes a new forward rectilinear gait, illustrated in Figure 1. The motion in Figure 1 is described as the snake's body segments expanding and contracting linearly with little to no vertical displacement, which

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage, and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
PerMIS'12, March 20-22, 2012, College Park, MD, USA.
Copyright © 2012 ACM 978-1-4503-1126-7-3/22/12...$10.00

allows most of the joint motion to be used in forward displacement. Note in Figure 1, the friction anchors at the ends of the robot are purposely actuated at their maximum height to better illustrate their motion during the gait sequence. In the sequence, (1) the rear friction anchor "plants" into the terrain to resist the reaction forces of the linear joints of the internal segments and ensure that the position of the rear end of the robot remains unchanged during segment expansion of step two. Next, (2) each internal segment of the robot expands to its maximum length – allowing the front of the robot to position itself a distance from the rear equal to the robot's original length plus the sum of the segments' displacement. Then, (3) the rear anchor is lifted from the terrain, permitting free sliding of the aft end of the robot. Finally, (4) the robot segments contract to their nominal length, causing the entire robot to advance and the gait cycle is complete.

The new design enables high speed operation. Fused Deposition Modeling (FDM) is used as the fabrication method for the prototype. The use of FDM reduces the number of features and parts needed for assembly and thereby reducing the weight. The FDM also enables manufacturing of 3D features and realization of parallel mechanism concepts. Four mechanism concepts were considered. To enable the forward gait concept, a method of anchoring segments of the robot to the terrain to provide positive forward displacement during extension is also developed and presented. In summary, this paper introduces four new parallel mechanism concepts, evaluates these designs and selects a new parallel mechanism to improve the function of the robot drive mechanism. Also, we will perform a complete kinematics and dynamics analysis for the new design. Finally we will compare the velocity performance of various materials for the new friction anchor mechanism and fabricate a complete robot using FDM.

2. RELATED WORK

Although the majority of snake-inspired robot designs are some form of wheeled robot or utilize a rectilinear motion based of lifting of its segments, robot designs that move using linear expansion and contraction of the robot's body do exist. The Slim Slime robot was an ACM composed of serially-connected modules driven by pneumatic actuators, which allowed it to perform in a 3D workspace [9]. Slim Slime robot was composed of six expandable modules. The robot maintained a high Degree of Freedom (DOF), while being pneumatically-driven without the use many air supply lines. Three flexible pneumatic actuators, known as bellows and a main distribution tube made up the actuation system of each module of the robot. Compressed air was provided into each bellows from the main tube through an inlet valve built in bellows. Inlet and outlet valves built in each bellows made the bellows stretch, shrink and lock its length; therefore the module could stretch and bend in any direction actively. Slim Slime Robot was capable of a maximum forward velocity of approximately 60 mm/s.

Another example of a robot which utilizes linear actuation-based rectilinear motion is the inchworm robot introduced by Chen et al. [10]. The robot consisted of interconnected actuating modules that can either deform in the direction of travel (extensors) or grip against walls in the robot's environment (grippers). The robot was designed for use in traveling and conducting tasks in narrow and highly constrained environments, such as pipes and conduits in industrial plants. Each module had a cart-like geometry moving along a horizontal track.

The design of the inchworm robot by Chen et al. led to the development a planar inchworm robot, called a Planar Walker, based on the basic inchworm motion [11]. The planar inchworm could mimic snake or inchworm-like creeping motions. In addition, the unique mechanical arrangement of the actuators allowed for quick change in travel direction and permitted rotational movement. The unit featured a simple closed-loop planar 8-bar mechanism formed by four linear cylinders and four revolute joints. When the four cylinders were actuated independently, the shape of the mechanism changed to a square, a rectangle, or an irregular quadrilateral. Four pneumatic suction/gripper modules were mounted below each of the revolute joint to hold the robot to the working surface. The robot was designed to be able to traverse forward, backward, and sideways a fixed distance or turn at a fixed angle. The robot had a maximum transverse stride length of 32 mm/cycle and a maximum turning angle of 25 deg/cycle. The robot achieved a maximum transverse speed of 1.07 mm/s (30 s per cycle) and a maximum turning gait speed of 0.42 deg/s (60 s per cycle).

Chen et al. introduced a rectilinear-gait based model, based on a finite state model, for a multi-segment inchworm-like robot capable of 1-dimensional motion in a confined channel [10]. The robot advances or retreats through the use of linear joint actuators called extensors and grippers. In the finite state model, joints are modeled only with binary values states "0" and "1". Gaits are generated for the subject robot by developing exhaustive search path finding algorithms for use on directed graphical representations of the body segment states. This gait generation approach and locomotion mechanism was further expanded to apply to a planar inchworm robot resulting in a new forward gait and separate turning gait [12].

The Telecubes were an example of self-reconfigurable robots which were able to assemble in configurations that could mimic snake-inspired locomotion [13]. Each Telecube robot module had two basic mechanical functions: contracting/expanding and connecting/disconnecting from the faces of neighboring modules. Each robot possessed six DOF through six prismatic joint which could individually expand or contract each face of the cube. Each face, known as a connection plate, had a mechanism and means to reversibly clamp onto the neighboring robot's connection plate and transmit power and data to the neighboring robot.

3. PARALLEL MECHANISM CONCEPT SELECTION AND DETAILED DESIGN

The conceptual design of the new parallel mechanism is based on an effort to couple of the output from two parallel, independently powered scotch yoke-like mechanisms. The basic idea is that when the two scotch yokes move in the same direction, at the same rate, the common link (parallel mechanism output link), which is connected to the output of the two scotch yokes, will move in a linear fashion. If the two scotch yokes move in different directions or at different rates, the common link will pivot appropriately (if properly constrained). In order to transition this idea into a working concept, several concept configurations for coupling the scotch yoke mechanisms were considered. Most of the configurations were quickly eliminated due to the fact that the output link would be over constrained or under constrained. However, four configuration were defined which may meet the design intentions needed for the joints for the robot. The four configurations are illustrated in Figures 2 through 5.

Concept A (Figure 2) – The mechanism is composed of two limbs (blue links in Figure 2), each consisting of a powered revolute joint mounted to the base link, as well as a passive revolute and prismatic joint connecting to the mechanism's output link (green link in Figure 2). Also, there is a third passive limb which consists of a revolute and a prismatic joint (center black link of the kinematic representation). The passive limb allows prismatic motion along the x-axis and pivoting motions (represented by the angle β) for the output link while resisting motion along the y-axis. Additional constraints are provided by the fact that the points A, B and C remain collinear throughout the range of motion of the mechanism. These constraints prevent the output link from pivoting while the revolute joints are held stationary.

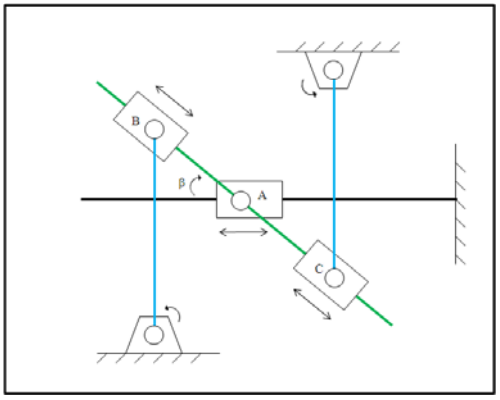


Figure 2. Parallel Mechanism Concept A

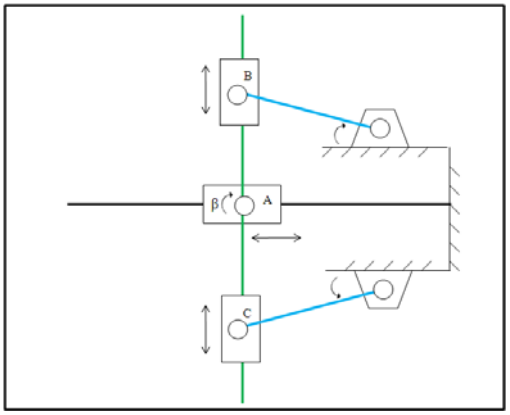


Figure 3. Parallel Mechanism Concept B

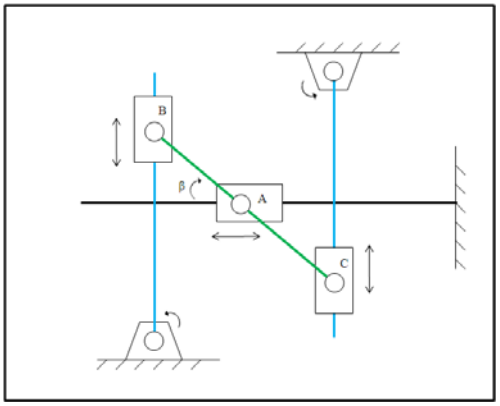


Figure 4. Parallel Mechanism Concept C

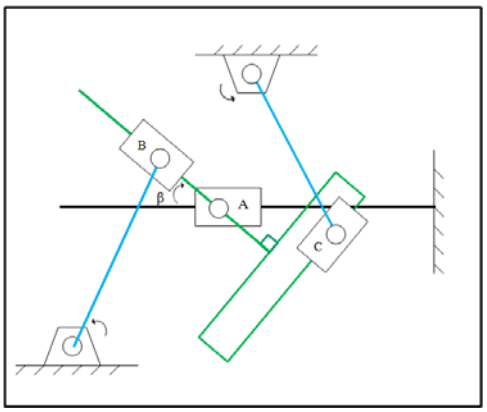


Figure 5. Parallel Mechanism Concept D

Concept B (Figure 3) – Similar to Concept A, the input to the mechanism is the rotational position of the two blue links driven by powered revolute joints. However, the location of the powered joints differs as they are mounted on near the center of the mechanism compared to Concept A, where the joint are mount near edge of the mechanism boundary. The two input links rotate outward from the mechanism with a range of 180 degrees and do not ever cross each other (Figure 3). Again, similar to Concept A, the green output link moves in a prismatic manner along the x-axis via the third black passive limb. The revolute joint located on the third limb allows for the pivoting motion characterized by angle β. As in Concept A, additional constraints are provided by points A, B and C remaining collinear throughout the range of motion to prevent unintentional pivoting of the output link.

Concept C (Figure 4) – In this mechanism, passive prismatic joints are attached and run along the length of the two blue input links (Figure 4); as opposed to running along the length of the green output link as in Concept A (Figure 2). In addition, the constraints defined by points A, B and C remaining collinear, as seen in Concepts A and B, are not present. Instead, the constraints that prevent the green output link from pivoting while the powered revolute joints are held stationary are imposed the prismatic joints on the blue input links. The remaining elements of the mechanism are very similar to Concept A, including the third passive limb and motion of the green output link.

Concept D (Figure 5) – This mechanism, though planar and parallel, differs from the previous options in that the two input limbs do not mirror one another. Similar to Concept A and B, the two input links are actuated by a powered revolute joint mounted to the base link and are a connected to the output link thorough a passive prismatic joint (on the output link side) and revolute joint (on the input link side). The third passive limb allows prismatic motion along the x-axis and pivoting motion for the output link while resisting motion along the y-axis. The primary difference between this mechanism and the mechanism in Concept A is that the sliding axes of the passive prismatic joints remain perpendicular to one another throughout the full range of motion, as seen in Figure 5. Through this kinematic arrangement the input link attached at the origin primarily influences the pivoting motion of the output link. The other input link primarily influences the extension of the output link. Due to the simple but unique arrangement, the constraints imposed by the orientation of the prismatic joints prevent any motion of the output link while the powered revolute joints are held stationary.

All four concepts demonstrate the ability to perform prismatic as well as revolute motion along the output link of each parallel mechanism. Each mechanism also couples the output link to coordinated motion between the two input links. Thus, in order to select a concept for the parallel mechanism, we must first examine the limitations of each concept. In Concept A, simply rotating the input links in opposite directions causes the output link to pivot, providing a wide range of revolute motion. In contrary, the linear expansion and contraction is significantly restricted due to the fact that the angle Beta defined in the nominal position, illustrated in Figure 2, must be maintained to produce pure translation motion. Concept B is capable of a wide range of pure translational and rotational motion. However, the input links pivot outwardly from the centerline of the mechanism, requiring that the mechanism have a large cross sectional area in order for the mechanism's output link to produce significant displacements. Concept C is capable of a wide range of pure translational motion and similar to Concept A and B, both input limbs contribute to load capacity of the mechanism. Contrary to Concept A and B, pivoting is only possible in inverse kinematics. Direct kinematics may only produce translational motion. Finally, Concept D possesses an output link which is capable of a wide range of translational and rotational motion. The primary limitation of this design is that forward limb contributes to rotational load capacity only, while the aft limb contributes to translational load capacity only. Due to this limitation, the mechanism's output link can carry significantly less load than the other three options assuming equally capable input motors.

A rating of each mechanism for various performance criteria is provided by Table 1. Note that the rating in each criterion is a ranking of one through four between options. A score of one is considered best out of the four options. All four designs are evaluated using a 63.5 x 63.5 mm cross-section to define the physical capacity of each mechanism on a common scale.

Table 1. Parallel Mechanism Design Comparison

	Length of Extension	Range of Rotation	Expansion Rate	Load Capacity	Easy of Fabrication
A	3	1	4	1	4
B	4	3	1	3	2
C	2	4	2	2	3
D	1	2	3	4	1

The performance criteria chosen were as follows: (1) Length of Extension, (2) Range of Rotation, (3) Expansion Rate, (4) Load Capacity and (5) Easy of Fabrication. Criteria 1, 2 and 3 are important because they defined the speed of the entire robot when executing the locomotion gaits. Criterion 4 defines the maximum length of the robot. The length of the robot is limited to the maximum number of segments that may be translated by a single mechanism. Finally, criterion 5 rates the effort required to mass produce the mechanisms. A simple snake robot consists of several mechanisms. A more capable robot will consist of significantly more mechanisms. In order to provide affordable, effective versions of the robot design, the segments must be relatively simple to fabricate. After thoroughly exploring the limitations of the four concepts and reviewing the scoring in Table 1, we decided that the need for a small cross sectional area, an easy of manufacturing and a large range of motion for the output link are necessary for the success of overall robot design. Therefore, Concepts A, B and C have been eliminated and Concept D was chosen for the basis of robot module. To address the issue of the load capacity, a powerful servomotor will be utilized as described in Section 5.2.

The detailed design of the selected mechanism concept utilizes slotted holes and sliding pin joints to replicate the functions of passive prismatic and revolute joints. These features allows for few parts, less assembly and a more compact design. Each parallel mechanism, pictured in Figure 6, is composed of two servomotors with servo arms attached to the output shafts acting as input links to the mechanism, see right image in Figure 6.

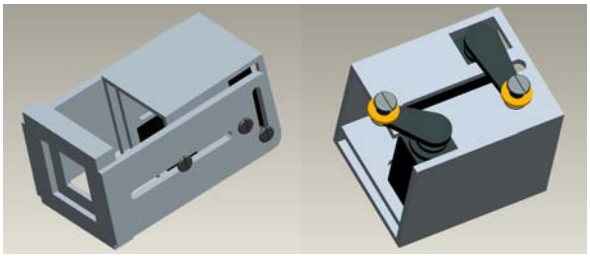


Figure 6. CAD Model of Parallel Mechanism

Each servo arm is attached to the output link of the mechanism (a U bracket) through a slotted hole and pin joint, see left image in Figure 6. Because the mechanism is a 3-D object, the passive limb (the pin in which the U bracket pivots) is replicated on the opposite side of the mechanism to provide support and stability for the U bracket throughout the range of motion. This configuration allows the output link to move in a prismatic and revolute manner depending of the location of the pin of each servo arm within its associated slotted hole.

A modular structure was devised in which two identical parallel mechanisms were assembled in a single module. The two mechanisms are stacked serially in a modular housing; with the mechanisms' orientation offset 90 degrees apart about the x-axis (direction of the linear expansion) of the module. Both mechanisms contribute to the total linear displacement of the adjacent module, while one mechanism is capable providing yawing motion and the other provides pitching motion. This assembly provides the potential for full spatial motion for the robot through modules being able to lift as well as pivot horizontally. In addition, this configuration allows all modules to contribute to the expansion-contraction capability of the robot, significantly increasing its speed.

4. PARALLEL MECHANISM ANALYSIS

Due to the fact that both input links actuate only on one side of the parallel mechanism, illustrated in Figure 6, the mechanism can be analyzed in 2-D space using a kinematic representation shown in Figure 5. In the kinematics illustration, the slotted hole and pin joints are represented using a passive revolute joint attached to a passive prismatic joint. From observation, it can be determined that the mechanism possesses 2 DOF: one translational and one rotational. Note that the axis of rotation moves along the translational axis. The DOF of the mechanism is confirmed using the Grübler criterion expressed in Equation 1.

$$F = \lambda(n - j - 1) + \sum_i f_i$$
$$F = 3(7 - 8 - 1) + 8 = 2 \quad DOF \tag{1}$$

Where λ is the degrees of freedom of space in which a mechanism is intended to function. The number of links in a mechanism, including the fixed link, is represented by n and j represents the number of joints in a mechanism, assuming that all the joints are binary. Finally, f_i is the number of degrees of relative motion permitted by joint i. With the planar nature of the mechanism confirmed, the kinematics and dynamics equations of motion for the mechanism were determined based on the kinematic representation of the planar parallel mechanism illustrated in Figure 5.

Referring to Figure 5, we assume that the center of mass of the output link is point A. The location of the moving platform can be specified in terms of the x-position of point A and an orientation angle β. The orientation angle β can be calculated using the known values of the position of point B (x_B, y_B) and A (x_A, y_A). Note that point A can only move in the x-direction due to the constraint imposed by the prismatic joint, therefore y_A is a constant. Thus there are only two unknowns to describe the 2-DOF motion of the planar parallel mechanism. Figure 7 shows the link lengths and joint angles of limb 1.

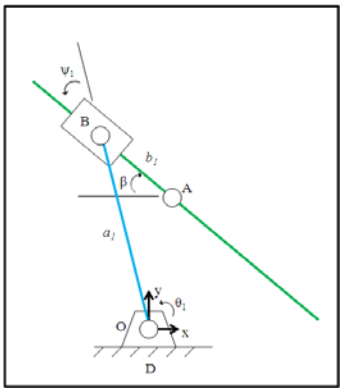


Figure 7. Limb 1 (RRP) Kinematic Representation

From the geometry of Figure 7 a vector-loop equation can be written as shown in Equation 2 and expressed in the fixed coordinate frame in Equation 3.

$$\overline{OA} = \overline{OD} + \overline{DB} + \overline{BA} \tag{2}$$

$$\begin{aligned} x_A &= x_D + a_1 c\theta_1 - b_1 c(\theta_1 + \psi_1) \\ y_A &= y_D + a_1 s\theta_1 - b_1 s(\theta_1 + \psi_1) \end{aligned} \tag{3}$$

Since D is located at the origin, $x_D = y_D = 0$. Since ψ_1 is a passive joint angle, it should be eliminated from Equation 3. Therefore, we substitute Equation 4 into Equation 3, yielding Equation 5:

$$\theta_1 + \psi_1 = 180 - \beta \tag{4}$$

$$\begin{aligned} x_A &= a_1 c\theta_1 + b_1 c\beta \\ y_A &= a_1 s\theta_1 - b_1 s\beta \end{aligned} \tag{5}$$

Note that b_1 represents a passive prismatic joint introduced by the slotted hole and pin joint. Therefore joint b_1 can be written as:

$$b_1 = \left(\frac{x_A - x_B}{c\beta}\right) = \left(\frac{x_A - a_1 c\theta_1}{c\beta}\right) \tag{6}$$

Next we substitute Equation 6 into Equation 5 and add the x- and y-terms which yields the geometric relationship for limb 1:

$$y_A - a_1 s\theta_1 + s\beta\left(\frac{x_A - a_1 c\theta_1}{c\beta}\right) = 0 \tag{7}$$

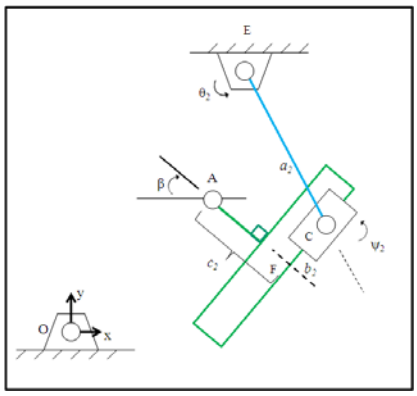


Figure 8. Limb 2 (RRP) Kinematic Representations

Similarly, the geometric relationship for limb 2, illustrated in Figure 8, is obtained. The vector-loop equation is shown in Equation 8 and expressed in the fixed coordinate frame in Equation 9.

$$\overline{OA} = \overline{OE} + \overline{EC} + \overline{CF} + \overline{FA} \tag{8}$$

$$\begin{aligned} x_A &= x_E - a_2 c\theta_2 + b_2 c(\theta_2 + \psi_2) + c_2 c(\theta_2 + \psi_2 + 90) \\ y_A &= y_E - a_2 s\theta_2 + b_2 s(\theta_2 + \psi_2) + c_2 s(\theta_2 + \psi_2 + 90) \end{aligned} \tag{9}$$

The passive joint angle, ψ_2, is eliminated from Equation 9, by substituting the expression in Equation 10. The representation of the limb 2 passive prismatic joint is shown in Equation 11.

$$\theta_2 + \psi_2 = 90 - \beta \tag{10}$$

$$b_2{}^2 = (x_F - x_C)^2 + (y_F - y_C)^2 \tag{11}$$

After summing the squares of Equation 9, Equation 11 is substituted into Equation 9 to yield the geometric relationship given in Equation 12.

$$c\beta(x_A - x_E) + s\beta(y_E - y_A) + a_2 c(\beta + \theta_2) + c_2 = 0 \tag{12}$$

From the geometric expressions in Equations 7 and 12, the inverse and direct kinematics equations for the mechanism were directly developed. Furthermore, by taking the derivative with respect to time of the geometric relationships, a Jacobian matrix was developed for the mechanism to relate input to output link velocities. Finally, the inverse dynamics are formulated using the Lagrangian approach and the complete equations of motion of this mechanism were derived. For the sake of brevity, these equations are not presented in this paper.

Table 2. Kinematics Constants

a_1	35.56 mm
a_2	38.10 mm
y_A	28.58 mm
y_E	57.15 mm
x_E	63.50 mm

c_2	31.75 mm

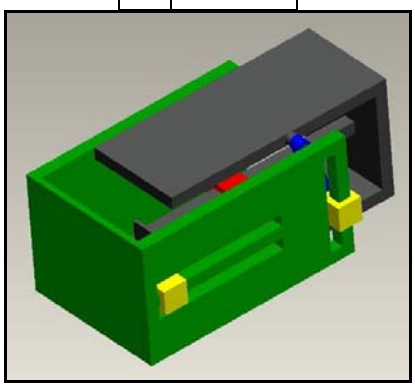


Figure 9. Pro-E Parallel Mechanism Representation

The derived equations of motion were validated against results generated by the Mechanism Analysis suite of Pro/Engineer Wildfire 4.0. The Pro/Engineer analysis is based on the solid model of the parallel mechanism illustrated in Figure 9. A sample of the validation results is provided by examining the comparison between the inverse dynamics equations and the Pro-E generated results. For the verification of the dynamics analysis, the values of the kinematics constants are given by Table 2. Figures 10 and 11 graphically illustrate the 91 sets of output link positions and forces, respectively, for the dynamics of the parallel mechanism. Note potential energy is included in the analysis. The graphic comparison of results between the Pro/Engineer solutions and the solutions from the derived equations of motion for input joint 1 and 2 are given by Figure 12 and 13, respectively.

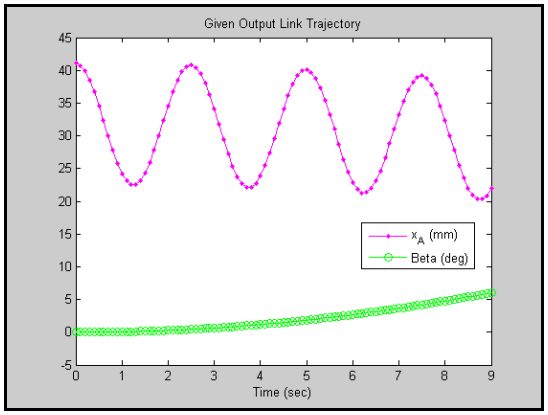


Figure 10. Inverse Dynamics Input: Output Link Kinematics

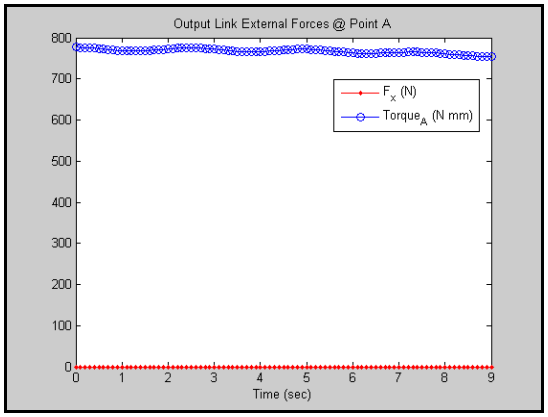


Figure 11. Inverse Dynamics Input: Output Link Forces

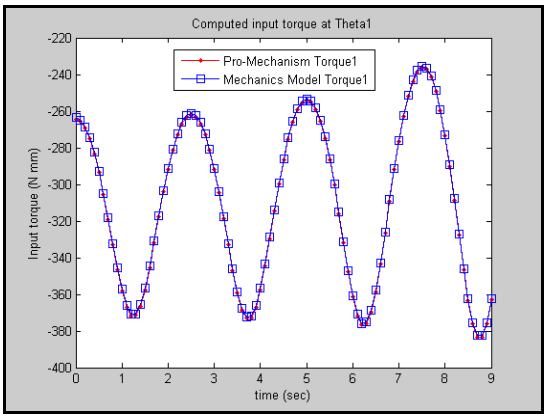


Figure 12. Inverse Dynamics Output: Pro-E vs. Model: Theta1

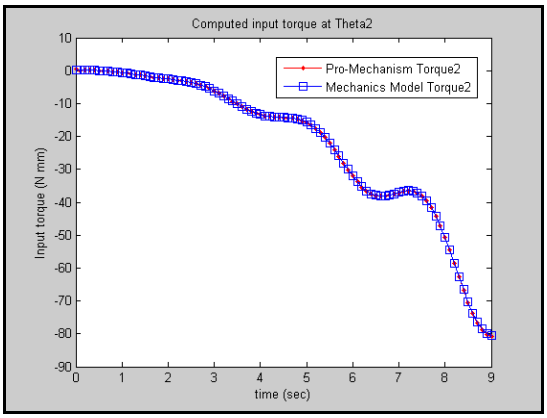


Figure 13. Inverse Dynamics Output: Pro-E vs. Model: Theta2

5. ROBOT PERFORMANCE

In this section, we explore robot performance in two ways: velocity performance based on various materials for the friction anchoring and robot performance using several performance metrics to characterize the robot design. The selection of the material for the friction anchoring concept is determined by conducting velocity trials using various materials. Once the "best" material is selected, the maximum velocity is measured for the overall robot and the performance of the new design is assessed using the performance metrics.

5.1 Friction Anchor Material Study

The variable static friction force concept is a simple yet effective method of anchoring one end of the robot to the terrain to provide a counter to the reaction forces of the powered joints of the modules during forward or turning gaits. In nature the anchoring or "planting" of such a device, i.e. a foot, is done by redistributing more of the animal's body weight across the surface of the foot to increase the friction force between the foot and the terrain. This concept adopts a similar approach. The surface of the friction anchor is covered in a material with a much higher coefficient of friction than the rest of the robot's housing material. The friction anchor is placed in contact with the terrain by the action of a powered revolute joint as illustrated Figure 14.

The friction force, a function of the normal force between the anchor and the terrain, is increased or decreased by varying the angle, θ_{FA}, of the revolute joint which changes the amount of the module weight being supported by the friction anchor. Figure 14a depicts the friction anchor in its nominal position, with the anchor's high coefficient of friction surface not in contact with the terrain, allowing the terminal end of the robot to freely slide under the force of the linear actuators. Figure 14b depicts the friction anchor surface in contact with the terrain with only a slight change of θ_{FA}, useful in low reaction force gaits. Figure 14c depicts a large change in θ_{FA}, useful in high reaction force gaits.

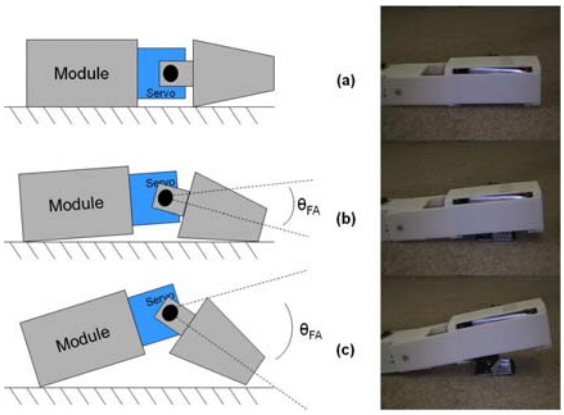


Figure 14. Kinematic Representation and Prototype Anchor

The actual material used for the pads was determined through an experimental study. In this study, 15 friction pad materials are evaluated using the forward gait. Three velocity trials are performed using each material candidate on a rough surface: carpet. The trials are conducted using a two module, aluminum prototype of the robot design and the results presented in Table 3.

Table 3. Forward Velocity Study Data

	Velocity (mm/s)			
Material	**Trial 1**	**Trial 2**	**Trial 3**	**Avg.**
Skid Guard ™ Tape	103.25	100.00	101.60	101.62
Waxman® Grip Pads	0.00	0.00	0.00	0.00
Vinyl Foam	0.00	0.00	0.00	0.00
EPDM Rubber	47.39	60.19	53.59	53.72
Emery Cloth: Fine	111.40	100.00	95.49	102.30
Emery Cloth: Medium	79.87	69.02	80.38	76.43
Emery Cloth: Coarse	83.55	71.35	78.88	77.93
Drywall Sanding Medium Screen	83.01	84.67	74.27	80.65
Al_2O_3 Paper 220 Grit	78.40	77.44	78.88	78.24
Al_2O_3 Paper 150 Grit	73.41	76.05	73.41	74.29
Al_2O_3 Paper 100 Grit	76.97	74.27	77.44	76.23
Al_2O_3 Paper 80 Grit	73.41	74.71	68.28	72.13
Al_2O_3 Paper 60 Grit	74.71	76.97	81.41	77.70
Duck ™ Friction Tape	74.71	70.17	75.60	73.49
Polyurethane Foam	41.23	34.23	37.13	37.53

Note that Waxman® Grip Pads and Vinyl Foam were rated zero velocity due to the ability of robot to gain traction on carpet using these materials. In addition, with the exception of EPDM Rubber and Polyurethane Foam, almost all of the remaining trial materials yielded very similar velocity results. Based on the results, Skid Guard ™ Tape and Fine Grit Emery Cloth are the best choices. Fine Grit Emery Cloth is technical better than Skid Guard ™ Tape in terms of overall average, as well as, the fact that the maximum velocity of 111.40 mm/sec seen throughout the study was observed during Fine Grit Emery Cloth trials. However, Skid Guard ™ Tape is more consistent per trial and coupled with the fact that Skid Guard ™ Tape is more durable, it was chosen as the preferred anchor covering.

5.2 FDM Fabricated Prototype

The new robot prototype, pictured in Figure 15, was fabricated using FDM manufacturing to demonstrate the new drive mechanism design and new friction pad material. Utilizing the recommended candidate material, Skid Guard TM Tape, the FDM prototype achieved a maximum velocity result of 196.65 mm/sec. The prototype robot is made primarily from ABS polymer. The robot has a 69.85 x 69.85 mm cross-section. The robot has a contracted length of 850.9 mm and a fully extended length of 1143 mm, as observed in Figure 15. The total mass of the robot is approximately 1.36 kg. The robot consists of three modules connected by six independent parallel mechanisms assembled in a serial configuration allowing each module to move in linearly and pivot with respect to the adjacent module. Each mechanism is capable of 90 degrees of motion and 48.68 mm of extension. Each parallel mechanism consists of two standard sized Hitec HS-985MG High Torque servomotors. They are capable of 12.40 kg-cm of maximum torque and a maximum speed of 0.13 s/60 deg.

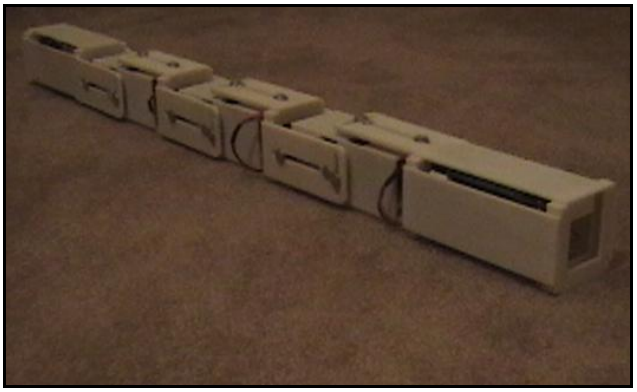


Figure 15. FDM Fabricated MSIR Prototype

5.3 Performance Metrics

Performance of the robot prototype is characterized through the use of three dimensionless performance metrics. The first metric is a measure of the robot's propulsive efficiency as defined in Equation 13.

$$\eta_{velocity} = \frac{measured_velocity}{predicted_velocity} \qquad (13)$$

The predicted velocity is calculated based on the gait and module velocity and is computed to be 243.4 mm/s. $\eta_{velocity}$ is calculated as 0.81. This value indicates that there is approximately an average of 20% slippage between the anchor and terrain. The next metric is a ratio between the modular input velocity to the gait and the resulting robot velocity and is defined by Equation 14.

$$Modular_ratio = \frac{n * robot_velocity}{3 * module_velocity} \quad (14)$$

The ratio is computed as $0.20n$, where n is the number of modules in the robot. This number defines the expected improvement in robot velocity based on increase in modular velocity or increase in number of modules (where n is greater than 3). Modular velocity improvements may be due to faster motors, higher voltage batteries or reductions in mechanism weight and friction. The final performance metric utilized in this work is the calculation of the Froude number, Equation 15. In robotics, the Froude number, *Fr*, is typically used to normalize walking speed of legged robots to provide a better comparison between the robots and animals.

$$Fr = \frac{v}{\sqrt{gl}} \quad (15)$$

Where v is the walking speed, l is the leg length, and g is gravity. Usually the formulation of the Froude number for snake-inspired robots is problematic due to the fact that simply growing the length of a snake-inspired robot might drastically affect the dimensionless value without changing the velocity; hence the Froude number for a snake-inspired robot is contrived [1]. However, in this design, increases in length directly lead to increases in velocity, as seen in Equation 14. This is due to the fact that the gait for this robot is similar to the strides made by walking robots. The Froude number range for the robot is calculated to be from $Fr = 0$ to $Fr = 0.12$. The Froude number ranges for some of the state of the art walking robots have been shown as: 'Rabbit' shows a speed range from about $Fr = 0.15$ to $Fr = 0.3$, 'Toddler' from $Fr = 0$ to $Fr = 0.09$ and the relatively fast and small `RunBot' from $Fr = 0.25$ to $Fr = 0.5$ [14]. In comparison Honda's Asimo has a speed range from $Fr = 0$ to $Fr = 0.3$ and humans from $Fr = 0$ to about $Fr = 1$ [14]. Although the current prototype has a relatively small range compared to other robots, this range can easily be increased the simply adding more modules, a modification which may not be as trivial for some of other the listed robot designs.

6. CONCLUSIONS

In this paper, FDM provided the opportunity to develop an effective drive mechanism for executing rectilinear motion based on a new parallel mechanism. FDM also was utilized to reduce the weight and thereby increase the speed potential of the snake-inspired robot design based on a rectilinear gait. A complete kinematics and dynamics analysis was performed and validated for the new mechanism design. A prototype robot was fabricated using FDM to demonstrate the robot architecture and gait concepts. The prototype executed the forward gait with a maximum velocity of 196.65 mm/sec. The prototype employs a cross section of 69.85 x 69.85 mm; allowing the robot to traverse small spaces. The benefit of this design is the enabling of new high speed applications for snake-inspired robots. One such application is the inspection of a structurally unstable building for trapped or incapacitated people prior to committing human rescuers. Although current snake-inspired robots are functionally capable of executing this mission, the critical factor is the time required to complete the inspection, as time in a rescue mission may mean the difference between life and death for both the occupants and the rescuers.

7. REFERENCES

[1] Anderson, F. C., and Pandy M. G., 2001, "Dynamic Optimization of Human Walking," Transactions of the ASME, 123.

[2] Dowling, K., 1997, "Limbless Locomotion: Learning to Crawl with a Snake Robot," Ph.D. Thesis, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA.

[3] Yim, M., Duff, D. G., and Roufas, K., 2000, "Modular Reconfigurable Robots: An Approach to Urban Search and Rescue," In the *Proceedings of the 1st International Workshop on Human-friendly Welfare Robotics Systems*, Taejon, Korea.

[4] Lipkin, K., Brown, I., Peck, A., Choset, H., Rembisz, J., Gianfortoni, P., and Naaktgeboren, A., 2007, "Differentiable and Piecewise Differentiable Gaits for Snake Robots," In the *Proceedings of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, San Diego, CA.

[5] Ikeda, H., and Takanashi, N., 1987, "Joint Assembly Moveable Like a Human Arm," US Patent 4683406, Assignee: NEC Corporation.

[6] Paap, K. L., Dehlwisch, M., Klaassen, B., 1996, "GMD-Snake: A Semi-Autonomous Snake-Like Robot," In the *Proceedings of the 1996 Conference on Distributed Autonomous Robotic Systems 2*, Springer-Verlag, Tokyo.

[7] USC Polymorphic Robotics Laboratory. http://www.isi.edu/robots/conro/

[8] Kamimura, A., Kurokawa, H., Yoshida, E., Murata, S., Tomita, K., and Kokaji S., 2005, "Automatic Locomotion Design and Experiments for a Modular Robotic System," IEEE/ASME Transactions on Mechatronics, 10.

[9] Ohno, H., and Hirose, S., 2000, "Study on Slime Robot (Proposal of Slime Robot and Design of Slim Slime Robot)," In the *Proceedings of the 2000 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 3.

[10] Chen, I-M., Yeo, S. H., and Gao, Y., 2001, "Locomotive Gait Generation for Inchworm-Like Robots Using Finite State Approach," Robotica, 19.

[11] Yeo, S. H., Chen, I-M., Senanayake, R. S., and Wong, P. S., 2000, "Design and Development of a Planar Inchworm Robot," In the *Proceedings of the 17th IAARC International Symposium on Automation and Robotics in Construction*, Taipei, Taiwan.

[12] Chen, I-M., and Yeo, S. H., 2003, "Locomotion of a Two-Dimensional Walking-Climbing Robot Using a Closed-loop Mechanism: from Gait Generation to Navigation," The International Journal of Robotics Research, 22.

[13] Suh, J., Homans, S., and Yim, M., 2002, "Telecubes: Mechanical Design of a Module for Self-Reconfigurable Robotics," In the *Proceedings of the 2002 IEEE International Conference on Robotics and Automation*.

[14] Hobbelen, D. and Wisse, M. Limit Cycle Walking. in Hackel, M. ed, *Humanoid Robots: Human-like Machines*, I-Tech Education and Publishing, Vienna, 2007, 277-294.

Emergency Response Robot Evaluation Exercise

Adam Jacoff[i]
301 975 4235
adam.jacoff@nist.gov

Hui-Min Huang[i]
301 975 3427
hui-min.huang@nist.gov

Ann Virts[i]
301 975 5068
ann.virts@nist.gov

Anthony Downs[i]
301 975 3436
anthony.downs@nist.gov

Raymond Sheh[ii]
raymond.sheh@robolit.com

[i]National Institute of Standards & Technology
100 Bureau Drive MS 8230
Gaithersburg, MD 20899

[ii]Robolit LLC
1829 Pine Street, Suite 404
Philadelphia, PA 19103

ABSTRACT

More than 60 robot test methods are being developed by a team led by the National Institute of Standards and Technology (NIST) with the sponsorship of U.S. Department of Homeland Security (DHS). These test methods are being specified and standardized under the standards development organization ASTM International. These standards are developed for the purposes of identifying the capabilities of mobile robots to help emergency response organizations assess the applicability of the robots.

The test methods are developed using an iterative process during which they are prototyped and validated by the participating researchers, developers, emergency response users, and robot manufacturers. We have conducted a series of evaluation exercises based on the test method implementations. These events were participated by representatives from all the different segments of the community. As such, these events present a unique opportunity for advancing the test methods, collecting capability data, and identifying robotic technology focusing issues. This paper describes an exercise event that this effort recently conducted.

Categories and Subject Descriptors

J.2 [physical sciences and engineering] unmanned systems performance

General Terms

Measurement, Performance, Design, Human Factors, Standardization, Verification

Keywords

capability, emergency response, evaluation, human-robot interaction, HRI, measure, metrics, mobility, power, radio communications, robot, performance, repetition, sensor, standard, task, test, test method, test suite, trial

(c) 2012 Association for Computing Machinery. ACM acknowledges that this contribution was authored or co-authored by a contractor or affiliate of the U.S. Government. As such, the Government retains a nonexclusive, royalty-free right to publish or reproduce this article, or to allow others to do so, for Government purposes only. PerMIS'12, March 20-22, 2012, College Park, MD, USA. Copyright © 2012 ACM 978-1-4503-1126-7-3/22/12...$10.00

1. INTRODUCTION

National Institute of Standards and Technology (NIST), with sponsorship from the Department of Homeland Security (DHS) Science and Technology Directorate, is developing a suite of DHS-NIST-ASTM International Standard Test Methods for Response Robots to quantitatively identify the capabilities of robots for emergency response applications, independent of robot size. These standard test methods identify robot capabilities in mobility/maneuvering, energy/power, sensing, radio communication, manipulation, human-robot interaction, logistics, and safety to provide point of comparison for a variety of robot sizes and configurations prior to testing in more realistic scenarios. Statistically significant test results captured within standard test methods measure incremental system improvements, highlight break-through capabilities, and support procurement and deployment decisions. More than sixty such test methods are under development with associated apparatuses, procedures, and performance metrics. They are being standardized through the ASTM International Standards Committee on Homeland Security Applications, Subcommittee on Operational Equipment, Robots Task Group (E54.08.01). Earlier publications [1, 2, 3] described these test methods development efforts.

1.1 Key Definitions

The term emergency response robot, or response robot, must be defined first. We define it as: a remotely deployed device intended to perform operational tasks at operational tempos that can serve as an extension of the operator to:

- improve remote situational awareness,
- provide means to project operator intent through the equipped capabilities,
- improve effectiveness and efficiency of the mission, and
- reduce risk to the operator.

Key features of a response robot include:

- Rapidly deployed
- Remotely operated from an appropriate standoff
- Mobile in complex environments

- Sufficiently hardened against harsh environments
- Reliable and field serviceable
- Durable and/or cost effectively disposable
- Equipped with operational safeguards

Repetition is a fundamental term used in the effort. It is defined as a robot's completion of the task as specified in the test method and readiness for repeating the same task when required.

Test event or event is defined as a set of testing activities—test methods at various stages of maturity or scenario tasks—that are planned and organized by the test sponsor and to be held at the designated test site(s).

Trial is defined as the identified number of repetitions to be performed by a testing robot for the test results to reach required statistical significance.

1.2 Test Method Focus

These test methods address high-priority tasks identified by emergency responders, including:

- Fast, light, and mobile reconnaissance tasks for throwable robots;
- Wide area survey tasks for hazardous material (HAZMAT) or other events for packable or luggable robots;
- Counter Improvised Explosive Devices (C-IED), Vehicle Borne IED (C-VBIED), and Personal Borne (C-PBIED) tasks for mobile manipulators;
- Aerial reconnaissance for small unmanned aerial systems (sUAS) conforming to the emerging Federal Aviation Administration (FAA) Group I class weighing less than 2 kg (4.4 lbs), less than 30 knots (56 km/hour) maximum speed in horizontal flight, and harmless upon impact;
- Underwater reconnaissance for small remotely operated vehicles (ROV).

For each of these application domains, the standard test methods enable quantitative robot evaluations, provide practice tasks, and help measure operator proficiency.

1.3 Development Process

The standards development process involves hosting periodic robot requirements workshops, standards committee meetings, and response robot evaluation exercises at responder training facilities. Emergency responders, robot developers, and test administrators are gathered around draft standard test methods to practice deployment scenarios. The evaluation exercise events allow emergency responders to articulate essential robot capabilities, validate proposed test methods, and refine performance thresholds and objectives based on objective performance data captured across a class of robots. Emergency responders involved in the process learn about the state-of-the-science in robotic capabilities and help ensure that the test method apparatuses and procedures address their application needs. These events also inform robot developers regarding the reliability and applicability of their robots for actual deployment scenarios, and the ease of use of their systems as they train responders within the test apparatuses. Robot developers involved in the process learn about emerging operational requirements and can demonstrate robotic capabilities by capturing statistically significant performance data within the resulting standard test methods.

2. EMERGENCY RESPONSE ROBOT EVALUATION EXERCISE

The seventh in a series of DHS/NIST Response Robot Evaluation Exercises was hosted at the emergency responder training facility known as Disaster City in College Station, Texas (TX). Thirty emergency responders from across the country participated—half representing DHS Federal Emergency Management Agency (FEMA) Urban Search and Rescue (US&R) teams and half representing bomb squads. They helped validate emerging standard robot test methods, became familiar with available robot capabilities, and advised robot developers regarding operational requirements. All applicable robots were invited to take part in these exercises including ground, aquatic, and the aforementioned sUAS. Robots' capabilities were identified within the implemented emerging standard test methods before being used to familiarize and train the responders with the capabilities and being deployed with the responders to perform operational tasks in the implemented practice scenarios. These designed correspondences between the test methods and the scenarios include:

- Test methods for energy endurance, mobility—covering obstacles and terrains, radio communications—covering line of sight, non-line of sight, and structure penetration, and sensors—covering video acuity, pan-tilt-zoom tasks, 2-way speech intelligibility, range imager resolution, and thermal imager resolution will prepare robots to perform operational tasks for down-range reconnaissance of hazardous material and passenger train wrecks from stand-offs greater than 150 m (500 ft). Figure 1 illustrates a mobility test course featuring crossing ramp terrains.
- Test methods for navigating, searching, and mapping (2D and 3D) complex environments will prepare robots for operational tasks in building interiors and exteriors, partially collapsed structures, and confined spaces in rubble piles.
- Test methods for mobile manipulation—covering non-contact inspection, access tool for window breaking and boring, and grasping/removal tasks will prepare robots for operational tasks to C-IED, C-VBIED, and C-PBIED.
- Test methods for towing trailers and gripper-dragging objects will prepare robots for operational tasks in C-IED, C-VBIED, C-PBIED, and US&R scenarios.
- Test methods for underwater navigation, station-keeping, and sensor acuity will prepare for operational tasks in vehicle reconnaissance in the onsite pond.
- Test methods for air-worthiness, station-keeping, and sensor acuity will prepare small unmanned aerial systems with vertical takeoff, hovering, and landing capabilities for operational tasks supporting several scenarios noted above.

These response robot evaluation exercises introduced emerging robotic capabilities to emergency responders within their own training facilities, while educating robot developers about the performance requirements necessary to be effective in these rigorous application domains. They also helped correlate the draft standard test methods with envisioned deployment tasks and laid the foundation for usage guides identifying a robot's applicability to particular scenarios. The results were the following:

- Refined and validated draft standard test apparatuses, procedures, and metrics

- Quantitative robot capability data to support test method balloting within the ASTM International Committee on Homeland Security
- Feedback for robot developers who allow user training/practice within test apparatuses
- Updated/Expanded Response Robot Capabilities Compendium capturing the trade-offs involved in tested robot configurations showing what robots can and cannot be expected to do reliably in the field

Disaster City is a 210,000 squared meter (52-acre) training facility designed to deliver the full array of skills and techniques needed by urban search and rescue professionals. As part of the Texas Engineering Extension Service (TEEX) at Texas A&M University and a training site for FEMA Texas Task Force (TF) 1 (TX-TF1), the facility features full-size collapsible structures that replicate community infrastructure, including a strip mall, office building, industrial complex, assembly hall/theater, single family dwelling, train derailment, three rubble piles, a C-VBIED scenario, and an underwater vehicle reconnaissance scenario.

2.1 Agenda

This event was held on Monday through Friday, including two days of robot practice and testing within the DHS-NIST-ASTM International Standard Test Methods for Response Robots, two days of robots deploying in operational scenarios with responders, and a final half-day ASTM standards committee meeting to capture feedback.

Day 1 and Day 2: Robot Practice and Testing
November 14-15, 8:00 am Safety Briefing - 5:00 pm Hot-wash

On site were robot developers and test administrators only. All participating robots ran through all applicable test methods, providing practice sessions prior to arrival of the emergency responders. “Expert” operators, chosen by the robot developers to capture baseline performance data and provide developer feedback regarding the test apparatuses and test methods, operated the robots. The robot capability data identified was not to be published. Rather the robot developers were exposed to the entire suite of responder-validated test methods and provided an opportunity to help refine the test methods prior to standardization. In other words, this event was the final opportunity for such refinement for this set of tests.

Day 3: Robot Testing and Operator Training
November 16, 8:00 am Safety Briefing - 5:00 pm Hot-wash

On site were emergency responders representing FEMA Task Force Teams and bomb squads from across the country, robot developers, and NIST administrators. The assembled responders rotated in small groups through all test methods to train on robots prior to deploying them into the US&R training props on site. They became familiar with robotic capabilities using the best performing robots in any given test method. While being exposed to the latest emerging technologies, the responders provided feedback to developers regarding necessary capabilities, operator interfaces, and realistic usage scenarios.

A lunchtime presentation focused on the use of robots in response to Japan’s multiple disasters this year. It was presented by the leadership of Japan’s International Rescue Systems Institute, who was also a professor at Tohoku University in Sendai where the devastating earthquake and tsunami did the most damage. In addition, a professor from the University of Tokyo discussed the response at the Fukushima Daiichi nuclear facility.

Day 4: Operational Scenarios
November 17, 8:00 am Safety Briefing - 5:00 pm Hot-wash

The emergency responders focused on the most applicable robots to perform targeted tasks in the operational practice scenarios around the site, which included embedded test methods practiced in the previous days. Robot developers accompanied the responders on scenario deployments as observers, advisors, and as operators in particularly difficult deployments to show the potential of robot capabilities. Robot developers onsite, including those whose robots were not selected by responders for deployment, watched the incident response scenarios and observed the robot deployments and absorb the lessons.

A lunchtime presentation focused on the use of standard test methods to provide rapid evaluations of ultra lightweight reconnaissance robots to identify the overall capabilities of the class in support of a rapid fielding initiative by the DoD’s Joint Improvised Explosive Device Defeat Organization (JIEDDO). It was presented by a representative from JIEDDO.

Day 5: ASTM International Standards Committee Symposium
November 18, 8:00 am 1:00 pm

The ASTM International Standards Committee on Homeland Security Applications; Operational Equipment; Robots (E54.08.01) hosted a Symposium for all participants to provide feedback on the proposed standard test methods, assess potential operational impact of robots, and define necessary improvements for robots to become useful tools for responders. Presentations included robot developers and other parties have used the standard test methods to measure, refine, and ultimately advertise their capabilities. Robot researchers presented cases where standard test methods helped refine assumptions about the domain tasks and focused their innovation, especially through international robot competitions which used the test methods as challenge arenas. Recent robot procurement efforts were also discussed which have used the test methods to quantify a class of robots or to specify certain combinations of capabilities demonstrated to statistical significance.

2.2 Test Stations and Test Methods

The following subsections describe the test stations and the associated test methods that were set up at the test site. Each of the test methods is noted with its standardization status, as follows:

- (ASTM ####): The document specifying the test method has completed its standardization process and is a published standard.
- (B): The draft document specifying the test method is in the balloting process.
- (V): The draft document specifying the test method is being validated within the ASTM Committee. Robots have begun testing within the test method and results are being collected for analysis.
- (P): The test method is being prototyped. Apparatuses might have been designed or developed.

A test method might also be noted with a Work Item number (WK####), which indicates that the test method has been registered officially with ASTM as a candidate standard and has received the designation. We typically do the registration when the test method is estimated to be about 12 months away from a ballot.

2.2.1 Dispatch Station

- Standard Terminology for Urban Search and Rescue Robotic Operations (ASTM E 2521–07A) [4]
- Standard Terminology for Federal/State/Local Bomb Squads (P)
- Standard Practice for Evaluating Cache Packaged Weight and Volume of Robots for FEMA Urban Search and Rescue Teams (ASTM E2592-07) [5]
- Standard Practice for Evaluating Cache Packaged Weight and Volume of Robots for Federal/State/Local Bomb Squads (P)

2.2.2 Mobility Terrains Station

- Maneuvering Tasks: Sustained Speed (ASTM E2829) [13]
- Maneuvering Tasks: Towing Grasped/Hitched Sleds (ASTM E2830) [14]
- Confined Area Terrains: Continuous Pitch/Roll Ramps (ASTM E2826) [10]
- Confined Area Terrains: Crossing Pitch/Roll Ramps (ASTM E2827) [11]
- Confined Area Terrains: Symmetric Stepfields (ASTM E2828) [12]
- Confined Area Terrains: Gravel (V)
- Confined Area Terrains: Sand (V)
- Confined Area Terrains: Mud (P)

2.2.3 Mobility Obstacles Station

- Confined Area Obstacles: Gaps (ASTM E2801) [6]
- Confined Area Obstacles: Hurdles (ASTM E2802) [7]
- Confined Area Obstacles: Inclined Planes (ASTM E2803) [8]
- Confined Area Obstacles: Stair/Landings (ASTM E2804) [9]
- Vertical Insertion/Retrieval Stack with Drops (V)

2.2.4 Energy/Power Station

- Endurance: Confined Area Terrains: Continuous Pitch/Roll Ramps (V) (W34433)
- Peak Power: Confined Area Obstacles: Stairs/Landings (P)

2.2.5 Radio Communications Station

The test site is at the Riverside Campus Airstrip, 20 minutes away.

- Control and Inspection Tasks: Line-of-Sight Environment (ASTM E2854) [15]
- Control and Inspection Tasks: Non-Line-of-Sight Environment (ASTM E2855) [16]
- Control and Perception Tasks: Structure Penetration (P)
- Control and Perception Tasks: Urban Canyon (P)
- Control and Perception Tasks: Interference Signal (P)

2.2.6 Manipulation Station

- Confined Area Inspection Tasks: Recessed Targets on Elevated Surfaces (V) (WK27851)
- Confined Area Grasping and Removal Tasks: Weighted Cylinders on Elevated Surfaces (V) (WK27852)
- Door Opening and Traversal Tasks (V) (WK27852)

2.2.7 Human-System Interaction Station

- Search Tasks: Random Mazes with Complex Terrain (B) (WK33259)
- Navigation Tasks: Random Mazes with Complex Terrain (ASTM E2853) [17]
- Mapping Tasks: Hallway Labyrinths with Complex Terrain (P)
- Mapping Tasks: Sparse Feature Environments (P)
- Operator Interface Constraints: PPE; Posture; Lighting (P)
- Operator Interface Indicators: Low Battery; Robot Tilt (P)

2.2.8 Sensors Station

- Video: Acuity Charts and Field of View Measures (ASTM E2566-08) [18]
- Video: Pan-Tilt-Zoom Tasks (V) (WK33261)
- Audio: Speech Intelligibility (Two-Way) (V) (WK34435)
- Audio: Spectrum Response Tones (Two-Way) (P)
- Range Imager Resolution (P)
- Thermal Imager Resolution (P)

2.2.9 Safety and Environmental Station

- Water Fording (V)
- Throw Distance Over a 2.4m (8ft) Wall (V)
- Washdown/Decontamination (V) (WK33262)
- Lost Communications Behaviors (P)

2.2.10 Aerial: Small Unmanned Aerial Systems (sUAS) Station

The initial stage were for Vertical Takeoff and Landing, FAA Group I, <2kg, 30knots, frangible.

- Maneuvering Tasks: Station-Keeping: Horizontal and Vertical (V)
- Energy/Power: Endurance (V)
- Safety: Crash Impact Forces (V)
- Safety: Lost Communications Behaviors (P)

2.2.11 Aquatic: Small Remotely Operated Vehicles Station

- Maneuvering Tasks: Sustained Speed (P)
- Maneuvering Tasks: Station-Keeping in a Current (P)
- Maneuvering Tasks: Bollard Thrust (P)
- Manipulation: Cutting Tasks: Rigid and Flexible (P)
- Manipulation: Lifting and Placing Tasks (P)
- Sensors: Video Acuity and Field of View (P)
- Sensors: Sonar Resolution (P)
- Safety: Gripper Drag

2.2.12 Counter Vehicle-Borne Improvised Explosive Devices Station

- Non-Contact Inspection Tasks:
  - Elevated Surfaces with Recessed Targets (0 and 90 degree approach) (P)

- Convex Surfaces with Recessed Targets (Vertical and Horizontal) (P)
- Vehicle Cabs (through window) (P)
- Vehicle Underbody (P)

- Grasping and Removal Tasks:
  - Elevated Surfaces with Weighted Cylinders (0 and 90 degree approach) (P)
  - Elevated Surfaces with Fuel Cans and Propane Tanks (P)
  - PBIED Gripper Drag and Roll-over (P)

- Payload Placement Tasks:
  - Vehicle Underbody Expulsion Disruptors (P)
  - Vehicle Interior Bottle Disruptors (P)
  - Vehicle Interior Overpressure Disruptors (P)

- Tool Deployment Tasks: (part of the robot configuration)
  - Window Breaking and Boring Drills (P)
  - PAN Disruptor Aiming (P)
  - Cutting Straps/Cloth (P)

- Trailer Towing and Placement:
  - Large Vehicle Bomb Disruptors (P)

2.2.13 Operational Scenarios

- Passenger Train Search and Package Removal Tasks
- Hazmat Train Reconnaissance and Retrieval Tasks
- Pancake Collapsed Structure
- Municipal Building and Parking Garage Collapse
- Rubble Piles #1, #2, and the Wood Rubble Pile
- Strip Mall Reconnaissance
- Aerial: Exterior Building Reconnaissance
- Aquatic: Submerged Vehicle Reconnaissance in the Lake

2.3 Test Administration Policy

The suite of standard test methods characterizes the capabilities of robots intended to operate in human-scale, complex environments with variable terrains, lighting, temperature, etc. These current tests are all teleoperation based, although new tests aiming for autonomy are being developed [19, 20]. Each test was assigned an operator station, which was positioned in such a manner as to insulate the operator from the sights and sounds generated at the test apparatus but was within the robot's communications range, except for the radio communications test methods. The operator was required to stay there and use her/his OCU to test the robot—see Section 2.6 for field reset situations. The robot configuration as tested shall be specified in detail to include its size, mass, manipulators, payloads, batteries, communications, etc. This configuration is subjected to the entire suite of test methods. Any variation in robot configuration must be retested across the entire suite of test methods to provide a comprehensive overview of performance characteristics and trade-offs for that particular robot variant. Systems with assistive capabilities or autonomous behaviors should demonstrate improved remote operator/robot performance, efficiency, or survivability of the robot under test. Although these test methods were developed for response robots, they may be applicable to other application domains with modest variations in terrains, targets, or tasks.

2.4 Apparatuses and Targets

The apparatuses associated with these test methods challenge specific robot capabilities in repeatable ways to facilitate direct comparisons of different robot models and particular configurations of similar robot models. Many of the test apparatuses use terrains, targets, and tasks that are intentionally abstract to facilitate the standardization process, which requires capture of repeatable results within a specific test facility and reproducible results across different test facilities. They are generally fabricated using readily available materials to facilitate fabrication by robot developers to support system innovation, refinement, and hardening, and for robot users to support robot evaluation and proficiency training. For example, many test apparatuses are constructed with oriented strand board (OSB) to provide a common friction surface similar to dust covered concrete. The specific terrains, targets, and tasks used can be modified or replaced with more operationally representative examples while using the same apparatuses and procedures to further support training, practice, and comparison of specific system capabilities. These test methods should be considered baseline evaluations and performed prior to more relevant operational tasks defined by robot users. Such operational tasks should leverage a specific set of test methods to establish that robots can perform the necessary capabilities to statistical significance.

Visual targets are used within the test apparatuses to evaluate the visual and color acuity of robots under test in lighted and dark conditions. Visual targets consist of Snellen visual acuity charts, also known as Tumbling E's, and standard hazardous material labels and placards. Snellen Tumbling E's are essentially line resolution tests that can be read through the remote operator station and announced by a robot operator to the test administrator. The test administrator then verifies the reading before scoring the result on the form. A correct reading of a particular line of four Tumbling E's produces a numeric measurement of the visual acuity that can be referenced to average human vision. The visual acuity test method uses comprehensive sets of Tumbling E charts to identify the robot's far field and near field visual acuity. Three line labels shown in Figure 4 are used within other test apparatuses as visual targets to provide an indication of the robot's visual acuity relative to human vision.

Hazardous materials labels provide a variety of standard visual targets that introduce modest complexity for visual identification tasks and operational relevance for some users. The labels contain four attributes including color, icon, text, and number. The text and numbers are sized for average human acuity. Identification of any three of four attributes is considered successful identification of the target.

More operationally relevant objects are used to provide targets for reconnaissance tasks, including simulated pipe bombs, simulated artillery shells, timer devices, cell phones, detonation cords, power sources, etc. Non-visual targets can also be used to test the capabilities of onboard sensors. For example, we have placed trace chemical, radiological, and explosive sources along with these visual acuity targets within the test apparatuses to identify proximity at initial detection and then localization accuracy of sources.

2.5 Test Trials and Statistical Significance

Performance data collections are conducted using the test apparatuses and associated test procedures to capture robot and remote operator performance across a statistically significant number of repetitions. Robots are tested to completion of certain tasks with "expert" operators designated by the developer to capture a task-based capability for a given robot in a given apparatus. The number of repetitions for each test method is determined by ASTM (or the test sponsor) using statistical principles while considering test administration practicalities for longer tests, such as the Endurance test method. The elapsed time of each test is typically not included as a standard metric to de-emphasize speed in favor of task completeness, although the test duration is captured for secondary comparison purposes. Timing measures are typically reported as an average time to perform each repetition, or as an average time to perform a particular sub-task within a test method that can produce varying levels of completeness so that novice operators can quantitatively establish their proficiency as a percentage of "expert" performance within the same test method.

Test trials typically consist of 30 repetitions to demonstrate statistical significance to at least 80 % reliability with 80 % confidence. Successful and failed trials are specifically noted. During the first trial at a particular apparatus setting, the Test Administrator may stipulate that the robot was dominating the apparatus at that setting after demonstrating the first 10 successful repetitions with no failures. However, if there are any failed repetitions, a second or even a third set of 10 repetitions would be required. For a trail to be noted as statistically significant, no more than 1 failure in 20 repetitions, or 3 failures in 30 repetitions, are allowed. This enables setting the apparatus to some known capability and quickly moving toward more aggressive apparatus settings to determine the limit of the robot's capabilities. All subsequent trials must be tested to 30 repetitions for a given apparatus setting.

2.6 Field Maintenance Resets During Test Trials

During a test trial robots may become stuck, inverted, or inoperable. The operator has the option to call a Field Maintenance Reset, which allows the operator to leave the operator station, reset the robot to the start position, and perform routine maintenance for up to 10 minutes (or other limits set by the sponsor). The goal is to allow some interaction with the robot in order to continue the trial to completion. The toolset captured in the cache packaging tools picture and list is allowed with the robot at the start point. No spare parts are allowed (excluding commonly available supply items such as tape and cable ties). A Maintenance/Repair form is to be filled out to include the information on the test method, indication of failure, the remedy, tools used, and overall time to perform the maintenance or repair. The maintenance interaction may be captured on video as well to be used later for training or other purpose. This is intended to be a field maintenance procedure, so the robot is considered to be downrange with some limited number of tools and personnel. However, any person or team of people may interact with the robot at the start point but the robot may not be removed from the start point. The actual list of field maintenance tools necessary to keep the robot operational is evident after the testing is complete along with likely points of failure.

2.7 Abstentions from Test Methods

Each robot configuration should be tested in all applicable test methods and may attempt each test as many times as necessary to attain a satisfactory result. Robots may abstain (through the developer's designee) from a particular test method when considered not applicable or choose not to release the resulting data from a specific test trial when considered not successful. This encourages robot developers to attempt test methods and learn about their systems. In either instance, the page is to be marked as "ABSTAINED" to indicate that the test method was available at test time and the manufacturer acknowledges the omission of performance data. Although some robot implementations may not be designed or equipped for particular test methods, (e.g., robots without manipulators in the manipulator test methods) this testing methodology makes no assumptions regarding capabilities. Specifics of particular robot configurations should be considered when the robot has abstained from a given test method. If the test method is considered critical to the operational needs of the sponsor or user, the test should be considered failed until the robot can demonstrate satisfactory performance at a later date.

If a robot returns to the test facility at a later date to quantify improvements in capability for a particular robot configuration, the robot is to be subjected to a subset of tests representing each of the test method suites. For example: Energy/Power: Endurance; Radio Comms: LOS & NLOS; Mobility Terrains: Crossing Ramps; Mobility Obstacles: Inclined Plane; Sensors: Pan-Tilt-Zoom Tasks, Human-Robot Interaction: Random Maze Navigation.

3. RESULTS

The event was conducted according to the schedule and was actively participated by all who registered. Over 150 test trials were conducted with the results captured to support the respective purposes. The following elaborates the results in detail.

3.1 Participation

An evaluation event like this presented a unique environment where participants with different roles integrated to evolve the technology and test methods for emergency response robots. The following are the composition of the participation (See Figure 2 and Figure 3 for group pictures):

(A) DHS Sponsor
The project sponsor is onsite to provide guidance.

(B) Robots
There were over 25 robot configurations (i.e., some particular robot models brought multiple units to the event) or robotic special tools participated, which can be categorized as:

- Over 20 ground robots or robotic special tools
- Two aerial robots Small Unmanned Aerial Vehicles (sUAS) (FAA ARC Group I under 2 kg, 30 knots, frangible)
- Four aquatic robots, or customarily called ROVs
- Over 22 robots or robotic special tools from the U.S. and 4 from overseas
- Over 21 commercially available and 5 from research organizations

Note that a robot might belong to multiple categories.

(C) Participating Emergency Responders
15 FEMA US&R members, representing the following teams: California (CA)-TF1, CA-TF2, CA-TF3, CA-TF6, Colorado (CO)-TF1, Florida (FL)-TF2, Indiana (IN)-TF1, New York (NY)-TF1, TX-TF1, Virginia (VA)-TF1, and Washington (WA)-TF1.

15 Bomb technicians, representing the following teams: Boca Raton, FL Police Department (Dept.), Chico, CA Police Dept., Florida State Fire Marshal's Office, Garland, TX Police Dept., Jacksonville, FL Sheriff's Office, Michigan State Police, Montgomery County, Maryland Fire/Explosive Investigations, New Jersey State Police Arson/Bomb Unit, Odessa, TX Police Dept., Sacramento, CA Sheriff Office, Santa Clara, CA Sheriff's Office, and Seattle, WA Police Arson/Bomb Squad.

(D) Research and Development, Test Method Design, Set Up, and Administration Personnel

- Representatives from Southwest Research Institute, USA and from Mitre Corp., USA – Mobility Test Methods
- Representatives from a USA robot company and from US Army Aberdeen Test Center, USA – Sensors Test Methods
- Representatives from Pennsylvania State University – Energy/power Endurance Test Method
- Representative from NIST, Boulder, Colorado site – Radio Communications Test Methods
- Representative from Jacobs University, Germany – Mapping Test Methods
- Representative from Nagaoka University of Technology, Japan – Safety and Environment Test Methods
- Representative from a robot company – Aquatic Test Methods
- Representative from Ryerson Univ., Canada – Aerial Test Methods
- Representative from Bureau of Procurement, Germany – General support and advice
- NIST team from the Gaithersburg, Maryland site – host of the event

(E) Site Support
Disaster City administration, TEEX, assigned a team to support the operation.

(F) General Audience
Many participated for general interests, representing various DOD organizations and other Government agencies, USA and International industries, and various research organizations.

3.2 Resulting Capabilities Compendium

Test results, over 150 sets, were organized for different purposes. They are used to support the repeatability analysis in the test method standards. They are also extremely valuable information for the emergency response communities. The following subsections describe these in detail.

3.2.1 Bar Charts

The graphical test forms associated with each test method provide an intuitive understanding of the robot's capabilities in order to facilitate side-by-side comparisons. However, there are dozens of test methods in the suite and users of the data benefit from comparisons across the entire class of robots. Bar charts such as those shown in Figures 5 through 8 help identify Best-In-Class robots in specific test methods, and allow initial identification of trade-offs for particular robot configurations. But once a search is narrowed to several robots, a detailed study of the associated performance data forms is recommended.

In Figure 5, each bar along the X axis clearly represents the robot's tested average speed in the continuous ramp test terrain. Figures 6 and 7 represent the robots' capabilities in the increasingly more difficult crossing ramp and stepfields terrains, respectively. For example, the leftmost robot's speeds were (10, 5 and 0) meters/minute from Figure 5 through 7. 0 means that the robot was not able to complete the test. Figure 8 shows the robots' combined test results in these three and all the other terrains as listed Section 2.2.2. Our goals of providing intuitive representations to facilitate capability identifications have been achieved through the illustrations of these charts.

3.2.2 Comparison and Trade-Off Software Tool

We are also developing a software tool called Response Robot Capabilities Compendium, which contains capability data from all robots that achieve statistically significance within the DHS-NIST-ASTM International Standard Test Methods for Response Robots (See Figure 9). Currently, NIST has conducted all the testing as part of the standards development process. Additional test facilities recently opened in San Antonio, Texas, Kobe, Japan, and Koblenz, Germany. They will start contributing tested robot capability data, soon. Yet additional organizations from various parts of the world are anticipated to interact with this effort and request our help to establish similar testing facilities, in the near future. Given the myriad combinations of robot sizes, weights, and capabilities, a software interface into the database is the best way to understand the implications of specifying certain attributes or performance thresholds. This interface allows the user to see which robots have demonstrated statistically significant performance for the highest priority capabilities necessary to perform their intended mission. They can quickly see the effects of specifying too stringent a requirement in any particular capability or attribute as the number of robots that have successfully demonstrated the specified combination are filtered. Backing off on the threshold for even one requirement can bring several more robots into consideration. So users quickly learn the trade-offs involved and what the state of the science can deliver with regard to the combination of attributes and capabilities they have in mind.

Figure 9 illustrates that the candidate robots were filtered through after a user identified the requirements in LOS, endurance, and stair traverse. The tool allows for setting up multiple levels of detail on the information display. Any confidential information will be activated or de-activated properly before the tool is delivered to a user.

4. SUMMARY AND FUTURE EFFORTS

The event was implemented as an integrated exercise environment. The emergency responders enjoyed great learning experience of the robotic capabilities. The robotic developers were provided great opportunities to exercise the robotic systems and to explore technology advancement opportunities. The test method developers and administrators were immersed in a great environment to evolve the test methods.

Overall, the event exercised our emphasis on repeatable and scalable testing and evaluation processes. The user communities

have already enjoyed successes in applying the results, including robot acquisition and responder proficiency training. We plan to continue expanding the scope of this process and methodology to include testing robots with various levels of autonomy, to further explore advanced robotic requirements, and to cover robots applying to additional domains.

COMPANY/PRODUCT DISCLAIMER

Certain trade names, products, or other types of identifying information might be used to facilitate communications. In no case does such an identification imply recommendation or endorsement by the NIST, nor does it imply that the names or products are necessarily the best available for the purpose. In addition, the data are presented to facilitate prototyping, validation, and standardizing the corresponding ASTM test methods undertaken by its E54.08.01 Robotics Task Group. In no case do the data and the associated representations imply any type of recommendation, endorsement, or judgment by NIST.

Figure 1: Mobility Test Station Crossing Ramps

Figure 2: Part of the Participants of the 2011 Event

Figure 3: Part of the Participants of the 2010 Event

Figure 4: Visual Target

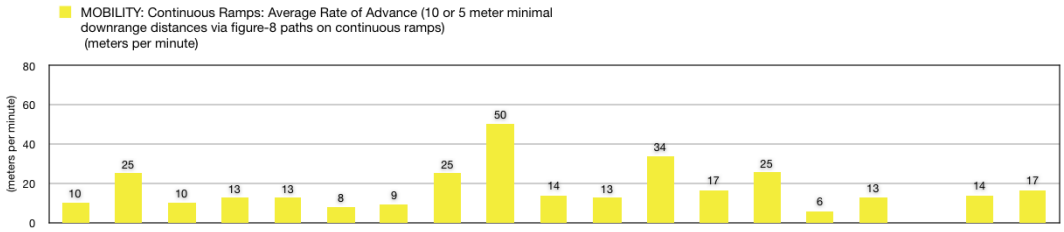


Figure 5: Continuous Ramps Terrain Test Results for Individual Robots

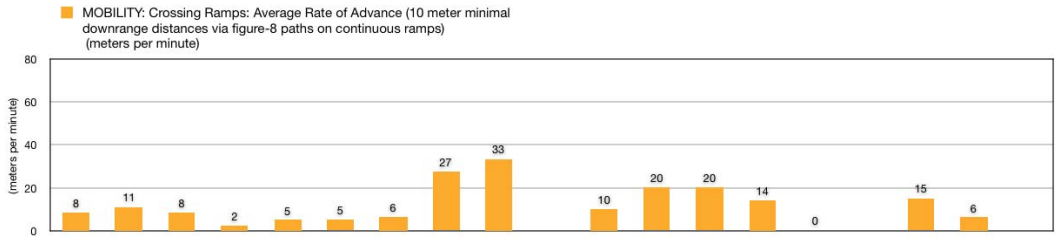


Figure 6: Crossing Ramps Terrain Test Results for Individual Robots

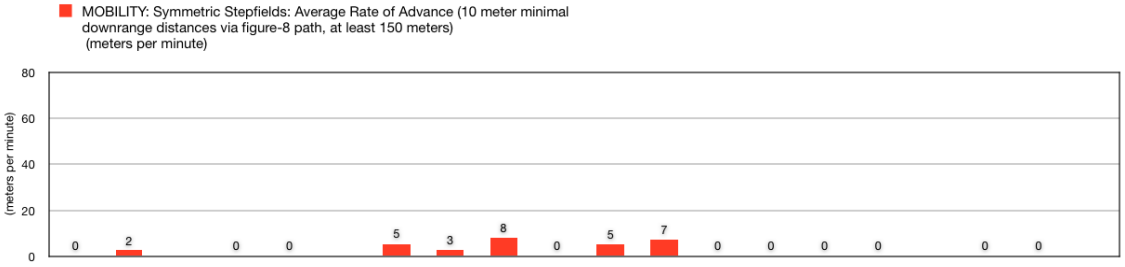


Figure 7: Stepfields Terrain Test Results for Individual Robots

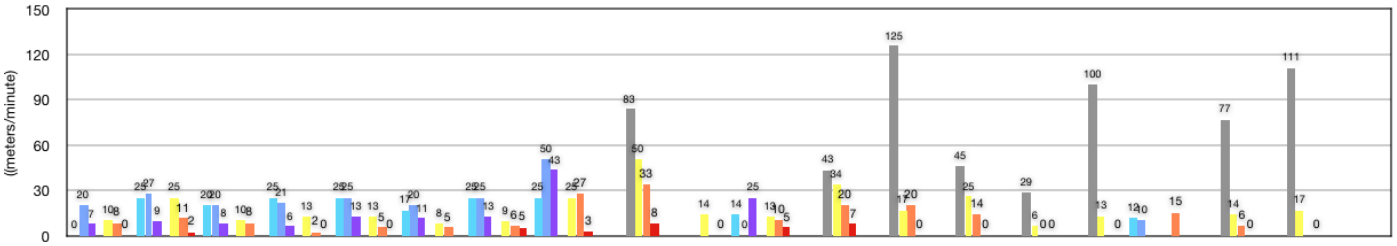


Figure 8: Combined Terrain Test Results for Individual Robots

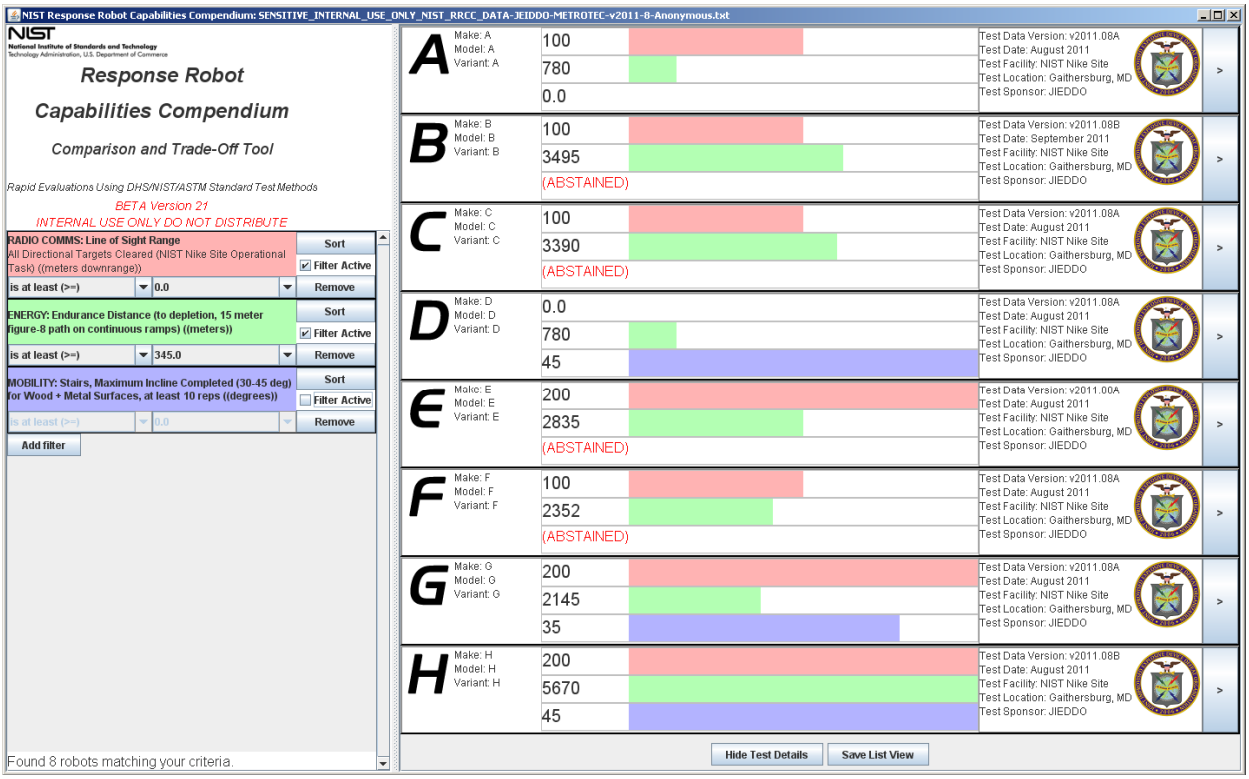


Figure 9: Compendium Illustration

REFERENCES

[1] Jacoff, A., et al., "Comprehensive Standard Test Suites for the Performance Evaluation of Mobile Robots," PerMIS Workshop, 2010

[2] Jacoff, A., and Messina, E, "Urban Search and Rescue Robot Performance Standards: Progress Update," SPIE Defense and Security Conference 2007.

[3] Messina, E. and Jacoff, A. S. "Measuring the Performance of Urban Search and Rescue Robots," IEEE Conference on Homeland Security Technologies, 2007

[4] ASTM International Standard E 2521 – 07a: Standard Terminology for Urban Search and Rescue Robotic Operations

[5] ASTM International Standard E 2592 – 07: Standard Practice for Evaluating Cache Packaged Weight and Volume of Robots for Urban Search and Rescue

[6] ASTM International Standard E2801: Standard Test Method for Evaluating Emergency Response Robot Capabilities: Mobility: Confined Area Obstacles: Gap

[7] ASTM International Standard E2802: Standard Test Method for Evaluating Emergency Response Robot Capabilities: Mobility: Confined Area Obstacles: Hurdles

[8] ASTM International Standard E2803: Standard Test Method for Evaluating Emergency Response Robot Capabilities: Mobility: Confined Area Obstacles: Incline Planes

[9] ASTM International Standard E2804: Standard Test Method for Evaluating Emergency Response Robot Capabilities: Mobility: Confined Area Obstacles: Stairs/Landings

[10] ASTM International Standard E2826: Standard Test Method for Evaluating Emergency Response Robot Capabilities: Mobility: Confined Area Terrains: Continuous Pitch/Roll Ramps

[11] ASTM International Standard E2827: Standard Test Method for Evaluating Emergency Response Robot Capabilities: Mobility: Confined Area Terrains: Crossing Pitch/Roll Ramps

[12] ASTM International Standard E2828: Standard Test Method for Evaluating Emergency Response Robot Capabilities: Mobility: Confined Area Terrains: Symmetric Stepfields

[13] ASTM International Standard E2829: Standard Test Method for Evaluating Emergency Response Robot Capabilities: Mobility: Maneuvering Tasks: Sustained Speed

[14] ASTM International Standard E2830: Standard Test Method for Evaluating Emergency Response Robot Capabilities: Mobility: Maneuvering Tasks: Towing: Grasped/Hitched Sleds

[15] ASTM International Standard E2854: Standard Test Method for Evaluating Emergency Response Robot Capabilities: Radio Communication: Line-of-Sight Range

[16] ASTM International Standard E2855: Standard Test Method for Evaluating Emergency Response Robot Capabilities: Radio Communication: Non-Line-of-Sight Range

[17] ASTM International Standard E2855: Standard Test Method for Evaluating Emergency Response Robot Capabilities: Human-System Interaction (HSI): Search Tasks: Random Mazes with Complex Terrain

[18] ASTM International Standard E2566 – 08: Standard Test Method for Determining Visual Acuity and Field of View of On-Board Video Systems for Teleoperation of Robots for Urban Search and Rescue Applications

[19] NIST Special Publication 1011-I-2.0 Autonomy Levels for Unmanned Systems (ALFUS) Framework Volume I: Terminology, Version 2.0, Huang, H., Ed., 2008 http://www.nist.gov/el/isd/ks/autonomy_levels.cfm

[20] NIST Special Publication 1011-II-1.0 Autonomy Levels for Unmanned Systems (ALFUS) Framework Volume II: Framework Models, Version 1.0, Huang, H., et al., Ed., 2007 http://www.nist.gov/el/isd/ks/autonomy_levels.cfm

Test Method for Measuring Station-Keeping With Unmanned Marine Vehicles Using Sonar or Optical Sensors

Asish Ghoshal
Computer Science & Eng
Texas A&M
College Station, TX 77843
1-979-845-8737
aghoshal@cse.tamu.edu

Avinash Parnandi
Computer Science & Eng
Texas A&M
College Station, TX 77843
1-979-845-8737
avinashparnandi@gmail.com

Robin R. Murphy
Computer Science & Eng
Texas A&M
College Station, TX 77843
1-979-845-8737
murphy@cse.tamu.edu

ABSTRACT

This paper proposes a test method for measuring the ability of a USV or ROV to fixate on an underwater object, i.e., station-keeping. Station-keeping is needed to permit an operator or domain expert to stay focused on an area in an image long enough to identify objects, such as submerged cars and debris, or a condition, such as scour eroding the underwater footing of a bridge. This problem is different from traditional robot control, as the point is not to measure the positions of the robot and sensor payload but rather how well the system maintains the position of the object in the image. The test method uses the Lucas-Kanade optical flow algorithm in OpenCV to track an inexpensive raised plywood and wire fiducial. The rotational, translational, and root mean square (RMS) error is measured over a 3 minute period as well as number of image frames in which the fiducial was not visible. The method was demonstrated using a DIDSON acoustic camera, but is generalizable to other types of sonars and underwater video cameras.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous; I.5.4 [**Computing Methodologies**]: Pattern Recognition—*Applications, Computer Vision*

General Terms

STANDARDIZATION

Keywords

unmanned marine vehicles, ROV, sonar

1. INTRODUCTION

Station keeping is the ability for an unmanned marine vehicle (UMV) to fixate on an object of interest. Two types of UMVs are designed specifically to work near underwater structures such as bridge footings: surface (USV) or tethered remotely operated underwater vehicles (ROVs). However, these two types of UMVs operate in a dynamic environment and are impacted by currents, winds, and waves. Unlike traditional robot control, station keeping considers how well the system maintains the position of the object in the image, not how well a UMV maintains a physical location. A UMV platform could maintain an accurate position in wind, waves, and currents while the sensor payload could react slowly and inaccurately, leading to a system that could not keep the object in view. A UMV with superior station keeping capabilities will be more valuable than one without; therefore it is useful to be able to quantitatively measure and compare station keeping performance. However, no standard test method exists for station keeping.

A standard test method for image-based station keeping should meet four criteria. It should

- *measure the translational and rotational error.* The method should measure both how well the UMV can keep the object of interest centered in the image (i.e., translational or displacement) and how well it can keep the same viewing angle (i.e. orientation).
- *be applicable to both sonar and video.* USVs and ROVs typically carry both imaging sonar to penetrate turbidity and video cameras, sometimes referred to as optical cameras, for visual inspection. A UMV may use either or both modalities for station keeping.
- *be inexpensive and easy to replicate.*
- *be automated.* The method should not require a human to manually compute or estimate the error.

This paper presents an imaging-based method for measuring station-keeping that meets the above criteria. The subsequent sections describes the related work, the approach, and the proposed test method followed by a demonstration of the test method for a DIDSON imaging sonar and discussion.

2. RELATED WORK

The related work in station-keeping for UMVs falls into two categories: station-keeping algorithms (either template matching or optic flow) and work measuring optical station-keeping performance using a fiducial. No papers were found

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
PerMIS'12 March 20-22, 2012, College Park, MD, USA
Copyright @2012 ACM 978-1-4503-1126-7-3/12/12 ...$10.00.

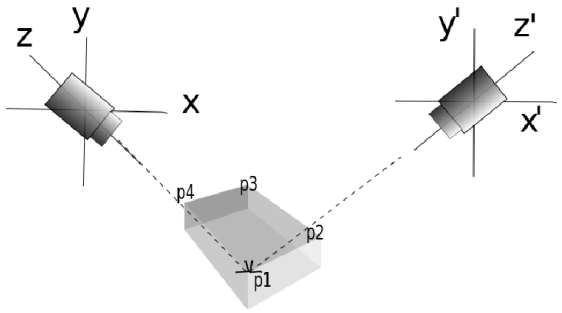


Figure 1: The frame of reference from the camera producing I and I'.

that discussed station-keeping with imaging sonar. This paper will use optic flow for determining movement of a fiducial in the image.

Four papers describe station-keeping using optical imaging, using downward facing cameras to stay focused on a particular patch of seabed. The systems construct a mosaic and then use either correlation or optic flow to drive the vehicle back to its desired relative location and orientation. [5] presented excellent results from sea trial experiments from near bottom optical station keeping. In their work 3D motion estimation was determined by constructing an image mosaic or map of the sea floor by using a downward pointing optical camera. Image mosaicing has also been used by [1] to present a region based matching method. While [2] used correlation techniques on sea bed imagery to determine correspondence between image sequences and limiting the the magnitude of motion by assuming an active controller. [3] incorporated optic flow to provide an initial estimate the motion parameters and presented a difference template matching method. They also incorporated learning to determine the appropriate motion model for the expected image deformations. Success in these cases was measured qualitatively.

A model based approach was presented in [4] in which a fiducial was deployed in the environment and 3D pose estimation was done using a forward facing optical camera. The approach relied on 3D reconstruction from the 2D planar image of the custom made fiducial which cannot be generalized to realistic conditions.

3. APPROACH

The approach taken to measuring how well an object stays centered in the image is to create a fiducial that can be seen by both an imaging sonar and optical camera and then measure the change in position and orientation of the object over a time interval. The measurement is done by computing the projective or affine transformation between features in the current image, $I^{'}$, and the initial image representing the desired position, I. Normally, this would mean computing the translation and all three Euler angles of rotation: roll(ψ), pitch(θ) and yaw(φ). However, as shown below, by assuming the UMV is at a constant depth and the camera motion is small, the transformation reduces to computing the difference in translation d and yaw φ.

3.1 Computation of Image Transformation

As shown in the figure 1 for each image the camera takes of the object, the pose of the object with respect to the camera coordinate system can be described in terms of a translation $\vec{t}$ and a rotation $\mathbf{R}$, where the rotation matrix is given by $\mathbf{R} = R_z(\psi)R_y(\varphi)R_x(\theta)$. Thus points on an object $P_i([X_i\ Y_i\ Z_i]^T)$ is related to that of points on the image plane $P_i^{'}([x_i\ y_i]^T)$ by (1)

$$P_i^{'} = \mathbf{R}(P_i - T) \tag{1}$$

So station keeping can be achieved by computing the roll(ψ), pitch(θ) and yaw(φ) angles and the translation vector which can then be fed to a closed loop controller to cancel the movement of the camera relative to the object. Such a transformation which maps points on a 2 dimensional planar surface to that of the image plane of the camera is called homogrpahy and is given by

$$\tilde{P}_i^{'} = sH\tilde{P}_i \tag{2}$$

Where $\tilde{P}_i^{'}$ and $\tilde{P}_i$ are in homogeneous coordinates and s denotes the scale factor upto which the homography is defined. But if the camera remains at a constant height with respect to the object and the camera motion is small from frame to frame then an affine transformation model can be used to model such transformations. So using an affine transformation model (2) can be written as

$$\tilde{P}_i^{'} = H^{'}\tilde{P}_i \tag{3}$$

where $H^{'}$ is given by

$$H^{'} = \begin{bmatrix} s\cos\varphi & -s\sin\varphi & t_x \\ s\sin\varphi & s\cos\varphi & t_y \\ 0 & 0 & 1 \end{bmatrix} \tag{4}$$

Since, the USV floats at a constant depth and the field of view of imaging sonars and optical cameras is small, viz. 30° [6] the above assumptions are justified.

However, it should be noted that the image obtained from an acoustic camera is highly sensitive to the camera's position. [6] has developed a model of a common imaging sonar, the DIDSON acoustic camera, which can be used to track the object of interest in camera coordinates. But, fortunately for small movements the projective distortions introduced by the acoustic camera can be assumed to remain same between the reference image and the current image. And since this method is only interested in finding the translational and rotational error in image coordinates, such an approximation is justified.

3.2 Choice of Fiducial and Features

The fiducial (shown in Fig. 2) is a raised rectangular surface which presents easily extractable corners in both imaging sonars and optic cameras. The rectangular surface provides sharp corners which are easy to detect and track with computer vision algorithms. The challenge for building a fiducial was to make one that was unambiguously visible to the imaging sonar. The corners are found using optic flow to bootstrap extraction of the polygon in the image to subpixel accuracy.

The fiducial was built out of 2'x1' plywood mounted on a metal wire mesh crab trap to raise the surface enough to provide a small grazing angle for imaging sonars. This fiducial provided only one surface that was visible to a sonar or

optical camera and provided discernible edges in the image. The rough surfaces such as plywood produce brighter images in an acoustic camera because of backscattering, while the spare thin wires of the crab trap were too thin for a sonar to see. Likewise the solid plywood surface is easy for a optical camera to detect as compared to thin wires. Raising the plywood surface provided the small grazing angle needed for an imaging sonar.

4. TEST METHOD

The proposed test method consists of four steps: initialization, nominal execution, termination, and exceptions. The implementation of the aforementioned procedure has been done in OpenCV.

Initialization. The proctor is given the recorded sensor output and bootstraps the tracking algorithm by selecting the rectangular face of the fiducial, in the initial image, I. This becomes the region of interest. The region of interest at any time represents the certainty with which the position of the fiducial is known in the image sequence.

Nominal execution. The noise in each image is subtracted by downscaling and upscaling the image in a Gaussian pyramid decomposition. The algorithm then tries to isolate the fiducial by finding edges using various thresholding levels and applying edge detection filters like Canny. Then the contour of the polygon is determined which is then approximated to a polygon using Douglas-Peucker approximation. Finally, the polygon is validated and after the polygon has been correctly validated the corners of the rectangle are tracked using pyramidal Lucas Kanade optic flow algorithm. The points returned by the Lucas Kanade algorithm are corrected at a determined timestep using the aforementioned procedure applied only to the established region of interest. The affine matrix is determined from the corresponding points in the reference image and the current image from which the angle of rotation φ and distance moved by the centroid $d = \sqrt{t_x^2 + t_y^2}$ is determined. The error statistics are overlaid on each image.

Termination. The image sequence ends after a set time (3 minutes, though this was arbitrary and could be changed). During the 3 minute interval the vehicle could be subjected to disturbances to simulate wind, waves, and currents. The minimum, maximum and root mean squared value of the error parameters φ and d are determined over the time window.

Exceptions. If the fiducial is lost, the region of interest is iteratively increased to relocate the fiducial by using the aforementioned procedures. The number of images where the image is lost can also be counted as a metric.

5. DEMONSTRATION WITH DIDSON IMAGING SONAR

The proposed test method was tested by capturing videos from a DIDSON acoustic camera with the fiducial in a swimming pool, generating the station keeping score, and then manually computing the object movement to measure the correctness of the autonomously generated score. The effort focused on determining if the method would work with sonar because it was assumed that it would work for optical cameras, given the maturity of object tracking for video.

The swimming pool measured 35' by 17' and the fiducial was placed at a depth of 5'. The acoustic camera's angle was kept so as to get the best view of the rectangular face of the fiducial upon initiating station keeping. The acoustic camera used was DIDSON 300 having a frequency of 1.1 MHz and a range of 30 meters. The algorithm was run offline on the captured video on a system running an Intel U4100 1.30 GHz processor and 2 GB of random access physical memory. The accuracy of the scoring algorithm was measured in pixel space by comparing the values returned by the algorithm to the values computed by annotating the video manually.

Figure 2: Demonstration setup showing the DIDSON imaging sonar mounted on the pool and the fiducial in the pool.

Two sets of trials were conducted; in the first set of trials the fiducial was moved with respect to the DIDSON by keeping the angle of incidence of the DIDSON constant. In the second set of trials the fiducial was fixed while the DIDSON was moved around the object by keeping the tilt angle of the DIDSON fixed. The fiducial was always in view of the DIDSON. For convenience, the DIDSON was manually mounted on the side of the pool rather than attached to a USV or ROV.

The tracking performance of the algorithm for a trial is shown in the graph. The yaw angle φ and the displacement d of the centroid returned by the algorithm is compared against the actual angle and displacement by manually annotating the video every 200 frames at 25 frames per second. The metric used was minimum, maximum and RMS deviation. Figure 3 shows the output of a typical trial in which the object was moved relative to the DIDSON. Figure 4 and 5 show the computed yaw angles and displacement with respect to ground truth in 1 minute windows.

6. DISCUSSION

The graphs show that the proposed method was generally able to track the object from an acoustic camera and extract the yaw angle and displacement. The disparities between the computed transform error and the manually computed error appeared to be due to drawbacks of imaging sonar: the images are highly sensitive to changes in the relative angle between the sensor and fiducial. This suggests that this method for measuring station-keeping ability would not work in an open water test bed versus a pool due to too much movement.

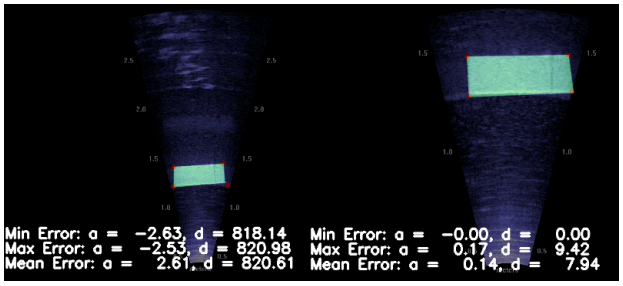


Figure 3: Sample output from two trials (side by side) showing the instantaneous computation of error.

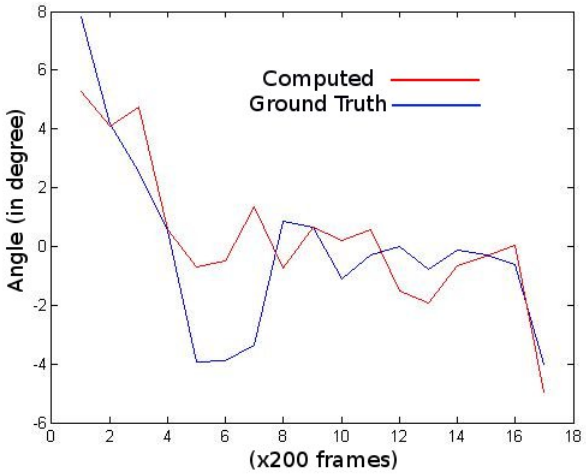


Figure 4: Yaw Angle

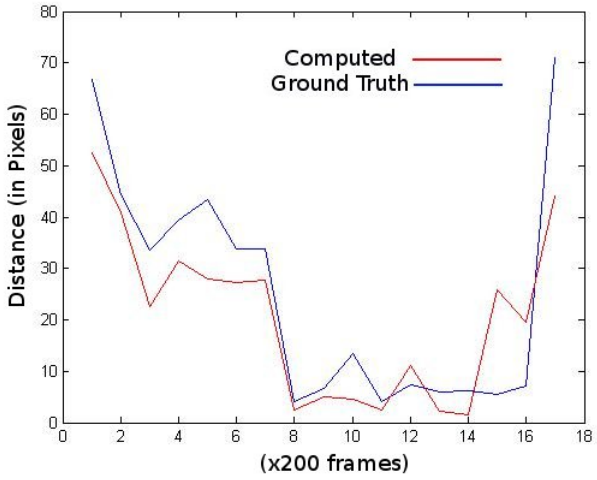


Figure 5: Displacement

The fiducial could be changed or improved. The size of the fiducial was determined empirically to give a good reflection at 5 to 30' of water; the size could be changed to present the desired size of surface for the depth of tests. The proposed method uses a symmetric (rectangular) fiducial which could create measurement problems if the fiducial is lost and the UMV approaches it from a different orientation. This could be handled with creating an irregular polygon.

The feature extraction algorithm could be changed to used the shadow rather than the surface returned by the sonar. The algorithm tracked the reflective surface of the object of interest, which provides sharp images for tracking but is limited by the field of view and is quite unstable. However, images obtained from DIDSON have shadows that are much more stable than the prime image obtained from reflection of sound waves [6]. Further, shadows are produced by all objects irrespective of orientation and shape and provide important cues for image recognition. Thus instead of sensing only the highlighted area, cues can be obtained from shadows which are present even when the highlighted area disappears. So from the shadows and the highlights in a DIDSON image, the 3D view of the object can be reconstructed thereby providing accurate object recognition and subsequently determining yaw and displacement from it.

7. CONCLUSION

This paper presented a proposed test method for station-keeping to maintain an object in the center of the image. The method from measuring the error in station-keeping, or amount of movement of the object in the image, is derived from computer vision and addresses the specific challenges of imaging sonars. The proposed method meets the four criteria for a test method for station-keeping. One, it measures the translational and rotational error (yaw), Two, the fiducial is visible to both sonar and video (assuming water clarity), though the fiducial and method were demonstrated only with imaging sonar which is the more challenging. Three, the fiducial is made from inexpensive, easy to obtain materials. Four, the method is fully automated: the method computes the error (minimum, maximum, RMS of the translational and rotational error; number of times out of view) by applying a computer-vision tracking algorithm to the recorded output. However, the performance of the autonomous scoring algorithm was lower than expected under benign conditions and more work is needed to improve the fiducial and fine-tune the algorithm.

8. ACKNOWLEDGMENTS

This work was an outcome of National Science Foundation Grant OISE-1029089 "NSF-JST-NIST Workshop on Rescue Robotics" and was supported in part by CNS-0923203 "MRI Acquisition of a Mobile, Distributed Instrument for Response Research (RESPOND-R)." Any opinions, findings, conclusions or recommendations expressed in this paper are those of the author and do not necessarily reflect the views of the National Science Foundation. The author would like to especially thank Sean Newsome for his helpful feedback as well as the anonymous reviewers.

9. REFERENCES

[1] X. Cufi, R. Garcia, and P. Ridao. An approach to vision-based station keeping for an unmanned underwater vehicle. In *Intelligent Robots and Systems, 2002. IEEE/RSJ International Conference on*, volume 1, pages 799 – 804 vol.1, 2002.

[2] R. Marks, H. Wang, M. Lee, and S. Rock. Automatic Visual Station Keeping of an Underwater Robot. In *Proc. IEEE Oceans 94 Osates*, Brest, France, 1994.

[3] S. van der Zwaan and J. Santos-Victor. Real-time vision-based station keeping for underwater robots. In *OCEANS, 2001. MTS/IEEE Conference and Exhibition*, volume 2, pages 1058 –1065 vol.2, 2001.

[4] Q. Wu, S. Li, Y. Hao, and F. Zhu. A model-based monocular vision system for station keeping of an underwater vehicle. In *Robotics and Biomimetics (ROBIO). 2005 IEEE International Conference on*, pages 450 –454, 2005.

[5] X. Xu and S. Negahdaripour. Automatic optical station keeping and navigation of an rov; sea trial experiments. In *OCEANS '99 MTS/IEEE. Riding the Crest into the 21st Century*, volume 1, pages 71 –76 vol.1, 1999.

[6] S.-C. Yu, T.-W. Kim, A. Asada, S. Weatherwax, B. Collins, and J. Yuh. Development of high-resolution acoustic camera based real-time object recognition system by using autonomous underwater vehicles. In *OCEANS 2006*, pages 1 –6, 2006.

Standard Test Procedures and Metrics Development for Automated Guided Vehicle Safety Standards

Roger Bostelman, Will Shackleford, Geraldine Cheok, Richard Norcross
National Institute of Standards and Technology
100 Bureau Drive, Stop 8230
Gaithersburg, MD 20899
(301) 975-3426

roger.bostelman@nist.gov

Abstract
The National Institute of Standards and Technology's Intelligent Systems Division has been researching automated guided vehicle (AGV) control based on advanced two-dimensional (2D) imaging sensors that detect dynamic, standard test pieces representing humans towards improving AGV safety standards. Experiments and results are presented in this paper showing the measurement of dynamic standard test pieces from an automated guided vehicle as compared to ground truth. The experimental results will be used to develop standard test methods and to recommend improved standard stopping distance exception language to AGV standards.

Categories and Subject Descriptors
B.7.0 [Advanced]; C.2.1 [Sensor Networks]

General Terms
Measurement, Performance, Design, Algorithms, Experimentation, Verification.

Keywords
2D/3D imagers, AGV, ANSI/ITSDF B56.5, ground truth

1 Introduction
The Mobile Autonomous Vehicles for Manufacturing (MAVM) Project at the National Institute of Standards and Technology (NIST) is evaluating the performance of advanced sensors as compared to a laser detection and ranging (LADAR) sensor typically used in industry and to ground truth. The American National Standards Institute/Industrial Truck Standards Development Foundation (ANSI/ITSDF) B56.5-2005 Safety Standard for Guided Industrial Vehicles and Automated Functions of Manned Industrial Vehicles "defines the safety requirements relating to the elements of design, operation, and maintenance of powered, not mechanically restrained, unmanned automatic guided industrial vehicles and automated functions of manned industrial vehicles." [1]

This paper is authored by employees of the United States Government and is in the public domain. PerMIS'12, March 20-22, 2012, College Park, MD, USA. ACM 978-1-4503-1126-7-3/22/12

NIST recently suggested improvements to the standard including a new test piece, test piece coatings, and non-contact sensor and vehicle performance requirements when detecting static test pieces in the vehicle path. This standard has recently passed ballot at the main committee level. However, the legacy standard still includes an exception for less than the minimum AGV stopping distance. The exception states: "Although the vehicle braking system may be performing correctly and as designed, it cannot be expected to function as designed and specified in para 4.3.1 should an object suddenly appear in the path of the vehicle and within the designed safe stopping distance. Examples include, but are not limited to, an object falling from overhead or a pedestrian stepping into the path of a vehicle at the last instant." Safe stopping distance refers to the distance the AGV travels after a stop command is given and before the AGV contacts an obstacle.

Therefore, the MAVM Project is now performing the second phase of experiments for the ANSI/ITSDF B56.5 standard for dynamic test pieces to once again develop safety standard procedures and metrics. Improved standard language to limit the exception is expected to evolve from the NIST experiments and include discussion of vehicle energy reduction. Initially, NIST must develop an understanding of the typical safety sensor and AGV control characteristics including how accurately the stop function reacts to standard obstacles entering the AGV path. The objectives of the second phase experiments were to:

- Dynamically position a standard test piece in the path of an AGV within the AGV stopping distance, and
- Compare the standard test piece detection point, dynamic test piece path and dynamic AGV path as measured on the vehicle to ground truth to establish a basis for standard test method development

This paper describes the second phase of the AGV experiments and test setup and presents some preliminary results and conclusions. The experimental results will be used to help develop further tests and standard test methods for inclusion in AGV standards, as well as to

develop improved standard stopping distance exception language.

2 Experimental Setup

The parameters investigated in the experiments included the type of test piece, the type of AGV stop (with controlled or e-stop braking), the speeds of the AGV and test piece, the path of the test piece relative to the AGV path, and operation in confined vs. open space. These parameters will be discussed in more detail in this section.

The experimental setup is graphically shown in Figures 1 and 2. Figure 1 shows the basic experimental motion system. The tape switch in Figure 1 triggered the sled motion. Figure 2 shows the test layout with labels describing: the AGV path, the perpendicular and angled test piece paths, sensor locations, and example time intervals showing the test piece crossing the AGV path within the maximum vehicle safe stopping distance.

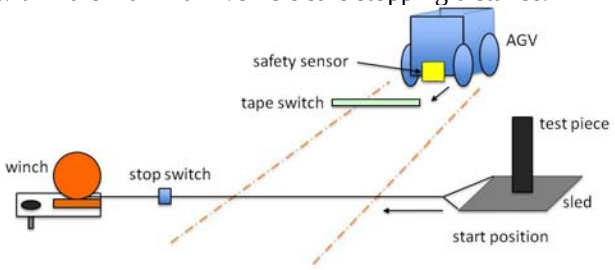


Figure 1 – Test setup showing the AGV, path and test piece sled.

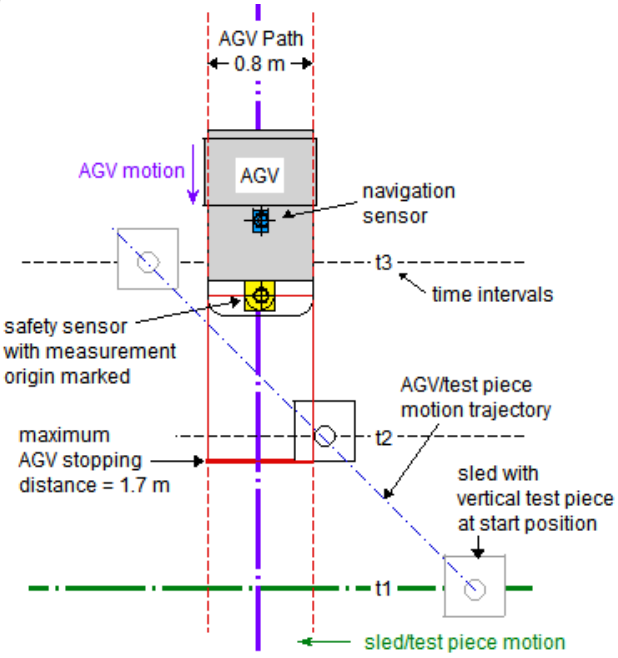


Figure 2 – Top view of test layout with labels describing: the AGV and test piece paths and sensor locations, and example time intervals (t1, t2 and t3) showing the test piece crossing the AGV path within the maximum vehicle stopping distance.

The automated guided vehicle (AGV) used in experiments was equipped with a NIST-built controller, based on the Mobility Open Architecture Simulation and Tools (MOAST) [2] control scheme.

Several sensors were mounted on the NIST AGV which was programmed to move in a straight line to a chosen navigational point. Both two- and three-dimensional (2D and 3D) sensors were used to collect data, including: a color camera, an infrared camera, two different types of 3D light detection and ranging (LIDAR) sensors and a 2D line-scan LADAR sensor. The safety sensor referred to in the discussion below is a 2D LADAR mounted to scan horizontally at a height of 10 cm above the floor. It is a sensor typically used in industry as a non-contact safety sensor for AGVs. The safety sensor's range measurement origin is approximately 70 mm behind the AGV's front foam-on-metal bumper. The data from the 3D imaging sensors will be used in future efforts to research their effectiveness in detecting obstacles, especially overhanging obstacles. For the experiments presented in this paper, the safety sensor data, collected simultaneously with the 3D sensor data, was used for dynamic obstacle detection and for AGV control. The safety sensor was used to detect ground-based obstacles and will later be used as ground truth for the other onboard sensors.

B56.5 states: "The determination of the vehicle's stopping distance … depends on many factors, such as other vehicle and pedestrian traffic, clearances, condition of the floor, and the stability and retention requirements of load(s). The prime consideration is that the braking system in conjunction with the object detection system and the response time of the safety control system shall cause the vehicle to stop prior to impact between the vehicle" and obstacles. Two main types of 'AGV stop' control tests, as described in ANSI/ITSDF B56.5, were performed: controlled braking and low-level emergency stop (e-stop) control.

B56.5 states: "Controlled braking may be provided. Controlled braking is a means for an orderly slowing or stopping of the vehicle." Controlled braking was used to demonstrate continuous AGV control to reduce AGV energy upon detection of an obstacle within the programmed AGV path and at any range detectable from the safety sensor. For example, using controlled braking, the AGV is under continuous control to decelerate to avoid contact with the test piece or other obstacles in the path. The low-level control function is bypassed to consider the effects of only controlled braking during controlled braking tests.

Low-level emergency stop (e-stop) control is required by the standard and is also a function of the NIST AGV which integrates the safety sensor directly into the AGV drive amplifiers. The safety sensor is typically programmed with slow and stop fields. For our tests, only the stop field was programmed and used.

When the safety sensor detected an obstacle in the stop field, the sensor caused the amplifier to be inhibited and the vehicle coasted a maximum distance of 1.7 m (if it was moving at its full speed of 1 m/s), i.e., no additional braking was provided since only stop control was being tested and braking may vary across AGVs due to size, weight, payload, etc. An additional function for the low-level e-stop control is that a timestamp is broadcast through a hardware device to the ground truth system at the instant an obstacle enters the safety sensor stop field. This may prove useful in future analysis.

A thin sled, shown in Figure 3, was designed and built so that it would not be detected by the safety sensor. A modular, laser-based measurement system with 0.01 mm accuracy was used to measure ground truth of the dynamic sled and AGV positions. Eight optical fanning laser transmitters surrounded the AGV/sled test area. Ground truth system receivers were mounted on 1.3 m high posts behind the test piece mounted to a sled Also, two ground truth receivers were mounted to the AGV. (see Figures 3 and 4).

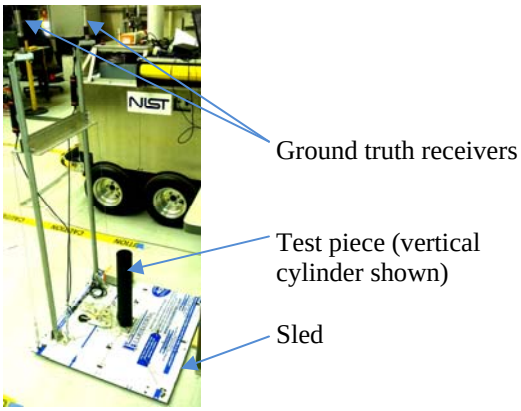


Figure 3 – Sled configuration shown with vertical cylinder test piece.

The sled base measured 64 cm^2 and was made of corrugated plastic between thin aluminum sheets and mounted onto 1.3 cm high teflon strips with their longitudinal axes parallel to the sled motion. The sled was pulled using a winch that began motion when the AGV tripped a tape switch on the floor. Interchangeable test pieces were fixtured to the sled with screws aligning the test piece vertical axis with the sled center point. Test pieces were mounted to the sled so that they entered the AGV path prior to the ground truth sensor posts entering the path.

The tape switch positions were chosen so that the test piece entered and passed through the programmed safety sensor stop zone before the AGV could strike the sled components. The stop zone, used for the low-level, e-stop tests, measured 2 m along the AGV path and 1.3 m perpendicular to the AGV path. The stop zone, used for the controlled braking tests, measured the maximum sensing range along the AGV path by 0.8 m wide. While the sensing range during run-time had no maximum, a 2 m limit was enforced by post-processing. The NIST AGV base measured 0.8 m wide, which sufficed for our tests. However, a roller table extended beyond the 0.8 m width by 0.1 m. This parameter will be included in our future 3D measurement tests and analysisThe AGV had a maximum stopping distance of 1.7 m.

Open and confined spaces (see Figure 4) were another parameter in the NIST experiments. B56.5 states that AGV areas with clearance less than 0.5 m are deemed hazardous and vehicle speed must be reduced.

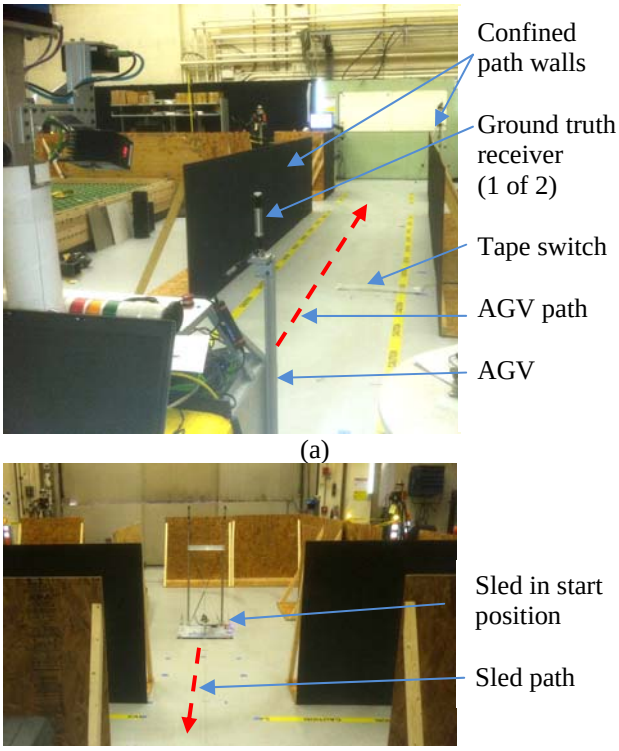


Figure 4 – Confined test course layout showing (a) the AGV paths and (b) the flat plate mounted on the test piece sled and ready to cross the AGV path. The flat plate is mounted with its ½ m square surface facing the AGV path.

Walls representing confined spaces or “hazard zones” were placed 0.15 m beyond the 1.3 m wide stop field making the distance between the walls a total of 1.6 m. Therefore, for confined space tests, the AGV was programmed with a velocity of 0.5 m/s instead of the open-space velocity of 1 m/s. The confined test course was converted to open space by simply removing the black walls and using the same AGV path. The sled path for the confined space tests was perpendicular to the AGV path as shown in Figure 4 (b).

Two cylindrical test pieces were used, as specified in the standard. A vertical cylinder 70 mm in diameter by 400 mm long represented the lower portion of a human leg. A horizontal cylinder 200 mm in diameter by 600 mm long represented the profile of a person lying down. A 0.5 m square flat plate was also used to represent flat, highly reflective materials in a manufacturing environment. The flat plate is part of the draft ANSI/ITSDF B56.5 standard. The cylindrical test pieces were coated with flat black paintwith a 4.6 % reflectivity, measured using a reflectance meter, which is below the maximum 6 % reflectivity allowable by the draft ANSI/ITSDF B56.5 standard. The walls in the confined section of the test course were painted with the same flat black paint to increase the probability of not detecting targets thereby making this a more severe condition.

The ground truth system was initially calibrated and each test piece was modeled so that data for the horizontal cylinder and flat plate reflected the first point that entered the AGV's safety sensor stop field, as shown in Figure 5. However, the vertical cylinder test piece ground truth reference point was on the central axis of the cylinder at mid-height. Since the radius of the cylinder is known, the point on the cylinder detected by the safety sensor can be determined.

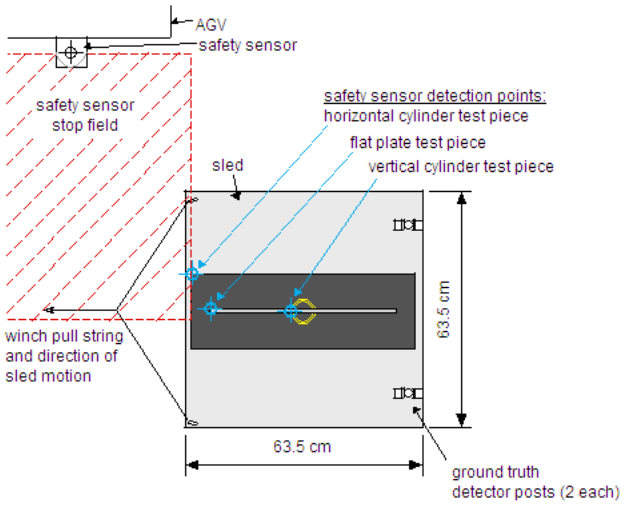


Figure 5 – Top view of initial safety sensor detection points for each test piece as they entered the moving safety sensor stop zone. Only one test piece was mounted on the sled for each test. The vertical cylinder test piece ground truth reference point was at the part center while the other two test pieces matched the points in the figure.

3 Data Collection and Software

The software developed to capture sensor data used the Mobility Open Architecture Simulation and Tools (MOAST) framework [2]. Software was developed using the C++ language for real-time AGV control and for data-collection. Java software was developed for real-time visualization and offline analysis. The Neutral Messaging Language (NML), part of the Real -Time Control Systems (RCS) Library, was used for storing and communicating the data [3]. Three computers were required for data-collection due to bandwidth limitations on each computer. One computer collected data from a high-resolution 3D imager. A second computer collected data from three low-resolution 3D imagers. A third computer controlled the AGV and collected data from the safety sensor, AGV wheel encoders, and navigation system. The Network Time Protocol (NTP) was used to synchronize the clocks on the three computers. [4]

The data collection software also controlled a wireless hardware device used for time synchronization with the ground-truth system. Ground truth was continually collected throughout each test. It was also time-stamped to record detector positions when the hardware device broadcast that the test piece was initially detected, i.e., first entered the stop zone. This information was used to correlate the safety sensor and ground truth detection locations.

4 Experimental Results

Twenty-five tests were completed using the draft ANSI/ITSDF B56.5 standard test pieces. Table 1 shows the tests that were performed (un-shaded cells) and tests that were not performed (shaded cells). The three tests that were not performed were:

- The static AGV (0 m/s) tests. These were performed in previous NIST experiments [6] and led to the initial B56.5 standard changes recently balloted and approved.
- Test piece orientation that is in-line with the AGV path. This test would simulate detection of the edge of the flat plate or horizontal cylinder. The researchers followed the B56.5 safety standard which includes only test methods for test pieces perpendicular to or at a 45° angle to the AGV path.
- A test piece that moves parallel to and towards the AGV in the same lane. This test has not been designed and may require breakaway test pieces and ground truth system components to ensure safety.

To closely model the in-lane tests not performed (explained in the last bullet above), the AGV safety sensor's stop field was programmed to be 1 m wide and within a passing AGV lane. The stop field, in this case, is a programmed safety sensor field that extends into the adjacent lane to sense when a passing test piece is detected. The safety sensor successfully detected the test piece as it paralleled the AGV in an adjacent lane. Results are shown in Tests 23-28.

Experimental results are summarized in Table 2 under the following column headings: test number, AGV velocity, test piece type, safety sensor to test piece range along the AGV path when the test piece first entered the

AGV path, and the difference in test piece location. The last column represents the distance between the location of the test piece as measured by the safety sensor (previous column) and the location as measured by the ground truth instrument. Potential sources of error potentially causing large distance differences are discussed later in this section.

Table 1 – Dynamic experiments performed.

test space	AGV control	AGV velocity	test piece	test piece orientation	test piece movement	test piece velocity	test piece/ AGV sep dist
open	controlled braking	0 - min. speed	vertical cylinder	perp. to path	perp. to and across the path	static	within 2 m
confined	low-level e-stop	50% - confined space speed	horizontal cylinder	45° to path	diag. to and across the path	0.5 mps (slow speed)	beyond 2 m
		100% - open space speed	flat plate	parallel to and in-line with path	parallel to and beside the path	1.0 mps (fast speed)	
					parallel to and in the path		

Seven of the tests listed in Table 2 included the low-level, e-stop control. These tests were used to demonstrate that low-level, e-stop control can reduce the AGV's kinetic energy within its maximum stopping distance and can also control an AGV stop. However, the stop position always occurred beyond the test piece path indicating that a stopped test piece in the path and entering the path within the maximum AGV stopping distance would have been struck. Eighteen controlled braking tests were also performed and demonstrated that once the test piece entered the AGV path within the maximum stopping distance, the AGV could decelerate to a stop without striking a stationary obstacle in the AGV path. For most tests, the tape switch was positioned to allow the test piece to exit the AGV path prior to potential contact and so the AGV slowed to a near stop in the test piece path. After a pause, the AGV began to accelerate again as there were no obstacles in its path. Tests 1, 2, and 12 demonstrated that the vehicle stopped prior to contact with the test piece while the AGV was in both controlled braking and low-level e-stop control modes. During a few tests, not listed in Table 2, the test piece stopped in the path or was struck by the vehicle. To avoid damage to the equipment and sensors, the researchers decided not to stop the test piece in the test path until an experimental setup for this case can be designed and implemented.

Table 2 – Experimental results of safety sensor range uncertainty. Abbreviated column information is as follows:

- *4th column: C = controlled braking and E = low-level, e-stop controlled.*
- *5th column: V = vertical cylinder, H = horizontal cylinder, FP = flat plate, X = 102 mm diameter cylinder.*
- *7th column: the difference in the location where the safety sensor measures the test piece and where the ground truth measures the test piece. This difference is measured along a line parallel to the AGV path.*

Test	Space	AGV/Test Piece Vel (m/s)	Control C or E	Test Piece	Sled Path	Safety Sensor-to-Test Piece Measured Distance (mm)	Diff. in Test Piece Location (mm)
1	Open	1 / 0	E	V	Static	1,702	147
2	"	1 / 0	C	V	Static	1,774	121
3	"	1 / 1	C	V	Perp.	1,205	116
4	"	1 / 1	E	V	Perp.	393	380
6	"	1 / 1	C	H	Perp.	1,158	102
7	"	1 / 1	E	H	Perp.	1,115	114
9	"	1 / 1	E	FP	Perp.	1,005	101
10	"	1 / 1	C	FP	Perp.	772	75
11	Confined	0.5 / 0.5	C	FP	Perp.	490	91
12	"	0.5 / 0	C	FP	Static	517	234
14	"	0.5 / 0.5	E	V	Perp.	460	784
15	"	0.5 / 0.5	C	V	Perp.	506	663
16	"	0.5 / 0.5	E	X	Perp.	380	842
17	"	0.5 / 0.5	E	H	Perp.	527	711
18	"	0.5 / 0.5	C	H	Perp.	1,030	2,240
19	Open	1 / 1	C	V	Angle	1,197	715
20	"	1 / 1	C	V	Angle	1,291	662
21	"	1 / 1	C	FP	Angle	1,409	728
22	"	1 / 1	C	FP	Angle	1,017	747
23	"	1 / 1	C	FP	Parallel	1,556	(281)
24	"	1 / 1	C	FP	Parallel	1,777	(329)
25	"	0.5 / 0.5	C	FP	Parallel	465	(96)
26	"	1 / 1	C	V	Parallel	1,820	(19)
27	"	1 / 1	C	V	Parallel	1,663	43
28	"	1 / 1	C	V	Parallel	506	(25)

Some post-processing was required to determine the safety sensor-to-test piece measured distance reported in the seventh column of Table 2. Post-processing was necessary because the range sensor provides the distance and angle to the test piece, instead of the test piece position when entering the programmed stop zone.

The AGV positions AGV_i (AGV position at time i) and AGV_{i-1} (previous AGV position before time i) are stored in a file and searched based on the timestamp of the obstacle point in another file. Distance was calculated as:

$$\text{distance} = \frac{(AGV_i - AGV_{i-1}) \bullet (TP - AGV_i)}{(AGV_i - AGV_{i-1})}$$

where:

- '$\bullet$' is the 2D XY vector dot product.
- AGV_i and AGV_{i-1} are 2D X and Y vectors.
- TP is the test piece location.

The result provides the distance along the vector that the AGV was travelling just before the obstacle was detected. The ground truth system provided the test piece travel line and was compared to the safety sensor measured location of the test piece.

The distance reported in the last column of Table 2 is the distance between where the safety sensor first detected the test piece and where the ground truth instrument measured the test piece. For the tests reported in this paper, the X-Y plane for the ground truth data corresponded to the lab floor, and the AGV navigation data were in the ground truth coordinate frame. For the tests in which the test piece was stationary, the distance in the last column in Table 2 was the distance between two points: the position of the test piece as measured by the safety sensor and its position as measured by the ground truth system. Since the test piece was stationary, the ground truth system recorded multiple measurements for test piece location and the average location was used in the distance calculation.

For the dynamic tests where the AGV path and the sled path crossed, the steps to calculate this distance in the last column of table 2 were as follows:

1) generate a line to represent the AGV path by best fitting a line through the AGV ground truth XY-data
2) generate a line to represent the sled path by best fitting a line through the sled ground truth XY-data
3) generate a line parallel to the AGV path through Pt. A (see Figure 6 a). Pt. A is the location reported by the safety sensor when it first detects the test piece.
4) Determine the intersection of the line generated in Step 3 with the Sled path. This is Pt. B in Figure 6 a and is the estimated location of the test piece using the data from the ground truth sensor.
5) The distance reported in the last column in Table 2 is the distance *d* between Pt. A and Pt. B in Figure 6 a.

For all of the tests where the path of the AGV and sled crossed, Pt. B was always beyond Pt. A as illustrated in Figure 6 a, i.e., the safety sensor underestimated the distance to the test piece.

For dynamic tests where the AGV and sled paths were parallel, the distance reported in the last column of Table 2 was calculated as follows and as shown in Figure 6 b:

1) generate a line to represent the AGV path by best fitting a line through the AGV ground truth XY-data
2) generate a line to represent the sled path by best fitting a line through the sled ground truth XY-data
3) generate a line, Line 1, through Pt. A perpendicular to the AGV path. Pt. A is the location reported by the safety sensor when it first detects the test piece.
4) create a point (Pt. B in Figure 6 b) on the AGV path that corresponds to the location of the safety sensor, based on safety sensor data, when the target was detected.

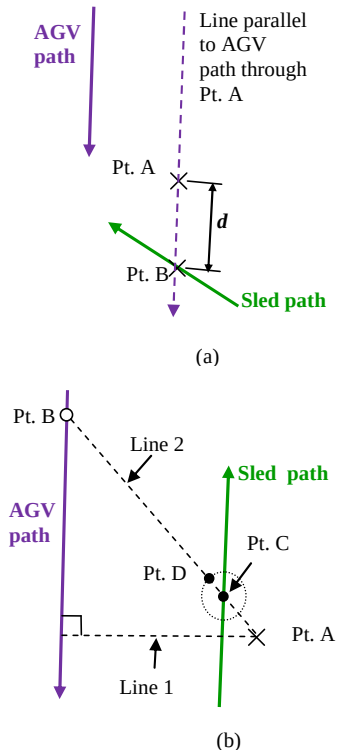


Figure 6. Schematic showing the procedure to determine uncertainty of the location of the test piece (Pt. A) as obtained by the safety sensor for (a) the test piece crossing the AGV path and (b) parallel paths.

5) generate a line through Pt. B and Pt. A.

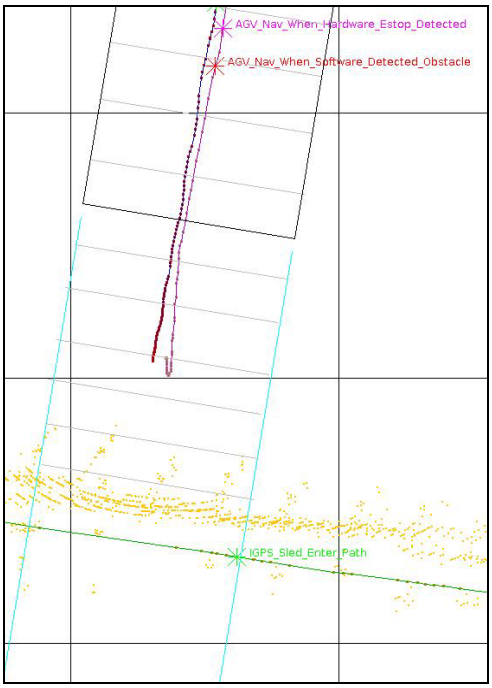
6a) For Tests 23-25 (flat plate)

1) the intersection of the line from Step 5 and the sled path is Pt. C which is also the point being tracked by the ground truth system
2) the distance reported in last column in Table 2 is the distance from Pt. C to Line 1 along the AGV path and mimicking the same situation as if the test piece was in the same path as the vehicle.

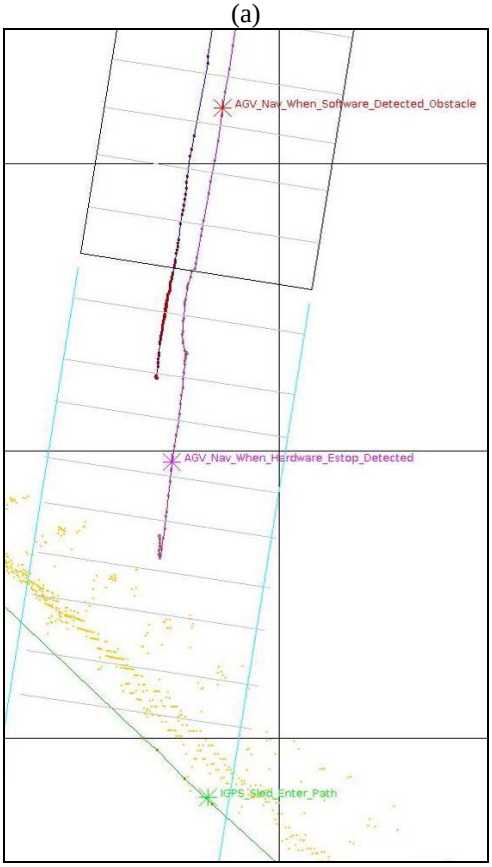
6b) For Tests 26-28 (vertical cylinder)

1) the intersection of the line from Step 5 and the sled path is Pt. C and is the center of the vertical cylinder being tracked by the ground truth system
2) create a point, D, offset from Pt. C by a distance equal to the cylinder radius along Line 2 (Figure 6 b)
3) the distance reported in the last column in Table 2 is the distance from Pt. D to Line 1 along the AGV path and mimicking the same situation as if the test piece was in the same path as the vehicle.

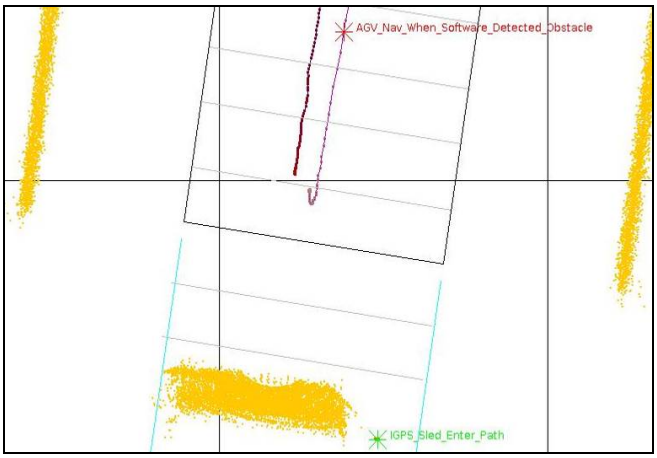
7) In keeping with the sign convention for the crossing paths tests, the distance in the last column in Table 2 is negative if Pt. C or Pt. D was above Line 1 and positive if below Line 1.



(a)



(b)



(c)

Figure 7 – Data from (a) Test 7, (b) Test 22, and (c) Test 12 showing the AGV path (gray lines and light green start position), AGV (black rectangle) at the position where it first detected the test piece (yellow crossing vehicle path). In Test 12, yellow lines represent the safety sensor detecting the confined walls and the static test piece in the AGV path. Ground truth of the AGV path centerline (purple) and test piece centerline (dark green) are also shown. Square, black grid lines are spaced at 1 m. Labels on starred points are as follows:

- *Red: AGV Nav when software detected obstacle*
- *Purple: AGV Nav when hardware (i.e., when a researcher pushed an e-stop button)*
- *Green: iGPS sled enter path*

Unlike the crossing path tests, in the parallel path tests (Test 23-28), in five out of the six tests, the safety sensor overestimated the distance to the test piece. This overestimation could result in the AGV hitting the test piece had the test piece been in the AGV path. Also, in the parallel path tests, the ground truth range difference was a lot higher for the Flat Plate compared to the Vertical Cylinder. Further data analysis is required to determine the reason for the large values (values > 500 mm) for the ground truth range differences in Table 2 – especially for Test 18.

Experimental results are shown in Figure 7 for three controlled braking tests (7, 22 and 11). Tests 7 and 22 were open space tests with AGV and test piece velocities set at 1 m/s.

Test 12 was a confined space test with 0.5 m/s AGV velocity and a static test piece. Test pieces used were the black horizontal cylinder for Test 7, black vertical cylinder for Test 22, and highly reflective flat plate for Test 12. Test pieces were perpendicular to the AGV path for Test 7 and 12 and at a 45° angle to the AGV path for Test 22. Test piece orientation was not a factor for Test 22 since it was a vertical cylinder. All tests that included moving test pieces had a test piece-to-AGV safety sensor separation distance of less than the AGV maximum stopping distance. Test 12 shows a point cloud at the test piece and a similar phenomenon resulted in Test 11 (not shown) with skewed data as well. Previous experiments using highly reflective test pieces and detected by light, instead of a laser range scanner as in tests 11 and 12, have

shown similar results. [7] Further analysis is required to interpret these laser-based results.

Potential sources of experimental errors, possibly causing large distance differences between ground truth and the safety sensor-to-test piece distances (i.e., Table 2, column 8), were:

- Poor logging frame rate. Some time gaps were as high as 0.3 s with more common gaps being 0.1 s. The frame rate error could therefore have caused data gaps of 0.15 m to 0.3 m.
- The manufacturer-specified safety system range error and angular resolution were 3 cm and 0.25°, respectively.
- The difference in time between when the navigation sensor and the ground truth sensor recorded the AGV location.
- The Z-value (vertical distance) was ignored for all data. Errors may have occurred if there were undulations in the test space floor that would cause the AGV to slightly tilt and therefore rotate the navigation and safety sensors.

Ground truth tracking of the horizontal cylinder and flat plate are shown in Figure 5. However, the point tracked for the vertical cylinder was the center of the cylinder at mid-height. Therefore, results for this test piece include an offset that varies up to 34 mm depending upon the distance from the safety sensor. The variation in range offset is due to where the test piece surface point is first detected at the angle measured by the safety sensor versus the test piece center point range perpendicular to the AGV path. The offset was used to correlate with the Figure 4 left-most point on the part. The distances in the last column in Table 2 account for this offset. Test 16 included a non- standard, white surface, vertical cylinder test piece with a 102 mm diameter x 1.5 m high with the vertical axis aligned with the sled center. Two similar vertical cylinders were placed on the AGV as well. Test 16 was used to compare two systems that may be used for ground truth measurements. Analysis of this comparative data will be published in a future paper.

5 Conclusions

The NIST Mobile Autonomous Vehicles for Manufacturing Project evaluated automated guided vehicle (AGV) control based on advanced 2D laser imaging safety sensors that can detect dynamic, standard test pieces representing humans. Experiments and results were presented. Both controlled braking and low-level e-stop braking control, as described in ANSI/ITSDF B56.5, were tested. Results showed that both control methods reduce vehicle energy as standard test pieces moved into or were placed in the AGV path and within the AGV's maximum stopping distance. Results also showed that controlled braking provided deceleration to minimize energy that would impact a test piece that appeared within the maximum AGV stopping distance and therefore, could be used to further improve safety near AGV's. Sources of measurement errors were listed after reviewing the results with the largest potential error source being data logging gaps. In most cases, the distance from the safety sensor to the test piece was less than the distance reported by the ground truth system, i.e., the test piece "appeared" closer that it actually was. This is the better case for AGV safety. The experimental results will be used to develop standard test methods and to recommend improved stopping distance exception language in AGV standards. NIST plans to perform more experiments with:

- low reflectivity test pieces beside similar colored walls,
- overhanging obstacles,
- various ground truth measurement systems,
- radio frequency identification (RFID) when used as proximity measurement devices for predicting pedestrian intent to enter the AGV path.

Also, NIST plans to analyze the 3D data for the experiments discussed in this paper and for the future experiments listed.

6 References

[1] ANSI/ITSDF B56.5 -2010 "Safety Standard for Driverless, Automatic Guided Industrial Vehicles and Automated Functions of Manned Industrial Vehicles."

[2] MOAST, http://sourceforge.net/apps/mediawiki/moast/index.php?title=Main_Page

[3] RCS Library, http://www.isd.mel.nist.gov/projects/rcslib/

[4] NTP: The Network Time Protocol, http://www.ntp.org/.

[5] Nikon iGPS, http://www.nikonmetrology.com/large_volume_tracking__positoining/igps/

[6] Bostelman, Roger,; Shackleford, Will, 2009. "Performance Measurements Towards Improved Manufacturing Vehicle Safety," NIST Intelligent Systems Division, PerMIS 09 Proceedings.

[7] Roger Bostelman, Will Shackleford, 2008. "Test Report on Highly Reflective Objects Near the SR3000 Sensor, NIST Internal Report to Consortium CRADA Partners, February 27.

Integrating Occlusion Monitoring into Human Tracking for Robot Speed and Separation Monitoring

William Shackleford, Richard Norcross, Jeremy Marvel, Sandor Szabo*
National Institute of Standards and Technology (NIST)
100 Bureau Dr.
Gaithersburg MD, 20899
011 1 (301) 975-4286, 011 1 (301) 975-3440, 011 1 (301) 975-4592, *retired
William.shackleford@nist.gov, Richard.norcross@nist.gov, Jeremy.marvel@nist.gov

ABSTRACT
Collaborative robots are used in close proximity to humans to perform a variety of tasks, while more traditional industrial robots are required to be stopped whenever a human enters their work-volumes. Instead of relying on physical barriers or merely detecting when someone enters the area, the collaborative system must monitor the position of every person who enters the work space in time for the robot to react. The TC 184/SC 2/WG 3 Industrial Safety group within the International Organization for Standard(ISO) is developing the standards to help ensure collaborative robots operate safely. [1][2] Collaborative robots require sophisticated sensing technologies that must handle dynamic interactions between the robot and the human. One potential safety risk is the occlusion of a safety sensor's field of view due to placement of objects or the movement of people in front of a safety sensor. In this situation the robot could shut down as soon as even a single sensor was partially occluded. Unfortunately this could greatly diminish the extent to which the robot could work collaboratively. In this paper we examine how a human tracking system using multiple laser line scanners [3]was adapted to work with a robot Speed and Separation Monitoring (SSM) safety system and further modified to include occlusion monitoring.

Categories and Subject Descriptors
C.1.2. Computing Methodologies / Artificial Intelligence /Robotics / Sensors

General Terms
Algorithms, Measurement, Performance, Standardization, Verification.

Keywords
Human Tracking, Laser Line Scanners, Robotics, Safety.

1. INTRODUCTION
The Intelligent Systems Division (ISD) of the National Institute of Standards and Technology (NIST) is part of the team preparing the portion of ISO technical specification (TS) 15066 that deals with a form of collaborative robot safety termed speed and separation monitoring (SSM). SSM prevents contact between a moving robot and any person in the workcell by limiting robot speed and maintaining an adequate separation distance.[4] NIST has developed a prototype SSM safety system that uses laser range and detection scanners to measure the position and velocity of humans (or any moving objects) and computes the separation distance between the human and robot based on the robot's reported position and velocity. The system issues stop or slow signals depending on a minimum separation distance equation proposed in the ISO TS.

2. TESTBED
Our system consists of an under-slung robot mounted on an overhead rail (Figure 1 & **Figure 2**). The human tracking is done using two laser line scanners mounted horizontally and facing each other from opposite ends of the work volume. The system uses two laser scanners, one mounted horizontally to the base of each column that supports the under-slung robot rail (see **Figure 2**). The scanners are mounted at 0.39 m and 0.41 m above the floor facing each other on opposite sides of the robot work volume 5.05 m apart. This configuration detects the entire robot work area and reduces stationary and moving object occlusions. Also, placing the scanners below the robot's reach eliminates the need to discriminate between the robot and other objects that have entered or moved since the system was initialized. The system distinguishes between people and static objects such as the legs of a conveyor table and the rail support structure by subtracting a previously recorded background scan from regular scans during normal operation. For collaborative operation, the tracking system sends the position and velocity of each person to the SSM safety system. The safety system slows or stops the robot based on the relative distance between the robot and the nearest human. This allows the robot to move through one part of the work-volume while a person moves through another part of the volume.

Figure 1 Under-slung Robot under Rail

.

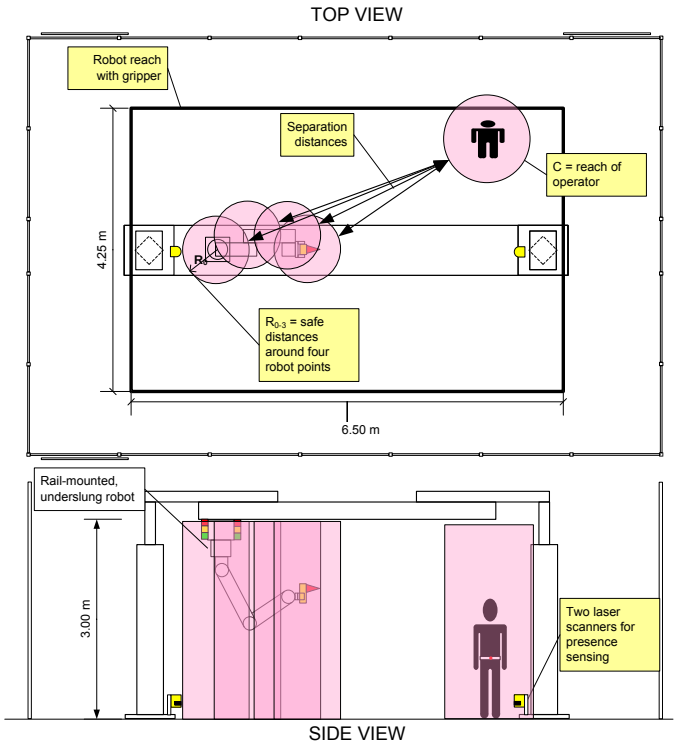


Figure 2 Robot Test-Bed Setup

3. SSM Controller

Equation (1) shows the collaborative form of the minimum separation distance equation.

$$S = K_H * \ T_R + T_B \ + K_R * T_R + B + C \qquad \textbf{(1)}$$

Where:

K_H = Speed of human
K_R = Speed of robot
T_R = Reaction time to detect human and issue a stop – a parameter measured during timing test.
T_B = Brake time – see below.
B = Brake distance – see below
C = $C_H + C_R$, the region surrounding the human and robot respectively. For the testbed, this region includes the uncertainty in position and dimension of each

For the SSM testbed, the brake distance is:

B = $(K_R^2)/2A$
T_B = K_R / A.

A = Acceleration: worst-case deceleration level measured during stopping tests

The robot reports its own position and velocity (K_R) while the human tracking system uses the laser scanners to report the positions and velocities (K_H) of each person or unaccounted for object detected in the work-volume. The distance between the robot and each human is computed by the SSM controller. The SSM controller issues a stop whenever the distance to any human is less the minimum separation distance (S).

.

4. HUMAN TRACKING

The human tracking system is an expanded version of a system we developed for inexpensive ground-truth measurement [3].

The tracker combines the range values into a single coordinate system. To accomplish this, the operator must first establish the position and orientation offset between the two sensors. This is done manually by visually aligning on a display the scans produced by each laser scanner. An object is placed in the Field of View (FOV) of both laser scanners. The operator drags the display of the object from one laser scanner over the display of the same object from the other laser scanner and rotates the object until the displays are aligned.

The background is recorded which contains all the static scanned objects in the FOV. Several frames of data are taken and combined to reduce sporadic noise. Objects seen during this background scan include the legs of a conveyor and the two columns supporting the robot. The tracker detects humans by detecting changes between the current range measurements and those recorded in the static background. Areas where background static objects exist are not processed by the tracker. This eliminates the problem where someone stands still in the robot work volume and eventually is considered part of the background. However, the operator needs to reestablish the background when static objects are moved. Otherwise a human could enter undetected through the previously occupied space. Future work will examine ways to automatically detect changes and automatically update the background.

The human tracking is calibrated to convert positions received from a coordinates system relative to each sensor to positions in the robot's coordinate system. The registration procedure uses a 10 cm (3.9in.) diameter x 91 cm (36 in) high tube placed in the robot's gripper facing down toward the floor. The robot is driven to three widely-spaced positions with the tube low enough to intersect the laser scanner plane. The robot's positions appear on the display along with the tracking system's measurement of the tubes. The operator uses display controls to manually align the robot position and the tube and software automatically calculates the transformation. All subsequent human tracker positions are transformed into the robot's coordinate system enabling the SSM controller to compute the correct separation distances.

During SSM operation, the tracker groups range values into leg groups and human (center of two legs) groups, matches groups from previous groups, maintains a history of the group, and filters the position of each human using a Kalman filter. The filter assumes constant velocity will be maintained and can be tuned by setting the expected acceleration variance and measurement variance. The final position and velocity of the human sent to the SSM controller are taken from the estimated state of a Kalman filter. The results of this tracking are shown in Figure 3.

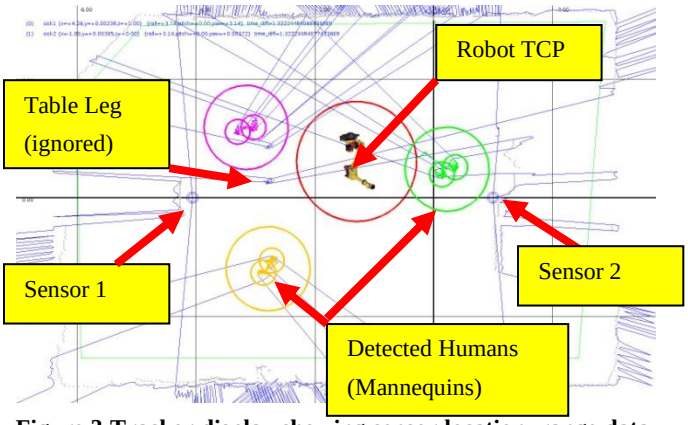


Figure 3 Tracker display showing sensor location, range data, locations of moving objects and the location of the robot's tool center point.

5. OCCLUSIONS

One issue is occlusions due to multiple objects or people blocking the laser scanner FOV. These occlusions can mask the approach of other people thereby preventing the SSM from issuing a stop. We extended the tracker to detect occluded regions. The results of the occlusion detection algorithm are shown in **Figure 4**. The figure shows regions occluded by static objects (yellow) computed from the background range data and regions occluded by dynamic objects (red) computed from the tracking range data.

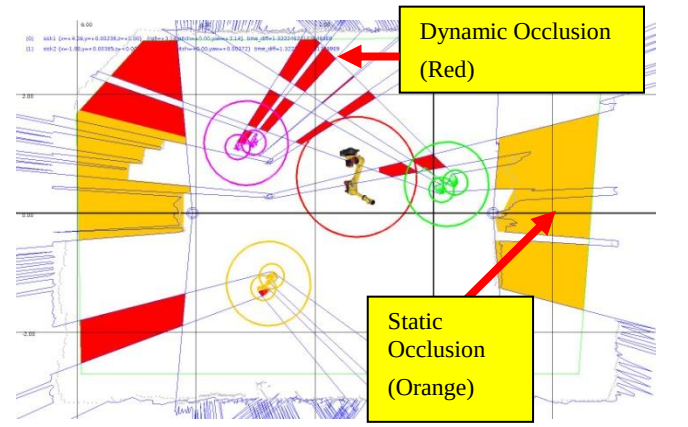


Figure 4 Detecting regions the sensors can't see due to occlusions by fixed objects and by moving objects.

6. GRAPH SEARCH ALGORITHM

To find the occluded areas the tracker creates a bidirectional graph network. Each node in the graph contains the location where the laser was reflected and a node number obtained by incrementing a global count as each node is added. The node is connected to the sensor location for the first and last element in each sensor's range scan. The sensor locations are added as nodes so the graph can be traversed more easily. Points other than the first and last element are connected to the next and previous node. The size of the graph is reduced by combining consecutive nodes of approximately equal range from the sensor. The size of the graph is also reduced by combining all consecutive points outside a manually chosen protected area polygon. The system creates a graph for each sensor. The graphs are combined by searching for intersecting rays between nodes. At each intersection the connections between the original nodes are broken and all involved points are connected to the new node at the intersection. The combined graph is searched to find all polygons. Too find a polygon, begin at any node, and then traverse to any node connected to it. After the first move always choose the next connected node with the smallest possible angle to the previous node. Repeat until you return to the starting node. If you go to every node and apply this to every connection, you will have many polygons stored redundantly. For example, the polygon found starting at node 2 of 2,3,5,6,7 in Figure 5 would also be found by starting at 6 as 6,7,2,3,5. To eliminate these redundancies each polygon is normalized by starting the polygon at the minimum node number. The polygons can then be compared to eliminate the redundant ones. The outer polygon (in the example 0,1,8,9,13,12,11,10) will also be found in this way and is eliminated by testing any point not on the edge of the polygon to determine if it is inside the polygon. Each polygon in the list is labeled as occluded or not by testing one internal point to determine if the polygon is visible to at least one of the laser scanners. The internal point is computed by averaging three consecutive points in the polygon with an internal angle less than 180º. The point is tested by comparing its distance to each sensor with the range reported by that sensor at the appropriate angle.

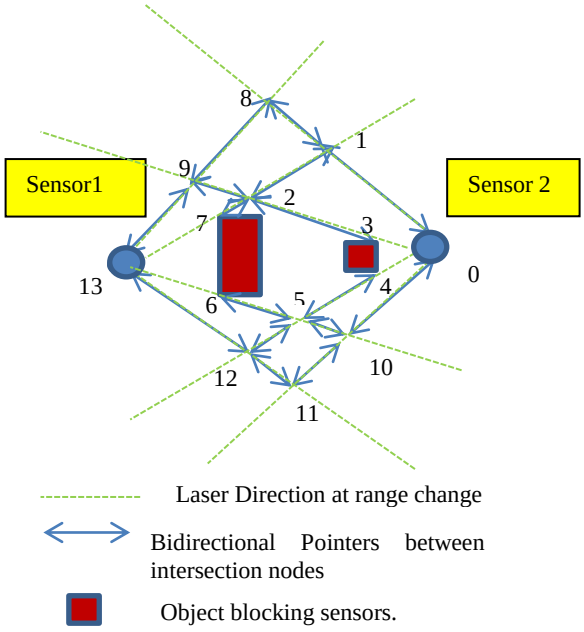


Figure 5. Example Graph network for occlusion analysis.

7. Protected Area Polygon

The sensors can see areas on the other side of the fence that are not of concern for safety. To reduce the processing time needed to find obstacles and occlusions, a polygon drawn approximately just inside the fence line is added to the graph. Objects and obstacles outside this protected area are ignored.

8. Simulation

A simulator was developed to test large combinations of obstacle locations. The simulator places a given number of 0.3 m (1 ft.) diameter circular obstacles at random locations within the protected area polygon. Obstacle locations that would overlap a

.

laser scanner are regenerated. For each range measurement a laser scanner produced, the simulator calculates the distance to the outside edge of the closest obstacle and adds 2.5 cm (1 in) standard deviation Gaussian noise to the range measurement. The noise parameter was chosen from the laser scanner's data sheet and the obstacle radius was chosen based on the approximate cross sections of our mannequins at the average height of the laser scanners. For each test the simulation generates one thousand combinations of obstacle locations.

9. Ground Truth Sampling

It is not really practical to use a high-precision range sensor to provide ground truth as to whether a given position should have been marked as occluded. Any displacement between the ground truth sensor and the laser scanner under test could make a position occluded for one sensor and not for the other. Instead, we use a simpler and more robust algorithm. This method works only at a single point in space. The distances to the point from the two laser scanners are compared against the range value provided by that scanner in the direction of the point. If any range measurement is greater than the distance to the point, the point is visible or else it is occluded. The area within the protected area polygon is randomly sampled and ground truth is only computed at those sample locations. Some points will be sampled within the radius C_H around a detected person or obstacle. Those points are ignored for purposes of occlusion ground truth since the robot would be required to stop as if there were a person there regardless of whether the point was occluded or not.

10. Performance Metrics

The following values were computed for each simulated or real-sensor data experiment as metrics for the effectiveness and/or efficiency of the system.

Processing time – average wall clock time measured as the system computes the occluded area. It does not include time for the robot to respond, nor for the raw data to be collected.[1]

Percentage Occluded – the percentage of the area as reported by the graph algorithm as occluded.

Percentage False Occluded – the percentage of sampled points labeled as occluded by the system under test but visible in the ground truth.

Percentage False Visible – the percentage of sampled points labeled as visible by the system under test but occluded in the ground truth.

Percentage of unseen obstacles - the percentage of obstacles that were more than C_H away from any detected person.

11. Simulation Results

The results of the first simulation set of tests are summarized in Table 1. One of the most disturbing results is the percentage of unseen obstacles with even two obstacles in the scene. The primary reason for this was that there was a blind area behind each laser scanner visible only to the laser scanner on the opposite side. Fortunately this area is not within the robot's work volume. However people in these areas could be moving towards the robot work volume while their positions and velocities were not being reported to the robot due to the occlusion.

[1] Tested on 2-core 2.1 GHz 32-bit laptop.

.

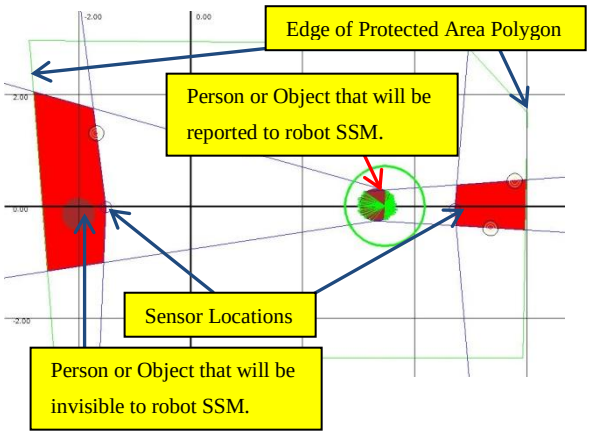


Figure 6. Image from simulation showing occluded areas behind each sensor and one unseen person/obstacle.

12. Real Laser Scanner Data Results

There are a number of problems with trying to reproduce simulated results with real sensor data. While it is easy to actually move people or mannequins around randomly, it is more difficult to ensure that their random positions were not biased to avoid or create occlusions. To ensure the positions were really chosen at random, the same computer program that generated obstacles for the simulation generated a set of obstacle positions to which the mannequins were then moved. The protected area polygon needed to be modified to eliminate areas that the mannequins could not be placed because the area was occupied by a conveyor table or a robot support. Figure 7 and Figure 8 show one snapshot taken from this data. Table 3 provides the cumulative averages for the entire set of tests, including 10 random position combinations and 100 frames of data collected for each combination.

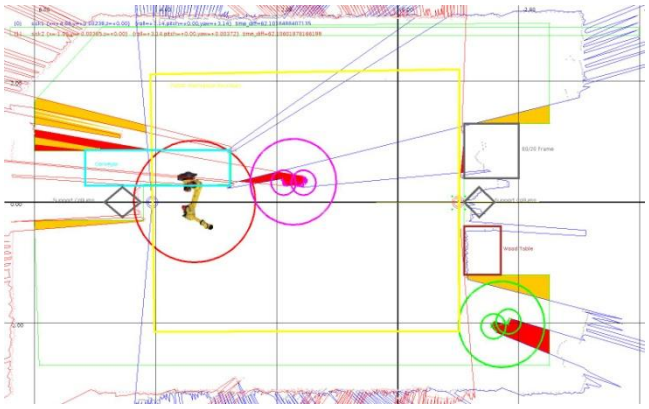


Figure 7. Snapshot of data collected from real laser scanners with mannequins placed at randomly generated obstacle positions.

Figure 8. Robot testbed with mannequins moved into places selected randomly by computer.

13. Occlusion Mitigation Strategies

There are several possible strategies for reducing or eliminating the risks of the robot failing to stop or slow because the person was in an occluded area.

1) Shut down the robot whenever the number of detected people exceeds some maximum. This assumes that no person would be completely occluded unless at least some number of people is detected. The system could be tested and proven to handle at least that number of people. Since the number of people detected was an output from the existing human tracking system, occlusions do not have to be analyzed in real-time. Equipment being carried that hangs down below the height of the laser scanners could be considered an additional person. This might cause unnecessary and unexpected shutdowns. Additional sensors could be added to allow more people to be detected and allowed in the area or to reduce the chances of a person being occluded. This was tested for our testbed in simulation. (See Table 2 for the results.) The additional sensors reduced the size of the occluded regions and the probability that a person would be fully occluded. The additional sensors also increased the amount of processing required to compute the occluded polygons.

2) Use physical barriers to prevent people from standing in areas that would cause a large area to be occluded. The laser scanners could also be used to enforce a policy where some areas of the work volume could be used for a collaborative activity and other areas would result in an immediate shutdown upon detection of people.

3) Occlusion software could execute in real-time if there are sufficient computing capabilities. Either the occlusion monitoring software or the SSM could use the list of occluded regions to compute the distance of the robot to the nearest occlusion and then compare the distance to the minimum separation distance given in Equation **(1)** as it does with the positions of people to determine when to shut down the robot. Since no estimate of a person's speed can be measured when they are occluded from the laser scanner, a constant maximum for K_H would have to be used. It may be necessary to use a less accurate, although faster, method of determining the occluded regions, such as sampling only the centers of grid squares.

14. Conclusions

Allowing humans to work in close proximity to robots will require an ability to detect people in and around the robot work volume. One technology already being used to protect people near robots is the laser line scanner. Although laser scanners are primarily being used only to shut down the robot, they can be adapted to provide real-time robot positions and velocities to allow the robot to adapt to the presence of people. One challenge in making this transition is accounting for the possibility that the laser scanners may be occluded. We presented a method for finding polygons of occluded areas and a way of testing such methods. This could be used either offline or online. Offline it could be used to show that the laser scanners are unlikely to be occluded for a region large enough to hide a person. Online the system could be used to stop or slow a robot before a person enters an occluded area.

15. REFERENCES

[1] ISO 10218-1 Robots and robotic devices – Safety requirements – Part 1: Industrial robots, July 1,2011, www.iso.org

[2] ISO 10218-2 Robots and robotic devices – Safety requirements – Part 2: Industrial robot systems and integration, July 1,2011, www.iso.org

[3] W.P. Shackleford, T.H. Hong, T. Chang, "Inexpensive Ground Truth and Performance Evaluation for Human Tracking using multiple Laser Measurement Sensors." *Proceedings of the 2010 Performance Metrics for Intelligent Systems (PerMIS) Workshop*. 2010. http://www.nist.gov/manuscript-publication-search.cfm?pub_id=906630

[4] Testbed for Evaluation of Speed and Separation Monitoring in a Human Robot Collaborative Environment, October, 2011, NISTIR

.

Table 1 Simulation results for two sensors with original layout

Number of Obstacles	Processing time (ms)	Percentage Occluded	Percentage False Occluded	Percentage False Visible	Percentage of unseen obstacles
1	26	2.28	0.03	0.00	0.00
2	37	4.92	0.05	0.00	1.15
3	61	7.10	0.06	0.00	2.43
4	84	9.71	0.08	0.00	3.32
5	110	11.74	0.10	0.00	4.78
6	133	14.01	0.10	0.01	6.11
7	163	16.38	0.12	0.02	6.22
8	179	18.69	0.11	0.01	8.58
9	224	20.85	0.11	0.01	9.12
10	226	22.93	0.14	0.01	10.41

Table 2 Simulation Results for four sensors.

Number of Obstacles	Processing time (ms)	Percentage Occluded	Percentage False Occluded	Percentage False Visible	Percentage of unseen obstacles
1	231	0.33	0.00	0.00	0.00
2	691	0.72	0.00	0.00	0.00
3	1490	1.15	0.00	0.00	0.03
4	2555	1.69	0.00	0.00	0.00
5	4002	2.28	0.01	0.00	0.00
6	5622	3.02	0.01	0.00	0.07
7	7752	3.77	0.01	0.00	0.17
8	9616	4.66	0.02	0.00	0.26
9	11143	5.56	0.03	0.01	0.27
10	12558	6.49	0.03	0.01	0.52

Table 3 Results using real sensors and mannequins

Number of Obstacles	Processing time (ms)	Percentage Occluded	Percentage False Occluded	Percentage False Visible	Percentage of unseen obstacles
2	45	5.60	0.17	0.11	0.00

.

Robotics Collaborative Technology Alliance (RCTA) 2011 Baseline Assessment

Barry A. Bodt
U.S. Army Research Laboratory
APG, MD 21005
+1 (410) 278-6659
barry.a.bodt.civ@mail.mil

Richard S. Camden
USARL/MPRI
APG, MD 21005
+1 (410) 278-2639
richard.s.camden.ctr@mail.mil

Marshal A. Childers
U.S. Army Research Laboratory
APG, MD 21005
+1 (410) 278-7996
marshal.a.childers.civ@mail.mil

ABSTRACT
This paper discusses the results of the Robotics Collaborative Technology Alliance (RCTA) 2011 baseline assessment conducted over three days in August at the Combined Arms Collective Training Facility at Fort Indiantown Gap, PA. The focus of the effort was on behavior primitives that are necessary for a small robot to autonomously perceive, "Look", and maneuver, "Move", which are two of the five fundamental UGV capabilities identified by the RCTA ("think," "look," "move," "talk," and "work"). An autonomous Talon UGV was challenged to autonomously navigate cul-de-sacs, avoid pedestrians, climb stairs, and negotiate drop-offs, doorways, alleys, and street clutter having both static and dynamic obstacles present. Each of these experimental vignettes had a more detailed set of experimental conditions to be varied (e.g., door width, stair configuration, and obstacle density in a scene.) In addition, a mapping capability was exercised within two buildings to locate walls and room clutter, and a "street view" camera function provided an opportunity for remote interpretation of vision acuity charts mounted on walls within the buildings. The intent was not to revisit physical capabilities of the Talon but rather to determine its autonomous performance over a set of primitive behaviors required for successfully operating in an urban environment with the potential to assist soldiers in reconnaissance or other tasks. The study was successful in identifying some limitations to autonomy, for example, with regard to pedestrian interactions, drop-offs, and doorways but also highlighted an intelligent control system able to overcome varying degrees of disruption in the planned route due to minor and major obstacles.

General Terms
Measurement, Experimentation, Performance

Keywords
Autonomous Ground Vehicle, Talon Platform, Urban Terrain

1. INTRODUCTION
The Robotics Collaborative Technology Alliance (RCTA) joins together government, academic, and industry partners to research and develop robotics technologies for future unmanned ground systems for military application. The RCTA follows previous successful efforts that developed autonomy for larger systems capable of navigating terrain ranging from complex urban environments to road systems to cross-country.[1, 2] The new RCTA extends consideration to smaller ground systems platforms, where teleoperation is currently the norm, and sharpens the operational focus to "think," "look," "move," "talk," and "work." [3] This first investigation under the new RCTA seeks to baseline small robot autonomous capability in an urban environment with respect to the "look" and "move" components of the RCTA vision.

The platform chosen for the initial investigation was the Qinetiq-NA Talon fitted for the Multi Autonomous Ground-robotic International Challenge (MAGIC) in 2010 by the Reconnaissance and Autonomy for Small Robots (RASR) Team, which had participation from RCTA members.[4] The platform offered 360 degree vertical and horizontal LADAR, 360 degree video, an innovative navigation unit, and small robot autonomy (see Figure 1).

Figure 1. RASR Talon.

The MAGIC challenge focused on robot team exploration and mapping, but over mostly flat indoor and outdoor surfaces. In contrast, the focus of the RCTA effort was on behavior primitives where the robot was challenged to autonomously navigate cul-de-sacs, avoid pedestrians, climb stairs, and negotiate drop-offs, doorways, alleys, and street clutter having both static and dynamic obstacles present. Each of these experimental vignettes had a more detailed set of experimental conditions to be varied (e.g., door width, stair configuration, and obstacle density in a scene.) In addition, a mapping capability was exercised within two buildings to locate walls

(c) 2012 Association for Computing Machinery. ACM acknowledges that this contribution was authored or co-authored by a contractor or affiliate of the U.S. Government. As such, the Government retains a nonexclusive, royalty-free right to publish or reproduce this article, or to allow others to do so, for Government purposes only. PerMIS'12, March 20-22, 2012, College Park, MD, USA. Copyright © 2012 ACM 978-1-4503-1126-7-3/22/12...$10.00

and room clutter, and a "street view" camera function provided an opportunity for remote interpretation of vision acuity charts mounted on walls within the buildings. The intent was not to revisit physical capabilities of the Talon but rather to determine its autonomous performance over a set of primitive behaviors required for successfully operating in an urban environment with the potential to assist soldiers in reconnaissance or other tasks.

An experimental strategy provided rationale for each vignette, specific test conditions, and other details of test operations.[5] The assessment was conducted over three days at the Combined Arms Collective Training Facility (CACTF) at Fort Indiantown Gap, PA (see Figure 2). Data were collected according to an experimental design, with balanced, randomized trials capable of supporting comparisons in performance among test conditions. No a priori map information was supplied. In the following, we will take each experimental vignette in turn, describing the experimental situation, providing summary statistics, and interpretation of results. The mapping vignette is not addressed in this paper; rather, it will be developed in a subsequent report.

Figure 2. Southwest portion of the CACTF.

2. EIGHT VIGNETTES

Eight experimental vignettes are reported in this paper: surfaces, cul-de-sac, pedestrians, marketplace, alley, doorway, drop-off, and stairs.

2.1 Surfaces

Traversal over different surfaces, climbing stairs, negotiating ditches, etc. is an inherent mobility concern. The platform capability of moving over pavement, concrete, gravel, stone, grass, mud, etc. can be explored in an operator controlled setting and was not within the scope of this experiment. However, surfaces, combined with slopes, could also induce slippage, affecting navigation solutions, or present challenges to perception depending on the coarseness of the surface (or in the case of grass, the height of perceived obstacles). The perception challenge can be indirectly assessed by measuring speed over course segments of available surfaces.

Three surfaces and four horizons determined the test conditions for the surface vignette. Low grass, gravel, and pavement surfaces were crossed with level, downhill, uphill, and side-hill horizons, resulting in 12 test conditions. Twenty-five runs were completed; each approximately 30 m long, with the exception of side-hill runs on gravel and pavement that were approximately 15 m long. The nominal threshold speed for all but three runs was set at 0.6 m/s. Three uphill runs (5.1, 10.1, and 22.1), one on each surface, were run at 1.2 m/s. One scheduled pavement side-hill run was skipped because of communication challenges with the test operations center, and another pavement uphill run was lost when a controller used to teleop the Talon into position was left on, causing the robot to stop for over a minute. Figure 3 shows the area on the CACTF where uphill, downhill, and side-hill gravel and grass runs were made. Table 1 summarizes the results in terms of average moving speed (m/s) when the robot was in motion, average run speed (m/s) taking into account stoppage times, and the percentage of total run time the robot status was recorded as stopped as opposed to moving. Statistics are based on two runs. Values with an asterisk (*) are based on only one run. All runs were completed except as already noted.

Figure 3. Gravel and grass surface runs.

Table 1. Surface vignette summary statistics

Surface	Horizon	Moving Speed	Run Speed	Stop Time %
Low Grass	Level	0.68	0.68	0.0
	Downhill	0.46	0.42	9.9
	Uphill	0.43	0.41	5.8
	Side-hill	0.30	0.25	14.4
Gravel	Level	0.70	0.68	2.3
	Downhill	0.61	0.61	0.0
	Uphill	0.61	0.61	0
	Side-hill	0.55	0.47	14.8
Pavement	Level	0.65	0.65	3.0
	Downhill	0.62	0.62	0.0
	Uphill	*0.60	*0.60	*0.0
	Side-hill	*0.64	*0.64	*0.0

The data suggest an interaction between surface and horizon on Talon speed and stoppage time. For pavement, the Talon traveled at approximately the nominal threshold with no degradation due to horizon. On gravel, the robot exceeded the threshold on level ground and slowed slightly with some stoppage time for the side-hill runs. The stoppages in both cases occurred just as the run was initiated, after which the Talon progressed without halting to the goal. On grass, greater speed differences were observed according to horizon, with the side hill runs taking the most time. Figure 4 shows the stoppages (in red) for run 7, a low grass, side-hill run. Stoppages occurred at various points along the run.

The three runs made at a nominal 1.2 m/s did not achieve that nominal threshold speed but in each case did exceed the speed of runs under the same conditions. Moving speeds of 0.54, 0.87, and 0.75 m/s were seen for uphill runs on grass, gravel, and pavement, respectively. This has the unfortunate implication that the threshold speed may have artificially

constrained the Talon, potentially masking speed effects for gravel and pavement.

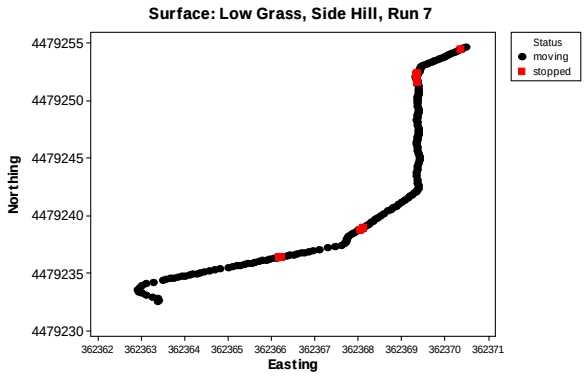


Figure 4. Run 7 stop time locations.

2.2 Cul-de-sac

Immediate obstacles do not have to drastically change the route of the Talon. It simply bypasses them and continues on the intended route. However, certain situations create major route disruptions that require the Talon to effectively establish a new route segment over a portion of the initial route or provide constraints on how the route is executed. These challenges suggest a need for higher-level intelligent control. For example, outside cul-de-sacs or dead ends within buildings call for a more substantive departure from the initial route. A maze of street clutter or interactions with buildings might have the same effect. Further, the Talon may seek to minimize line of sight to an enemy location or prefer a surface (e.g., sidewalks) when operating in a busy urban setting. Additional constraints provide a challenge to perception and planning.

Three cul-de-sac locations and three severity levels for route disruption determined the test conditions for the cul-de-sac vignette. Cemetery, office building, and maze locations were crossed with slight, moderate, and full severity in route disruption, resulting in nine test conditions. Severity was a subjective determination of the degree of departure from the original route that would be necessary to reach the goal. A total of 22 runs were made, with 18 complete, 3 listed as "did not finish (DNF)" and 1 instance, run 46, when the Talon bypassed the moderate challenge at the maze altogether. Run 46 was excluded from analysis. Figure 5 shows the office building challenge, with routes skirting the edge of the office building or passing through it. (The cemetery is in the foreground.)

Figure 5. Office building location.

Table 2 lists summary statistics for the cul-de-sac vignette. In this analysis we again relied on average speed while moving (m/s), average run speed including stoppage time (m/s) and the percentage of time the Talon status was stopped. Given that autonomous mobility is a study focus, it is important to also note the ratio of completed runs (Comp.) where the robot achieved the goal to runs attempted. Distance of the runs varied according to the location and challenge. The slight route interruption at the cemetery was approximately 30 m. The longest routes were approximately 100 m.

Table 2. Cul-de-sac vignette summary statistics

Location	Severity	Comp. /Run	Moving Speed	Run Speed	Stop Time %
Cemetery	Slight	2/3	0.51	0.49	4.1
	Moderate	2/2	0.49	0.37	26.1
	Full	2/3	0.46	0.32	29.0
Office Building	Slight	2/2	0.50	0.43	15.2
	Moderate	2/2	0.54	0.48	11.5
	Full	2/3	0.47	0.32	30.5
Maze	Slight	2/2	0.56	0.40	30.9
	Moderate	2/2	0.72	0.52	19.9
	Full	2/2	0.59	0.46	22.0

The data do not suggest a difference in average moving speed over severity, and only a modest gain in average moving speed occurs at the maze, which was the only location on pavement. Much of the higher number for moving speed under moderate severity at the maze is attributable to run 46.1, which included an additional waypoint to encourage the Talon to get to the maze directly; it did so more quickly. For the cemetery and office building locations, there is more stoppage time associated with full severity of the route disruption.

Figures 6–8 show Talon paths at all locations and note the paths by the severity of the route disruption. Figure 6 shows paths through the only opening in the cemetery wall. In two runs, the Talon proceeded down the line of headstones in search of a way out. In one case it found it; in one it did not. Another run did not finish when the Talon tried to go between headstones with high grass in its field of view. Figure 7 shows different route choices by the Talon around the office building. In one case the Talon did not finish the route when it veered off and entered an open door in a neighboring building. Figure 8 shows the paths through the maze. More turns are evident with the greater the severity disruption. The most complex maze is shown in Figure 9.

2.3 Pedestrians

The potential for an unmanned ground system to interact with pedestrians in an urban environment is great. A long-standing interest of the RCTA has been the safe interaction with pedestrians. Unlike with larger robots, where pedestrian safety was the paramount consideration, the focus with small robots is their ability to successfully navigate in the close proximity to pedestrians. This capability is required for the robots to function in teams with soldiers as well as to avoid civilians in an area of operations.

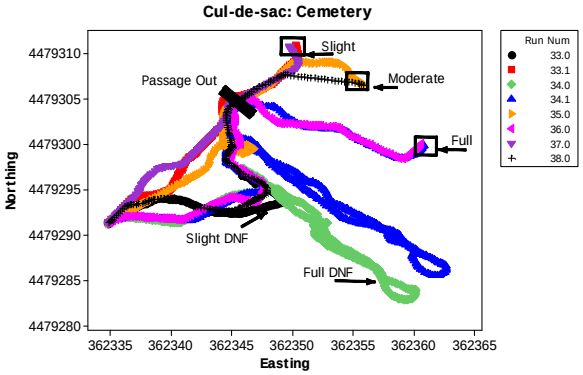


Figure 6. Talon paths out of the cemetery.

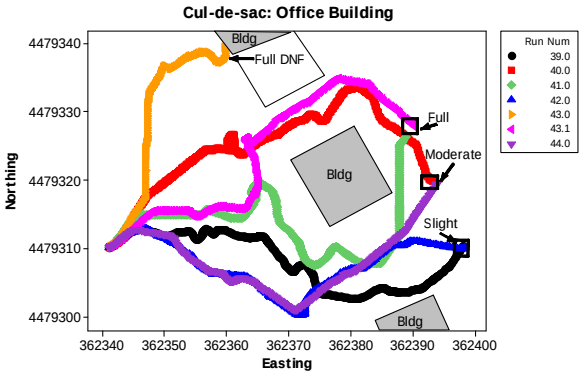


Figure 7. Talon paths around the office building.

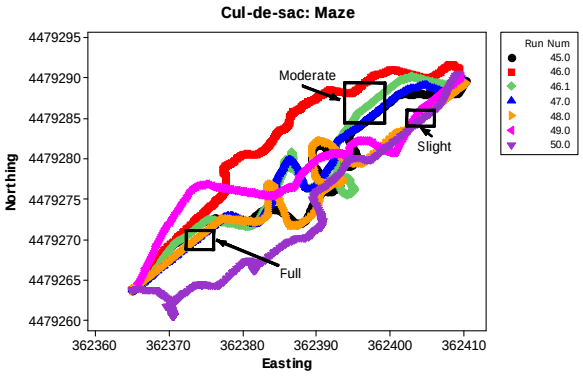


Figure 8. Talon paths through the maze.

Figure 9. Full severity for the maze cul-de-sac.

Two pedestrian speeds and five modes of interaction with the Talon determined the test conditions for the pedestrian vignette. Speed (walking or jogging) was crossed with five distinct pedestrian paths relative to the path of the Talon: front perpendicular, front parallel, side perpendicular, side parallel, and orbit. The purpose of this vignette was to determine how the Talon would react to pedestrian paths close to or crossing its route. Figure 10 shows a side-parallel pedestrian path that passes and cuts into the path of the Talon to within 2-3 m. The Talon would be expected to slow or veer. Perpendicular paths would intersect the Talon's path immediately in front or would stop right at the side of the Talon, but would still be within its 360 degree field of view. Orbit runs required the pedestrian to walk around the Talon throughout its run.

Figure 10. Side-parallel pedestrian path.

A total of 21 runs were conducted in the pedestrian trials, executed in what we will refer to as three phases that corresponded to our evolving understanding of run results during the experiment. All but two of the runs were complete in that the goal was reached, but the interaction between the pedestrian and the Talon was probably only observed in the third phase. In the first phase, five runs were conducted east to west (right to left in Figure 11), with the pedestrian to the left of the Talon. Initially, it was thought that the Talon responded in a sensible way, giving the pedestrian wide birth as it moved to the right and followed the curb until angling toward the goal. However, it was gradually realized that something was amiss when even in an orbiting run, the Talon chose to move toward the pedestrian when his path was between the Talon and the curb. A conjecture as to the cause moved us to Phase 2.

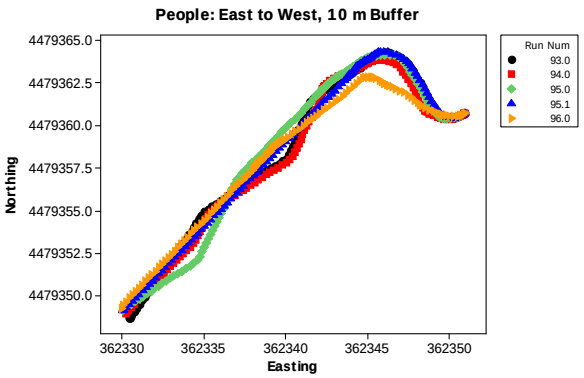


Figure 11. Pedestrian runs east to west, 10 m buffer.

In the next phase, the start and end points were switched, making the movement west to east (left to right in Figure 12). The belief by this time was that a 10 m obstacle buffer was in

effect, causing the Talon to react to a retaining wall and building to the same side as the pedestrian. (This parameter had a role in outside mapping at MAGIC, where it was useful to keep the robot away from buildings.) The advantage of switching start and end points was that the retaining wall did not begin until approximately half way through the run. Still, as seen in Figure 12, the Talon gradually moved away from the retaining wall side of the street, but now toward the pedestrian, still on the Talon's left. Two runs of eight did not finish, one where the robot became high centered on the curb, the other where an obstacle in the map at start caused it to wander over the curb down an embankment. The 10 m buffer was masking any pedestrian interaction.

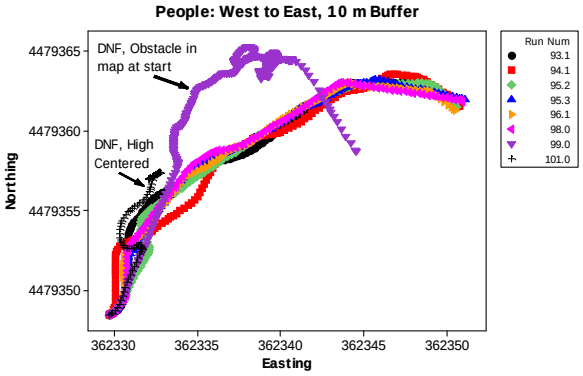


Figure 12. Pedestrian runs west to east, 10 m buffer.

Finally, in the third phase the obstacle buffer was set to 1 m to take the retaining wall out of play for the remaining eight runs shown in Figure 13. Local perturbations throughout the routes were now seen and likely the result of interaction with the pedestrian. This is strongly suggested in Figure 14, which shows all four pedestrian orbit runs. In Figure 14, run 95.3, under the 10 m buffer, maintains a smooth path to the goal free of local perturbations, while the other routes under the 1 m buffer clearly show the Talon adjusting to the pedestrian's presence in orbit about the Talon.

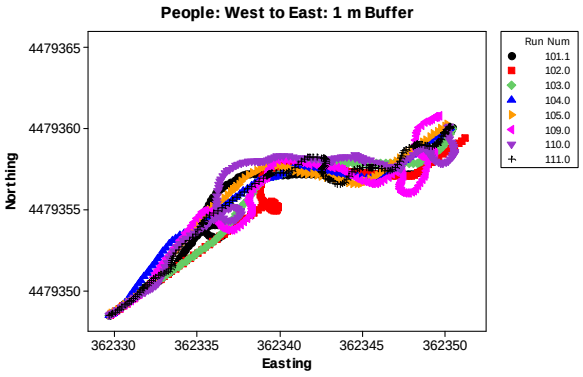


Figure 13. Pedestrian runs west to east, 1 m buffer.

Perpendicular approaches to the Talon and orbits of the Talon resulted in stoppages during the route as the Talon avoided the pedestrian. There appeared to be no difference in performance caused by pedestrians jogging rather than walking. Table 3 lists the raw data for the eight runs in Phase 3. Average moving and run speeds (m/s) were reduced for orbit runs and the percentage of stoppage time was increased. The Talon successfully avoided the pedestrian in all instances.

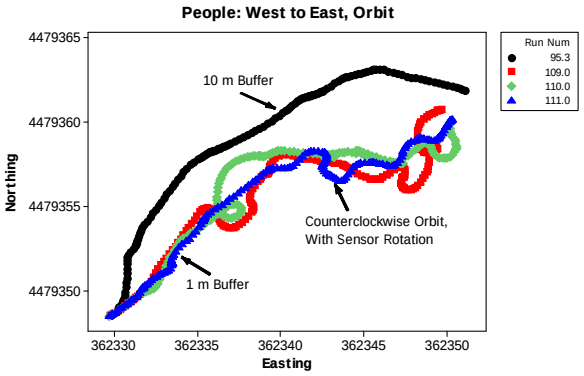


Figure 14. Pedestrian orbit runs, west to east.

Table 3. Phase 3 pedestrian runs

Run	Path	Moving Speed	Run Speed	Stop Time %
101.1	Side ‖	0.61	0.61	0.0
102	Side Per.	0.55	0.54	3.2
103	Side ‖	0.63	0.63	0.0
104	Front ‖	0.65	0.65	0.0
105	Side Per.	0.63	0.57	11.6
109	Orbit	0.45	0.35	21.9
110	Orbit	0.43	0.37	15.1
111	Orbit	0.42	0.28	32.9

2.4 Marketplace

The marketplace vignette combines elements of the cul-de-sac and pedestrian vignettes. The Talon had to avoid static obstacles that if dense enough in the scene could force it to reroute, not just avoid obstacles; it also was required to interact with pedestrian paths, this time from several pedestrians, not just one. An unmanned system acting in a busy urban environment and participating on a team with soldiers and other robots is expected to handle this situation en route to a goal.

Only one factor representing both density and type of obstacles determined the test conditions for this vignette. Two runs were conducted in a clear scene, four with a dense array of static obstacles, and two more that added "Dynamic" obstacles, (six pedestrians), to the existing static obstacles on the course. Pedestrian paths were completely unscripted, but participants were encouraged to reasonably interact with the Talon's path forward, for example, by crossing in front of it or walking closely beside it. Three runs (two static and one static plus dynamic) were repeated as "return" trips through the marketplace. Return trips benefited from the retention of the detected objects on the map from the initial trip; otherwise, the map was cleared for each run.

Figures 15–17 show the individual runs according to the obstacles faced. They illustrate the impact of obstacles on the path of the Talon. Figure 17 compares the initial and return trips under dynamic obstacles. Stoppages are noted.

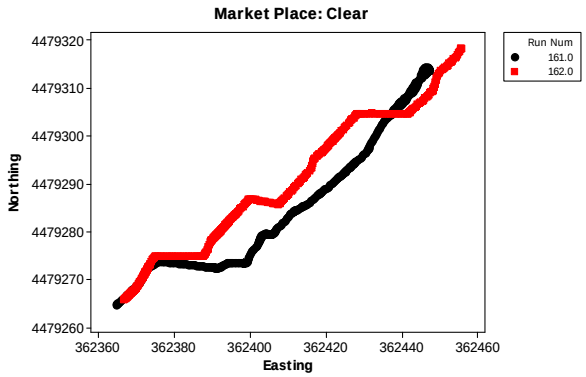


Figure 15. Marketplace runs with no obstacles.

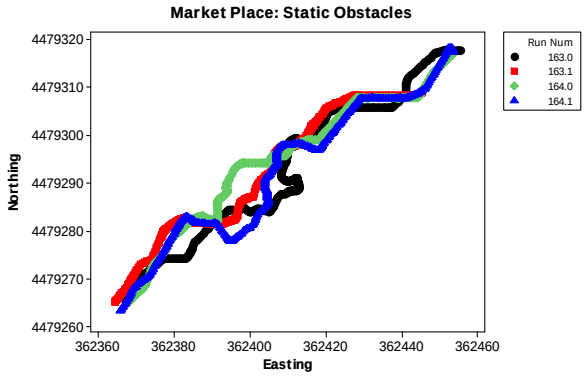


Figure 16. Marketplace runs with static obstacles.

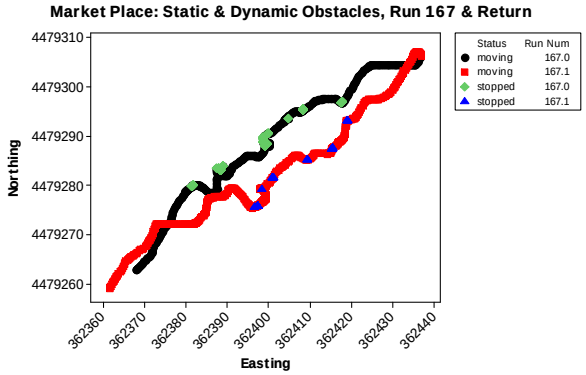


Figure 17. Marketplace runs with dynamic obstacles.

Table 4 shows data recorded for the eight individual runs in terms of average moving and run speed (m/s) and stop time percentage for each test condition. Dynamic in the table includes the static obstacles. The run numbers ending in ".1" are the return trips for the corresponding integer-valued run. Runs were generally fast, as the course was on pavement. Moving speed appears unaffected by the introduction of obstacles. In run 161, the Talon made a big move to the right and ended up stopping five times over the course for as much as 21 seconds. In run 163, a teleop was required after an extended stop; thus, the run was not counted as complete and is noted with an asterisk (*). Other runs accumulated stoppage time when interacting with the obstacles. The last four runs suggest that retaining the map obstacles may have improved average moving speed or reduced the stop time percentage.

Table 4. Marketplace vignette summary statistics

Run	Obstacle	Moving Speed	Run Speed	Stop Time %
161	Clear	0.79	0.53	33.2
162	Clear	0.76	0.72	6.5
163*	Static	0.60	0.33	45.3
163.1	Static	0.62	0.58	7.1
164	Static	0.68	0.62	7.9
164.1	Static	0.76	0.71	6.7
167	Dynamic	0.60	0.46	24.0
167.1	Dynamic	0.68	0.59	13.5

2.5 Alley

The alley vignette provided a quick look at how narrow a passageway the Talon was willing to autonomously pass through. The platform is approximately 22 inches wide, but whether or not it will pass autonomously is a question of perception in how the robot sees itself in the physical environment.

This vignette consisted of only six runs. The Talon was to pass through an alley approximately 30 m long with width varied according to the run: 48, 40, or 32 inches. The initial alley width was 48 inches, but was constricted to one of the stated widths early in the run by an artificial plywood wall (see Figure 18). Two replications of each width were run in a randomized order.

Figure 18. Talon approaches 32-inch wide alley.

Table 5 lists the individual run results in terms of average moving and run speed (m/s) and the percentage of stop time. The Talon was able to navigate the 48- and 40-inch passage, but would not pass through the 32-inch alley. Runs with asterisks (*) did not finish. How time was used to accomplish the approximately 30 m route is of keen interest in operations. Due to the tight space, the percentage of time stopped is increased relative to that of other vignettes.

2.6 Doorway

The doorway vignette challenged the Talon to pass through a variety of doorway configurations and approach angles. In many respects, the challenge is similar to the alley vignette,

but differences are present with respect to the angle of approach and the door position.

Table 5. Alley vignette summary statistics

Run	Width	Moving Speed	Run Speed	Stop Time %
51	48”	0.50	0.27	45.7
55	48”	0.57	0.28	51.0
52	40”	0.52	0.26	50.2
54	40”	0.45	0.18	59.4
53*	32”	0.42	0.16	62.4
56*	32”	0.47	0.20	57.3

Three door widths (36”, 30”, 24”), two door positions (60°, 180°), and three approach angles (0°, 45°, 90°) determined the original set of test conditions, which with 2 replications led to 36 planned runs. The front door of the CACTF police station was chosen as the site for this study because it provided a double doorway, which opened away from the approaching robot, thus providing the structure to explore door position settings. The door was held wide open (180°) or partially open (60°). (See Figure 19.) A full sheet of plywood was used to create the various doorway widths.

Figure 19. Doorway open 60°.

For this vignette, challenges were taken in the order of easy to hard, based on doorway width. Table 6 lists summary statistics for the configurations tested. At the door width of 36” and the door wide open (180°), all six attempts were successful (two approaching from 0°, two from 45°, and two from 90°). When the door opening was reduced to 60°, the robot was unsuccessful in finding an acceptable lower level plan through the opening and did not attempt to go through the doorway. An upper level plan did exist. The average time (sec) and percentage of stop time are listed. Speeds are not reported because of expected variability over only a 10 m distance.

When the door width was reduced to 30”, two attempts were made at the head on (0°) approach angle and the door wide open (180° door opening). In both runs, the robot explored the area around the door and behind the nearby filing cabinet but could not generate a low level plan through the door even though there was sufficient physical clearance for the robot to go through the 30” doorway. All other conditions under the 30” and the 24” doorways would have generated the same unsuccessful run.

Table 6. Doorway vignette summary statistics

Door Width	Door Open	Approach Angle	Complete /Run	Average Time	Stop Time %
36”	180°	0°	2/2	29.0	29.3
		45°	2/2	32.8	48.1
		90°	2/2	15.8	4.8
	60°	All	0/6	…	…
32”	180°	0°	2/2	15.3	29.5
		45°	2/2	25.0	17.0
		90°	1/2	9.8	15.4
	60°	All	…	…	…
30”	180°	0°	0/2	…	…

As an excursion to explore the minimum achievable doorway width, six runs were made at the doorway width of 32” and the three approach angles (0°, 45°, 90°). The door opening was held at 180° because the robot was not able to go through the 36” door with a 60° door angle.

On all six runs, the robot went through the doorway. On the last run the robot track got wedged in the doorway after it had cleared ¾ of the doorway. This establishes a baseline doorway performance, for this configuration of the Talon autonomous mobility software; a 32” door is the smallest width for which it can autonomously plan and execute a passage through a wide-open doorway.

2.7 Drop-off

The drop-off vignette provides an opportunity to observe what autonomous decision the robot will make if the path forward involves an abrupt negative change in elevation. Especially in an urban environment, manmade structures often have this characteristic.

Two landing surfaces (plywood, sod) and three notional heights (low, medium, and high) were initially intended to define the test conditions. Fixing exact heights, however, was more difficult than anticipated with available materials, so a few convenient heights were run. The main question answered was whether the Talon would step down. In addition, it was to be noted if the Talon, after stepping down, finished far from the goal with the rationale that such a finish might indicate the drop disrupted the navigation solution. A drop-off to sod is shown in Figure 20. Table 7 lists the results for runs completed.

The Talon successfully attempted and negotiated the drop-off in 8 of 12 runs overall, 3 of 6 to a plywood landing and 5 of 6 to a sod landing. The distance to the goal did not suggest a navigation disruption in any of the runs, even with the Talon landing hard for the greater drop-offs.

2.8 Stairs

The stairs vignette addresses an obvious challenge in an urban environment. The Talon is not built for stair climbing, but a few runs were attempted. The stair configuration used had an open style riser and either a 4” or 8” height. In the 4” height, a second 4” step was encountered after a 24” tread. Closed risers, additional heights, and tread widths were shelved for subsequent study.

Figure 20. Talon approaches a drop-off to sod.

Table 7. Drop-off vignette summary statistics

Landing	Drop Height (in)	Complete /Trials
Plywood	5.75	2/2
	7.5	1/3
	9	0/1
Sod	6.5	1/2
	7.0	1/1
	7.5	3/3

The results were simply that in the two runs for the 4" riser up two steps that the Talon completed the task with only slight hesitation on one run. It was noted that it did not square up properly to the steps. For the 8" riser and one step, 3 runs were attempted, but none were completed. For the last run with an 8" riser, a 2 x 4 board was added to effectively lower the riser to 6 ½", but the Talon would still not climb.

3. DISCUSSION

There were a few limitations in this study. Metrics continue to challenge robot experimentation. Slippage, mentioned in the abstract, was never formally pursued because there was not a good way to measure it; the internal or external systems capable of measuring robot navigation accuracy were not available. In addition, autonomous mobility leads to considerable variability among routes taken, even under the same conditions. Variability has the potential to mask the effect of other variables. The speed threshold setting of 0.6 m/s was an unfortunate choice in some instances. We may have learned more without this artificial threshold. Despite these limitations, performance was observed and measured in a variety of relevant situations and data recorded for future comparisons.

The 2011 baseline study provides a starting point for considering capabilities of a small autonomous system working within an urban environment. A cross section of behavior primitives challenged the platform, perception, and intelligent control. The RASR Talon was successful in meeting most of these challenges with regard to run completion, but challenges were sufficiently difficult to provide an opportunity for discovery of limitations. Going forward, findings will be used to address limitations in autonomy described in Section 2. We expect to return to the CACTF to measure progress in all areas explored here as a component of RCTA integrated research assessments.

4. ACKNOWLEDGEMENTS

Many individuals contributed to the success of the baseline study, either in the planning phase, execution, or data management. The authors would like to thank Craig Trice and Nicoleta Florea (MPRI); Robert Mitchell, Brad Stuart, and Robert Dean (General Dynamic Robotic Systems); Alberto Lacaze, Karl Murphy, Anne Schneider, and Nenad Uzunovic (Robotic Research, LLC); Mark Del Giorno (Del Services LLC); and Justin Teems (QinetiQ-NA).

5. REFERENCES

[1] Bodt, B.A., Childers, M.A., Hill, S.G., Camden, R.S., Gonzales, J.P., Dean, R.M., Dodson, W.F., Kreafle, G., LaCaze, A., Sapronov, L., *Unmanned Ground Vehicle Two-Level Planning Technology Assessment*, ARL-TR-5331, September 2010.

[2] Childers, M, Bodt, B. and R. Camden, "Assessing Unmanned Ground Vehicle Tactical Behaviors Performance," *International Journal of Intelligent Control Systems*, V16, No 2, pp 52-66, 2011.

[3] Mitchell, R. and Bornstein, J., Robotics Collaborative Technology Alliance FY11 Annual Program Plan. http://www.arl.army.mil/www/pages/392/rcta.fy11.ann.prog.plan.pdf

[4] Brooks, P. *The Challenge: A Small Business Perspective on Entering an Unmanned Systems Competition*, Association of Unmanned Systems International, 2011.

[5] Bodt, B.A., *Robotics Collaborative Technology Alliance (RCTA) 2011 Baseline Assessment Experimental Strategy*, ARL-TN-457, September 2011.

Using Competitions to Advance the Development of Standard Test Methods for Response Robots

Adam Jacoff, Raymond Sheh & Ann-Marie Virts
National Institute of Standards and Technology (NIST)
100 Bureau Drive, Gaithersburg, MD 20899, USA
+1 301 975 [4235|5068|3533]
adam.jacoff@nist.gov, raymond.sheh@robolit.com,
ann.virts@nist.gov

Tetsuya Kimura
Nagaoka University of Technology, Fujihashi, Nagaoka, Niigata, Japan
+81 258 47 9708
kimura@mech.nagaokaut.ac.jp

Johannes Pellenz
Bundeswehr Technical Center for Engineer and General Field Equipment, 56070 Koblenz, Germany, +49 261 876 7470
pellenz@uni-koblenz.de

Sören Schwertfeger
Jacobs University Bremen
28759 Bremen, Germany
+49 421 200 3155
s.schwertfeger@jacobs-university.de

Jackrit Suthakorn
Mahidol University, 25/25 Puttamonthon 4 Road, Salaya, Nakorn Pathom 73170, Thailand, +66 2 889 2138
egjst@mahidol.ac.th

ABSTRACT

Competitions are an effective aid to the development and dissemination of standard test methods, especially in rapidly developing, fields with a wide variety of requirements and capabilities such as Urban Search and Rescue robotics. By exposing the development process to highly developmental systems that push the boundaries of current capabilities, it is possible to gain an insight into how the test methods will respond to the robots of the future. The competition setting also allows for the rapid iterative refinement of the test methods and apparatuses in response to new developments.

For the research community, introducing the concepts behind the test methods at the research and development stage can also help to guide their work towards the operationally relevant requirements embodied by the test methods and apparatuses. This also aids in the dissemination of the test methods themselves as teams fabricate them in their own laboratories and re-use them in work outside the competition.

In this paper, we discuss how international competitions, and in particular the RoboCupRescue Robot League competition, have played a crucial role in the development of standard test metrics for response robots as part of the ASTM International Committee of Homeland Security Applications; Operational Equipment; Robots (E54.08.01). We will also discuss how the competition has helped to drive a vibrant robot developer community towards solutions that are relevant to first responders.

©2012 Association for Computing Machinery. ACM acknowledges that this contribution was authored or co-authored by a contractor or affiliate of the U.S. Government. As such, the Government retains a nonexclusive, royalty-free right to publish or reproduce this article, or to allow others to do so, for Government purposes only. PerMIS'12, March 20-22, 2012, College Park, MD, USA. Copyright ©2012 ACM 978-1-4503-1126-7-3/22/12...$10.00

Keywords

autonomous robots, competitions, field robotics, manipulation, mobile robots, performance metrics, response robots

1. INTRODUCTION

The Intelligent Systems Division of the National Institute of Standards and Technology (NIST) has been making use of international competitions as part of its standards development process for response robots. These standards, part of ASTM International Committee of Homeland Security Applications; Operational Equipment; Robots (E54.08.01), are developed with the active input of robotics researchers, developers, and test administrators and are based on requirements formulated in consultation with first responders. They measure the performance of operationally relevant aspects of whole robot systems, as a deployable configuration.

Recent world events, ranging from natural disasters such as the 2011 Tohoku Earthquake through to various counterterrorism operations and military deployments in places like Iraq and Afghanistan, have thrown the spotlight onto the use of robots order to reduce the risk to humans. The ability to evaluate the different aspects of response robot performance in a relevant, objective manner is crucial to ensuring that first responders and other end users have an accurate understanding of the capabilities of current robots and are able to make an informed choice when procuring such robots.

In administering competitions, NIST gains a valuable understanding of the performance of upcoming best-in-class technologies and an opportunity to perform rapid iterative refinement on the test methods and apparatuses. The competitions also serve as a proving ground where new test methods and apparatuses may be conceived and refined in the presence of researchers and developers that have a deep understanding of new capabilities still at the research stage. In return, the development community gains an insight into the needs of first responders and an opportunity to finetune their approach and promote their work. As they replicate the test methods, they also help to disseminate their use

through the academic community.

Academic competitions are unique in their ability to gather a wide variety of research platforms in the one location, at the same time, to tackle the same problem. As a target for standard test metrics, response robots present a unique challenge since they must be tested end-to-end as complete robotic systems and evaluated in terms of their performance in a particular aspect of their task. Unlike other fields, it is not practical to derive this performance from the performance of the individual components. For example, the test methods for vision are affected not only by the quality of the camera but also the power supply, communications system, human-robot interface, and any directed perception mechanisms. Competitions inherently test such systems end-to-end and are ideal for gaining exposure to complete, highly innovative, experimental implementations.

In the rest of this paper, we will discuss the role of the RoboCupRescue Robot League (RoboCup RRL) competition, community, and associated events, in the development of standard test methods for response robots. In particular, we will focus on several examples of test methods that were conceived in, or refined at, the competition and have since become or are soon to become standards. For an overview of the competition itself, including performances from the most recent competition, the reader is invited to refer to [8]. More detailed information about the competition and arena are available from the rules outline and arena construction manual [6, 10].

2. THE ROBOCUPRESCUE ROBOT LEAGUE

The International RoboCup Competition is best known for its soccer playing robots, where the challenge is to build robots that are able to play soccer, according to World Cup rules, better than the winners of the 2050 human World Cup. However, since 2001, RoboCup has also played host to the RoboCup RRL, a NIST-administered event where the task is to develop robots to solve challenges from the field of Urban Search and Rescue robotics. The RoboCup RRL sees over 100 teams of undergraduate and graduate students and researchers from around the world compete in regional competitions. These culminate in between 15 and 25 international teams competing over a week of intense competition, development, evaluation, and collaboration. These competitions ''test the test methods'' in the presence of cutting edge, experimental technologies.

The competition takes place in an arena that represents a building in various stages of collapse. It consists of a variety of standard test method apparatuses for response robots and an example is shown in Figure 1. Many of the test methods and apparatuses that make up current ASTM standards started in these competitions and were tested and refined at these events. The goal of each team is to reach simulated victims, report their state, and build a map that would allow a human rescuer to reach them. The victims are strategically placed such that in order to reach them, robots must overcome the test methods. To add structure to the competition, the arenas are separated into three sub-arenas, denoted by the colors Yellow, Orange, and Red and representing challenges posed by Autonomy, Structured Obstacles, and Advanced Mobility Terrain.

The RoboCup RRL is unique in its emphasis not on finding a champion, but rather on building a community that works together to advance the state of the art in Urban Search and Rescue robotics: ''A League of Teams with one goal: to Develop and Demonstrate Advanced Robotic Capabilities for Emergency Responders.'' To this end, the competition is carefully administered to encourage the participation of a wide cross-section of the developer community that has something to contribute to this application domain, even if they do not have the resources for a championship team.

Specialized teams, which often produce highly sought-after Best-in-Class solutions to particular challenges in this domain, rarely have the broad based resources to compete well across the whole competition. As we will discuss, mechanisms that encourage teams to collaborate and special awards for Best-in-Class solutions, have resulted in many such specialized teams competing. From the perspective of the standards process, the participation of these teams is highly desirable as exposure to a wide variety of the best approaches to each test method have proven to be invaluable in their refinement. These specialized developers are often able to suggest, and in many cases demonstrate, many alternative approaches to the test method, and contribute to improving its ability to properly represent real world performance. In some cases these specialist developers have become part of the standards process. A particular advantage to incorporating these groups into the standard process is that, as academic institutions, they often have more freedom to experiment and develop innovative implementations to solve problems, without the commercial pressures and restrictions that affect commercially developed solutions. In the background, their exposure to the test methods encourages development towards the operationally relevant requirements embedded in the test methods and encourages the dissemination of the test methods through their adoption by the academic community.

The broader goals of promoting and advancing the state of the art are also well served by supporting such a vibrant community of robot developers as significant cross-pollination of ideas and capabilities occurs at such events. Many of the more general teams have begun to demonstrate capabilities that were formerly only demonstrated by highly specialized teams; conversely several highly specialized teams have begun to branch out into more general capabilities. As they gain a greater understanding of the challenges faced by the first responders, through their exposure to the test methods, they have also become more involved in the test method development process. In Section 4 we will discuss how this has included bringing advanced robotic hardware to robot evaluation exercises to demonstrate to first responders, government, and robot developers what the state of the art may look like in the future.

In the rest of this section we will briefly outline the way in which the competition is run; for a more detailed discussion please refer to the League Overview [8], Rules Outline [6], and Arena Construction Manual [10]. We will then discuss, in Section 3, some salient examples of test methods that were conceived or refined during the competition, while in Section 4 we will discuss examples of further integration between the competition and the wider standards process.

2.1 Preliminary Missions

The RoboCupRescue Robot League is a point-scoring exercise. Teams run several time-limited ''missions'' within the

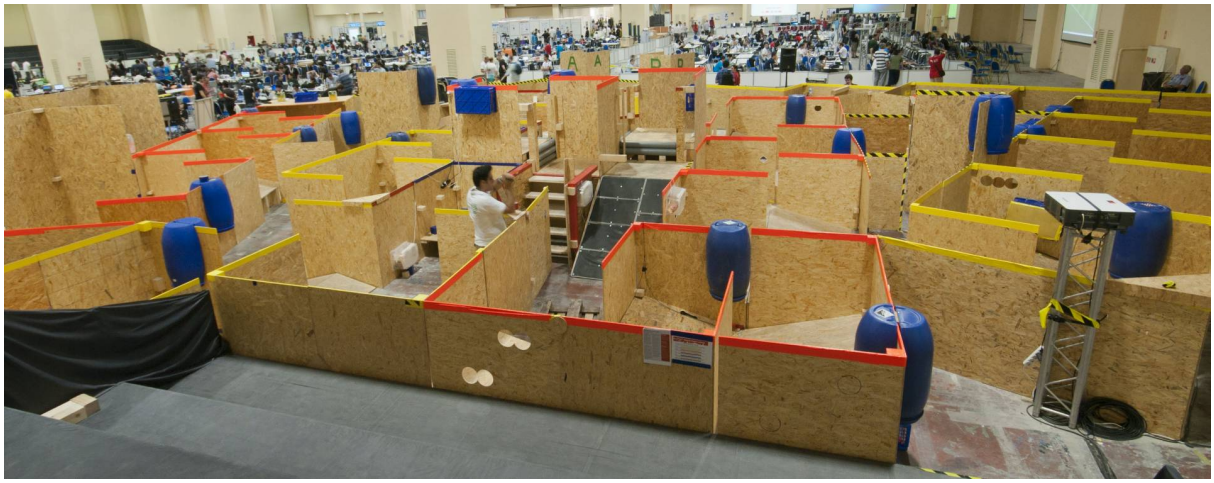


Figure 1: The RoboCupRescue Robot League arena from the 2011 International competition, held in Istanbul, Turkey. Several test method apparatuses may be seen in this photo. The blue barrels are the fiducials used in the Map Quality metric, together with the walls that form the maze. These will be discussed in Section 3.5. The ramp, stairs and raised platform in the center, along with the mismatched ramps in areas highlighted with orange tape, form the structured obstacles of the Orange arena. The continuous pitch-and-roll ramps highlighted by walls with yellow tape form the Yellow arena. The boxes hung against the walls house victims that robots must reach and are discussed in Section 3.4.

arena over the course of several preliminary, semi-final, and final rounds. In each mission, they deploy robots in order to find and characterize the victims, which are distributed throughout the arena. Teams are awarded points based on the quality of the information that they obtain about each victim. This includes the ability to bring back high resolution imagery of the victim, take their temperature or return a thermal image, sense the presence of carbon dioxide, and detect if the victim is speaking. Points are also awarded for the quality of the map in which the victim is reported.

Depending on the number of teams, all teams participate in four to six Preliminary missions, each taking around 15 to 20 minutes. These missions are held in half of the arena, allowing two teams to run missions concurrently. This gives all teams the best opportunity to demonstrate their capabilities to their full potential and allows them to gain experience with the test method apparatuses. In the process, valuable data is generated on the performance of a wide variety of robots in the test apparatuses.

2.2 Championship

The Preliminary missions act as a qualifying round for the Championship, which selects the Champion, 2nd, and 3rd place teams. Usually run as a Semi-finals and Finals round, the Championship takes place in an arena twice the size of that in the preliminary missions, giving teams a greater incentive to rapidly cover as much of the arena as possible. The determination of the Championship is also based on points, set to zero for all teams at the start of the Championship and earned in the same way as in the Preliminaries.

A unique aspect of the League is in the qualification process, which aims to be as inclusive and forgiving as possible to ensure that teams have the freedom to experiment and push their implementations. In the process, teams are able to push the test method apparatuses to their limits. The qualification process ignores the worst of each team's preliminary missions and the qualification cutoff is decided once the distribution of preliminary scores is known. While this means that the number of teams in the Championship is variable, it ensures that there is a clear performance gap between the best performing eliminated team and the worst performing qualified team.

To further encourage the participation of specialist teams, which often fail to qualify due to their narrow focus, and to promote the dissemination of Best-in-Class implementations, the League encourages qualified teams to combine with a team that was eliminated and progress through the Championship as a joint team. On winning or placing, awards are given to both teams.

2.3 Best-in-Class Awards

The Best-in-Class awards, which rank equal in status to the championship, are designed to reward the demonstration of Best-in-Class performance in specific challenges posed by Urban Search and Rescue robotics. Currently, there are three Best-in-Class awards for Mobility, Autonomy, and Manipulation. Each of these awards is also decided on the basis of points, half of them coming from the demonstration of the relevant capabilities in the preliminary missions and half coming from a special Best-in-Class round of the competition, for which an entire day is often dedicated.

2.3.1 Best-in-Class Mobility

The Best-in-Class Mobility award is given to the team that demonstrates proficiency in the test method apparatuses relating to advanced robot mobility. Half of the score for this award is based on the number of victims located by the team in the Red part of the arena during the preliminary missions. This part of the arena tests the mobility of the robots and consists of stepfields, which will be discussed in detail in Section 3.1. The second half of the score is based

on points scored during a special Best-in-Class Mobility run, the nature of which varies from year to year to expose new test method apparatuses to a variety of robots and highly motivated operators. In recent years, this run has been based on the number of laps of the Mobility: Stepfields standard test method within a 10 minute time limit, with the operator out of sight of the robot. In this way the Mobility run becomes an iteration of the standard test. Prior to that, points have instead been awarded for the number of times they can traverse particular test method apparatuses in a 10 minute time limit. For example, traversing the Mobility: Obstacles: Stairs, the Mobility: Obstacles: Hurdles, the Mobility: Inclined Plane, and each pallet of the Mobility: Terrains: Stepfields in each direction would earn a point each.

2.3.2 Best-in-Class Autonomy

The Best-in-Class Autonomy award is given to the team that demonstrates proficiency in challenges relating to autonomous navigation, autonomous detection of objects of interest, and autonomous map building. Half of the score for this award is based on the number of victims found by completely autonomous systems in the preliminary missions. The second half is based on the combined quality and coverage score resulting from a dedicated Best-in-Class Autonomy run through an enlarged maze. This is a direct application of the Map Quality metric [7], currently under development as a standard test method. Autonomous robot mapping is still a highly specialized area so the competition provides a valuable opportunity to gather data from a wide variety of very different approaches from all over the world on the same apparatus and perform rapid iterative refinement as part of the standard test method development process. It also involves researchers into the process; most of the teams that participate in this challenge have also been active in providing input to the standard test method development process.

2.3.3 Best-in-Class Manipulation

The Best-in-Class Manipulation award is given to the team that demonstrates proficiency in challenges relating to manipulating objects in the arena. Half of the score is determined based on the number of objects that teams are able to place with victims in the arena -- teams may retrieve objects, representing such things as radios, water, or supplies, from a shelf and place them with victims that they find. This task by itself is analogous to the placement task in the Manipulation: Grasping Dexterity test. This leads in to the second half of the score, where teams must retrieve objects from one shelf and place them in holes in another shelf as many times as possible in a fixed time period.

3. EMBEDDED TEST METHODS AND APPARATUSES

The field of Urban Search and Rescue provides many challenges. The test method apparatuses represent these challenges, as gathered through extensive consultation with first responders and distilled into separate, reproducible physical challenges. Many of these appear in the RoboCupRescue Robot League arena. Due to the variety of challenges, it is rare for a single team to be able to perform well across all of them. Indeed, it is often the case that good performance in a particular set of challenges is an open research problem and the team that demonstrates best-in-class performance in those challenges needs to dedicate all of their effort and expertise towards solving that particular challenge.

Emerging, draft, and standard test method apparatuses that have appeared, in full or adapted form, in the RoboCupRescue Robot League arenas over the past years include those for the following ASTM standard, validating (V), balloting (B), and prototyping (P) test methods:

- Confined Area Terrains: Continuous Pitch/Roll Ramps (ASTM E2826)
- Confined Area Terrains: Crossing Pitch/Roll Ramps (ASTM E2827)
- Confined Area Terrains: Symmetric Stepfields (ASTM E2828)
- Confined Area Obstacles: Hurdles (ASTM E2802)
- Confined Area Obstacles: Inclined Planes (ASTM E2803)
- Confined Area Obstacles: Stair/Landings (ASTM E2804)
- Confined Area Inspection Tasks: Recessed Targets on Elevated Surfaces (V) (WK27851)
- Confined Area Grasping and Removal Tasks: Weighted Cylinders on Elevated Surfaces (V) (WK27852)
- Search Tasks: Random Mazes with Complex Terrain (B) (WK33259)
- Navigation Tasks: Random Mazes with Complex Terrain (V) (WK33260)
- Mapping Tasks: Hallway Labyrinths with Complex Terrain (P)
- Video: Acuity Charts and Field of View Measures (ASTM E2566-08)
- Audio: Speech Intelligibility (Two-Way) (V) (WK34435)
- Thermal Imager Resolution (P)

In the rest of this section, we will discuss several salient examples of standard test methods that were conceived at, or saw rapid development within, the RoboCupRescue Robot League competition, and which have subsequently become standard test methods or are in the process of becoming standard test methods.

3.1 Symmetric Stepfields

Stepfields are blocks of wood of specified lengths, arranged in a grid such that the tops form an uneven surface. They represent rubble, steps, and other arbitrary terrain in a way that is easy to reproduce [5]. First used to analyze the movement of cockroaches [4], stepfields have since been used in robot evaluations to evaluate the performance of robots of all sizes. A major challenge in the use of stepfields as a standard test method apparatus is specifying the dimensions and standard configuration. Since 2005, the RoboCupRescue Robot League has played a vital part in the evolution of the patterns up to the present day standard. Several iterations of the stepfield appear in Figure 2.

The initial Random Stepfield apparatuses, sized for Urban Search and Rescue robots, consisted of blocks with a footprint of 10x10 cm and varying in length from 5 cm to 40 cm. Early iterations of the stepfields consisted of these blocks arranged in a 10x10 pallet with a prescribed pattern of tall blocks and surrounded by randomly placed blocks, following the rule that adjacent blocks should differ by no more than 20 cm. The tall blocks therefore form ridges or pillars that the robots

(a) (b) (c) (d)

Figure 2: The evolution of the Stepfields standard test method apparatus through the course of the RoboCup competitions. (a) The initial Random Stepfield, as introduced in the 2005 competition. (b) The Symmetric Stepfields, the patterns for which were developed and refined during the 2008 competition. (c) The latest version of the Symmetric Stepfields, adapted for competition. (d) The final form of the Symmetric Stepfields in the test method.

need to navigate around or over and represent larger objects such as pipes or large rocks among smaller, random rubble. The fluid nature of the RoboCupRescue Robot League arena, the large number of missions and the wide variety of robot geometries enabled an immediate comparison between the different schemes and the ability to identify and address shortcomings in the design of the apparatus that were not apparent until they were exposed to particular, unusual robot geometries. Through the course of several competitions, different variants evolved.

These random stepfields proved to be very effective in producing a challenging terrain for advanced mobility robots and helped shape the evolution of the robot geometries from those that were mostly optimized for climbing stairs and curbs to those that could also handle more general rough terrain. However, their randomness impaired the repeatability of the trials and made it difficult to use as a standard test method apparatus. To overcome this, in the 2008 competition symmetric stepfields were introduced where the entire stepfield pallet consisted of a prescribed, symmetric pattern. The competition enabled the evaluation and refinement of a variety of patterns over the course of over 100 missions with a wide range of robot geometries. The final incarnation of the Stepfields test method apparatus appeared in the 2010 competition. The terrain and figure-of-8 pattern, which now forms ASTM Standard E2828 [3], was tested both in the main competition as well as in the Best-in-Class Mobility competition.

3.2 Continuous Pitch/Roll Ramps

In its early days of the RoboCup RRL, the competition was dominated by wheeled robots, usually variants of floor robots used in the lab for research into navigation and planning. In order to make the environment more challenging and closer to what might be encountered in the real world, continuous pitch-and-roll ramps were introduced. Several incarnations of this apparatus appear in Figure 3. These ramps force robots to demonstrate sufficient power and control to position themselves on a non-flat surface, enough degrees of freedom to direct perception when the base is not horizontal, and 3D-aware sensing and mapping in order to generate maps that are correctly registered despite the attitude of the robot changing. However, they are not so hard as to act as a barrier to entry for teams that are not specialized in mechanical engineering.

The competition provided a vital proving ground for the pitch-and-roll ramps, where different layouts could be tested relative to targets that the robots had to approach and inspect or paths that the robots needed to traverse. Different heights of ramps were also tried before the current standard was settled on. Much experience was gained from observing a wide variety of robot geometries perform, from tracked to wheeled to legged robots, performing in the arena. This experience has guided subsequent test method development incorporating pitch-and-roll ramps [1].

3.3 Crossing Pitch/Roll Ramps

Crossing pitch-and-roll ramps, shown in Figure 4, are an evolutionary branch from the continuous variety, first introduced in the 2008 competition in response to the need to develop a terrain that of a difficulty between that of the continuous pitch-and-roll ramps and the stepfields. The resulting terrain should be traversable by wheeled robots only if driven carefully, providing an incentive for teams that were focusing on autonomy to add terrain analysis and more advanced autonomous terrain negotiation.

Once again the fluid nature of the competition arena allowed the rapid evaluation of a wide variety of configurations of the crossing ramps and the ability to observe the way in which a variety of different robot geometries and control methodologies responded to them. These observations have shaped the final crossing ramps test method apparatus that now appears as a middle difficulty test terrain in the standard test method suite [2].

3.4 Inspection Tasks, Acuity Charts, and Thermal Imager Resolution

Points in the main RoboCupRescue Robot League competition are scored based on the robots ability to get close to the "victims" in the arena and obtain information about their state. Examples of victims are shown in Figure 5. All teams make extensive use of visible light cameras and many teams make use of novel thermal sensing techniques.

All of the teams currently in the League are academic teams or in some way associated with universities and high schools. The sensing abilities of the robots that they develop differ significantly from those encountered in the field because as previously mentioned they are significantly less constrained by commercial and practical limitations and are instead focused on research into particular areas of specialization. This has resulted in a wide variety of exotic and experimental sensors, coupled with innovative ways of transferring this

(a) (b) (c)

Figure 3: The evolution of the continuous pitch-and-roll ramps. (a) Their first appearance in 2006. (b) Their subsequent refinement and use throughout the arena. (c) Continuous ramps now appear in many of the standard tests.

(a) (b) (c)

Figure 4: The evolution of the pitch-and-roll crossing ramps. (a) Their conception during the 2008 competition, where they were called "Wacky World". (b) A different form during the 2009 competition matching full and half height ramps. (c) Their final form in the standard test method.

(a) (b) (c)

Figure 5: Examples of victims placed throughout the arena as targets for teams to reach, identify, and localize in their map. (a) An open victim in the stepfield terrain. (b) An open victim in a car being inspected by a robot. (c) A victim hidden in a wall, accessible through holes. The visual acuity eyechart and heating pad (white, back of box) can be seen; the doll in the foreground acts as a secondary visual target and provides an occlusion that shadows the heating pad for ad-hoc thermal imager evaluation.

data, fusing it, and presenting it to the remote operator.

Exposing the test method apparatuses to these robots yields a glimpse at how deployed robots of the future may perform and ways in which the tests may need to scale. For example, several of the top performing robots in the League have vision capabilities that far exceed that of humans, a capability that only exists in the largest of commercially available response robots. Several teams are also experimenting with unusual thermal sensors, some of which detect objects of interest without producing conventional images. Making thermal imager tests relevant to these classes of devices ensures that new sensors that may become widespread in the near future can be meaningfully compared with those that responders are already familiar with.

Another innovation that is well developed in the league but almost unheard of in deployed robots is assistive and full autonomous behavior. These range from controllers to help steady a camera and assist the operator in directing it to where they desire, right through to vision algorithms that can detect, recognize, and interpret objects of interest in the scene. By encouraging teams to incorporate these developments into their robotic entries and fielding them in the emerging test methods, the test method development process gains valuable early insight into the capabilities that are possible and may soon become available. This helps to ensure that when these capabilities are being fielded and marketed that the standards are ready for them. For example, there are now draft test method apparatuses available that test the visual acuity of autonomous and semi-autonomous systems, in a way that is directly comparable to that of teleoperated systems.

An equally valuable side-effect of this insight is that knowledge of these developments can be passed back through the standards process to the first responders, whose needs direct the whole standards process. It is often the case that their requirements are unmet by commercially available robots. Yet, unbeknownst to both responders and the commercial developers, such problems may have already been solved in the research community and are just waiting for commercialization. This is particularly important in the sensing, inspection from mobile platforms, autonomy, and human-robot interaction fields where the abilities shown by implementations in the lab far exceed those currently in deployment. A push from an end user may be all it takes for a robot vendor to bring such developments to life.

3.5 Human-Robot Interfaces, Mapping, and Autonomy

The arena is arranged as a labyrinth, or maze, of hallways that teams must navigate, with test methods embedded at strategic points; between the test methods the robots must navigate portions of maze consisting of mostly continuous pitch-and-roll ramps. Three groups of test methods make use of the maze: Human-robot interfaces, mapping, and autonomy.

Robots that provide their operators with good levels of situational awareness through their human-robot interface, and which respond to the operator's controls in an appropriate manner, tend to perform well in the maze. This is because they are able to drive through the maze without colliding with the walls, a task that is made more difficult due to the introduction of continuous pitch-and-roll ramps, which can make the robot behave in an unpredictable manner. Particularly good user interfaces and predictable controls, especially those that overlay some autonomous behaviors such as automatically moving downrange without colliding with obstacles, also allow the operator to cut corners closer than they might otherwise, further improving their performance. In contrast, operators using interfaces that do not provide good situational awareness tend to misjudge the positions of corners relative to the robot's edges and thus waste time colliding with the walls or taking wide or slow turns.

The maze is also used to test the ability for robots, teleoperated or autonomous, to build 2D and 3D maps of their environments using a variety of algorithms and sensors such as laser range scanners, range imagers and lidars, and various forms of structure-from-vision techniques. For the purpose of evaluating these abilities, the maze is augmented with fiducials [7] that allow various metrics, such as map coverage and consistency, to be measured in a quantitative manner. Finally, the maze is used to evaluate the performance of robots with the ability to autonomously navigate, search, and map a complex environment.

The RoboCup RRL plays a particularly important role in the development of this test method because many of the capabilities being tested -- advanced human-robot interfaces, robot mapping, and autonomous navigation combined with robot platforms that are able to overcome non-flat flooring -- are almost exclusively available only in research robots.

4. INTEGRATION IN THE STANDARDS PROCESS

Participation by the RoboCup RRL Community in the standards process extends beyond the competition. The standards process is enhanced by the involvement of RoboCup RRL teams at response robot evaluation exercises, teaching camps, and in the standards development process itself. The latter is a natural fit as most teams replicate a subset of the standard test method apparatuses in their own labs in order to practice and aid their research.

4.1 Response Robot Evaluation Exercises

NIST hosts response robot evaluation exercises that bring together robot developers, researchers, first responders, procurement officials, and test method developers and administrators. These events, usually held at fire and rescue training facilities, see developers bringing robots to be tested in current and emerging test method apparatuses as well as more unstructured, operationally significant scenarios. First responders and procurement officers observe the robot performances and experience them hands-on, under test conditions, and within the operationally significant scenarios. In the process, data is collected on robot performance in the test methods which allow them to be further developed, refined, and validated.

The Best-in-Class winners of the RoboCup RRL are invited to bring their robots and equipment to these events, in order to give robot developers, responders, and procurement officials a valuable glimpse at the performances that are possible within the standard test methods. This allows them to put the results of deployable robots into a proper context, relative to what is possible based on emerging technology. It also provides data points for the test methods that are often well beyond those achieved by deployed robots.

4.2 Teaching Camps and Summer Schools

Several key features are incorporated into the competition that encourage teams with a wide variety of specializations to enter and to collaborate with each other, investigate the test methods, and contribute to their testing and development. However, it is still a high pressure environment, with teams usually focused primarily on ensuring their entries do well. Ironically, the competition structure, which gives all teams a chance at competing right to the final day so that they have the best chance of showing off their capabilities, also means that teams don't usually have much truly free time. The RoboCupRescue Robot League hosts teaching camps and summer schools several months after the competition, that allow competitors to reflect on and become more familiar with the test methods and the best-in-class implementations that were demonstrated in them. These events are also a vital part of the standards development process as teams, who are encouraged to bring their robots, are able to experiment with the test methods in greater detail and with more freedom than at the competition.

4.3 Standards Process Involvement

Virtually all teams that compete in the RoboCup RRL fabricate at least some of the standard test method apparatuses in their own labs. As noted earlier, this is a very effective way of disseminating the use of the standard test methods through academia. This is further amplified by the inevitable sharing of the facilities that happens at academic institutions, resulting in the standard test methods being used in projects that are not directly related to the competition. As these results are published, the test methods become known to research communities outside the standards process itself. As teams become even more intimately familiar with the test methods, they have also become involved in the test method development process and in many cases team members have subsequently worked directly with NIST on developing standard test methods [7]. There have also been examples of new proposed test methods coming up through the RoboCup process [9]. Some labs that participate in the competition have even opened standard testing facilities in their home countries, based on the standard test methods.

5. FUTURE DIRECTIONS AND CONCLUSIONS

The RoboCup RRL continues to see new test method apparatuses rotating in for exposure to the robotic implementations that teams bring and refine at the event. Test methods for autonomy, 2D mapping, and 3D mapping are being further developed with the assistance of expertise from the RoboCup RRL community -- one of few with such a wide variety of expertise in this area. Likewise, improvements to test methods for visual acuity and other vision based sensing are being made with assistance from the League. Existing and new test methods will continue to be refined through the competition, which continues to see new teams joining and contributing their expertise.

Planning is also underway for the development of new test methods for entirely different classes of response robots, that of robots for fighting fires in the home, and the RoboCup RRL will serve as an integral part of this test method development effort. It will integrate currently prototyping sensing and autonomy test methods with new test methods specific to domestic early fire intervention such as the detection of fire-specific signs and the simulated delivery of suppressant.

The RoboCup RRL has been, and continues to be, an effective tool for aiding the development of standard test methods for response robots. In particular, it provides a venue where current and prototypical test method apparatuses and procedures may be evaluated in the presence of a wide variety of implementations, it brings researchers into contact with the test methods and encourages them to assist in their dissemination, and it allows them to contribute their expertise to the test method development process. The competition also assists the wider effort of NIST in promoting research and development in capabilities for robotic Urban Search and Rescue equipment.

6. REFERENCES

[1] ASTM International. E2826 Standard Test Method for Evaluating Emergency Response Robot Capabilities: Mobility: Confined Area Terrains: Continuous Pitch/Roll Ramps. Technical report, ASTM International, 2011.

[2] ASTM International. E2827 Standard Test Method for Evaluating Emergency Response Robot Capabilities: Mobility: Confined Area Terrains: Crossing Pitch/Roll Ramps. Technical report, ASTM International, 2011.

[3] ASTM International. E2828 Standard Test Method for Evaluating Emergency Response Robot Capabilities: Mobility: Confined Area Terrains: Symmetric Stepfields. Technical report, ASTM International, 2011.

[4] R. J. Full, K. Autumn, J. I. Chung, and A. Ahn. Rapid negotiation of rough terrain by the death-head cockroach. *American Zoologist*, 38:81A, 1998.

[5] A. Jacoff, A. Downs, A. Virts, and E. Messina. Stepfield Pallets: Repeatable Terrain for Evaluating Robot Mobility. In *Performance Metrics for Intelligent Systems Workshop*, 2008.

[6] A. Jacoff, S. Tadokoro, E. Mihankhah, T. Kimra, J. Pellenz, A. Birk, J. Suthakorn, M. Hofbauer, and A. L. Gutierrez. RoboCupRescue Robot League: Rules 2011.2. `http://www.nist.gov/el/isd/upload/Robocup_Rules_2011.pdf`, 2011.

[7] S. Schwertfeger, A. Jacoff, C. Scrapper, J. Pellenz, and A. Kleiner. Evaluation of Maps using Fixed Shapes: The Fiducial Map Metric. In *Performance Metrics for Intelligent Systems Workshop*, 2010.

[8] R. Sheh, T. Kimura, E. Mihankhah, J. Pellenz, S. Schwertfeger, and J. Suthakorn. The RoboCupRescue Robot League: Guiding Robots Towards Fieldable Capabilities. In *International Workshop on Advanced Robotics and Social Impacts*, 2011.

[9] K. Shimaoka, K. Ogane, and T. Kimura. An Evaluation Test Field Design for a USAR Robot considering a Collapsed Japanese House. In *International Symposium on Safety, Security and Rescue Robotics*, 2011.

[10] A. Virts, A. Jacoff, and A. Downs. RoboCupRescue Arena Assembly Guide 2011. `http://www.nist.gov/el/isd/upload/2011_Assembly_Guide.pdf`, 2011.

Validation of the dynamics of an humanoid robot in USARSim

Sander van Noort
Intelligent Systems Lab Amsterdam
Science Park 904
1098 XH Amsterdam, NL
+31205257460
Alexander.vanNoort@student.uva.nl

Arnoud Visser
Intelligent Systems Lab Amsterdam
P.O. Box 94323
1090 GH Amsterdam, NL
+31205257532
A.Visser@uva.nl

ABSTRACT

This paper describes a model to replicate the dynamics of a walking robot inside USARSim. USARSim is an existing 3D simulator based on the Unreal Engine, which provides facilities for good quality rendering, physics simulation, networking, a highly versatile scripting language and a powerful visual editor. To model the dynamics of a walking robot the balance of the robot in relation with the contact points of the body with the environment has to be calculated. To guarantee a fast frame rate several approximations in this calculation have to be tried, and the performance (both in dynamics and computational effort) is evaluated in a number of experiments. This extension is made and validated for the humanoid robot Nao. On this basis many other applications become possible. A validated simulation allows us to develop and to experiment with typical robotic tasks before they are tested on a real robot.

Categories and Subject Descriptors

I.2.9 [**Artificial Intelligence**]: Robotics—*Kinematics and dynamics*; I.3.5 [**Artificial Intelligence**]: Computational Geometry and Object Modeling—*Physically based modeling*; I.6.4 [**Simulation and Modeling**]: Model Validation and Analysis

General Terms

Design, Verification, Performance

Keywords

simulation, NAO, dynamics, collisions

1. INTRODUCTION

Robotic simulation is essential in developing control and perception algorithms for robotics applications. Simulation creates the environment with known circumstances, which allows rapid prototyping of applications, behaviors, scenarios, and many other high-level tasks. Robot simulators have been always used in developing complex applications, and the choice of a simulator depends on the specific tasks we are interested in simulating. Yet, the level of realism of a simulator is also important in this choice.

A 3D simulator for mobile robots must also correctly simulate the dynamics of the robots and of the objects in the environment, thus allowing for a correct evaluation of robot behaviors in the environment. Moreover, real-time simulation is important in order to correctly model interactions among the robots and between the robots and the environment. Since simulation accuracy is computationally demanding, it is often necessarily an approximation to obtain real-time performance.

In this paper the focus is on the humanoid Nao robot, which is selected by the RoboCup organization as the standard platform for the Soccer competition. This robot (see Fig. 1) is widely used in many research institutes around the globe.

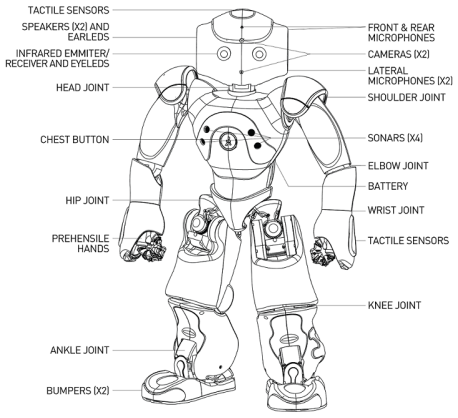


Figure 1: Schematic overview of the Nao (Courtesy of Aldebaran Robotics).

A model is described to replicate the dynamics of the Nao robot in USARSim[2]; an existing 3D simulator based on the Unreal Engine. Inside USARSim robots are simulated on the sensor and actuator level, making a transparent migration of code between real robots and their simulated counterparts possible. USARSim is an open source project, available on

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
PerMIS'12 March 20-22, 2012, College Park, MD, USA.
Copyright 2012 ACM ACM 978-1-4503-1126-7-3/22/12 ...$10.00.

sourceforge[1]. It includes a powerful editor to create worlds and allows experiments, benchmarks and competition scenarios to be set up easily.

2. RELATED WORK

There are many robotic simulator platforms available. The simulators listed here are selected because they provide support for the Nao robot (shown in Fig. 2).

2.1 NaoSim

NaoSim is the official 3D simulator supported by Aldebaran. The simulator is based on the game development framework Unity[2] and is developed by Cogmation Robotics[3]. NaoSim is closed source and uses Nvidia PhysX as a physics engine.

The Nao is controlled using the NaoQi framework, which is the native interface of the Nao. This means that the same code can be used for both the real and simulated Nao. Furthermore the user can manipulate the Nao (move, rotate, etc) and add basic primitives (cubes, spheres, triangles, etc).

A downside is that, currently, it is not possible to create custom environments or simulate more than one Nao in NaoSim. Another potential downside is that the simulator is specifically developed for the Nao robot and as a result no heterogeneous teams of robots can be simulated.

2.2 SimSpark

SimSpark[4] is the official 3D RoboCup simulator and is primarily made for this goal. SimSpark is used as the official simulator in the RoboCup 3D Soccer Simulation League. The simulator is open source and freely available. It uses a client-server architecture, where agents (i.e. robot controllers) are the clients that communicate with the simulation server. Several robots (including the Nao) are supported and SimSpark makes it easy to add new robots with `rsg` files that describe the physical representation of a robot.

SimSpark always starts a football simulation, including a soccer field, game states and referee. The robots are controlled using a custom protocol, not the native interface of the Nao.

Noteworthy is the abstraction of the physics layer, which is supposed to make it easy to switch between different physic engines[5]. Currently SimSpark only supports *Open Dynamics Engine* (ODE) as physics engine.

2.3 Webots

Webots[5] is a commercial closed source robot simulator for educational purposes[7]. It uses the ODE physics engine for the simulation of the dynamics of the robots.

A Webots simulation is composed of a world, one or several controllers and optional physics plugins to modify the regular physics of Webots. A world describes the environment and the properties of the robots. Using the included world editor new environments can be made.

Controllers are programs to control the robots in those worlds. These controllers are started as separate processes and have limited privileges in terms of interacting with the simulation. Multiple robots and controllers can be used at the same time in Webots.

Webots also includes a controller that allows us to connect with the simulated Nao robot using the NaoQi framework.

2.4 SimRobot

SimRobot is a free open source general robot simulation and uses ODE as physics engine[6]. SimRobot consist of several modules linked to a single application, which differs from the commonly chosen client/server based approach. This approach offers the possibility of halting or stepwise executing the whole simulation without any concurrency.

The specification of the robots and the environment (*simulation scene*) is modeled via an external XML file and loaded at runtime. This xml file uses the specification language *RoSiML* (Robot Simulation Markup Language), which was developed in an effort to create a common interface for robot simulations.

Controllers allow us to command the robots and implements a sense-think-act cycle and is called each step by the core component of the simulation to read the commands for the robot it controls.

SimRobot is an initiative of a team from the RoboCup Standard Platform League, B-Human, and they provide more information in their Team Report and Code Release[8].

Figure 2: Screenshots of the different simulators in action: NaoSim (top left), SimSpark (top right), Webots (bottom left), SimRobot (bottom right)

3. SIMULATION MODEL

The RoboCup version of the Nao (H21 model) has 21 joints, resulting in 21 degrees of freedom (DOF). There is also an academic version with 25 degrees of freedom, which has 2 additional DOF in each hand. See Fig. 1 for a complete schematic overview of the Nao robot.

The movement of each joint can be described by a rigid body equation[1]. The first step is to definition of unconstrained motion as described in equation (1). This equation

[1]http://usarsim.sourceforge.net
[2]http://unity3d.com/
[3]http://www.cogmation.com/naosim.html
[4]http://simspark.sourceforge.net
[5]http://www.cyberbotics.com/
[6]http://www.informatik.uni-bremen.de/simrobot/

contains four vectors, it takes both the spatial information $x(t)$, $R(t)$ and the linear and angular momentum $P(t), L(t)$ into account. $F(t)$ and $\tau(t)$ are external forces and the input to solve this equation. The linear and angular speed $v(t)$, $\omega(t)$ can be derived from the linear and angular momentum when the total mass M and the inertia tensor $I(t)$ of a rigid body is known.

$$\frac{d}{dt}Y(t) = \frac{d}{dt}\begin{bmatrix} x(t) \\ R(t) \\ P(t) \\ L(t) \end{bmatrix} = \begin{bmatrix} v(t) \\ \omega(t)^* R(t) \\ F(t) \\ \tau(t) \end{bmatrix} \quad (1)$$

The inertia tensor $I(t)$ is time dependent, but can be calculated from the inertia tensor I_{body} in body space, which is a fixed property, by taking the orientation of the body into account $I(t) = R(t)I_{body}R(t)^T$.

$$\left[\; v(t) = \tfrac{P(t)}{M}, \quad \omega(t) = I(t)L(t) \; \right]^T \quad (2)$$

The next step is to take contacts into account. When the rigid body encounters a contact it imposes a constraint on the movement.

Two different types of contacts can be distinguished. The first is a contact caused by bumping into another rigid body or into the world. The other type of contact is caused by having a joint defined between two rigid bodies.

3.1 PhysX Dynamics

Nvidia PhysX is the underlying physics engine of Unreal and USARSim. A physics engine gives an approximate simulation of rigid body dynamics (or any other physical related system). In PhysX a simulation is executed within a scene. A scene is basically a container for actors, joints and effectors. It allows the user to simulate multiple scenes in parallel without objects influencing each other over large distances.

The simulation of a scene is advanced one time step at a time. Advancing a time step means the properties of the objects in the simulation change (i.e. the position and velocity of the objects). The choice of the time-step settings is important for the stability of the simulation. In general longer time steps lead to poor stability in the simulation, while shorter time steps can lead to poor system performance.

The motion of a rigid body can either be constraint by contacts (with the static world or other rigid bodies) or joints. The PhysX constraint solver limits the motion of rigid bodies (and satisfies the constraints) by reiterating the constraints a number of times.

The following three important aspects of PhysX are highlighted: actors, materials and joints. Collision detection is described in Section 3.3.

3.1.1 Actors

Actors define objects that are capable of interacting with the world and other objects. In PhysX actors can have two roles: static objects (fixed in the world reference frame), or dynamic rigid objects. Importantly, actors can have a shape assigned, which is used for collision detection. Static objects (like the environment) always have a shape assigned, since they are only used for collision detection. Rigid objects on the other hand do not always need to have a shape. In this case they represent an abstract point mass (can serve as connections between joints) and the properties of the rigid body must be assigned manually.

An object is represented by an inertia tensor I_{body} and by a point of mass M located at the center of mass. The inertia tensor describes the rigid bodies' mass distribution. For our simulated Nao robot, care has been taken so that each body part has the actual mass as specified in Aldebaran's documentation[7].

3.1.2 Materials

Materials describe the surface properties of actors. These properties are used when two actors collide. The result of a collision will influence the simulation and result in the actors bouncing, sliding, etc.

3.1.3 Joints

Joints connect two rigid bodies and limit the movement between those two bodies. How the movement is limited is specified by the type of joint. PhysX supports a large number of different joints including Revolute, Prismatic and 6 Degrees of Freedom Joint (which can again be configured to any of the earlier joints).

3.2 Joint definition and convention

As said in the previous section, a joint connects two rigid bodies and limits the movement in some way. The type of movement limitation results in different types of joints, like a rotational joint, translational joint (also called prismatic joint), spherical joint, screw joint, etc.

A rotational joint, also called *revolute joint*, is as the name suggests capable of rotating around an axis. This type of joint allows one degree of freedom (DOF) between the two rigid bodies, namely the range of motion around the specified axis. In case of this type of joint the motion is usually also limited to a specified range around the axis.

It is important how the relative position and orientation of the frames is characterized. A commonly used convention to describe this is the *Denavit Hartenberg* (DH) notation. This convention uses homogeneous transformation matrices to describe the relative positions of the frames (coordinate systems). This convention is used in USARSim. A full description can found in the book Robotics, chapter 2.2.10, by K.S Fu *et al.*[4].

3.3 Collision Detection

The Unreal Engine is designed to build multi-user games, which means that they apply an approach called the *generalized client-server model*. The task of networking is to keep the world state synchronized between the different users. In the case of *generalized client-server model* there is a server that is authoritative over the evolution of the world state and only the server knows the true state of the world. Clients maintain an accurate local subset of the world state and predict change of the world state by executing the same code as the server. Servers then need to send information about the world state to the client to correct the client world state, which is smaller than when the server would need to send full updates. The problem of approximating the world state between server and client is called *replication* by Unreal Engine.

This networking model implies the physics simulation runs on both the server and client, where the physics simulation on the server represents the true state of the simulation. The

[7] http://users.aldebaran-robotics.com/docs/site_en/reddoc/hardware/masses_3.3.html

server will send updates about the rigid body states to the client. In the case of the Unreal Engine such a state consists of the position, orientation, linear velocity and angular velocity.

Each client has its own scene, which contains actors, joints and effectors. Actors are world related objects which can interact with the world and other actors. Actors are *ticked* once per frame. During such a *tick* they can update their logic, including their physics.

The PhysX engine is only one component of the Unreal Collision engine. There are actually various physics modes which allow actors to move around in the world, where PhysX is one of them. Most of the other physics modes involve simplified physics driven by game logic.

These alternative physics modes are implemented by the Unreal Engine and do not use the collision detection system of PhysX. For this reason each actor (with physics) has two collision representations. One collision representations is intended for the Unreal Engine and the other one for the physics engine (PhysX).

The first collision representation is intended for *static meshes* in Unreal Engine. Static meshes are a type of meshes that are not dynamic. This name does not imply they cannot move or interact with the world. The advanced option for static meshes is to check collisions per polygon against the static mesh 3D model itself and is potentially expensive to use. There is also a (simplified) collision hull option, but this option is not used for robots inside USARSim. Additionally there is a collision representation which is intended for *skeletal meshes* in the Unreal Engine. Skeletal meshes are used for game characters, not for USARSim robots.

Figure 3: The left picture shows the PhysX collision model, the right picture the Unreal Engine collision model.

The second collision representation is intended for PhysX and is created in the same way as the advanced static mesh version. The PhysX collision model is used in the physics simulation. However sensors will usually involve collision detection with the first representation. For example a simulated sonar sensor uses Unreal Engine tracing to detect objects in the world, which uses the Unreal Engine collision model. Care has been taken (as can be seen in Fig. 3) to keep both representations equivalent for the Nao robot.

PhysX collision detection algorithm.

The first step in collision detection is to find out which pairs of objects could collide. This stage is usually called the *Broad Phase*. In case of PhysX this is the *Sweep and Prune* algorithm[3]. This algorithm detects potentially colliding pairs by comparing the bounding boxes of rigid bodies. The starts (lower bound) and ends (upper bound) of the bounding boxes are sorted along a number of arbitrary axes. When a rigid body moves the bounding box may overlap with another bounding box of a rigid body (done by comparing the starts and ends). If the starts and ends of two of such bounding boxes overlap in all axes it means a pair of possible colliding rigid bodies is found.

In the case of simulating large scenes with a huge number of rigid bodies it is not feasible to check all possible pairs. If there are n shapes it means this algorithm would roughly have a complexity of $O(n^2)$. Instead PhysX divides the world in partitions and only checks pairs that are nearby each other. Once nearby pairs of shapes are identified the collision detection can move on to the *Near Phase* algorithm. In the Near Phase the exact collisions are computed. Details about the PhysX Near Phase algorithms are not available because they are part of PhysX's intellectual property.

4. EXPERIMENTS

The experiments are divided into two categories; experiments which check general properties for constrained rigid body motion and experiments that are directly related to the proposed Nao model.

4.1 Basic Experiments

This first experiment section describes preliminary experiments that do not directly involve the Nao robot. Yet, these experiments on fundamental properties of constrained body motion need to be performed before more advanced experiments are done, because they can have major influence on the dynamics of a robot that has to maintain its balance.

In section 4.1.1 the gravity of the simulation is verified. Gravity is one of the main factors influencing the balance of the robot. In section 4.1.3 the effects of the frame rate on the correctness of the simulation is tested.

4.1.1 Gravity

This first experiment is to verify the gravity in USARSim. The reason for this initial experiment is that changing the gravity at a later point would affect the way the Nao behaves due the balance of the robot changing. Another reason for doing this experiment is because prior USARSim versions were still using the default Unreal Engine gravity parameter, contradicting the gravity documentation[8] of USARSim.

One real meter is converted to Unreal Engine units by multiplying the value 250 times. Additionally Unreal Engine scales the gravity of rigid bodies two times by default (*rigid body gravity scale*).

The experiment was performed by dropping a block from a high distance and measuring the fall distance after a number of different times. Then using the gravity formulas, the distance the block was supposed to fall was computed (*expected fall distance*). This *expected fall distance* assumes there is no force slowing down the falling block. Using the *expected fall distance* and fall distance from the experiment the *correction* value can be computed. Results were averaged over ten runs.

[8]http://usarsim.sourceforge.net/wiki/index.php/Gravity_Documentation

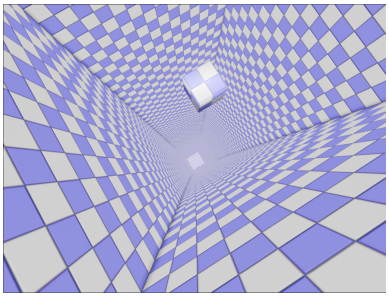


Figure 4: Experiment setup for testing gravity fall distances.

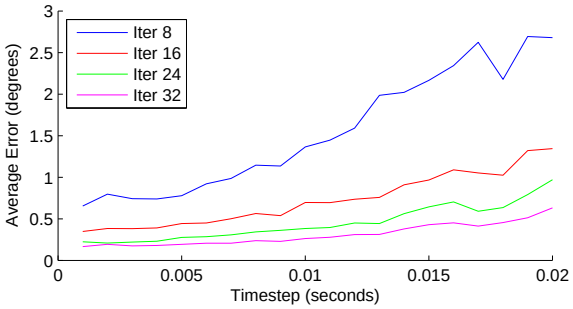


Figure 5: PhysX Time Step experiment results

The default setting of the Unreal Engine is -520uu with the *rigid body gravity scale* set to 2.0. This setting results in the block not falling far enough; the result has to be corrected with a factor of 2.5. Next we used a more realistic gravity setting based on g; the standard acceleration due to free fall of an object in vacuum. Near the surface of the Earth this constant is $9.8m/s^2$ which corresponds for the Unreal Engine gravity parameter value of $-250uu \times 9.8 = -2450uu$ and the *rigid body gravity scale* set to 1.0. With those values the object falls the expected distance.

The result of this experiment shows -2450uu is a realistic and correct gravity setting and the physics engine behaves as expected with regard to the gravity.

4.1.2 Simulation Timing

The second experiment is to investigate how the simulation timing settings affects the simulation. Considering the complexity of the simulation (21 DOF robot) the default simulation timing in the Unreal Engine might not be sufficient for a correct simulation.

The PhysX simulation is updated by calling the simulation function with the 'elapsed time'. This function runs a number of TimeSteps to synchronize the physics behavior with the rendered frame rate. Longer time steps lead to poor stability in the simulation.

For this experiment a test setup was made with several rigid bodies connected through joints. Of these joints only one is movable. The experiment consists of setting the one movable joint to a specified angle and measuring the error between the desired target angle and measured angle. In this position the gravity will push the blocks down to the ground, while the joints will have to try to satisfy the constraints. This real angle is measured by taking the rotation between the bottom and next block in the chain.

The experiment was executed for twenty different time steps. Because we have a number of rigid bodies connected we also added four different solver iteration count settings. For each timestep and solver iteration count setting the experiment was repeated five times. The measured error was averaged. The setup of this experiment is similar to the rigid bodies chained in, for example, the leg of the Nao.

In figure 5 the results are shown. The average errors for these tests vary between 2 and 3 degrees for the default timestep in UDK ($\frac{1}{50}$ second with solver iteration count set to 8). Although such an error may seem small, the error accumulates through the chained joints. Making the timestep smaller and the solver iteration higher results in a lower average error. Based on the results we used a default physics timestep of $\frac{1}{200}$ second, combined with a solver iteration count of 32.

4.1.3 Frame rate and simulation correctness

Another important aspect of a simulator is how well it runs on different machines. This might not seem so trivial because USARSim uses UDK, which is primarily intended as games development kit. The main issue this choice causes is that the update logic is tied to the frame rate at which the games engine is running. In other words actors are ticked once per frame and during this tick they update their logic. The primarily logic that is affected by the frame rate can be summed up as follows:

1. USARSim can only receive commands from the external control at most once per frame. These command updates include the updated joint parameters for the Nao, which must be sent at a high rate to execute the correct movement. Sending more than one command per frame will result in the commands to be processed all at once in a frame, making all commands except the last received one useless.

2. USARSim only sends status updates at most once per frame. These status updates include the current joint angles for the Nao.

3. PhysX only simulates the physics at most once per frame. Although it always executes the same number of time steps within a physics simulation call, it still means it is not possible to update the joint parameters between frames.

To find out the effects of the frame rate on the correctness of the simulation a simple experiment was performed. The HeadYaw joint of the Nao performed an angle interpolation at different fixed frame rates and the sensor HeadYaw angle values were measured by the controller. For reference the HeadYaw trajectory of a real Nao was also added. The results are plotted in Fig. 6.

The blue line shows the desired HeadYaw angle sent to the Nao. The red line shows the trajectory of the HeadYaw angle for a real Nao. At 5 and 2 frames per second (FPS) the effects of a low frame rate become clearly visible. The trajectories become jagged and there is a delay between the desired and real angles. At 25 and 50 FPS (the yellow and green line respectively) effects of a lower frame rate are almost fully gone. When looking closer at both results it is

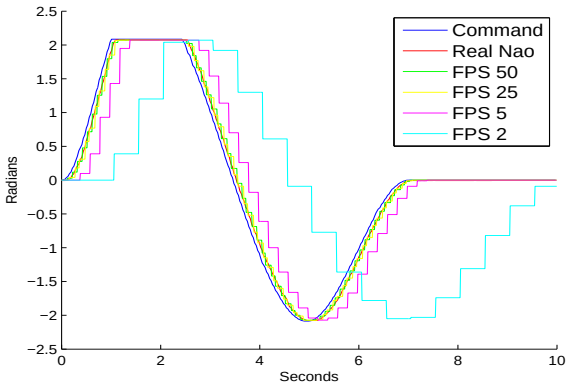


Figure 6: The effects of a lower frame rate become visible as jagged lines and a delay between the desired trajectories appears.

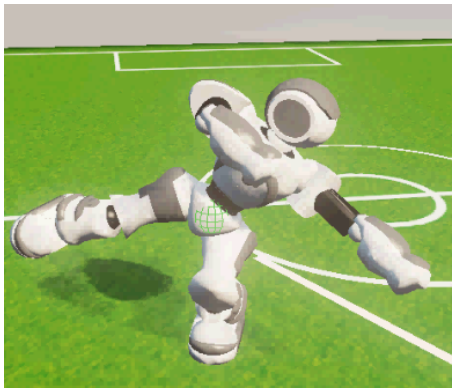


Figure 7: Nao performing the Tai Chi Chuan dance. The center of mass of the Nao is visualized as the green sphere.

still possible to note differences between 50 and 25 FPS in terms of smoothness, although the measured angle is a very acceptable representation of the simulated angle.

4.2 Advanced Experiments

In this section experiments are done with the simulated and real Nao. The results of these experiments are compared to see how close they resemble each other. The experiments all consist of the combined movement of multiple joints. A more simple version of this experiment would be the movement of a single joint (for instance turning the head). Such simple experiments are performed and show close correspondence. The more advanced experiments are more interesting in the sense that they show sometimes unexpected results due to the interaction of the constraints in between joints. Alternative advanced experiments would be kicking the ball and collisions between two robots, as demonstrated by Zaratti *et al.*[11] for the four legged Aibo robot.

In section 4.2.1 a fixed motion is executed by both the real and simulated Nao. The center of mass is visualized and the joint angles are recorded for several runs, averaged and compared. Section 4.2.2 includes several walking experiments. The walking behavior of the real and simulated Nao are compared by looking at the walk distances, joint angles and walk trajectories.

4.2.1 Tai Chi Chuan

In this experiment the real and simulated Nao were set to perform the Tai Chi Chuan dance. (i.e. play a sequential set of commands). During this animation the Nao first balances on one leg by stretching the other leg and keeping the arms in a specific position to keep balance. The animation repeats this motion for the other leg. Playing this dance is interesting for several reasons.

First is to perform the animation correctly the simulated Nao must maintain balance. The balance of the Nao is largely determined by the center of mass. An incorrect center of mass during movements can cause the Nao to be unable to maintain balance and as a result fall down to the ground. To correctly perform this in the simulation the center of mass must be above the supporting leg to ensure balance (visualized as the green sphere in Fig. 7).

Second because the motion is a fixed animation the experiment can be repeated for several runs, so the results over several runs can be averaged and compared against the joint angles between the simulated and real Nao. Finally, this animation is used by the manufacturer Aldebaran as diagnostic behavior; as long as a Nao is able to execute the Tai Chi Chuan no major malfunction in the motors and gears is expected.

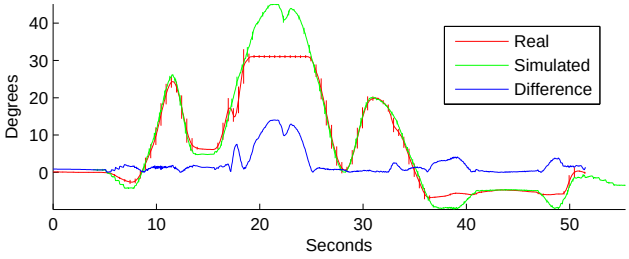


Figure 8: Joint angles and standard deviation of the RAnkleRoll joint while executing the Tai Chi Chuan dance. Results were averaged over ten runs. The red line shows the angles trajectory of the real Nao, while the green line shows the same for the simulated Nao. The blue line shows the difference between the two angles trajectories.

Fig. 8 shows the average joint angles for the RAnkleRoll joint. This joint is interesting because it shows a difference in the angles trajectories of the real and simulated Nao.

The command angle around 22 seconds is about 45 degrees. The real Nao joint is unable to follow the command angles. Most likely this is caused by the movement of other joints, resulting in a force being put on the parts around the joint. When sufficient force is put on the joint it will be unable to maintain the correct position (due to the motor not putting enough force in maintaining that position). In the case of the simulated Nao RAnkleRoll joint there is either not enough force pushing on the joint or the force of the joint used to maintain the position is too high.

4.2.2 *Walking*

Realistic walking comparable to the walking behavior of the real Nao is crucial in a robot simulation. During a RoboCup match a robot will have to walk a large part of the time.

For this experiment several walking and turning tests were done for the simulated and real Nao using the included walk engine of the Nao provided by Aldebaran. This walk engine uses a simple dynamic model inspired by work of Kajita *et al.*[6] and is solved using Quadratic programming[10]. When walking at full speed it can reach a velocity of $9.52cm/s$ and $42deg/s$ when turning.

In the first test the Nao was set to do a single full step with the left leg. The joint angles of the real and simulated Nao were recorded and compared.

Fig. 9 shows the average joint angles of the LKneePitch joint (i.e. the left knee) with standard deviation over ten recordings of the real and simulated Nao. In contrast to section 4.2.1 the standard deviation for the real Nao is lower than the simulated Nao. The same behavior is also seen for the standard deviations of the other joints.

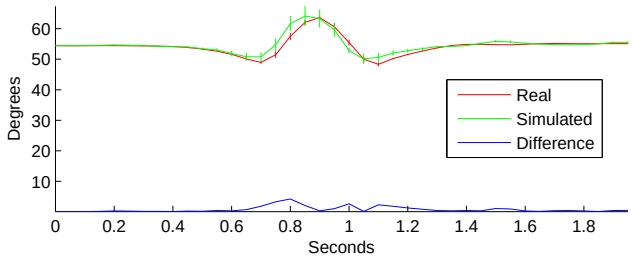


Figure 9: Average joint angles with standard deviation of the LKneePitch joint while executing a single step. Joint angles were averaged over ten runs for the real (red) and simulated (green) Nao. The blue line shows the difference between the joint angles trajectories.

For both the real as simulated Nao the forward walking was recorded ten times. The real Naos all walked around the expected distance (0.48 meter), while the simulated Naos only reached about 0.37 meter. This result for the simulated Naos could be tweaked (for instance by enlarging the motor force), but this makes the robot less stable.

In the third test the Nao was set to turn at full speed for five seconds. This means the Nao should turn about 210 degrees. This test was again executed ten times for the real and simulated Nao. During this test the real Nao reached the full 210 degrees turning, while the simulated Nao only reached about half.

In the last experiment the Nao was set to walk in a circle. Commands were generated by making one real Nao walk in a circle with a radius of 60cm. These commands were then replayed by the real and simulated Naos. Fig. 11 and 10 shows the path trajectories of three different real Naos and a simulated Nao walking in a circle using the same walking commands. Each real Nao executed the walk five times, while the simulated Nao was set to repeat the walk ten times.

Most of the real Naos successfully walked a circle like shaped path when replaying the commands, although there is a lot of variation in the paths.

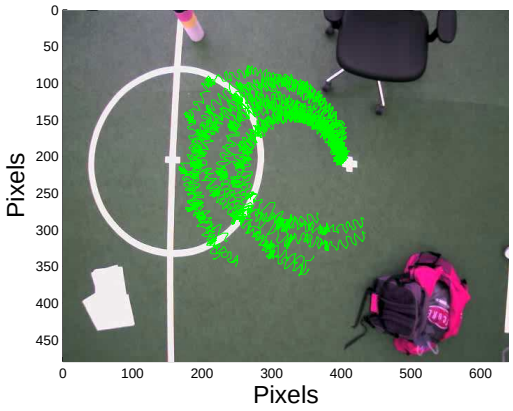


Figure 10: Trajectory of the simulated Nao walking in a circle with the same diameter as the white circle, repeated ten times for each Nao.

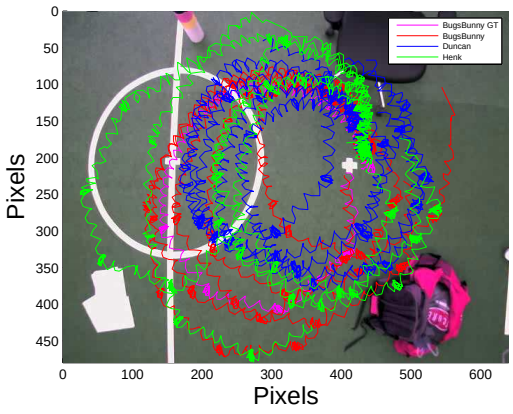


Figure 11: Trajectory of three different Naos walking in a circle, repeated five times for each Nao (recording using camera ground-truth).

On the other hand none of the simulated Naos were able to complete the circle. Considering the results of the forward walking and turning of the simulated Naos this is not totally unexpected.

5. FULL APPLICATION EXPERIMENT

To test how well the performance is for real applications, the source code of the Dutch Nao Team[9] has been tested with USARSim.

This application not only involves walking around, but also perception and dedicated behaviors like kicks and standing up.

To test real applications an intermediate program has been created, UsarNaoQi, which works as a proxy server, converting NaoQi messages in USARSim messages and vice versa. NaoQi is the framework provided by Aldebaran and allows the user to control the Nao in various programming

languages (C++, Python, C# or Urbi).

The source code of the Dutch Nao Team is written in Python, and could be directly applied. The code was fully functional, the robots could standup, position themselves on the field, locate the ball and kick the ball. The only observed difference is in the approach of the ball; the Dutch Nao Team code makes a number of small steps to get in a good position behind the ball. In simulation those steps are too small; the Nao needs too much time to position itself.

The experiment was performed by putting a number of Nao robots in the simulated RoboCup environment. The average frames per second (FPS) was recorded for two different scenarios. In the first scenario the Nao is simply standing and doing nothing. In the second scenario we executed the Nao with robot controller from the Dutch Nao Team. The controller was set in *play* mode. In this mode the Naos will walk around scanning for the ball.

The experiment was performed on a computer with an Intel iCore 7 920 processor and an AMD Radeon HD 6850 graphics card. USARSim was used in combination with the UDK December build 2011. UsarNaoQi was set to use a time step of 10ms; the Naos in USARSim sent status updates at a rate of 100 times per seconds (joint angle updates).

Table 1 shows the frame rate of the simulation with different numbers of Naos. The base FPS shows the frame rate when the Naos are standing on the ground doing nothing, while FPS DNT shows the Naos in the *play* state of the game.

Number of Naos	base FPS	FPS DNT
0	320	320
1	120	110
2	100	55
3	65	30
4	50	10

Table 1: Frame rate results with UsarNaoQi time step of 10ms

Without any Naos the scene is rendered at a FPS of 320. With one and two Naos the FPS drops to around 110 and 55 respectively, which is enough for running a decent simulation. With three Naos the FPS drops to 30, which is still acceptable (see section 4.1.3). With four Naos the simulation frame rate drops to 10 FPS, resulting in incorrect movements.

To find the performance bottlenecks in the simulation various profiler tools provided by UDK are used (PhysX statistics and UnrealScript code profiler). Using these tools reveals that when simulating four Naos half of the frame time is spent in the physics. The remaining part of the time goes to the sonar sensor (tracing), receiving and processing messages in the bot connection with the controller, sending the current status to the controller (joint angles) and updating the current joint angles.

6. CONCLUSION

In this paper we demonstrated that the simulation of the Nao in USARSim resembles reality quite closely. Our current model is usable in practice on the condition that one keeps in mind the limits of the method; like the walking behavior and the scaling issues with the number of Naos. The combination of Unreal/USARSim provides several advantages over other robot simulators. The simulation is at such a level that transparant migration of code between real robots and their simulated counterparts is possible. In this paper this is demonstrated with an intermediate program, UsarNaoQi, which enables access to the simulated robot with its native interface. Using this interface several experiments have been performed with both the real and simulated robot. The experiments consisted of movements where most of the 21 DOF were needed to maintain balance, which allowed us to monitor unexpected correllation between joints. The model developed for this humanoid robot demonstrates that robots with complex dynamics could be realistically modeled inside USARSim, which could be the basis of the introduction of other models of complex robots into USARSim like two-arm manipulators and/or service robots.

7. ACKNOWLEDGMENTS

The authors like to thank Hayley Hung for proofreading the manuscript. Part of the research is funded by the Dutch IIP Cooperation Challenge 'Sensor Intelligence for Mobility Systems'.

8. REFERENCES

[1] D. Baraff. An introduction to physically based modeling: rigid body simulation I - unconstrained rigid body dynamics. SIGGRAPH Course Notes, 1997.

[2] S. Carpin, M. Lewis, J. Wang, S. Balakirsky, and C. Scrapper. Usarsim: a robot simulator for research and education. In *Proceedings of the International Conference on Robotics and Automation (ICRA'07)*, pages 1400–1405, 2007.

[3] J. Cohen, M. Lin, D. Manocha, and M. Ponamgi. I-collide: An interactive and exact collision detection system for large-scale environments. In *Proceedings of the Symposium on Interactive 3D graphics*, pages 189–196. ACM, 1995.

[4] K. Fu, R. Gonzalez, and C. Lee. *Robotics: control, sensing, vision, and intelligence*. McGraw-Hill, 1987.

[5] A. Held. Creating an abstract physics layer for simspark. Studienarbeit im Studiengang Informatik, Universität Koblenz-Landau, November 2010.

[6] S. Kajita and K. Tani. Experimental study of biped dynamic walking. *Control Systems Magazine, IEEE*, 16(1):13–19, 1996.

[7] O. Michel. Cyberbotis ltd - webots^{TM}: Professional mobile robot simulation. *International Journal of Advanced Robotic Systems*, 1(1):39–42, March 2004.

[8] T. Röfer et al. B-human team report and code release 2011. Published online, November 2011.

[9] C. Verschoor et al. Dutch nao team - code release 2011 and technical report 2011. Published online, Universiteit van Amsterdam, October 2011.

[10] P. Wieber. Trajectory free linear model predictive control for stable walking in the presence of strong perturbations. In *Proceedings of the International Conference on Humanoid Robots*, pages 137–142, 2006.

[11] M. Zaratti, M. Fratarcangeli, and L. Iocchi. A 3D simulator of multiple legged robots based on USARSim. In *Robocup 2006: Robot Soccer World Cup X*, volume 4434 of *Lecture Notes in Artificial Intelligence*, pages 13–24. Springer, 2007.

Evaluation of Robotic Minimally Invasive Surgical Skills using Motion Studies

Seung-kook Jun
University at Buffalo (SUNY)
Dept. of Mech. & Aero. Engg.
1-716-645-1434
seungjun@buffalo.edu

Madusudanan Sathianarayanan
University at Buffalo (SUNY)
Dept. of Mech. & Aero. Engg.
1-716-645-1434
ms329@buffalo.edu

Abeer Eddib, *MD*
Childrens Hospital
Obstetrics & Gynecology
1-716-878-7138
aeddib@buffalo.edu

Pankaj Singhal, MD
Millard Fillmore Suburban Hospital
Obstetrics & Gynecology
1-716-689-8398
psinghal@buffalo.edu

Sudha Garimella, *MD*
University at Buffalo (SUNY)
Pediatrics Nephrology
1-716-878-7275
sgarimella@buffalo.edu

Venkat Krovi
University at Buffalo (SUNY)
Dept. of Mech. & Aero. Engg.
1-716-645-1430
vkrovi@buffalo.edu

ABSTRACT

Robotic minimally-invasive-surgery (rMIS) is the fastest growing segment of computer-aided surgical systems today and has often been heralded as the new revolution in healthcare industry. However, the surgical performance-evaluation paradigms have always failed to keep pace with the advances of surgical technology. In this work, we examine extension of traditional manipulative skill assessment with deep roots in performance evaluation in manufacturing industries for applicability to robotic surgical skill evaluation. This method relies on defining task-level segmentation of modular sub-procedures called "*Therbligs*" that can be combined to perform a given task. Performance metrics including intra- and inter-user performance variance can by analyzed by studying surgeons' performance over each sub-tasks. Additional metrics on tool-motion measurements, motion economy, and handed-symmetry can be similarly expanded over this temporal segmentation to help characterize performance. Our studies analyzed video recordings of surgical task performance in two settings: First, we examine performance of two representative manipulation exercises (peg board and pick-and-place) on a da Vinci surgical (SKILLS) simulator to afford a relatively-controlled and standardized testbed for surgeons with varied experience-levels. Second task-sequences from real surgical videos were analyzed with a list of predefined "*Therbligs*" in order to investigate its usefulness for real implementation.

Categories and Subject Descriptors

H.5.2 [**User Interfaces**]: Evaluation/Methodology; H.1.2. [**User/Machine Systems**]: Human Information Processing; I.2.10 [**Vision and Scene Understanding**]: Motion, Video Analysis

General Terms

Measurement, Performance, Experimentation, Verification

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage, and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

PerMIS'12, March 20-22, 2012, College Park, MD, USA.

Copyright © 2012 ACM 978-1-4503-1126-7-3/22/12...$10.00

Keywords

Robotic minimally invasive surgery, skill assessment, Therbligs, motion study, dexterity

1. INTRODUCTION

Surgical skill assessment and clinical evaluation has predominantly remained subjective [1] and developing quantitative assessment tools has been a topic of considerable importance [2, 3]. Medical education has long relied on subjective evaluations or in some cases semi-quantitative (like Likert-scale based) due to the lack of reliable, accurate and stable objective and quantitative performance metrics [4, 5]. Key challenges to assessment and accreditation of surgeons include (i) creating appropriate clinically relevant scenarios and settings and (ii) developing uniform, repeatable, stable, verifiable performance metrics; at manageable financial levels for ever increasing cohorts of trainees [6-8].

Given the lack of reliable performance assessment, the Accreditation Council for Graduate Medical Education (ACGME) [9] had only required maintenance of surgical logs for granting accreditation in certain procedures. The growth of computer integration in minimally-invasive-surgery (MIS) especially in the form of rMIS [10, 11] now offers a unique set of opportunities to comprehensively address this situation. A variety of physical variables can now be transparently monitored via instrumented tool-usage in both simulated and real-life operational settings [12, 13].

The recent incorporation of simulation based training into ACGME guidelines [14] is considered to be part of this transformation process. Satava [15] notes virtual-simulation-based training has benefitted from the concomitant revolutions of objective assessment of procedural skills and transition from an apprenticeship-based to criterion-based training model. The growing acceptance of virtual simulators stem from ability to (1) Control presentation of stimuli to trainees [16]; (2) Accurately and transparently monitor user responses [12, 17]. For example, in the Fundamentals of Laparoscopic Surgery (FLS) program [18], laparoscopic training competency is measured based on metrics such as time to task completion (TTC) [13, 16, 19], tool-path length precision (TPL) [17, 20] and dexterity of motions [21] using standardized box trainers [22, 23].

While quantitative metrics are clearly superior to subjective assessment, it is unclear as to WHICH data, at WHAT spatial and temporal resolution needs to be collected from the vast choices of

physical measurements possible. Improper understanding of the underlying relationships, coupled with insufficient computational support has led to present scenario focused on easy to measure but simplistic spatial and temporal aggregated measures (such as TTC, TPL [24], number of tool collisions, object-drops [12, 25] etc).

In this manuscript, we examine an alternate method of dexterous manipulative skill evaluation using micro-motion studies, with deep roots in manipulative performance evaluation in manufacturing industries. The well-established motion studies' methodology leverages segmenting of a primary task into basic-motion elements (*Therbligs*), recording the sequence of elements and key subtask performance details in process-charts, which are then statistically analyzed. In this work, we examine the extension of this traditional motion-study methodology to encompass assessment of Minimally Invasive Surgical procedures from videorecordings in two settings. First, we examine performance of two representative manipulation exercises (peg board and pick-and-place) on a da Vinci surgical (SKILLS) simulator to afford a relatively-controlled and standardized testbed for surgeons with varied experience-levels. Second task-sequences from real surgical videos were analyzed with a list of predefined "*Therbligs*" in order to prove its usefulness for real implementation. Performance metrics are obtained, including intra- and inter-user performance variance, by analyzing surgeons' performance over each sub-procedure. Additional metrics on tool-motion measurements, motion economy, handed-symmetry can be similarly extended over this temporal segmentation to help characterize performance and are being investigated.

2. BACKGROUND

In recent times, standardized objective methods for assessing technical skills were introduced and accepted for use in surgical training programs. The Objective Structured Assessment of Technical Skills (OSATS) as well as Objective Structured Clinical Examination (OSCE) emphasize the quantitative assessment processes without relying on expert evaluators. These methods though require appropriate hardware (measurement device) such as Imperial College Surgical Assessment Device (ICSAD) and Advanced Dundee Endoscopic Psychomotor Trainer (ADEPT) in order to perform surgical dexterity analysis. Most of these methods used TTC and TPL as the primary measures.

To our knowledge, the validation studies for "acceptance" of commonly used surgical measures are very limited [19, 26-28] and their actual relationship against skill levels specific to the robotic surgical simulators is not yet clear. On one hand, the oversimplification inherent in using aggregated/cumulative measures may result in loss of desirable user-specific discriminative characteristics. But more importantly, their use to provide feedback with a training curriculum might even lead to "wrong" skills to be learnt [29, 30]. Though several methods for analysis have been proposed, most of these are considered to be inadequate, inconsistent, non-standardized and in most cases, invalidated (or improperly validated) [4, 31, 32].

On the other end, several studies in the recent past showed that segmenting the surgical videos into sub-tasks (defined as surgemes in [2, 33]) can aid in automated performance and skill assessment. One of the most relevant work along this aspect has been the automated motion recognition using Hidden Markov Modeling (HMM) [34, 35] for simulated surgical tasks using da Vinci Trainer (dV-Trainer). Nonetheless, the basis and requirements of the surgical task segmentation has not been dealt with detail. It is essential to define these building blocks in a unified and generalized way to allow not only segmentation and further analysis of complex surgical procedures but also to establish meaningful metrics for skill and expertise.

Within industrial engineering practice [36, 37], motion studies are a well-established method used to characterize, simplify and improve the efficiency and effectiveness of manual tasks. Since originating from early twentieth century, such motion studies have been employed to characterize and quantify sub-tasks within a larger task context with a view to both characterize expertise as well as to eliminate inefficiencies. This decomposition potentially allows for a finite state automaton representation of a complex activity as in [35] that could form the discrete basis for linguistic representation as well as fault-detection and correction.

In this work, we seek to examine the applicability and usefulness of such a technique to assess surgical performance and help create a viable, robust yet quantitative basis for grounding the surgical-skill assessment process. Care needs to be taken to accommodate changes in data-acquisition environment (da Vinci surgical robot vs. various trainers) or training modalities (animals, cadavers, simulators, ex-vivo organs/tissues samples, simulated tasks etc). We will address these issues by defining subtasks in a systematic and generalized manner, estimating the efficacy of *Therblig* based micro-motion analysis of recorded videos (of simulated- as well as real- surgical procedures).

3. MOTION ANALYSIS

3.1 *Therbligs*

The traditional time and motion studies is based on the hypothesis that: any manipulation or assembly task can be subdivided into smaller individual units called "*Therbligs*" as coined by Frank Gilbreth [37]. He cataloged a set of "*Therbligs*" into effective and ineffective motions that served as building blocks of all manual manipulative activities in a factory shop floor. At its core, these basic elements allow for decomposition of a large complex manual job sequence into sub-parts that could then be individually examined.

Left hand Description	Sym	Time (ABS)	Time (Rel)	Time (Rel)	Time (ABS)	Sym	Right hand Description
Reach	RE	**1.5**	1.5	1.6	**1.6**	RE	Reach
Grasp	G	**1.8**	0.3	0.1	**1.7**	G	Grasp
Move	M	**2.5**	0.7	0.9	**2.6**	M	Move
Release	RL	**2.8**	0.3	0.2	**2.8**	RL	Release
Reach	RE	**3.5**	0.7	1.6	**4.4**	RE	Reach
Avoidable delay	AD	**4.4**	0.9	0.1	**4.5**	G	Grasp
Reach	RE	**5.0**	0.6	0.8	**5.3**	M	Move
Avoidable delay	AD	**5.4**	0.4	0.2	**5.5**	RL	Release
Reach	RE	**7.1**	1.7	1.5	**7.0**	RE	Reach
Grasp	G	**7.4**	0.3	0.2	**7.2**	G	Grasp
Reach	RE	**9.1**	1.7	0.4	**7.6**	M	Move
Grasp	G	**9.3**	0.2	1.6	**9.2**	AD	Avoidable Delay
Move	M	**9.8**	0.5	0.6	**9.8**	M	Move
Release	RL	**10.0**	0.2	0.1	**9.9**	RL	Release
Reach	RE	**11.0**	1.0	1.4	**11.3**	RE	Reach
Avoidable delay	AD	**12.0**	1.0	0.3	**11.6**	G	Grasp
Reach	RE	**13.8**	1.8	0.6	**12.2**	M	Move
Grasp	G	**13.9**	0.1	0.1	**12.3**	RL	Release
Move	M	**14.6**	0.7	1.4	**13.7**	RE	Reach
Release	RL	**14.7**	0.1	0.1	**13.8**	G	Grasp
Reach	RE	**16.3**	1.6	0.7	**14.5**	M	Move

Fig. 1: Process Chart

In an effort to develop a specialized but well defined set of RS *Therbligs*, we base the development on already established set of basic motion elements. In select cases, such as 'Use Tool', this motion was further classified into alternate classes specifically as 'Cut Tissue', 'Open Tissue', 'Scissors' and 'Cauterize Tissue'. At the same time, several of original *Therblig* series were deemed inappropriate and not included (refer Table 1).

3.2 Process Charts

Traditional motion studies are captured in a process chart [38] where *Therbligs* are hierarchically grouped into work elements and then ultimately into meaningful tasks, which has proved adequate to offer a primary discretization of industrial manipulation tasks. At its core, the process chart consists of a table with listing as shown in Fig. 1.

Enhancements to this basic process chart now involve taking advantage of bilateral symmetry (left/ right) or increased discretization or agglomeration of tasks as well as performing varying levels of statistical analyses for the collected information. The principle of motion economy [39] can now be applied to analyze, assess, simplify and improve the task efficiency and effectiveness to analyze RS training and skill assessment. In order to estimate the skill levels surgical performance markers can be captured by analyzing for the motion economy and dexterity of surgeons' capability to use their hands (or tools). Though importance of both these aspects has been recognized, detailed micromotion analysis has not yet been performed especially within the context of robotic surgeries.

So, the extent of each *Therblig* in a surgical task is captured using the process chart (refer Fig. 1) that is typically used to track both the tools' and hands' motions. The results of our *Therblig* analysis of individual hands captured by a Two Hand Chart proved to be valuable in terms of identifying specific pointers of skill-deficiency as well as surgical efficacy.

Table 1: List of *Therbligs* (Effective and Ineffective)

Name	Sym.	Description
Reach	**RE**	Reaching for object with empty hand.
Move	**M**	Moving an object using a hand motion
Grasp	**G**	Grasping an object by contacting and closing the finger of the active hand
Release	**RL**	Releasing control of an object
Hold	***H***	Holding an object
Pre-Position	PP	Positioning and/or orienting an object for the next operation
Position	*P*	Positioning and/or orienting an object in the defined location
Use	U	Manipulating and/or applying a tool in the intended way (UC- cutting tissue, UO- cut-open tissues, US- scissors cut, UZ- cauterize tissues)
Assemble	A	Joining the two parts together to form an assembled entity
Disassemble	DA	Separating multiple components that were previously joined in some way
Search	*SH*	Attempting to find an object using the eyes or hand
Select	*SL*	Choosing an object from a group
Inspect	*I*	Determining the quality of characteristic of an object
Plan	*PL*	Deciding a course of action
Unavoidable Delay	*UD*	Wait due to the factors beyond the control of the worker
Avoidable Delay	***AD***	Waiting that is within the worker's control
Rest to overcome Fatigue	R	Rest to overcome fatigue

Italicized: ineffective *Therbligs*, **bold***:* used in the current study

4. EXPERIMENTAL SETUP

In order to benchmark the performance of different surgeons both for intra- and inter-subject comparative analyses and evaluate improvements over a period of time, it is desirable to conduct these studies in a relatively controlled and standardized testbed. In this particular study, the da Vinci Surgical System-Si (dVSS-Si) was used along with its SKILLS simulator system [40] as in Fig. 2. In addition, using the dVSS-Si enabled recording of stereoscopic video images for post-processing as each task was being performed. Since, our objective was to develop a system skill assessment methods for a generic surgical robotic device, only the video feeds were used as input to our evaluation scheme.

Fig. 2: Da Vinci SKILLS Simulator

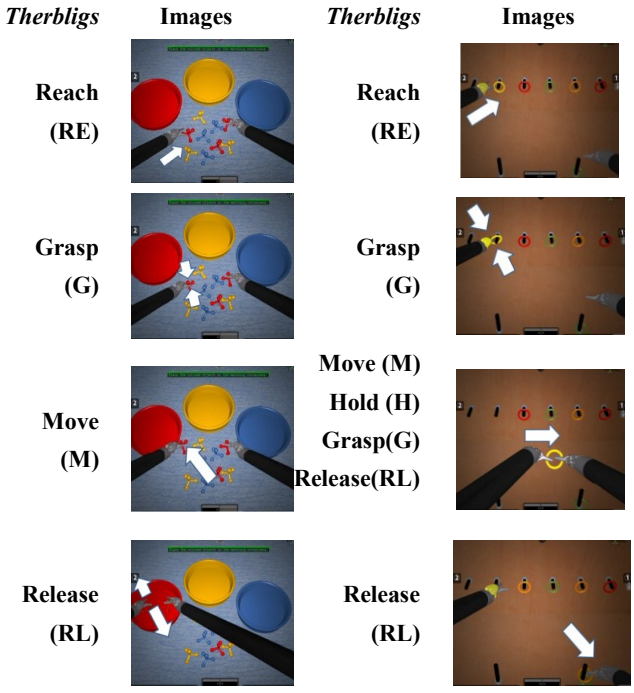


Fig. 3(a) Peg Board (PegI) (b) Pick-N-Place (PnP) [40]

Overall, the experiments were conducted using six subjects with varied levels of expertise (2 experts, 2 intermediates, 2 novices). Though the number of subjects is limited in this study, recruitment of more surgeons is currently underway for validation of our metrics. Two representative but simple simulator tasks were chosen to: (i) ensure only a subset of entire list of "*Therbligs*" are required for our analysis (ii) keep the manual

labeling segmentation process tractable; and (iii) ensure possibility of conducting physically simulated tasks to correlate with this analysis in the future. Each surgeon was assigned to perform these two simulated tasks (i) Pick-and-place (Fig. 3.a) and (ii) Peg board (Fig. 3.b) a minimum of 10 times. Finally, to demonstrate the applicability of this method, we also performed subtask segmentation for a real robotic-surgical procedure performed by one of the experts. In both the cases (simulated and real surgical tasks), two sequences of the videos were recorded from dVSS while the tasks are performed for use in motion study.

5. RESULTS

The pick-and-place tasks need only the 4 *Therbligs*— Reach (RE), Grasp (G), Move (M) and Release (RL) while the Peg-Board tasks required a total of 5 elements– in addition to these four elements as earlier, Hold (H) *Therblig* is included. For all cases, the two hand chart in form of text files were generated using *Therblig* labeling software developed in our lab (refer Fig. 4). These data files were then analyzed for each subject, each task and each *Therblig* based on the distributions of time to task completion. In order to anonymize the subject information, the following symbols were assigned during our analysis – experts (E_1 and E_2), intermediates (I_1 and I_2) and novices (N_1 and N_2). An immediate observation of the final results based on this analysis reveals that higher task complexity resulted in improved discriminative characteristics of surgical efficacy between experts and novices using this method.

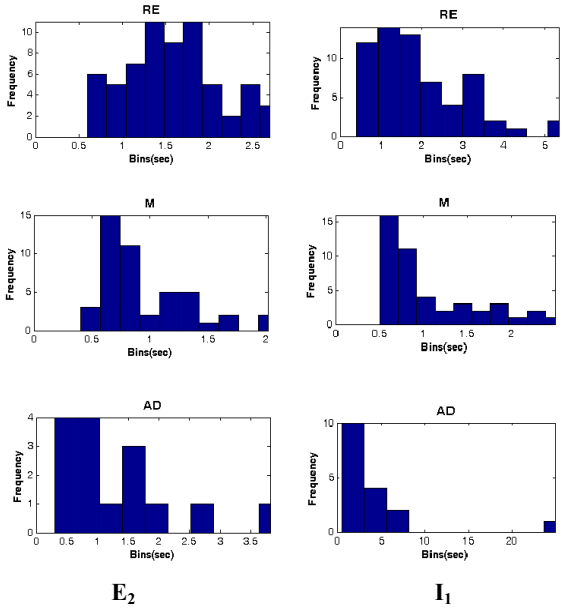
Fig. 4. Manual *Therblig* Labeling Segmentation Application

Case 1.a Pick-n-Place Simulator Task

Pick-n-place experiments formed the simplest task within our study and the motion analysis for these videos is shown in Table 2. The values in each cell actually correspond to cumulative average and standard deviations of time taken for each *Therblig* (column) and each subject (row). The data was normalized over the number of times a *Therblig* is observed within a task, total number of times the task has been carried out as well as number of tools used (in this case, right and left arms only). It can be seen that skill can be characterized in terms of the average time required for various *Therbligs*. Even for the simple pick-n-place tasks, there are distinct differences to be noted in the performance of novices with that of experts and intermediates based on the *Therblig* segmentation. E.g., while overall time based characteristics showed only marginal differences, expertise discrimination is possible by analyzing RE and M *Therbligs* for pick-and-place transfer task as in Table 2.

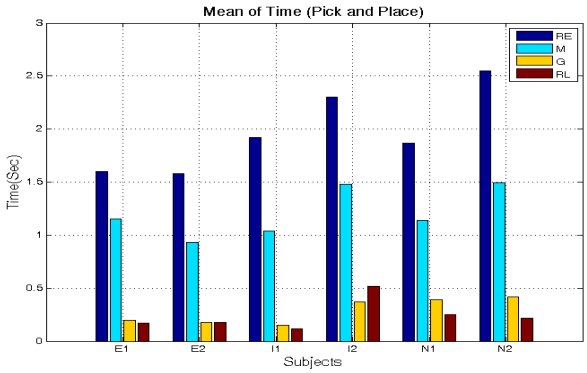
Case 1.b Peg Board Simulator Task

The discriminative capability of relative distribution of time within various *Therbligs* (to serve as expertise skill marker) is increasingly evident in the peg board simulator tasks. As the complexity of the task increases, the inappropriate and inefficient techniques of novices are more evident (based on the more scattered distributions of time spent on RE, M and H *Therbligs*) as shown in Table 3.



E_2 I_1

(a)



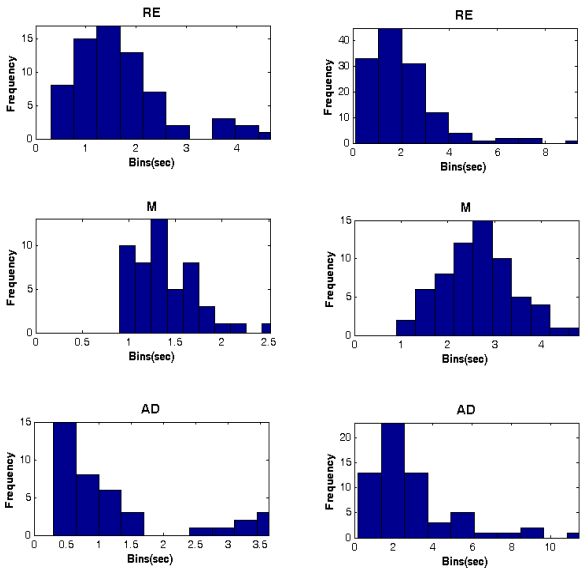
(b)

	RE	M	G	RL	AD
E_1	1.6 +/-0.66	1.15 +/-0.47	0.2 +/-0.09	0.17 +/-0.06	1.67 +/-1.13
E_2	1.58 +/-0.53	0.93 +/-0.41	0.18 +/-0.09	0.18 +/-0.07	1.24 +/-1.01
I_1	1.92 +/-1.14	1.04 +/-0.56	0.15 +/-0.07	0.12 +/-0.05	4.14 +/-5.92
I_2	2.3 +/-0.84	1.48 +/-0.78	0.37 +/-0.21	0.52 +/-0.57	1.07 +/-1.02
N_1	1.87 +/-0.85	1.14 +/-0.58	0.39 +/-0.24	0.25 +/-0.11	0.87 +/-0.55
N_2	2.55 +/-1.11	1.49 +/-0.53	0.42 +/-0.21	0.22 +/-0.06	1.56 +/-1.24

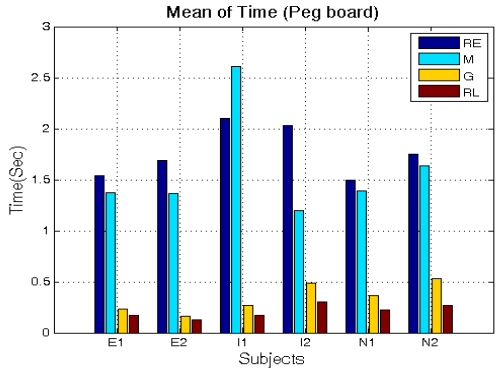
(c)

Table 2. Analysis of TTC for each of 5 *Therbligs* of Pick-n-Place Tasks (a) Frequency Distribution for Major *Therbligs* for E_2 and I_1 (b) Histograms of Means (c) Means and Standard Deviations

Clearly, for these representative examples, we note that the traditional TTC (per *Therblig*) proves inadequate. Deconstructing



(a)



(b)

	RE	M	G	RL	H	AD
E_1	1.54 +/-0.51	1.37 +/-0.46	0.23 +/-0.16	0.17 +/-0.06	1.23 +/-0.41	1.11 +/-0.69
E_2	1.69 +/-0.98	1.36 +/-0.4	0.16 +/-0.06	0.13 +/-0.07	0.43 +/-0.22	1.19 +/-1.03
I_1	2.1 +/-1.43	2.61 +/-0.85	0.27 +/-0.42	0.17 +/-0.1	1.52 +/-1.88	2.92 +/-2.29
I_2	2.03 +/-0.93	1.2 +/-0.47	0.49 +/-0.56	0.3 +/-0.16	2.11 +/-2.28	1.01 +/-0.42
N_1	1.5 +/-0.71	1.39 +/-0.53	0.36 +/-0.19	0.22 +/-0.13	1.49 +/-0.52	1.66 +/-1.09
N_2	1.75 +/-1.96	1.64 +/-0.59	0.53 +/-0.28	0.27 +/-0.13	1.02 +/-0.69	2.92 +/-2.88

(c)

Table 3: Analysis of TTC for each of 6 *Therbligs* of Peg Board Tasks (a) Frequency Distribution for Major *Therbligs* for E_2 and I_1 (b) Histograms of Means (b) Means and Standard Deviations

this total TTC to level of specific sub-motion elements offers a more effective quantitative characterization of achievable expertise. As noted in earlier tabulations (Table 2 and Table 3), however, there still exist discrepancies in terms of expertise identification even in the case of controlled simulation experiments using this method. Henceforth, it is necessary to develop "smarter" metrics to capture low-level characteristics based on these raw *Therblig* data.

Dexterity and Bimanual Synchronization

In surgical skill assessment, it is equally essential to characterize the dexterity of surgeons. In general, one measure of dexterity in bimanual tasks can be defined as synchronized and effective usage of both the hands leading to reduction in cost related factors such as time, object drops/ collisions etc. Therefore, within the quantitative framework, it can be expressed as total achievable overlap of effective *Therbligs* between the two hands. Similarly, sequence of tool motions that does not lead to minimizing the cost function, can be easily quantified in terms of amount of inefficient and wasteful motions. These might include improper handling of tools or objects such as hesitant motions, dropping the objects as well as failed object grasps/ transfers.

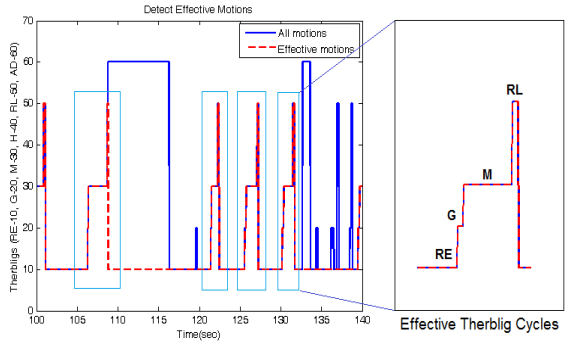


Fig. 5. Micromotion *Therblig* Analysis for Effective Motion Detection

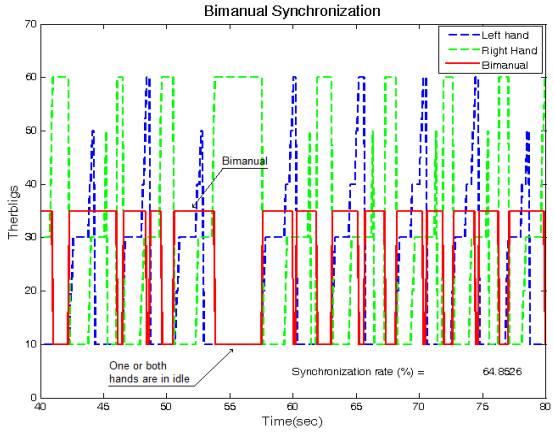


Fig. 6. Micromotion *Therblig* Analysis for Bimanual Synchronization (Dexterity)

In surgical skill assessment, it is equally essential to characterize the dexterity of surgeons. In general, one measure of dexterity in bimanual tasks can be defined as synchronized and effective usage of both the hands leading to reduction in cost related factors such as time, object drops/ collisions etc. Therefore, within the quantitative framework, it can be expressed as total achievable overlap of effective *Therbligs* between the two hands. Similarly, sequence of tool motions that does not lead to minimizing the cost function, can be easily quantified in terms of

amount of inefficient and wasteful motions. These might include improper handling of tools or objects such as hesitant motions, dropping the objects as well as failed object grasps/ transfers.

The first step in this process is to separate the entire *Therblig* data into efficient and wasteful motions. This can be done by performing a template matching for a given task in hand with that of *Therblig* results as in Fig. 5. For example, for a peg transfer task, the *Therblig* template for left and right hands are RE-G-M-H-RL and RE-G-M-RL. After subtracting out the essential components, the net avoidable delay and wasteful motions can be easily categorized. Within the efficient motion *Therbligs*, the times at which both hands are actively involved gives an indication of surgical dexterity. This procedure is shown in the Fig. 6.

	AD (sum)	Inefficient*	Synchronized*
E_1	25.1%	44.8%	57.3%
E_2	10.9%	23.8%	80.8%
I_1	28.9%	40.3%	52.3%
I_2	6.1%	21.0%	88.8%
N_1	6.9%	22.2%	90.2%
N_2	15.1%	29.2%	73.5%

Table 4: Inefficient *Therbligs* and Bimanual Synchronization for Pick and Place Tasks

So, by carefully analyzing the overlap of effective *Therbligs* between right and left tool movements from the high level analysis results, it is possible to extract the effective and efficient sections for quantifying skill dexterity in a way. Table 4 summarizes this result for pick and place transfer tasks and Table 5 for peg board tasks across 6 subjects. The extent of skill discrimination is not easily visible for the pick and place task as expected. However, for the peg board tasks, the surgical dexterity estimates correlates nicely with the actual surgical expertise. Therefore, by building on top of our motion bases, it is now possible to define newer metrics that is related to both the experience and expertise of surgeons and trainees.

	AD (sum)	Inefficient*	Synchronized*
E_1	17.8%	27.9%	65.4%
E_2	18.4%	25.1%	64.9%
I_1	25.9%	41.8%	50.6%
I_2	36.2%	39.9%	56.9%
N_1	26.4%	34.1%	55.0%
N_2	33.6%	45.8%	37.7%

Table 5: Inefficient *Therbligs* and Bimanual Synchronization for Peg Board Tasks

Case 2. Real-life Surgical Task Analysis

The generalized and simple definition of our basic motion elements (i.e. *Therbligs in* Table 1) aids the extension of skill assessment even to real surgical scenarios. Though this method could be used for any MIS type procedures, in this work, we focus on sections of hysterectomy surgical procedure conducted by an expert surgeon using da Vinci robot.

Even though using time alone to develop our metrics seem to be limited, this study reinforced our confidence in this *Therblig* Analysis micro-motion analysis method. The extension of the same using motion analysis as well as 3D kinematic information estimated from videos is already in progress to define newer metrics for quantifying surgical expertise especially in real-surgical scenarios.

The start and end frames of a sequence for each *Therblig* is shown in Table 6 that clearly illustrates the direct applicability of our micromotion analysis for real-surgical scenarios. The analysis results for the entire section of the video are shown by the frequency distributions of time-consumed for each motion elements in Fig. 7.

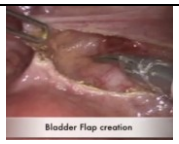
	Start Frame	End Frame	Time
Bladder Flap Creation			1:08:24
Right IP Isolation and transection			1:04:54
Transection right round ligament			0:41:16
Skeletonizing and transaction Rt Uterine arteries			1:06:38
:	:	:	:

Table 6: *Therblig* Analysis of Hysterectomy Robotic Surgery (Start and End Frames with Manual Annotation)

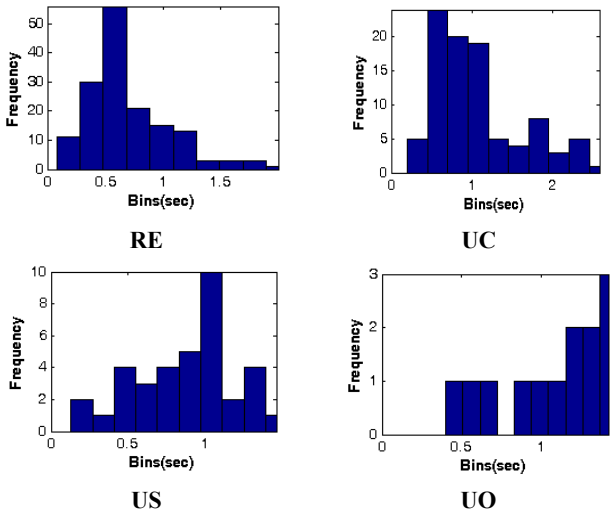


Fig. 7. Manual *Therblig* Analysis of Real-life Robotic Surgical Procedure (Hysterectomy) by E_2

6. Conclusion

These preliminary studies helped us gain insight about skill evaluation based on dexterity as well as motion economy at the micro-motions level. In addition to applying for simulated tasks,

the power of *Therbligs* was demonstrated by using it for a real-robotic surgical scenario.

It should also be remembered that the 2D and 3D kinematic data estimated from two camera feeds (that can be obtained from dVSS-Si as well as other commercial trainers) has not yet been incorporated into our analysis. With this additional information, we believe that the discriminative performance of our method can only improve. We plan to include this information, not only for improving the performance of this algorithm but also to implement automated recognition of these *Therbligs* given a stream of kinematic data.

REFERENCES

[1] T. T. Tsue, J. W. Dugan, and B. Burkey, "Assessment of Surgical Competency," *Otolaryngologic clinics of North America,* vol. 40, pp. 1237-1259, 2007.

[2] H. C. Lin, I. Shafran, D. Yuh, and G. D. Hager, "Towards automatic skill evaluation: detection and segmentation of robot-assisted surgical motions," *Computer Aided Surgery: Official Journal Of The International Society For Computer Aided Surgery,* vol. 11, pp. 220-230, 2006.

[3] P. D. van Hove, G. J. M. Tuijthof, E. G. G. Verdaasdonk, L. P. S. Stassen, and J. Dankelman, "Objective assessment of technical surgical skills," *British Journal of Surgery,* vol. 97, pp. 972-987, 2010.

[4] B. M. A. Schout, A. J. M. Hendrikx, F. Scheele, B. L. H. Bemelmans, and A. J. J. A. Scherpbier, "Validation and implementation of surgical simulators: a critical review of present, past, and future," *Surgical Endoscopy,* vol. 24, pp. 536-546, 2010.

[5] J. A. Aucar, N. R. Groch, S. A. Troxel, and S. W. Eubanks, "A Review of Surgical Simulation With Attention to Validation Methodology," *Surgical Laparoscopy Endoscopy & Percutaneous Techniques,* vol. 15, pp. 82-89, 2005.

[6] A. Amodeo, A. Linares Quevedo, J. V. Joseph, E. Belgrano, and H. R. H. Patel, "Robotic laparoscopic surgery: cost and training," *Minerva Urologica E Nefrologica = The Italian Journal Of Urology And Nephrology,* vol. 61, pp. 121-128, 2009.

[7] D. Wagner and M. L. Lypson, "Centralized Assessment in Graduate Medical Education: Cents and Sensibilities," *Journal of Graduate Medical Education,* vol. 1, pp. 21-27, 2009.

[8] M. Teryl Nuckols, MSHS; and M. José J. Escarce, PhD, "Potential Cost Implications of Changes to Resident Duty Hours and Related Changes to the Training Environment," Division of General Internal Medicine and Health Services Research, David Geffen School of Medicine at UCLA., Los Angeles, CA, USASeptember 28, 2010.

[9] ACGME. (2011). *Accreditation Council for Graduate Medical Education, General Surgery Program Guidelines*. Available: http://www.acgme.org/acwebsite/rrc_440/440_prindex.asp

[10] M. J. Mack, "Minimally Invasive and Robotic Surgery," *JAMA: The Journal of the American Medical Association,* vol. 285, pp. 568-572, February 7, 2001 2001.

[11] D. B. Camarillo, T. M. Krummel, and J. J. K. Salisbury, "Robotic technology in surgery: Past, present, and future," *The American Journal of Surgery,* vol. 188, pp. 2-15, 2004.

[12] M. A. Lerner, M. Ayalew, W. J. Peine, and C. P. Sundaram, "Does training on a virtual reality robotic simulator improve performance on the da Vinci surgical system?," *Journal of Endourology,* vol. 24, pp. 467-72, Mar 2010.

[13] K. Tanoue, M. Uemura, H. Kenmotsu, S. Ieiri, K. Konishi, K. Ohuchida, M. Onimaru, Y. Nagao, R. Kumashiro, M. Tomikawa, and M. Hashizume, "Skills assessment using a virtual reality simulator, LapSim™, after training to develop fundamental skills for endoscopic surgery," *Minimally Invasive Therapy & Allied Technologies,* vol. 19, pp. 24-29, 2010.

[14] N. Brown, S. Helmer, C. Yates, and J. Osland, "The revised ACGME laparoscopic operative requirements: how have they impacted resident education?," *Surgical Endoscopy,* pp. 1-7.

[15] R. Satava, "Historical Review of Surgical Simulation—A Personal Perspective," *World Journal of Surgery,* vol. 32, pp. 141-148, 2008.

[16] P. Kanumuri, S. Ganai, E. M. Wohaibi, R. W. Bush, D. R. Grow, and N. E. Seymour, "Virtual Reality and Computer-Enhanced Training Devices Equally Improve Laparoscopic Surgical Skill in Novices," *JSLS, Journal of the Society of Laparoendoscopic Surgeons,* vol. 12, pp. 219-226, 2008.

[17] J. D. Hernandez, S. D. Bann, Y. Munz, K. Moorthy, V. Datta, S. Martin, A. Dosis, F. Bello, A. Darzi, and T. Rockall, "Qualitative and quantitative analysis of the learning curve of a simulated surgical task on the da Vinci system," *Surgical Endoscopy,* vol. 18, pp. 372-378, 2004.

[18] "Fundamentals of Laparoscopic Surgery (FLS) Written Instructions and Performance Guidelines," 2011.

[19] M. Pellen, L. Horgan, J. Barton, and S. Attwood, "Construct validity of the ProMIS laparoscopic simulator," *Surgical Endoscopy,* vol. 23, pp. 130-139, 2009.

[20] J. H. Chien, M. M. Tiwari, I. H. Suh, M. Mukherjee, S.-H. Park, D. Oleynikov, and K.-C. Siu, "Accuracy and speed trade-off in robot-assisted surgery," *The International Journal Of Medical Robotics + Computer Assisted Surgery: MRCAS,* vol. 6, pp. 324-329, 2010.

[21] S. D. Bann, M. S. Khan, and A. W. Darzi, "Measurement of Surgical Dexterity Using Motion Analysis of Simple Bench Tasks," *World Journal of Surgery,* vol. 27, pp. 390-394, 2003.

[22] K. R. Rosen, "The history of medical simulation," *Journal of Critical Care,* vol. 23, pp. 157-166, 2008.

[23] M. Mulla, D. Sharma, M. Moghul, O. Kailani, J. Dockery, S. Ayis, and P. Grange, "Learning Basic Laparoscopic Skills: A Randomized Controlled Study Comparing Box Trainer, Virtual Reality Simulator, and Mental Training," *Journal of Surgical Education*.

[24] R. Aggarwal, J. Ward, I. Balasundaram, P. Sains, T. Athanasiou, and A. Darzi, "Proving the Effectiveness of Virtual Reality Simulation for Training in Laparoscopic Surgery," *Annals of Surgery,* vol. 246, pp. 771-779 10.1097/SLA.0b013e3180f61b09, 2007.

[25] P. A. Kenney, M. F. Wszolek, J. J. Gould, J. A. Libertino, and A. Moinzadeh, "Face, content, and construct validity of dV-trainer, a novel virtual reality simulator for robotic surgery," *Urology,* vol. 73, pp. 1288-92, Jun 2009.

[26] R. Aggarwal, T. Grantcharov, K. Moorthy, T. Milland, and A. Darzi, "Toward feasible, valid, and reliable video-based assessments of technical surgical skills in the operating room," *Annals of Surgery,* vol. 247, pp. 372-9, Feb 2008.

[27] C. Basdogan, M. Sedef, M. Harders, and S. Wesarg, "VR-Based Simulators for Training in Minimally Invasive Surgery," *Computer Graphics and Applications, IEEE,* vol. 27, pp. 54-66, 2007.

[28] M. Schijven and J. Jakimowicz, "Construct validity: experts and novices performing on the Xitact LS500 laparoscopy simulator," *Surgical Endoscopy,* vol. 17, pp. 803-810, 2003.

[29] A. G. Gallagher, A. B. Lederman, K. McGlade, R. M. Satava, and C. D. Smith, "Discriminative validity of the Minimally Invasive Surgical Trainer in Virtual Reality (MIST-VR) using criteria levels based on expert performance," *Surgical Endoscopy,* vol. 18, pp. 660-665, 2004.

[30] S. Maithel, R. Sierra, J. Korndorffer, P. Neumann, S. Dawson, M. Callery, D. Jones, and D. Scott, "Construct and face validity of MIST-VR, Endotower, and CELTS," *Surgical Endoscopy,* vol. 20, pp. 104-112, 2006.

[31] F. Carter, M. Schijven, R. Aggarwal, T. Grantcharov, N. Francis, G. Hanna, and J. Jakimowicz, "Consensus guidelines for validation of virtual reality surgical simulators," *Surgical Endoscopy,* vol. 19, pp. 1523-1532, 2005.

[32] S. S. Van Nortwick, T. S. Lendvay, A. R. Jensen, A. S. Wright, K. D. Horvath, and S. Kim, "Methodologies for establishing validity in surgical simulation studies," *Surgery,* vol. 147, pp. 622-630, 2010.

[33] C. E. Reiley, E. Plaku, and G. D. Hager, "Motion generation of robotic surgical tasks: Learning from expert demonstrations," in *Engineering in Medicine and Biology Society (EMBC), 2010 Annual International Conference of the IEEE*, 2010, pp. 967-970.

[34] J. Rosen, J. D. Brown, L. Chang, M. N. Sinanan, and B. Hannaford, "Generalized approach for modeling minimally invasive surgery as a stochastic process using a discrete Markov model," *Biomedical Engineering, IEEE Transactions on,* vol. 53, pp. 399-413, 2006.

[35] J. Rosen, B. Hannaford, C. G. Richards, and M. N. Sinanan, "Markov modeling of minimally invasive surgery based on tool/tissue interaction and force/torque signatures for evaluating surgical skills," *Biomedical Engineering, IEEE Transactions on,* vol. 48, pp. 579-591, 2001.

[36] G. Salvendy(ed), *Handbook of Industrial Engineering: Technology and Operations Management*: Wiley-Interscience, 2001.

[37] R. M. Barnes, *Motion and Time Study: Design and Measurement of Work*. Univ. of California, Los Angeles: Wiley, August 1980.

[38] B. Niebel and A. Freivalds, *Methods, Standards, and Work Design*, 10th ed., 2003.

[39] M. P. Groover, *Work systems and the methods, measurement, and management of work*: Prentice hall, 2006.

[40] Intuitive-Surgical and Inc. *da Vinci Surgical Robot: Si and SKILLS Simulator*. Available: www.intuitivesurgical.com/

Multi-Relationship Evaluation Design: Modeling an Automatic Test Plan Generator

Brian A. Weiss
National Institute of Standards and Technology
100 Bureau Drive, MS 8230
Gaithersburg, Maryland 20899
+1 (301) 975-4373
brian.weiss@nist.gov

Linda C. Schmidt
University of Maryland
0162 Glenn L. Martin Hall, Building 088
College Park, Maryland 20742
+1 (301) 405-0417
lschmidt@umd.edu

ABSTRACT
Advanced and intelligent systems within the manufacturing, military, homeland security, and automotive fields are constantly emerging and progressing. Testing these technologies is crucial to (1) inform the technology developers of targeted areas for improvement, (2) capture end-user feedback, and (3) verify the degree of the technology's capabilities. Evaluation designers have put forth considerable effort in developing methods to speed test-plan generation. The Multi-Relationship Evaluation Design (MRED) methodology is being created to gather multiple inputs from several source categories and automatically output evaluation blueprints that identify the pertinent test-plan characteristics. MRED captures input from three categories including the evaluation stakeholders, the technology state, and the available resources. This information and the relationships among these inputs are combined as input into an algorithm that will yield specific test plan characteristics. This paper reviews the MRED methodology as it enters its final stages of development, including new discussion of the relationships among the various inputs and the chosen method of Evaluative Voting to capture *Stakeholder Preferences.* An example focusing on the design of test plans to evaluate a robotic arm is also presented to bring further clarity to the latest MRED developments.

Categories and Subject Descriptors
B.8.0 [**Performance of Systems**]: *measurement techniques, modeling techniques, performance attributes*

General Terms
Measurement, Performance, Design, Experimentation, Verification

Keywords
MRED, performance evaluation, model, test plan design

1. INTRODUCTION
Advanced and intelligent systems within the manufacturing, military, homeland security, and automotive industries are constantly emerging and progressing. Evaluating these technologies is vital to (1) inform the technology developers of targeted areas for improvement, (2) capture end-user feedback, and (3) verify the degree of the technology's capabilities.

(c) 2012 Association for Computing Machinery. ACM acknowledges that this contribution was authored or co-authored by a contractor or affiliate of the U.S. Government. As such, the Government retains a nonexclusive, royalty-free right to publish or reproduce this article, or to allow others to do so, for Government purposes only. PerMIS'12, March 20-22, 2012, College Park, MD, USA. Copyright © 2012 ACM 978-1-4503-1126-7-3/22/12...$10.00

Evaluation events provide useful data that both update the state of the technology and support future testing. In this paper, the term *test* refers to a planned evaluation event or exercise focused on capturing data to generate performance metrics of a specific technology under scrutiny. Evaluation designers put forth extensive efforts in generating methods to speed the test-plan development process. These efforts are most apparent when designers must create comprehensive test plans to evaluate advanced and intelligent technologies.

The Multi-Relationship Evaluation Design (MRED) methodology will allow evaluation designers to hasten the test-plan development process. MRED gathers multiple inputs from several source categories and automatically outputs evaluation blueprints that identify pertinent test-plan characteristics. MRED captures input from three categories including the evaluation stakeholders, the technology state, and the available resources. This information and the relationships among these inputs are combined as input to an algorithm that will yield specific test plan characteristics.

This paper is organized as follows: Section 2 presents the overall MRED methodology; Section 3 discusses the preference capture method of 'Evaluative Voting' and how it will be implemented with MRED; Section 4 shows an example application of 'Evaluative Voting' integrated into MRED; and Section 5 concludes the discussion.

2. MULTI-RELATIONSHIP EVALUATION DESIGN (MRED) - METHODOLOGY
MRED's goal is to automatically produce evaluation test plans based upon multiple inputs [12]. MRED is an interactive algorithm that processes information from multiple input categories and outputs one or more evaluation blueprints including their constituent test plan elements (Figure 1). During this process MRED invokes the relationships among the inputs and the impacts the inputs have on the outputs. The overall methodology was proposed in [11], while the output blueprint evaluation elements were defined in [9] and [10]. The relationships between specific inputs and outputs were presented in [12] and [13]. This section briefly presents the MRED model inputs (including the *Technology State, Resources,* and *Stakeholder Preferences*) and outputs (including *Technology Test Levels, Metrics, Resources, Evaluation Scenarios,* and *Explicit Environmental Factors*). Greater detail can be found in the afore-mentioned references. The remainder of Section 2 gives an overview of MRED's process and presents new work characterizing the relationships among the various inputs.

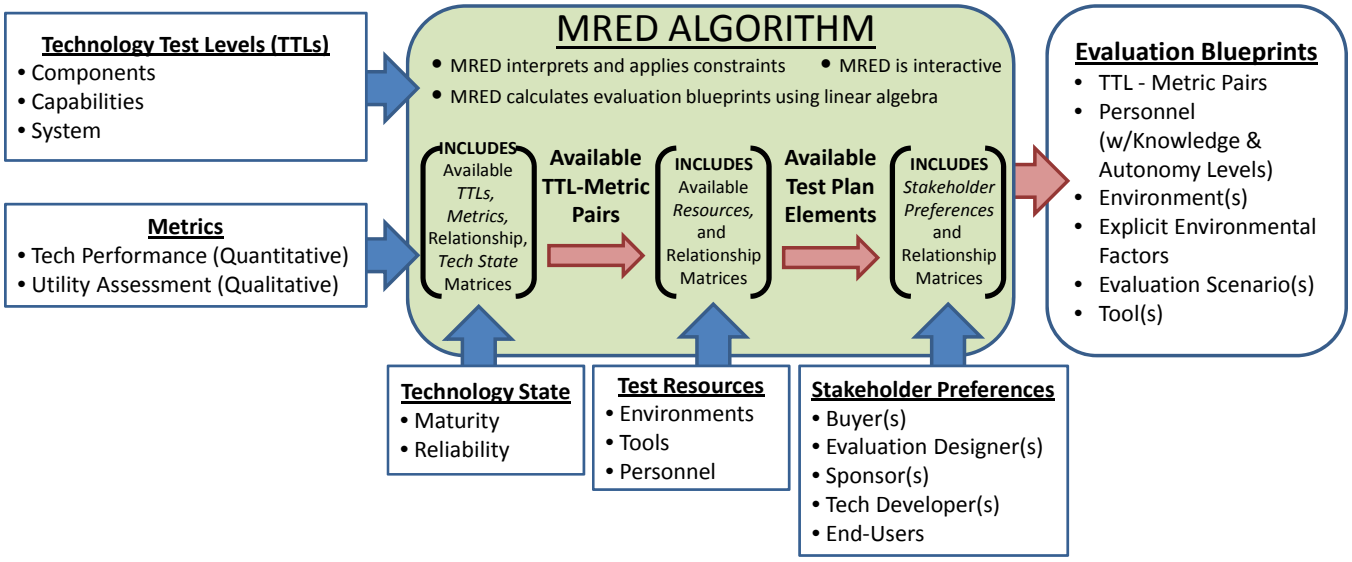


Figure 1. MRED Model with Input (TTLs, Metrics, Technology State, Test Resources, Stakeholder Preferences) and Output (Evaluation Blueprints)

2.1 Input

The most significant inputs into the MRED model are the *Technology Test Levels (TTLs)* and corresponding *Metrics*. *TTLs* are defined as the technology's constituent *Components* and *Capabilities* along with the *System* as a whole [9]. Specifically, they can be described as:

- *Component* – Essential part or feature of a *System* that contributes to the *System*'s ability to accomplish a goal(s).
- *Capability* – A specific ability of a technology. A *System* is made up of one or more *Capabilities*. A *Capability* is enabled by either a single *Component* or multiple *Components* working together.
- *System* – Group of cooperative or interdependent *Components* forming an integrated whole to accomplish a specific goal(s).

Pertinent *Metrics* are also input for each *TTL*. *Metrics* fall into one of two groups:

- *Technical Performance* – *Metrics* related to quantitative factors (e.g., accuracy, precision, time, distance, etc.).
- *Utility Assessments* – *Metrics* related to qualitative factors that express the condition or status of being useful and usable to the target user population.

Technology State features another set of inputs: *Maturity* and *Reliability* of the individual *TTLs*. In the context of MRED, they are defined as:

- *Maturity* – The state of development of individual *Components*, *Capabilities*, and the *System*. *Maturity of a* technology's *Components* must be provided by the *Technology Developer(s)*, whereas *Maturity* of *Capabilities* and the *System* could either be provided by the *Technology Developer(s)* or calculated by MRED given *Component Maturity* and the *Component – Capability* matrix (presented in Section 2.4). *Maturity* information provided by the *Technology Developer(s)* is either classified as *Fully-Developed*, *Functional*, or *Non-Functional*.
- *Reliability* – The probability that a specific *Component*, *Capability*, or the *System* (as a whole) will continue to function under certain conditions for a certain time. Similar to *Maturity*, *Component Reliability* must be directly provided by the *Technology Developer(s)* or by the *Evaluation Designer(s)* from prior test efforts. *Reliability* of specific *Capabilities* and the *System* can either be obtained from the *Technology Developer(s)*, the *Evaluation Designer(s)* (also from prior testing), or through MRED calculations using *Component Reliability* and the *Component – Capability* relationship matrix. The nature of the specific *Reliability* measure is dependent upon the technology in question.

Further details on *Technology State* including *Reliability* and *Maturity* can be found in [13].

*Test Resources r*epresents the availability of the viable *Environments*, *Personnel*, and *Tools* for data collection and analysis. Discussion of these inputs is presented in [9] and [10].

The last significant input category is that of the *Stakeholder Preferences*. Initially presented in [12], this includes the preferences from five specific individuals (or groups) presented in Table 1. *Stakeholder preferences* are captured with respect to *TTL-Metric* pairs[1], *Environments*, *Tools*, *Personnel*, *Explicit Environmental Factors*, and *Evaluation Scenarios* [10] [11].

Note that colors are used in tables throughout this document to assist the reader in distinguishing data among the rows and columns. Colors do not indicate information of greater or lesser importance.

[1] *TTL-Metric* pairs are specific *Technology Test Levels* and *Metrics* that are coupled together. Multiple *TTLs* can be coupled with the same *Metrics* and vice-versa.

Table 1 – Stakeholders [12]

STAKEHOLDER GROUPS	WHO THEY ARE…
Buyers	Stakeholder purchasing the technology
Evaluation Designers	Stakeholder creating the test plans by determining MRED inputs
Sponsors	Stakeholder paying for the technology development and/or evaluation
Technology Developers	Stakeholder designing and building the technology
Users	Stakeholder that will be or are already using the technology

2.2 Output Elements

MRED is designed to automatically output sets of evaluation blueprints complete with specified elements (Figure 1). Each set of blueprints will include one (or more) *TTL-Metric* pairs, an *Environment* for testing, *Tools* to support the capture of data to generate the necessary *Metrics*, *Personnel* including those that will interact with the technology and those that will execute the evaluation, *Knowledge* and *Autonomy Levels* dictating what specific *Personnel* can and cannot do during the evaluation [12], *Evaluation Scenarios* describing the types of exercises that will occur [10], and *Explicit Environmental Factors* which provide guidance as to the level of *Feature Complexity* and *Feature Density* within the *Environment* [10].

2.3 MRED Process

MRED generates the most preferred evaluation blueprints by using an interactive process between:

- Interacting with the *MRED Operator* to collect the necessary information and *Stakeholder Preferences* and
- Processing the collected information and preferences by calculating pertinent *Technology State* information, assessing the feasibility of blueprint elements, generating potential blueprints, and scoring the feasible blueprints.

This multi-step process shown in Figure 2 is summarized below. The term *MRED Operator* is defined as the individual that inputs data, information, and preferences into MRED. This is usually the *Evaluation Designer* or another facilitator who is guiding the blueprint generation process.

1. *MRED Operator* inputs the technology's *TTLs* and corresponding *Metrics* that are considered for testing.
2. *MRED Operator* defines the *Components-Capabilities* and *Metrics-TTLs* relationship matrices.
3. *MRED Operator* inputs *Component Tech. State* data.
4. *MRED* calculates the *Technology State* data for the *Capabilities* and the *System*.
5. *MRED* eliminates *TTLs* and *Metrics* based upon the *Technology State* data input in 3 and calculated in 4.
6. *MRED Operator* inputs the *Available Resources* including *Environments*, *Tools*, and *Personnel*.
7. *MRED Operator* defines the *TTLs-Environment* and *Metrics-Tools* relationship matrices.
8. *MRED* eliminates *TTLs*, *Metrics*, *Environments*, *Tools*, and *Personnel*.
9. *MRED* captures *Stakeholder Preferences* as to which *TTL-Metric* pairs should be tested.
10. *MRED* scores and groups the pairs based upon the *Stakeholder Preferences*
11. *MRED* eliminates low scoring *TTL-Metric* pairs.
12. *MRED* captures *Stakeholder Preferences* as to which *Personnel* should evaluate the remaining candidate *TTL-Metric pairs*.
13. Step 12 is sequentially repeated with the remaining blueprint elements until *MRED* outputs the most preferred blueprints.

The noted relationship matrices are elaborated upon in Section 2.4 while the overall process will be formalized in future work.

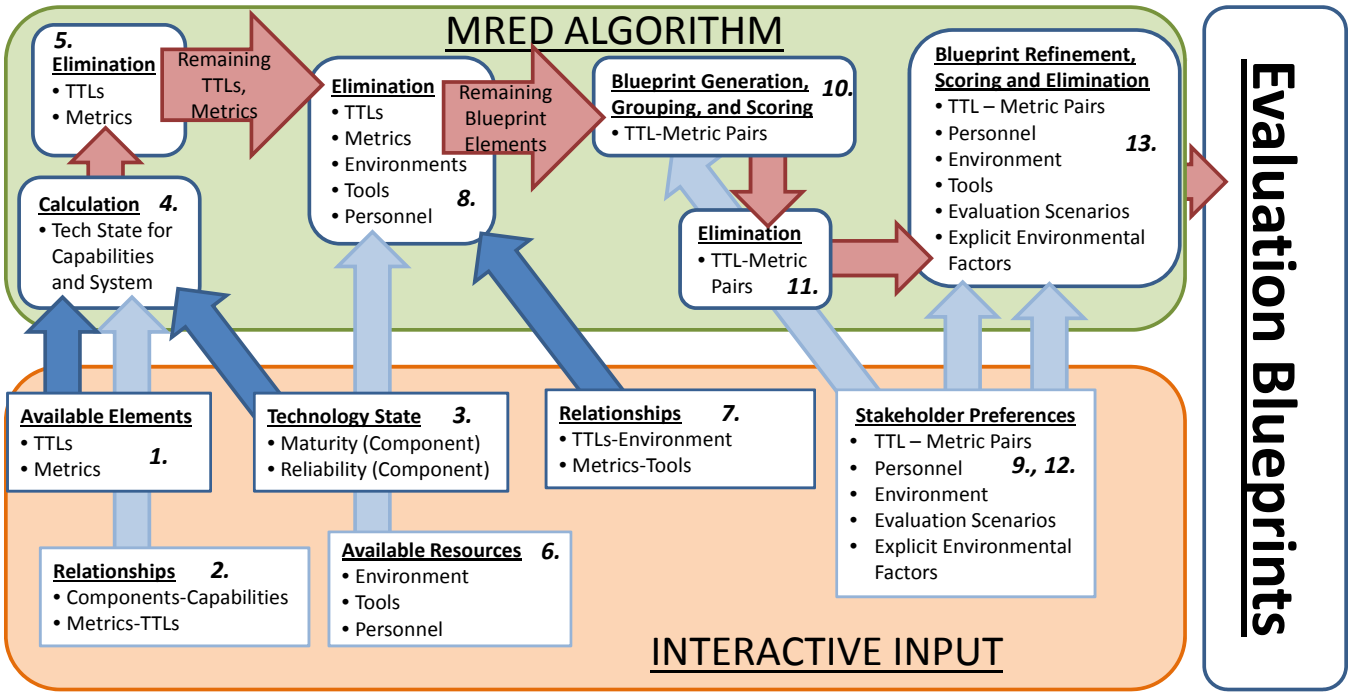


Figure 2. MRED Process Flow Diagram

2.4 Key MRED Relationships

MRED exploits the numerous relationships that exist among the various inputs. Two types of relationships are: (1) physical (the two *Components* of an engine and a transmission work to affect the vehicle's *Capability* of acceleration); and, (2) performance-based (the *Reliability* of the vehicle's acceleration is a function of the *Reliability* of the vehicle's engine and transmission). Since each technology being considered for evaluation is unique, these relationships must be defined by the *MRED Operator* with input from other *Stakeholders*. These relationships (or lack thereof) are critical to MRED's success whereby they are integrated with the inputs defined in Section 2.1. Each set of relationships is represented by one or more matrices within the MRED Algorithm. This section will present these specific relationships.

An example robotic arm will be used to clearly illustrate the relationships as they are defined below. The example robotic arm, shown in Figure 3, is illustrated as a *System* with seven *Components* (C_1, C_2, C_4, and C_6 are revolute joints; C_3 and C_5 are prismatic joints; C_7 is a gripper). These seven *Components* function to provide seven *Capabilities* (P_1, P_2, and P_3 are translation in X, Y, and Z of the end-effector; P_4, P_5, and P_6 are roll, pitch, and yaw of the end-effector; and P_7 is grasping).

MRED interacts with the *Operator* to obtain many of the relationships discussed throughout this section. This is important to note considering that relationships are technology specific and have the potential to change as a technology evolves to its final iteration. MRED's design also contains natural constraints and intrinsic relationships. These are discussed where present.

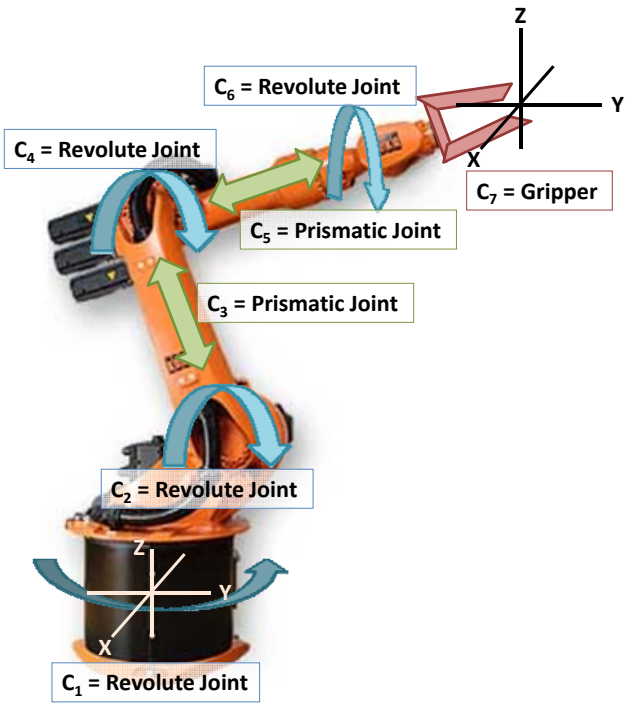


Figure 3. Robotic Arm[2] Example

The first relationship defined in MRED is that between the *Components* and *Capabilities*. This relationship exists because *Capabilities* are only produced through the function of one or more *Components*. This relationship is similar to that between Functional Requirements and Design Parameters as defined by Suh in his theory of Axiomatic Design [7]. An example of this binary matrix is shown in Table 2. In the *Components – Capabilities* Matrix, a "1" cell indicates that the corresponding *Component* contributes to (influences) the corresponding *Capability*. A "0" indicates that no such relationship exists between the *Component* and *Capability*.

Table 2 - Example *Components – Capabilities* Relationship Matrix for Robotic Arm

	CAPABILITIES						
COMPONENTS	X (P_1)	Y (P_2)	Z (P_3)	Roll (P_4)	Pitch (P_5)	Yaw (P_6)	Grasp (P_7)
Rev 1 (C_1)	1	1	0	0	0	1	0
Rev 2 (C_2)	1	1	1	1	1	0	0
Pris 1 (C_3)	1	1	1	0	0	0	0
Rev 3 (C_4)	1	1	1	1	1	0	0
Pris 2 (C_5)	1	1	1	0	0	0	0
Rev 4 (C_6)	0	0	0	1	1	1	0
Gripper (C_7)	0	0	0	0	0	0	1

The *Components – Capabilities* relationship is critical when MRED defines the *Maturity* and *Reliability* for *Capabilities* and the *System*. If these *Maturities* and *Reliabilities* are not provided by the *Technology Developer(s)* or *Evaluation Designer(s)* at these *Technology Test Levels*, then they must be calculated given the *Maturity* and *Reliability* of the *Components* along with the *Component* and *Capability* relationship matrix. If unknown, MRED calculates the *System Maturity* and *Reliability* matrices based upon the *Maturities* and *Reliabilities* for the various *Capabilities*. The *Maturities* and *Reliabilities* for each *Component*, *Capability*, and the *System* must be above certain thresholds in order for a specific *TTL* to be considered further for evaluation. If these thresholds are not met, then MRED eliminates these *TTLs* from further testing consideration.

The second set of relationships captured by MRED is that between the *Metrics* and *Technology Test Levels*. This relationship is documented in two matrices; one binary matrix whose columns display all of the *Technology Test Levels* with the rows indicating potential *Technical Performance Metrics* (an example is presented in Table 3); the second binary matrix's columns present the *Capability* and *System Technology Test Levels* with the corresponding *Utility Assessment Metrics* highlighted in the matrix's rows. Note that one of the MRED constraints is that *Utility Assessment Metrics* can only be captured for *Capabilities* and the *System* while *Technical Performance Metrics* can be captured for all three *TTL* groups [9].

Table 3 - Example *Metrics (Technical Performance) - TTL* Relationship Matrix for Robotic Arm

	C_1	C_2	C_3	C_4	C_5	C_6	C_7	P_1	P_2	P_3	P_4	P_5	P_6	P_7	System (S)
Max Force	0	0	1	0	1	0	1	1	1	1	0	0	0	0	0
Max Linear Velocity	0	0	1	0	1	0	0	1	1	1	0	0	0	0	0
Max Torque	1	1	0	1	0	1	0	0	0	0	1	1	1	0	0
Max Angular Velocity	1	1	0	1	0	1	0	0	0	0	1	1	1	0	0
Range of Motion	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Max Lift Capacity	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1
Speed	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
Force	0	0	0	0	0	0	0	1	1	1	0	0	0	1	1

[2] Robot arm image courtesy of www.robots.com

The goal of establishing the *TTL – Metric* relationship matrices is to indicate which *Metrics* can be obtained from testing the various *TTLs*. MRED utilizes the data within these relationship matrices numerous times throughout the test plan generation process. In addition, MRED uses this matrix numerous times to eliminate either *TTLs* or *Metrics* if the other is eliminated in a prior step (presented in Section 2.3). For example, if a *TTL* is eliminated because its *Maturity* and *Reliability* do not meet the designated threshold, then MRED would eliminate any *Metrics* that solely correspond to this *TTL* which would only be shown in the *Metrics – TTL* relationship matrix.

The third set of binary relationship matrices captured in MRED are the *TTL – Environment* matrices. The three specific *TTL – Environment* matrices are: 1) *Components and Capabilities* (rows) – *Lab Environment* (columns), 2) *Components*, *Capabilities* and *System* (rows) – *Simulated Environment* (columns), and 3) *Capabilities* and *System* (rows) – *Actual Environment* (columns). The necessity of these three matrices is brought upon by MRED's constraints that only *Components* and *Capabilities* can be tested in *Lab Environments* and only *Capabilities* and the *System* can be tested in *Actual Environments* [10] [11]. A *TTL – Environment* relationship matrix is presented using the robotic arm example in Table 4.

Table 4 – Example *Components* and *Capabilities* - *Environment* (*Lab*) Matrix for Robotic Arm

		LAB ENVIRONMENTS		
		ABC Controls Lab	ABC Robotics Lab	DEF Force/Torque Lab
COMPONENTS and CAPABILITIES	C_1	1	0	1
	C_2	1	0	1
	C_3	1	0	1
	C_4	1	0	1
	C_5	1	0	1
	C_6	1	0	1
	C_7	1	0	0
	P_1	0	1	0
	P_2	0	1	0
	P_3	0	1	0
	P_4	0	1	0
	P_5	0	1	0
	P_6	0	1	0
	P_7	0	1	0

The goal of establishing the *TTL – Environment* matrices is to indicate which of the candidate *TTLs* could be tested in the various environments. Figure 4 presents a screen capture from the interactive MRED interface (created in Matlab[3]) that enables the *Evaluation Designer* to indicate the available *Environments* (shown in the top half of the figure) and specify the *TTL – Environment* relationship matrices (in the bottom half of the figure). If there are no candidate *Environments* available to test a specific *TTL*, then MRED eliminates this *TTL* from further testing consideration. If MRED eliminates *TTLs* at this stage because there are no viable *Environments* that exist, then MRED checks the *Metrics – TTL* relationship matrix and eliminates those *Metrics* that only correspond to the eliminated *TTL(s)*.

The fourth set of binary relationship matrices captured in MRED are the *Metric – Tools* matrices. The two relationship matrices in this category are: 1) *Technical Performance Metrics – Tools* and 2) *Utility Assessment Metrics – Tools*. The first matrix only includes those data collection and analysis tools that support the generation of *Technical Performance Metrics* while the second includes those tools that support the production of *Utility Assessment Metrics*.

Table 5 – Example *Technical Performance Metrics – Tools* Matrix for Robotic Arm

		TOOLS		
		Tension Sensor	Dynamometer	LADAR
TECHNICAL PERFORMANCE METRICS	Max Force	1	1	0
	Max Linear Velocity	0	0	1
	Max Torque	0	1	0
	Range of Motion	0	0	1
	Max Lift Capacity	1	0	0
	Speed	0	0	1
	Force	1	1	0

The benefit of these matrices is that they indicate if any *Tools* are unnecessary (in that they do not support any of the *Metrics*) and/or if *Metrics* cannot be obtained (if the appropriate *Tools* are unavailable). Similar to the *Environment – TTL* relationship matrices, the *Metric – Tools* matrices are used to eliminate *Metrics* if there are no candidate *Tools* available to capture the required data. If MRED eliminates *Metrics* due to a lack of *Tools*, then MRED checks the *Metrics – TTLs* relationship matrix and eliminates those *TTL(s)* that only correspond to the eliminated *Metric(s)*. MRED accesses the data in this set of matrices several times throughout the test plan generation process.

[3] Certain commercial companies, products and software are identified in this paper in order to explain our research. Such identification does not imply recommendation or endorsement by NIST, nor does it imply that the companies, products, and software identified are necessarily the best available for the purpose.

Figure 4. Example Interactive MRED Screen from Matlab Presenting the Available Environments and corresponding *TTL – Environment* relationship matrices

Additional relationship matrices, including those relating *Personnel* and *Environments* are still being finalized and will be discussed in future work.

The last set of inputs into MRED comes from the *Stakeholders* in the form of *Stakeholder Preferences*. MRED presents the list of the candidate blueprint elements to the *Stakeholders* based upon those elements that are still available after *Maturities* and *Reliabilities* are calculated, relationships are defined, and available *Resources* are input. It is critical that their subjective preferences are appropriately captured and reflected in MRED. If not, the output evaluation blueprints will not accurately reflect the wishes of the *Stakeholders*. Preference is the topic of the next section.

3. PREFERENCE CAPTURE

The MRED inputs shown in Figure 1 are objective with the exception of the *Stakeholder Preferences*. These subjective preferences are supported by each *Stakeholder's* knowledge of the facts. Providing preferences to ultimately select evaluation blueprints is different than what is encountered in product development. Each class of *Stakeholders* could potentially select entirely unique blueprints with very different test plan elements. This is not the case in product development where preferences provided on constituent attributes (product size, weight, etc.) all contribute to the same overriding goal of profit for the business. In product development, the decision-makers are usually all employees of the same entity. In the typical development of an evaluation, input from different *Stakeholders* (often with competing interests) is collected and valued.

Accurately capturing and representing the preferences of the various stakeholders is critical to MRED's success. The *Stakeholder Preferences* are central to further reducing the set of candidate *TTLs* and *Metrics* down to those that are most valuable for testing at the present time. Likewise, these preferences also play a crucial role in determining what *Environment(s)* the *TTLs* should be tested, what type of *Evaluation Scenarios* will be used, and who (*Personnel*) will be using the technology during the evaluation exercises. Further, analyzing the preferences from multiple stakeholders to select the most preferred options is another key step within MRED. This step reflects that of group decision-making.

This section will present background on several preference capture and group decision-making methods, introduce the preference capture method of 'Evaluative Voting' that MRED is adopting, and discuss how it will be implemented into MRED's algorithm.

3.1 Background

Preference capture is a topic that has been studied for decades by researchers in many fields including economics and engineering design. Preference can be defined as *the power, right, or opportunity of choosing*[4] and as a *positive regard for something*[5]. In turn, preference capture is the act of obtaining an individual's or group's desires on one or more options. Each proposed preference capture method attempts to find out what an individual or group really wants. Many group decision-making methods have been produced and refined over many years of study. There are numerous challenges to effectively capturing group preferences including [8]:

- Delineating between weak and strong preferences for alternatives
- Comparing preferences between group members if there is minimal to no overlap on preferences of discrete alternatives

[4] http://www.merriam-webster.com/dictionary/preference

[5] http://www.merriam-webster.com/thesaurus/preference

- Weighting the importance of the attributes to one another that compose the alternatives
- Weighting the importance of each group member's preferences to one another
- Competing objectives or priorities held by different group members (this raises issues of fairness or equitable distribution if members do not share a common objective) so a Pareto Optimal frontier cannot be defined [8]
- Lack of a method for aggregating individual rankings "that does not directly or indirectly include interpersonal comparisons of preference" which does not resolve Arrow's Impossibility Theorem [8]

One method of preference capture and group decision-making is the Borda count, which is often referred to as a voting method [1] [2] [6]. The Borda count was developed as a method to allow a group of individuals to rank order candidates and select the 'most preferred' candidate among the members. This method is implemented by first asking the voters to individually rank the *n* candidates from 1 to *n* with the candidate being ranked number 1 the most preferred and the candidate being ranked *n* being the least preferred. If a voter chooses not to rank one of the candidates (whether they are indifferent or don't have enough information), then this candidate is ranked last (so multiple candidates could be ranked last). The Borda Count then turns the individual rankings into scores by giving *n* points to the candidate ranked 1st, *n-1* points to the candidate ranked 2nd, etc. Voter's scores for each candidate are added together and the candidate that receives the highest score is considered the winner (or 'most preferred'). This is a simple method to implement.

There are several drawbacks to this method that eliminated it from consideration with MRED. In general, the Borda Count satisfies Arrow's first four axioms yet violates Arrow's fifth axiom, *Independence of irrelevant alternatives*[6] [2]. Specifically, it is susceptible to agenda manipulation [1] in that it does not account for majority preferences at all. This method is strictly ordinal and it does not enable MRED *Stakeholders* to delineate the distance between adjacently-ranked alternatives. In this sense, a candidate that a *Stakeholder* is indifferent on would be scored the same as a candidate the *Stakeholder* finds least appealing (last).

Pairwise comparison is another method of preference capture and can be used to achieve a group decision when combined with other methods [2]. Pairwise comparison is predicated upon all alternatives being compared one-to-one. Although this method has been proven effective in some applications, it is not practical for integration with MRED. Specifically, the vast number of alternatives to be compared during the various steps of the *Stakeholder Preference* capture process would result in an extremely time-consuming process. It's possible that *Stakeholders* would have to compare over 20 alternatives which would require nearly 200 pairwise comparisons. Further, Arrow's Impossibility Theorem restricts aggregation of pairwise comparison [3].

There are many other methods available to capture individual preferences and produce a group decision. One such category includes methods in the area of Multi-Attribute Decision-Making (MADM) and Multi-Criteria Decision-Making (MCDM) [5] [14]. These methods have been proven beneficial when a selection must be made among various alternatives where each alternative is valued against one or more attributes.

This category of methods does not appear to be suitable for use with MRED. One important reason is MADM would require all possible blueprints to be input as the list of alternatives. This would potentially lead to a combinatory explosion of blueprints. If this were done for the robotic arm example introduced in Section 2.4, then it is likely hundreds, if not thousands of blueprints would have to be considered. This example includes up to:

- 15 *TTLs* (7 *Components*, 7 *Capabilities*, and 1 *System*)
- 6 *Metrics* (on average and including both *Technical Performance* and *Utility Assessment Metrics*)
- 5 *Environments* (on average, across the *Lab, Simulated* and *Actual Environments)*
- 3 types of *Technology Users* (part of the *Personnel* input)
- 3 types of *Evaluation Scenarios*
- And consideration to additional *Personnel* and *Explicit Environmental Factors*.

The above information would yield approximately 4050 sets of blueprints (15 x 6 x 5 x 3 x 3). Only by stepping through MRED, would one know exactly how many blueprints are being considered since *TTLs* and/or *Metrics* can be grouped together, test plan elements could be eliminated based upon *Maturity* and *Reliability*, etc.

Another reason that MADM is not suitable for integration with MRED is because the blueprints and diversity among *Stakeholder Preferences* is too complex to produce an objective function. The objective function is determined from the output of the tests since there's no way to indicate a preference rating in MADM.

Asking each *Stakeholder* to rate all of these blueprints would be tremendously time-consuming, especially considering that not all *Stakeholders* will care to test every *TTL*, generate every potential *Metric*, etc.

Realizing that MRED has the potential to generate an unnecessary and excessive amount of blueprints, it is important to identify a method that will capture the *Stakeholders' Preferences* in an inexpensive and timely manner, along with the ability to eliminate undesirable test plan elements prior to final blueprint selection to further streamline the process.

3.2 Evaluative Voting

MRED will leverage the method of Evaluative Voting to enable *Stakeholder Preference* capture on an independent cardinal scale [4]. Evaluative Voting is a method where voters (*Stakeholders*) score each alternative on an integer scale to signify their preference for, neutral, or against testing a particular *TTL-metric* pair. Using Hillinger's [4] general election *EV-3* scale (-1,0,1), a *Stakeholder* would give each alternative a score of '-1' (against the alternative), '0' (neutral stance), or '1' (for the alternative). An initial example of applying the *EV-3* scale to MRED would be asking the *Stakeholders* to score each of the available *TTLs* in regarding their agreement to the statement of "This *TTL* should be evaluated." A *Stakeholder* would vote '-1'to indicate they are against testing a *TTL* (they disagree with the statement); '0' to indicate they are indifferent as to if the *TTL* should be tested; or '1' to indicate they believe the *TTL* should be tested (they agree with the statement). The EV method provides a score of '0' if a voter decides not to cast their vote regarding a specific candidate. In the case of MRED, if a *Stakeholder* chooses not to vote on a specific element (due to a lack of information), the vote remains at the default of 'NV' to indicate they are recusing themselves from

[6] *Independence of irrelevant alternatives* (IIA) is defined as: If the aggregate ranking would choose A over B when C is not considered, then it will not choose B over A when C is considered.

scoring that specific element. This is different from the originally-defined EV method in that MRED does not average in a score of '0.' However, MRED does average in a score of '0' if a *Stakeholder* actively scores a specific element as neutral. The rationale behind this decision is that neutral preferences have a mathematical impact on the overall scores, where their lack of inclusion can present misleading data.

There are numerous benefits to integrating Evaluating Voting with MRED to capture *Stakeholder Preferences* [4]. They are:

- Enables the aggregation of judgments on a cardinal scale
- Avoids highly scoring a minority candidate which could occur with the Borda Count, Plurality Voting, and other voting methods
- Simple to implement and for the *Stakeholders* to understand
- Method is comparable to other judgments expressed on cardinal scales such as grades (given in schools, universities, etc.) which are often aggregated through averaging
- Successfully implemented using scales larger than (-1,0,1)
- Accounts for a *Stakeholder* that chooses not to vote on a specific element in such a manner that does not incorrectly inflate or deflate an element's score

Hillinger recommends using the *EV-3* scale (-1,0,1) for general elections (selection of a single candidate) and the *EV-5* scale (-2,-1,0,1,2) for expert decisions. A German political survey institute adopted an 11-point scale (-5,-4,-3,-2,-1,0,1,2,3,4,5) when asking survey respondents to rate their satisfaction with politicians. The University of Michigan Survey Research Center used a much larger integer scale (0 to 100) to capture voters' perceptions of candidates [4]. The 11-point scale (also known as the Forschungsgruppe Wahlen scale after the German institute that devised this scale) is selected for use with MRED. This decision is made based upon the amount of *TTL-metric pairs* that *Stakeholder Preferences* would be solicited and that multiple elements will be selected for consideration (while the lowest scoring elements will be eliminated from further consideration),

The following section will discuss how Evaluative Voting will be integrated with MRED to ultimately output preferred blueprints given *Stakeholder Preferences*.

3.3 MRED Implementation

An iterative approach is used with respect to implementing Evaluative Voting with MRED. This iterative process consists of 1) capturing *Stakeholder Preferences* of a single set of test plan elements, 2) aggregating these cardinal scores whereby the weakest scoring elements are eliminated from further consideration, and 3) the remaining test plan elements are then considered with another set of test plan elements for further preference capture. This process is repeated until a series of candidate test plan elements is output. Figure 5 illustrates this approach with respect to the robotic arm example. This figure presents the Matlab MRED interface for capturing Stakeholder Preferences for the *Metric-TTL* pairs. The process begins with 1) all of the *Stakeholders* inputting their preferences for each *Metric-TTL* pair on the selected Evaluative Voting scale, 2) these preference scores being aggregated where those *Metric-TTL* pairs scoring lower than '0' (or another threshold set by the *Evaluation Designer*) being eliminated and 3) the remaining *Metric-TTL* pairs being passed through to the next set of test plan elements.

This approach offers numerous benefits in both capturing *Stakeholder Preferences* and using these preferences to both eliminate low-scoring blueprint elements and highlight high-scoring blueprint elements. This approach will be discussed further , followed by its advantages and disadvantages.

Implementing Collaborative Evaluative Voting (CEV) in MRED begins with having the *Stakeholders* score each of the available

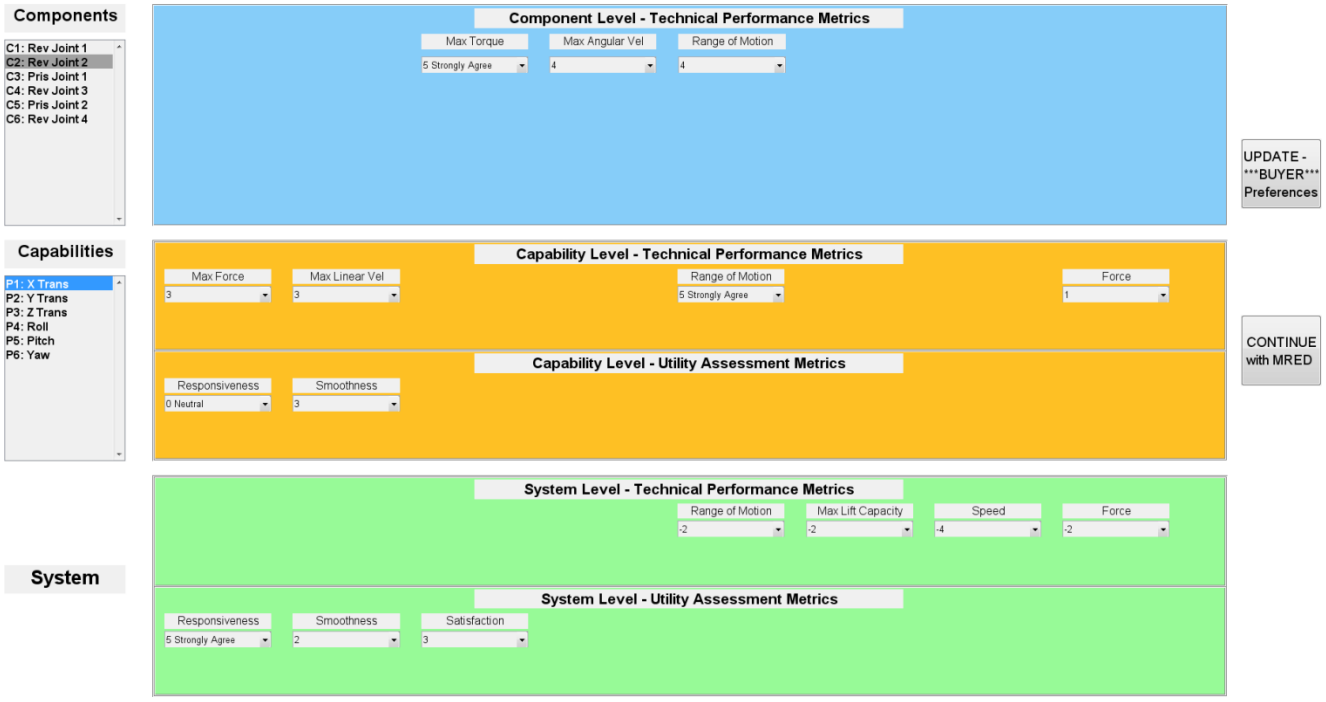


Figure 5. Example Collaborative Implementation with respect to capturing Stakeholder Preferences of *Metric-TTL* pairs for the robotic arm

TTL-Metric pairs on the 11-point scale. The *Stakeholders Preference* scores for the *TTL-Metric* pairs are averaged and *TTL-Metric* pairs with an average score less than 0 are eliminated from further consideration. A negative average score indicates that the group's aggregate preference is to not evaluate this *TTL-Metric* pair. The only exception where a negatively scoring *TTL-Metric* pair could still be considered for further evaluation is if it's grouped with other *TTL-Metric* pairs (either of the same *TTL* or same *Metric*) that were scored above '0.'

MRED then requests *Stakeholder Preferences* on the possible *Personnel/TTL-Metric* pair combinations based upon the *TTL-Metric* pairs that scored above '0,' the available *Personnel*, and MRED's constraints on which *Personnel* can realistically interact and/or evaluate the different types of *TTL-Metric* pairs. The scores for each *Personnel/TTL-Metric* pair combination are averaged and those combinations scoring a '0' or lower are eliminated from further consideration. In some instances, it may be desired to set the elimination threshold to a higher value (e.g., '1' or '1.5'). This would be at the *MRED Operator's* discretion given the total amount of *TTL-Metric* pairs being considered, the number of pairs with positive averages, etc.

Once the *TTL-Metric* pairs are combined with additional blueprint elements, it's plausible that some of the *Stakeholders* may not have preferences regarding specific combinations. This situation is likely due to a *Stakeholder* being asked to rate a combination whose *TTL-Metric* pair the *Stakeholder* rated poorly or did not have an opinion. To counteract this situation, *Stakeholders* have the power to issue a 'NV' for an entire group of blueprint elements, in addition to individual elements.

The CEV process of 1) preference capture, 2) averaging, and 3) elimination is repeated with *Personnel*, *Knowledge*, and *Autonomy Levels*, then *Environments*, followed by *Evaluation Scenarios* and finally *Explicit Environmental Factors*. The final output of this process is a series of blueprints ordered based upon those receiving the highest scores throughout the CEV process. This process is demonstrated in an example throughout the following section.

4. MRED EXAMPLE

The CEV process is demonstrated using the robot arm example presented in Section 2.4. A subset of the *TTLs* and *Metrics* are paired up according to the relationships presented in Table 3 where the *Stakeholders* provide their preferences to evaluate each *TTL-Metric* pair according to the Evaluative Voting process defined in Section 3.2. The *Stakeholder Preference* scores are presented in Table 6.

The reason a subset of the potential *TTL-Metric* pairs are used in this example is so that the process could be shown in detail. The full set of *TTL-Metric* pairs is easily scored, averaged, and processed in Matlab code that is being developed. The overall robotic example is not as large or complex (relatively speaking) as compared to other technologies. An autonomous ground vehicle would be an example of a more complicated technology for evaluation.

Table 6 - Evaluative Voting Scores for TTL-Metric Pairs for Robotic Arm

EVALUATIVE VOTING	STAKEHOLDERS				
TTL-Metric Pairs	Buyer	Eval Designer	Sponsor	Tech Dev	User
C_1 - Max Torque	NV	4	1	4	NV
C_1 - Max Angular Velocit	NV	4	1	4	NV
C_1 - Range of Motion	NV	5	1	5	NV
C_2 - Max Torque	NV	4	1	2	NV
C_2 - Max Angular Velocit	NV	4	1	2	NV
C_2 - Range of Motion	NV	5	1	5	NV
P_3 - Max Force	3	3	5	4	4
P_3 - Linear Velocity	2	3	5	3	5
P_3 - Range of Motion	4	5	5	5	4
P_4 - Max Torque	2	3	5	3	0
P_4 - Max Angular Velocit	1	3	5	3	-1
P_4 - Range of Motion	4	5	5	5	4
S - Max Lift Capacity	5	-2	-1	-4	5
S - Speed	4	-4	-1	-5	4
S - Force	4	-4	-1	-4	3

Recall that 'NV' indicates No Vote. This means that the average of the first *TTL-Metric* pair presented in Table 6 is (4+1+4)/3=3 since two *Stakeholders* cast an 'NV' score. In this example, if an 'NV' score was counted as '0' and averaged with the other scores, then this *TTL-Metric* pair average would be (0+4+1+4+0)/5=1.8. Removing 'NV' scores from the averages enables the *Stakeholders* to not impact the option to evaluate or not to evaluate a given *TTL-Metric* pair (at this step in the overall process) if they believe they are not equipped to make an informed decision. Table 7 presents the average scores from Table 6.

Table 7 - Evaluative Voting Averages for TTL-Metric Pairs for Robotic Arm

TTL-Metric Pairs	AVERAGE
P_3 - Range of Motion	**4.60**
P_4 - Range of Motion	**4.60**
P_3 - Max Force	**3.80**
C_1 - Range of Motion	**3.67**
C_2 - Range of Motion	**3.67**
P_3 - Linear Velocity	**3.60**
C_1 - Max Torque	**3.00**
C_1 - Max Angular Velocity	**3.00**
P_4 - Max Torque	**2.60**
C_2 - Max Torque	**2.33**
C_2 - Max Angular Velocity	**2.33**
P_4 - Max Angular Velocity	**2.20**
~~S - Max Lift Capacity~~	**0.60**
~~S - Speed~~	**-0.40**
~~S - Force~~	**-0.40**

The bottom three *TTL-Metric* pairs are removed from further consideration given the negative scores of the last two pairs and the range between these averages and the rest of the pairs.

It is not surprising to see the *System* excluded from consideration (shown in Table 7). This is likely early on in the development process and during the first round of evaluations where *Components* and *Capabilities* are still undergoing significant changes. Whether or not the *System* is ready for testing at this point is heavily dependent upon the type of *Technology*, the *Maturity* and *Reliability* of its constituent *TTLs*, etc.

The next step in the CEV process is to have each *Stakeholder* assign their preference scores for the possible *Personnel* that could use and/or evaluate each of the *TTL-Metric* pairs. Given

that there are numerous *Personnel* options available for testing, the *Evaluation Designer* must consider the practicality of grouping pairs by *TTL* or pairs by *Metric*. In what manner they should be grouped (*TTL* vs. *Metric*) and even if they should be grouped at all is technology-specific and driven by the quantity of *TTL-Metric* pairs.

Groupings by *Technology Test Level (TTL)* are established in Table 8. This appears to be a logical decision considering that the 12 remaining *TTL-Metric* pairs are split among four unique *TTLs*. Note that Table 8 not only presents the individual pair averages within each group, it also shows the group average of these pairs and the pair average max within a group. Both of these values are important to consider when moving deeper into the CEV process so it's easily identifiable as to what groups, on the whole, are important to evaluate and which groups have the most critical elements.

Table 8 - TTL Groupings of Remaining TTL-Metric Pairs for Robotic Arm

TTL GROUPINGS		METRICS	Pair Averages	Group Average	Pair Average Max
	P_3	Range of Motion	4.60	4.00	4.60
		Max Force	3.80		
		Linear Velocity	3.60		
	P_4	Range of Motion	4.60	3.13	4.60
		Max Torque	2.60		
		Max Angular Velocity	2.20		
	C_1	Range of Motion	3.67	3.22	3.67
		Max Torque	3.00		
		Max Angular Velocity	3.00		
	C_2	Range of Motion	3.67	2.78	3.67
		Max Torque	2.33		
		Max Angular Velocity	2.33		

5. CONCLUSION

The definition of key relationships exploited within MRED and the integration of Collaborative Evaluating Voting (CEV) are fundamental pieces in the finalization of the MRED methodology. These relationship matrices, along with the *Evaluation Designer's* ability to set the specific relations, enable MRED to eliminate test plan elements based upon the availability of their relations. Likewise, CEV allows MRED to capture the *Stakeholder Preferences* of the various test plan elements which will ultimately lead to the generation of the most preferred sets of evaluation blueprints. The next efforts will finalize the MRED methodology to include scoring the output sets of evaluation blueprints so it's evident which are most preferred. MRED is proving to be an invaluable tool towards the generation and rapid re-iteration of evaluation blueprints to test complex, advanced, and intelligent systems.

6. ACKNOWLEDGMENTS

The authors would like to acknowledge the support of NIST's Intelligent Systems Division.

7. REFERENCES

[1] Dummett, M., 1998, "The Borda Count and Agenda Manipulation," *Social Choice and Welfare*. 15, 2 (1998), 289-296.

[2] Dym, C., Wood, W. and Scott, M., 2002, "Rank Ordering Engineering Designs: Pairwise Comparison Charts and Borda Counts," *Research in Engineering Design*. 13 (2002), 236-242.

[3] Geanakoplos, J., 2005, "Three brief proofs of Arrow's Impossibility Theorem," *Economic Theory*. 26 (2005), 211-215.

[4] Hillinger, Claude, 2004, "Voting and the Cardinal Aggregation of Judgments," Munich Discussion Paper 2004-9 (May 2004), http://epub.ub.uni-muenchen.de/353/.

[5] Olcer, A. and Odabasi, A., 2005, "A new fuzzy multiple attribute group decision making methodology and its application to propulsion/maneuvering system selection problem," *European Journal of Operational Research*. 166 (2005), 93-114.

[6] Saari, D., 2006, "Which is Better: the Condorcet or Borda Winner?" *Social Choice and Welfare*. 26, 1 (January 2006), 107-129.

[7] Suh, N.P., 1998, "Axiomatic Design Theory for Systems," *Research in Engineering Design*. 10,4 (December 1998), 189-209.

[8] Thurston, D.L., 2001, "Real and Misconceived Limitations to Decision Based Design with Utility Analysis," *Journal of Mechanical Design*. 123 (June 2001), 176-182.

[9] Weiss, B.A., Schmidt, L.C., Scott, H., and Schlenoff, C.I., 2010, "The Multi-Relationship Evaluation Design Framework: Designing Testing Plans to Comprehensively Assess Advanced and Intelligent Technologies," *Proceedings of the ASME 2010 International Design Engineering Technical Conferences (IDETC) – 22ND International Conference on Design Theory and Methodology (DTM)*.

[10] Weiss, B.A. and Schmidt, L.C., 2010, "The Multi-Relationship Evaluation Design Framework: Creating Evaluation Blueprints to Assess Advanced and Intelligent Technologies," *Proceedings of the 2010 Performance Metrics for Intelligent Systems (PerMIS) Workshop*.

[11] Weiss, B.A. and Schmidt, L.C., 2011, "The Multi-Relationship Evaluation Design Framework: Producing Evaluation Blueprints to Test Emerging, Advanced, and Intelligent Systems," *ITEA Journal*. 32, 2 (June 2011), 191-200.

[12] Weiss, B.A. and Schmidt, L.C., 2011, "Multi-Relationship Evaluation Design: Formalizing Test Plan Input and Output Elements for Evaluating Developing Intelligent Systems," *Proceedings of the ASME 2011 International Design Engineering Technical Conferences (IDETC) – 23RD International Conference on Design Theory and Methodology (DTM)*.

[13] Weiss, B.A. and Schmidt, L.C., 2011, "Multi-Relationship Evaluation Design: Formalizing Test Plan Input and Output Blueprint Elements for Testing Developing Intelligent Systems," *ITEA Journal*. 32, 4 (December 2011), 479-488.

[14] Xu, Z., 2007, "Multiple-Attribute Group Decision Making with Different Formats of Preference Information on Attributes," *IEEE Transactions on Systems, Man, and Cybernetics*. 37, 6 (December 2007), 1500-1511.

An IEEE 1588 Performance Testing Dashboard for Power Industry Requirements

Julien Amelot
NIST
100 Bureau Dr.
Gaithersburg, MD 20899
1 (240) 668-4660
julien.amelot@gmail.com

Ya-Shian Li-Baboud
NIST
100 Bureau Dr.
Gaithersburg, MD 20899
1 (301) 975-5319
ya-shian@nist.gov

Clement Vasseur
NIST
100 Bureau Dr.
Gaithersburg, MD 20899
1 (301) 975-5787
clement.vasseur@nist.gov

Jeffrey Fletcher
University of Michigan
2350 Hayward St
Ann Arbor, MI 48109
jeffflet@umich.edu

Dhananjay Anand
University of Michigan
2350 Hayward St
Ann Arbor, MI 48109
danand@umich.edu

James Moyne
University of Michigan
2350 Hayward St
Ann Arbor, MI 48109
moyne@umich.edu

ABSTRACT

The numerous time synchronization performance requirements in the Smart Grid necessitate a set of common metrics and test methods. The test methods help to verify the ability of the network system and its components to meet the power industry's accuracy, reliability and interoperability criteria for next-generation substations. In order to develop viable metrics and test methods, an IEEE 1588 Testbed for the power industry has been established. To ease the challenges of testing, monitoring and analysis of the results, a software-based testing dashboard was designed and implemented. The dashboard streamlines the performance testing process by converging multiple tests for accuracy, reliability and interoperability into a centralized interface. The dashboard software enables real-time visualization and analysis of the results. The paper details the design and implementation of the IEEE 1588 Power Industry Performance Testing Dashboard as well as an update of the preliminary findings from the testbed.

Categories and Subject Descriptors

C.2.2 **[Network Protocols]**: Protocol verification
H.5.2 **[User Interfaces]**: Standardization, Benchmarking Evaluation/methodology

General Terms

Measurement, Performance, Reliability, Experimentation, Standardization, Security, Verification.

Keywords

IEEE 1588, time synchronization, test methods, conformance testing, PMU

(c) 2012 Association for Computing Machinery. ACM acknowledges that this contribution was authored or co-authored by a contractor or affiliate of the U.S. Government. As such, the Government retains a nonexclusive, royalty-free right to publish or reproduce this article, or to allow others to do so, for Government purposes only. PerMIS'12, March 20-22, 2012, College Park, MD, USA.
Copyright © 2012 ACM 978-1-4503-1126-7-3/22/12...$10.00

1. INTRODUCTION

Enabling the next-generation automated substation's ability to gather multitudes of data from intelligent electronic devices (IEDs) will require sufficient contextual data quality. Improved data quality will minimize uncertainty when processing the data to establish situational awareness for more efficient and reliable substation control. The impact of data quality on distributed control algorithms over an asynchronous network have been shown to impact the quality of state estimation [1]. Accurate time-stamps are required for merging data from heterogeneous sources. This data is essential in determining cause and effect. The network of substation end devices will require time synchronization with worst-case accuracy on the order of ±1 μs [2]. Reliable, high accuracy time synchronization continues to be difficult to achieve in complex systems [3]. Wide-Area Monitoring Systems (WAMS) can benefit from monitoring the accuracy of time synchronization and assessing the quality of the WAMS applications based on timing accuracy achieved [3]. Similarly, substation monitoring and control applications, which propagate information to the wide area network, also need to have accurate time synchronization as one factor in achieving high quality control models by reducing measurement uncertainty. The IEEE 1588 Precision Time Protocol (PTP) provides a promising solution for enabling network time synchronization over the data line within a substation network. However, IEEE 1588 is a nascent standard. Concerns regarding the ability to reliably maintain 1 μs accuracy need to be addressed. The IEEE 1588 testbed for the power industry provides a neutral venue to characterize factors impacting IEEE 1588 performance and to develop test methods to verify the metrics.

In order to streamline the testing process against the numerous requirements with respect to accuracy, reliability and interoperability, a software-based IEEE 1588 performance testing dashboard has been developed. The dashboard, through a Graphical User Interface (GUI), enables performance monitoring of the IEEE 1588 devices on the network, while providing centralized execution of the test methods, data visualization and performance analysis. The dashboard is designed to readily integrate into any IEEE 1588-compatible network as it is based upon the Management Node messages in the 1588 version 2

standard [4]. The metrics used are based on industry requirements [1,5]. This paper introduces a novel means of enhancing the management node features to provide an automated testing dashboard for assessing conformance to the IEEE 1588 standard and IEEE 1588 power industry profile requirements. Additionally, the paper details test methods and results from new test scenarios including ring topology, ring topology link failure, traffic load, interoperability, and security.

2. PERFORMANCE CRITERIA

The dashboard test suites assess the performance of IEEE 1588 devices on the network where the performance criteria are accuracy, reliability and interoperability as depicted in Figure 1. The implementation currently focuses on the ability of IEEE 1588 devices to reliably maintain the required synchronization under a variety of plausible scenarios.

Factors impacting reliability include the implementation's capability to maintain synchronization over time under all conditions ranging from ideal, to stressed, to failure conditions in substation topologies such as linear, star and ring. Stressed conditions include traffic bursts on the network that could create packet delay variation (PDV), which significantly degrade the synchronization accuracy. Failure modes include loss of network connectivity, during which the substation must maintain synchronization of its network for as long as possible and as close to UTC (Coordinated Universal Time) as possible. The current metrics used to assess the reliability of the synchronization include synchronization offset with respect to the GM (Grandmaster), mean path delay between the GM and the OC (Ordinary Clock), and out-of-specification probability of 10^{-4}. Additionally, the number of security vulnerabilities is also considered a reliability metric.

Cybersecurity is pertinent to the Smart Grid. The National Institute of Standards and Technology (NIST) has developed guidelines for Smart Grid cybersecurity [6]. Therefore, reliability of the synchronization is also dependent upon the ability of the slave node and network to detect and to defend against cybersecurity attacks. Thus far, the dashboard includes test methods for Denial of Service (DoS), masquerade, delay, and multicast poisoning. The dashboard can also serve as a tool to warn the admin of potential deviations.

Interoperability among the IEEE 1588 nodes not only impacts accuracy and reliability, but also has implications on ease of system integration and interchangeability of substation devices. The IEEE 1588 standard specifies many requirements. As a proof of concept, a few requirements have been selected for the current testbed. Among the interoperability specifications that can impact performance, one IEEE 1588 parameter selected for testing is the synchronization (sync) interval. The evaluation method would include a scorecard of the required and optional functions for a specific IEEE 1588 node. Other interoperability metrics include ease of integration and ease of interchangeability.

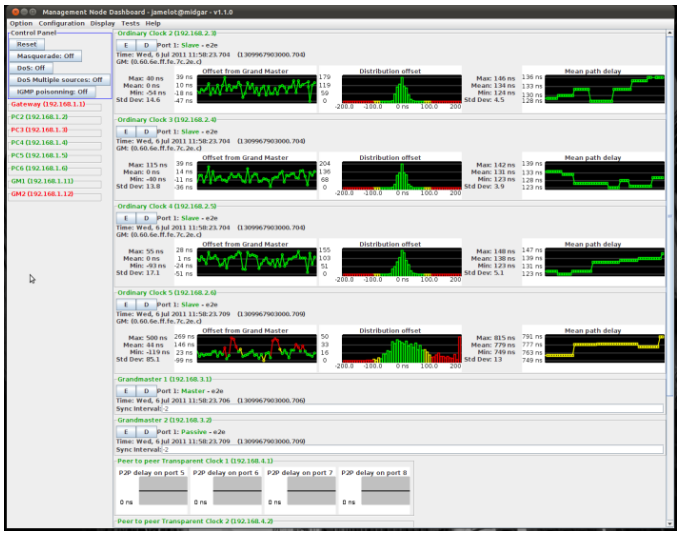


Figure 2: IEEE 1588 Dashboard using an enhanced management node for network configuration, test scenario deployment and results analysis.

3. DASHBOARD DESIGN AND IMPLEMENTATION

The dashboard, shown in Figure 2, provides a centralized monitoring interface, automated means of executing test scenarios, visualizing the data in real-time, as well as real-time analysis statistics of the key metrics identified for the IEEE 1588 Power Systems profile [2]. The dashboard also enables remote configuration of IEEE 1588 nodes in the network.

3.1 Management Node

The foundation of the test dashboard relies on the management messages. The IEEE 1588 management messages provide the ability to set and obtain data regarding the performance and status of the IEEE 1588 devices in the network. The management messages provide the ability to remotely and dynamically monitor and configure the network and each IEEE 1588 device.

3.2 Traffic Simulation

Another component of the testbed that is integrated into the dashboard is the traffic generator. The dashboard enables execution of the traffic generator through the GUI. In order to provide practical test methods, traffic loads representative of next-generation substations need to be simulated. As the traffic characteristics of next-generation substations are not yet available, the first set of simulations is based upon G.8261 Timing and Synchronization Aspects in Packet Networks [7]. The traffic patterns include static, square and ramp. The simulator can generate traffic at up to 100 percent of the network bandwidth using a specified traffic model. The objective is to inject traffic based on the IEC 61850 standard [8] and to simulate networks under heavy duress during a fault occurrence. It is expected that

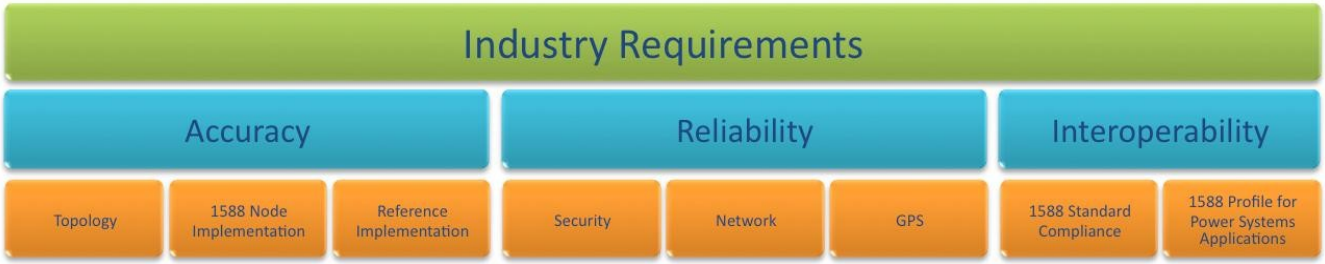


Figure 1: Performance criteria of the IEEE 1588 testbed.

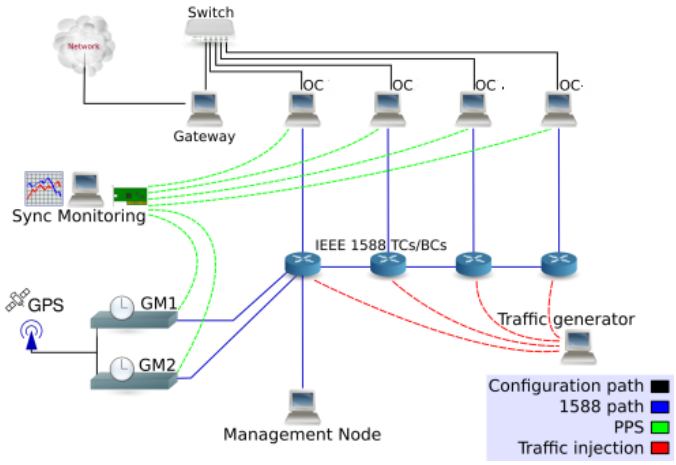


Figure 3: Testbed schematic updated with traffic generator.

during a fault occurrence, the network will experience frequent traffic bursts. It is imperative to have good synchronization during a fault occurrence to be able to accurately correlate the cause and effect.

3.3 Graphical User Interface (GUI)

The dashboard monitors the offset of synchronization and mean path delay between the Grandmaster and the ordinary clocks. To see the reliability over time, a histogram displaying the distribution of the synchronization offset is enhanced with color-coded outliers to highlight the frequency of occurrence. The mean path delay is also shown. When nodes are in peer-to-peer (P2P) mode, the delays between the peers are displayed. The dashboard monitors the current status of the IEEE 1588 devices including the current elected Grandmaster and whether the ordinary clock is synchronized. The status of all the IEEE 1588 nodes, synchronization offsets over time, the distribution of the offsets, and mean path delays are visualized through the GUI in real-time as shown in Figure 2. The dashboard alerts the user when the offset approaches 75 ns and 100 ns, by color-coding the points yellow and red, respectively. The alert thresholds are configurable by the user via the GUI.

3.4 Test Execution

In order to ease the testing process to be able to automate the execution and repetition of the tests to provide data for analysis, the dashboard enables remote configuration of and automates the execution of the tests. The dashboard is capable of executing an entire test suite, a combination of different types of tests and parameter configurations. The following IEEE 1588 parameters and configuration variables can be set through the GUI: syncInterval, announceInterval, DelayReqInterval, DelayMechanism, and port status. Other test scenarios that can be executed via the dashboard include the security and conformance test methods, Grandmaster switchover, as well as network holdover and convergence. Data plots of the synchronization offset and mean path delay from the slave are also automatically generated for each test scenario.

Table 1: Synchronization accuracy and reliability using RSTP in ring topology with link failures in a 2 hour test

	OC2	OC3	OC4	OC5
Maximum Offset (ns)	1206	186	191	262
1 µs out-of-specification probability	4.15×10^{-4}	0	0	0

3.5 Scalability through simulation

The simulation aims to incorporate virtual versions of common Smart Grid devices, specifically the PMU (phasor measurement unit), into the testbed. PMUs are becoming increasingly important in wide area monitoring and protection schemes. PMUs provide voltage and current phasor measurements to detect anomalies in the grid [9]. PMUs depend on synchronized time for accurate measurements. Therefore accurate clock synchronization on the order of 1 µs of UTC is needed, which is within PTP capabilities. The simulation provides the ability to incorporate realistic synchrophasor traffic into the network [10].

4. PERFORMANCE RESULTS

As shown in Figure 3, the testbed is comprised of redundant PTP Grandmasters (denoted as GM1 and GM2) synchronized to the Global Positioning System (GPS). For the results described in this paper, we used four PTP switches with two different implementations. The PTP switches can be configured as Transparent Clocks (TCs) or Boundary Clocks (BCs). The PTP network currently has five ordinary clocks (OCs) configured in slave mode. OC2, OC3 and OC4 are based upon the same implementation. OC3 has an oven-controlled crystal oscillator (OCXO), while OC2 and OC4 have temperature-controlled crystal oscillators (TCXOs).

4.1 Ring topology link failure

A series of tests, as configured in Figure 4, were conducted to determine the synchronization accuracy and reliability of the ring topology with two different protocols, Rapid Spanning Tree Protocol (RSTP) and Media Redundancy Protocol (MRP). A

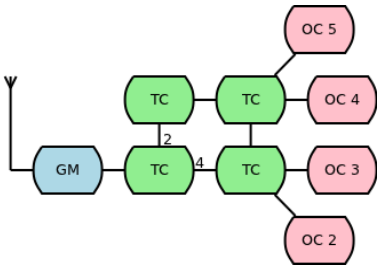


Figure 4: Ring topology scenario with two link failures at ports 2 and 4.

failure scenario where link failures force the packets to traverse in two different directions is investigated.

Figures 5 and 6 provide a comparison of the results from the link failure scenario between the two protocols. The area shaded in red denotes the time duting which port 2 is closed for three minutes and opened and subsequently, port 4 is closed for 3 minutes and opened such that all nodes in the ring would be affected at least once. With RSTP, the network did not maintain the accuracy and reliability requirements. When the link fails in RSTP, packets are not routed correctly, leading to packet loss. The protocol takes about tens of seconds to be able to recover from the failure scenario. The poor synchronization performance is due to lost packets during the link failure. Regardless of the quality of the quartz, the synchronization accuracy of the slave node can be impacted by the loss and re-route of the packets. The re-route of the packets can introduce PDV depending on the location of the

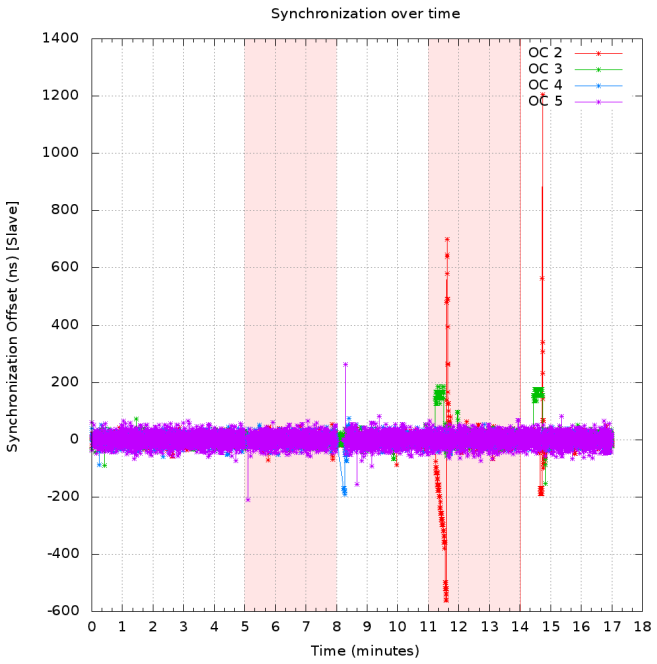


Figure 5: Synchronization offset with link failure in ring topology using RSTP.

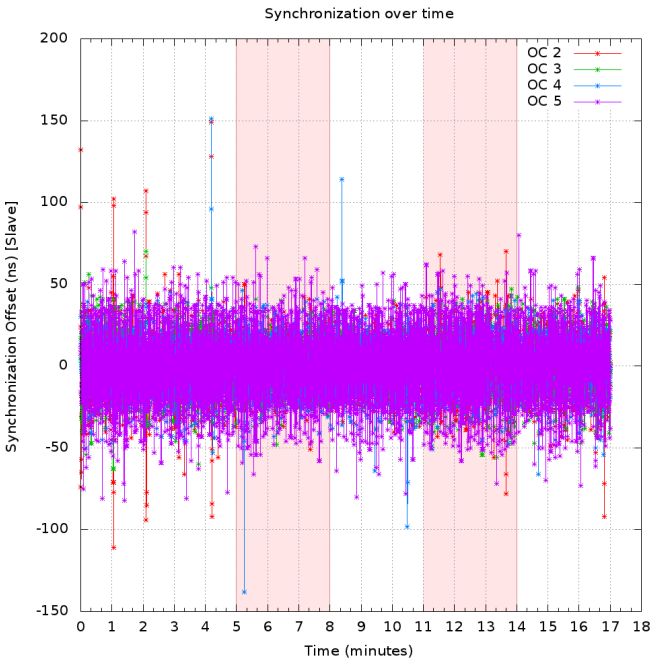


Figure 6: Synchronization offset with link failure in ring topology using MRP.

node. In Table 1, the maximum offset and out-of-specification probability are shown. For a single run, one OC did not meet the 10^{-4} out-of-specification requirement. In contrast, with MRP, where there is a deterministic response to a link failure on the order of tens of milliseconds, the results after a link failure expectedly show a consistent synchronization offset within the hundreds of nanosecond range after 50 runs. When a link failure occurs in an MRP ring, the topology is able to maintain the synchronization performance over the network of four switches. Packet loss is minimized, thus maintaining the communication between the GM and the OC. Therefore, the accuracy of the synchronization is not affected.

4.2 Network traffic bursts

Due to fault conditions in the substation, which may result in short but frequent bursts of traffic, this test scenario emulates what would occur when substation data is sampled at high frequencies in order to detect transient fault occurrences. We conjectured that static heavy traffic loads would not impact IEEE 1588 because TCs are able to compensate for the jitter by time-stamping at the ingress and egress ports, therefore removing the PDV. An accurate implementation of the TC should be able to maintain the synchronization accuracy over the four hops. The traffic bursts occur over the duration of two hours. The traffic is injected as square steps, with a period of 1 h, where the minimum network load threshold is at 5 percent and a maximum network load threshold is at 95 percent with each load lasting for 30 minutes. The traffic injected is based upon the traffic model 1 of G.8261/Y.1361 [7]. As shown in Figure 7, the IEEE 1588 devices were configured in a linear topology with three hops, with the slave nodes on the last hop to assess the synchronization performance. The traffic generator node injects packets at the specified percentages into the first hop and absorbs the extraneous traffic from the third hop. To ensure the correct level of traffic is being generated, a network packet analyzer was used to verify the quantity and sizes of the packets. We tested two device implementations on the third hop, TC A and TC B. Results from both TCs indicate that there were no significant time synchronization performance setbacks due to the bursts of traffic as shown in Figure 8. The slaves were able to maintain similar variation in mean path delay with a maximum offset of less than 200 ns. Heavy traffic, with use of TCs, did not have impact on the synchronization of the slaves and the ability of the TCs to time-stamp the messages.

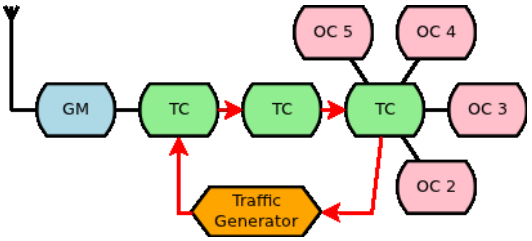


Figure 7: IEEE 1588 topology for network traffic scenario.

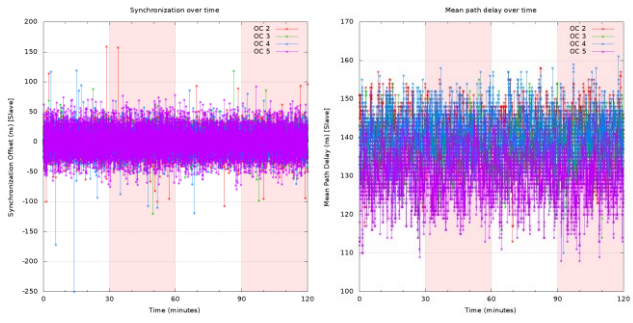


Figure 8: Mean path delay and synchronization offset between Grandmaster and slave nodes through TC B.

4.3 C. Holdover and convergence

The holdover tests provide a view of how the IEEE 1588 nodes would fare without a Master clock. The holdover durations tested include 10 s, 100 s, and 1000 s. With accurate time-stamping in the TC, the IEEE 1588 OCs were able to support holdover between 10 to 100 s while remaining within 1 µs accuracy. Table 2 provides a sample of the synchronization offsets after the node establishes contact with the Grandmaster. OC3 holdover ranged from 200 ns to 2.5 µs at 10 s and 1000 s respectively, whereas a less stable clock, OC4, drifted 448 ns in 10 s to a drift of 4.7 µs in 1000 s. OC5, which is compromised by a TC introducing a large timing error drifted significantly with a 2.6 µs offset at 10 s. At 1000 s, the maximum offsets of all three OCs went significantly above the 1 µs threshold. It is important to note that since the dashboard relies on the offset responses from the IEEE 1588 slave nodes, it is currently not recording data when it is not synchronized to a Grandmaster. The dashboard will integrate the hardware measurement to be able to provide data during the holdover. In contrast to results from [5], the automation of test deployment enabled more data to be obtained on holdover and convergence patterns. Figure 9 indicates a consistent convergence pattern and duration within seconds over ten iterations, with an hour stabilization period. The holdover dispersion between runs indicates a large range of uncertainty in the behavior of the OC. While the pattern is consistent, the amount of drift can vary significantly. The variation is due to conditions such as ambient temperature, which can contribute to the variation in the drift rates. To address the issue of ambient temperature, using more robust quartz such as an OCXO would guarantee a smaller margin of error. Analyzing and isolating the factors impacting the variation could ensure greater repeatability. However, the initial results indicate devices could benefit from robust shielding to be able to handle ambient conditions within the substations without adversely affecting the synchronization performance.

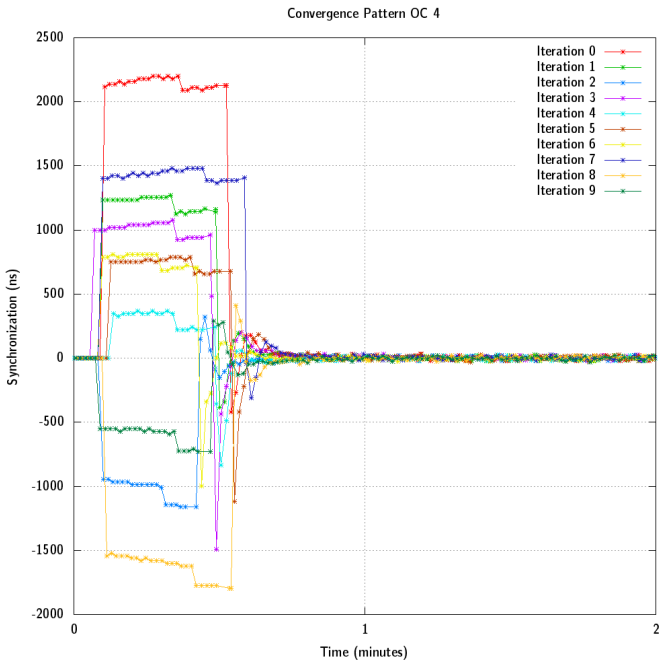


Figure 9: Convergence patterns from ten iterations for OC4 after a 5 minutes holdover duration.

4.4 Security

The method of testing IEEE 1588 security is by exposing the network to attacks and detecting vulnerabilities. Several security vulnerabilities of the IEEE 1588 protocol have been identified [11], [12], [13] and [14]. The attacks implemented include masquerade, DoS, and multicast poisoning. Masquerade enables the attacker to control the synchronization, while DoS and multicast poisoning would leave the slave clocks without a master. The dashboard provides a basic framework to readily deploy the security tests and can be readily extended to include more test methods. Using default configurations, the devices tested succumbed to masquerade, but not the basic DoS attack.

The goal of the masquerade attack is to become the best master clock such that all the IEEE 1588 devices synchronize with the rogue clock. In order for the best master clock algorithm to select this clock, the rogue clock sends an announce message describing itself as the best clock within the network. Once it has been selected as the best master clock it becomes the GM. It can disrupt the accuracy of the time synchronization by periodically sending the sync messages and responding to the delay request. The results have repeatedly indicated the nodes, by default, would synchronize to the new GM. The rogue GM can introduce both obvious offsets, which can be verified by other clocks or it can introduce subtle variations. With IEEE 1588 slaves in default configuration, the vulnerability existed on all devices in the network.

For multicast poisoning, IEEE 1588 is using multicast packets to communicate between the devices. This attack aims at isolating a device from the IEEE 1588 multicast group. It continuously sends Internet Group Management Protocol (IGMP) Leave packets, which notify the network that a device is leaving a multicast group. The vulnerability would prevent an OC from receiving any multicast IEEE 1588 messages, and therefore compromise the synchronization. The multicast poisoning attack will only work if the IEEE 1588 BCs and TCs are taking into account the IGMP messages. The testbed is currently configured for broadcast messages, so the vulnerability does not exist by default.

To realize the DoS attack, the test overloads the Grandmaster with IEEE 1588 delay request messages from different slave nodes, which prevent the Grandmaster from sending the Sync packets to synchronize the other devices. This attack was partially successful on our testbed, some transparent clocks were able to detect the DoS attack and close the port where it was coming from. A more advanced DoS attack, where the packets are disguised as originating from multiple sources, was successful on all the nodes.

Table 2: Comparison of holdover synchronization offsets

	OC3	OC4	OC5
10 s	200 ns	448 ns	2589 ns
100 s	430 ns	1099 ns	53831 ns
1000 s	2487 ns	4710 ns	707651 ns

4.5 Interoperability

Interoperability test methods are being developed by implementing the IEEE 1588 management messages for retrieving and configuring IEEE 1588 parameters. Additionally, requirements specified by the profile can also be included into the dashboard conformance test method suite. The purpose is to determine whether the required or optional functions are available to enable both improved performance and ease of management. The interoperability tests evaluate both the percentage of required functions available as well as the percentage of optional functions available. To extend to the power industry requirements, the performance of the IEEE 1588 devices can be compared against the IEEE 1588 Power Profile requirements.

In addition to verifying the existence of the required features, the dashboard provides additional analysis capabilities to verify the implementation performs to the specified configuration. For example, one interoperability test includes the ability to query the synchronization frequency available and then for each frequency determine the actual number of synchronization packets received within a specified window of time. Figure 10 displays the results from the synchronization rate where the left column is the specified log sync interval, the middle column show the rates and actual number of packets received when the test goes from a log interval of -3 to 3, and the last column show the rates and actual number of packets where the interval range is 3 to -3 to ensure the ability for rapid transition between interval specifications in both directions.

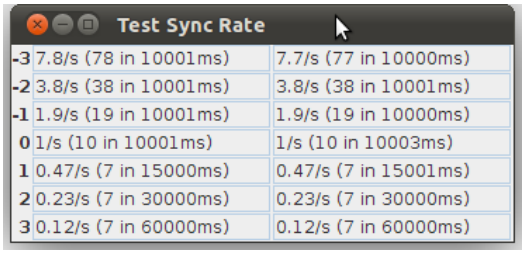


Figure 10: Interoperability test for synchronization rate verifying both the options available and the actual rate.

5. CONCLUSION AND FUTURE WORK

The IEEE 1588 Test Dashboard enables network time synchronization performance monitoring and streamlines performance testing through a centralized GUI. It automates the execution of test methods for evaluating the accuracy, reliability and interoperability performance criteria against the IEEE 1588 version 2 standard and IEEE 1588 profile for the power industry. The dashboard is also easily extensible to include IEEE 1588 profiles from other industries. The dashboard has significantly eased the testing and data acquisition process. It enables the deployment of a series of test scenarios and the ability to readily repeat the series of tests to optimize the consistency of the different iterations by minimizing the variables introduced when running the tests manually. Increasing the number of repeatable runs ensures sufficient data can be collected and statistically analyzed. The dashboard can also be utilized by vendors and customers to measure the performance of their network of IEEE 1588 devices based on the criteria discussed. An open-source version of the IEEE 1588 dashboard software is planned for release to allow testing against the IEEE 1588 standard as well as the power profile.

Additional test scenarios were implemented and investigated through the new dashboard software. The dashboard enables remote configuration of a ring network allowing automated testing of a ring topology by opening and closing the ring to simulate link failures. Additionally the traffic generator was integrated into the testbed network, where the dashboard can execute the script to enable various types of traffic loads to deploy various traffic patterns and models.

Furthermore, the dashboard provides a prototype of how vulnerability testing can be developed and deployed. Though only a limited number of devices were available for test, by default, each node was vulnerable to at least some of the cybersecurity attacks. Therefore, it is imperative for the network administrator to ensure perimeter security for the IEEE 1588 devices in the network given the cybersecurity requirements of the Smart Grid [6]. To protect the network against these attacks, one solution is to implement Annex K of IEEE 1588 [6]. However, vulnerabilities have also been found and must be addressed [15]. A complete solution would be a secure protocol along with a security policy for the entire network [6].

Interoperability can also be a significant challenge to achieving the performance and reliability necessary to meet the power industry requirements. The dashboard implementation provides a prototype of how conformance testing can be executed via the Management Node messages in addition to profile requirements. In addition to verification of IEEE 1588 capabilities required in the profile, such as the accuracy requirement, the dashboard can also serve as a means to display the status of all the IEEE 1588-enabled based on the Management Base Information (MIB) Objects [2].

Future work on the test dashboard will include integration with the hardware synchronization offset measurement [5]. The focus will also include development of test methods for security, interoperability as well as conformance to the IEEE 1588 Power Profile industry requirements. Additional security tests, such as replay and delay attacks, as well as countermeasures will be implemented. The performance impact of the countermeasures will also be analyzed. A substation network simulation will also be integrated. The current IEEE 1588 simulation is limited in to replicating the effect of the synchronization protocol on each node's simulated local time. Future work will involve replicating the IEEE 1588 protocol down to each individual packet within the simulation. Along with the bridge between the physical testbed and simulation, this will allow the virtual nodes to act as IEEE 1588 slaves, exchanging synchronization messages with a real world grandmaster clock. The simulation would transition towards building a virtual substation network model synchronized with IEEE 1588. The testbed will also continue to expand to characterize new metrics impacting the performance criteria of IEEE 1588.

6. ACKNOWLEDGMENT

We would like to thank Jerry Stenbakken, James Gilsinn, Kevin Brady, Galina Antonova, Kang Lee and the IEEE PSRC WG, and numerous others who provided support and suggestions on the development of the testbed.

7. REFERENCES

[1] D. Anand, J.G. Fletcher, Y. Li-Baboud, and J. Moyne, “A practical implementation of distributed system control over an asynchronous Ethernet network using time stamped data,” IEEE Conference on Automation Science and Engineering, August 21-24, 2010, Toronto, Canada.

[2] IEEE Standard Profile for use of IEEE 1588TM Precision Time Protocol in Power System Applications, IEEE Power System Relaying Committee and Substations Committee, July 2011.

[3] S. Meliopoulos and A. Bose, “Substation of the Future: A Feasibility Study,” Power System Engineering Research Center Publication 10-17, October 2010.

[4] IEEE 1588-2008, Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems, IEEE Instrumentation and Measurement Society, TC-9, The Institute of Electrical and Electronics Engineers, Inc., New York, NY, 24 July 2008.

[5] J. Amelot, J. Fletcher, D. Anand, C. Vasseur, Y. Li-Baboud, J. Moyne, "An IEEE 1588 time synchronization testbed for assessing power distribution requirements," Precision Clock Synchronization for Measurement Control and Communication (ISPCS), 2010 International IEEE Symposium, pp.13-18, Sept. 27-Oct. 1 2010.

[6] NISTIR 7628, Guidelines for Smart Grid Cybersecurity, September 2010.

[7] G.8261/Y.1361 Timing and Synchronization Aspects in Packet Networks. ITU-T Recommendation.

[8] IEC 61850 Communication Networks and Subsystems in Substations.

[9] R.E. Wilson. “PMUs”, IEEE Potentials, vol. 13, pp. 23-26, 1994.

[10] M. Chenine. Wide Area Monitoring and Control Systems – ApplicationCommunication Requirements and Simulation. PhD Thesis, KTH Royal Institute of Technology, 2009.

[11] A. Treytl, G. Gaderer, B. Hirschler, R. Cohen, "Traps and pitfalls in secure clock synchronization," Precision Clock Synchronization for Measurement, Control and Communication, 2007. ISPCS 2007. IEEE International Symposium on, vol., no., pp.18-24, 1-3 Oct. 2007.

[12] J. Tsang, K. Beznosov "A Security Analysis of the Precise Time Protocol," Technical Report, Vancouver, Canada, Laboratory for Education and Research in Secure Systems Engineering (LERSSE), University of British Columbia, LERSSE-TR-2006-02, 4 December, 2006, pp.20.

[13] G. Gaderer, A. Treytl, and T. Sauter, "Security Aspects for IEEE 1588 based Clock Synchronization Protocols." IEEE International Workshop on Factory Communication Systems (WFCS06), Torino, Italy, pp. 247-250, June 2006.

[14] M. Ullmann, M. Vogeler, "Delay attacks-Implication on NTP and IEEE 1588 time synchronization," IEEE International Symposium on Precision Clock Synchronization for Measurement, Control and Communication, Brescia, Italy, pp.97-102, Oct. 12-16, 2009.

[15] A. Treytl, B. Hirschler, "Security flaws and workarounds for IEEE 1588 (transparent) clocks," *IEEE International Symposium on Precision Clock Synchronization*, Brescia, Italy, pp.1-6, Oct. 12-16, 2009.