Available online at www.sciencedirect.com
SCIENCE DIRECT
Robotics and Autonomous Systems 49 (2004) 7–12
Robotics and Autonomous Systems
www.elsevier.com/locate/robot

Knowledge representation for a trash collecting robot: results from the 2004 AAAI Spring Symposium

Stephen Balakirsky[a,*], Elena Messina[a], Craig Schlenoff[a], Scott Smith[b], Michael Uschold[b]

[a] *National Institute of Standards and Technology, Intelligent Systems Division, 100 Bureau Drive, Stop 8230, Gaithersburg, MD 20899, USA*
[b] *Boeing, Phantom Works, P.O. Box 3707, mls 7L-40, Seattle, WA 98124, USA*

Received 27 July 2004; accepted 27 July 2004
Available online 12 October 2004

Abstract

The attendees of the Knowledge Representation and Ontologies for Autonomous Systems Symposium[1] applied their collective intelligence in an attempt to solve one of the universe's great challenges: how do you autonomously collect garbage from an airport facility? The attendees divided into three pre-selected, cross-disciplinary groups that were headed by Stephen Balakirsky (NIST), Elena Messina (NIST), and Scott Smith (Boeing). The teams were given limited direction and 3 h to develop a solution to the problem. Results from the groups were briefed the following morning at a plenary session.

Keywords: Challenge problem; AAAI spring symposium; Knowledge representation; Ontology; Autonomous systems

1. Scenario

The challenge problem addressed by the working groups was to design the knowledge architecture to enable a team of five autonomous robots (known as TrashBots) to perform trash removal at an airport. This involves identifying and removing trash from the floor and seating area. The airport corridors contain lanes (marked by yellow tape) that the TrashBots must follow when observing or picking up a piece of trash that is not in the lane. People in the airport are not restricted from entering or leaving the lanes.

The TrashBot must be able to distinguish between people, trash, and other TrashBots. For each piece of trash that is identified, the TrashBot must keep a record

[1] Part of the 2004 American Association for Artificial Intelligence (AAAI) Spring Symposium.

* Corresponding author. Tel.: +1 301 975 4791; fax: +1 301 990 9688.

E-mail addresses: stephen@nist.gov (S. Balakirsky), elena.messina@nist.gov (E. Messina), craig.schlenoff@nist.gov (C. Schlenoff), scott.d.smith@boeing.com (S. Smith), michael.f.uschold@boeing.com (M. Uschold).

0921-8890/$ – see front matter © 2004 Elsevier B.V. All rights reserved.
doi:10.1016/j.robot.2004.07.013

of all pertinent information about that object and why it classified it as trash.

When possible, the TrashBots are expected to recycle. There are specially marked trash receptacles around the airport for paper, glass, and plastic recycling. There are also numerous trash receptacles for all other forms of trash. The TrashBot must identify the type of trash it finds and place it in the proper receptacle.

The TrashBots must cover the entire airport multiple times every day. The TrashBots should know where other TrashBots are at all times, and coordinate their activities to ensure that they remove trash from the airport as efficiently as possible.

The TrashBots must have full awareness of their own health status and know what action they are performing at all times. They must be able to communicate this information with the other TrashBots and proceed to a repair facility when something goes wrong.

The TrashBots need to be cognizant of people who are walking through the hallways, and for safety considerations should be no closer than a predetermined number of feet from any person at any time. Hence, the TrashBot must make near-term predictions as to where any person will be at points in the future to ensure that it will never be too close.

While collecting trash, the TrashBots are also tasked with identifying any suspicious package that they encounter, based upon qualities such as location, color, odor, shape, and size. When encountering a suspicious package, the TrashBot has a procedure that it must follow. This procedure includes not getting too close to the package, assess the package's criticality, and providing an emergency alert to a central base. The TrashBot must also immediately inform the other TrashBots as to the location of the package.

Each TrashBot is equipped with:

- A location system which informs it of its approximate location at all times.
- A sensory processing system that allows it to:
  - segment objects in the environment;
  - identify the color, shape, size, and odor of objects in the environment;
  - know the exact location of objects in the environment;
  - identify lane markings in the corridors.
- A communication system that allows it to:
  - transmit information to other TrashBots;
  - receive information from other TrashBots.
- Separate finite-capacity holding containers for glass, plastic, paper, and all other forms of trash.
- A robotic arm that can:
  - grasp any type of object and place it in the proper holding container;
  - remove a bag from the appropriate holding container and place it in the appropriate trash receptacle;
  - re-bag each holding container.
- A mobility system that can move the TrashBot in any direction.

2. Group's task

Each group was tasked with defining a knowledge architecture that provides a mechanism to capture all pertinent information for the TrashBot to perform its duties. The architecture would include specifications for the different types of knowledge that will be included, how that knowledge will be represented, and what type of interfaces will be needed between the knowledge sources. Note that the task was not to design the TrashBot itself, but only those aspects related to representing and reasoning with knowledge related to the TrashBot environment and task. This included defining the nature and contents of the TrashBot's "world model", and how the TrashBot will use that world model information.

As expected, the three groups chose different approaches to tackle the problem with two of the groups focusing on creating a task decomposition, and the third focusing on creating a knowledge decomposition.

2.1. Group 1's approach

Group 1 started by creating a complete task decomposition tree that would allow for the derivation of a set of low-level tasks (basis tasks) that are shared by several higher-level tasks (see Fig. 1). The knowledge requirements of these tasks could then be determined. It was quickly realized that due to the combination of time constraints and beautiful California weather, a complete decomposition would be impossible, thus efforts were refocused on deriving the knowledge requirements for a small subset of the total task set.

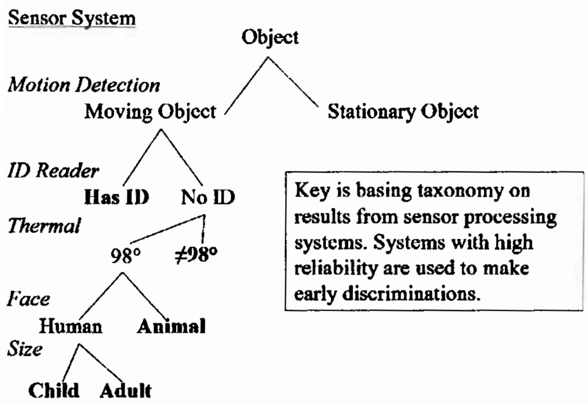

Fig. 1. Basic decision tree developed by Group 1.

Object classification was identified as a central task for the trash-collecting robots. How will the TrashBot tell the difference between people, trash, suspicious objects, and airport infrastructure? This determination was required for various higher-level tasks such as people avoidance, recycling, and suspicious object reporting. This led to three major knowledge requirements. First, the TrashBot should have an accurate a priori map including all of the stable infrastructure in the airport environment. The map could be created by prior entry and augmented by learning. A second requirement is that the mobile infrastructure in the airport environment (e.g. wheelchairs, janitorial carts) should be tagged with radio frequency (RF) beacons. A database of RF ids matched to object classes would enable gathering of important information about the mobile objects. For example, knowing that a particular object was a janitorial cart implies that there will likely be nearby trash, and that it does not move quickly or often. This gives rise to the third knowledge requirement: an object knowledge database for storing these characteristics.

The TrashBot's sensor suite would have to be relied on to recognize the remaining objects in the airport environment (that were neither part of the infrastructure nor tagged with RFIDs). Group 1 decided to implement a decision tree (similar to an object taxonomy) to aid in this decision process (a fourth knowledge requirement). The leaf nodes of this tree represented individual object classes, and the branching decisions were tied to the sensor systems of the TrashBot. The decision tree was intentionally constructed to have the most reliable object discriminators (based on available sensors) occur early in the decision process. Fig. 1 depicts a subset of this decision tree. As depicted, decisions continue to be made until the finest granularity of object that is required by any of the classification consumers has been reached. For example, if the system detects a RF tag on a moving object, classification is complete. However, if a determination has been made that the object is an airport patron, further classification as to if it is an adult or child is required so that the trajectory prediction module can better predict the object's future trajectory. One possible discriminator for this attribute was determined to be the objects size. Of course, the actual determination of what an object is only allows for easier access into additional knowledge bases that help to determine what should be done about the particular object (i.e. knowing that a widget is located in front of you does not tell you what to do with it). Further work would need to be done to inform the TrashBot on how to react to the object.

2.2. *Group 2's approach*

This group started by examining the specific knowledge that a TrashBot would need in order to accomplish its mission. The knowledge was grouped into a number of categories:

- Static spatial information—map of the airport, location of lanes, trash cans, recharging stations, gates, etc.
- Dynamic spatial information—location, velocity, direction, and time of people, TrashBots, packages, and other objects.
- Vehicle health knowledge—amounts of trash and supplies, performance of subsystems, power level, failure modes, etc.
- Object recognition knowledge—characteristics of trash, suspicious objects, trash cans, fixtures, people, as well as manipulation models of objects that must be picked up.
- Situation recognition knowledge—trash that is in use, flight loading/unloading, unattended bags/objects, people trying to communicate, emergencies, vehicle health problems.
- Operational knowledge—mission plan, current object or goal, area of responsibility, status of other TrashBots, model of own capability, exception handling and recovery method, when and how to call for help.

After exploring the knowledge that the TrashBot would need, the group looked at two specific tasks that would have to be performed; global activity planning for all TrashBots and picking up trash that a TrashBot located. For each of these tasks the specific knowledge that would be required was identified, methods of representing the knowledge were discussed, and alternative reasoning algorithms were proposed. In the course of this exercise, the group realized that the representation of the knowledge was tightly coupled to the reasoning algorithms that would be used. For instance, the knowledge representations required to perform object recognition might be very different when using a neural network versus a feature-matching method. There was a fair amount of unhappiness in the group about this observation. It was also observed that in "real world" engineering, a lot of interface representations would be defined by outside entities and that this would constrain design choices. In addition, system designed choices needed to be made to focus the representation issues. For instance, the knowledge representation would be very different between a system that incorporated global activity planning for all of the TrashBots, verses one that utilized emergent behaviors, like swarming, for coordination. Overall, the scenario provided an interesting way to explore the knowledge representation and reasoning issues involved in designing an autonomous system.

2.3. *Group 3's approach*

The third group began with analyzing the functional requirements for the TrashBot, with the intention of defining the knowledge architecture to meet the requirements. Some of the major functional requirements were investigated in more depth, to tease out the knowledge implications: what knowledge is required, what is the role of knowledge, and what reasoning is required? During the course of the breakout session, alternative views of the problem were considered; for example, breaking down the problem into the W's: "who, what, when, where, and how." Also, a scenario-based approach was taken, in which a group member played the TrashBot, and asked questions about how it would go about doing specific tasks.

The functional requirements of "looking and identifying objects" and "mobility" were looked at in more detail. From examining the "identifying objects" task, it became clear that classification of objects was necessary, based on situation and context. For instance, a newspaper is trash under some circumstances, but not others. Therefore, this type of knowledge could use attributes and relationships, all of which include uncertainty, for classification. Attributes could include color, shape, size, odor, location, orientation, texture, etc. (including derivatives). Relationships could be proximity, attachment, and context. Rules can be used to infer when something is trash. These rules could be learned over time. Indeed, humans could be queried by the TrashBot about whether something is trash. The analysis of the mobility task helped discover the a priori knowledge requirements, such as spatial representation, with feature layers and semantic tagging of locations. The variability of traffic patterns and trash density within the airport dictate that support for dynamic representations is required. This information can be shared among the TrashBots.

Several alternatives were explored in the overall architecture design. Three main areas were identified: ontology and classification, a spatial reasoning system, and a procedural, rule-based system. Once the principal knowledge modules were identified, the group considered what reasoning should be done where. However, there was not consensus reached on this point. For instance, specialist modules that aid specific functions, such as determining whether an object is trash, were thought to be useful, but there was not agreement about whether these specialists contained the reasoning or just fed sensed attributes to an ontology that made the classification decision. The third group had very interesting and spirited discussions during the breakout session, with numerous approaches and perspectives evaluated. However, given the short amount of time and diversity of opinions, no single approach was developed in much depth.

3. Summary

At the beginning of the breakout session when the groups were being briefed on their tasks, one of the participants in the audience stated "You mean all we have to do is determine the knowledge requirements for the TrashBot, not design the TrashBot itself. That sounds too easy." As the symposium attendees learned throughout the course of this exercise, determining and representing the knowledge for an autonomous system is not as easy as it may appear.

As expected, the three groups chose different approaches to tackle the problem with two of the groups focusing on creating a task decomposition, and the third focusing on creating a knowledge decomposition. This provides insight that there is no "magic bullet" in knowledge representations, and that different techniques offer different advantages and disadvantages. Allowing the participants to "get their hands dirty" by addressing the challenge problem also confirmed the belief that knowledge representation for autonomous systems is a tough problem, and should receive more attention from the community.

While no group was expected to complete the entire challenge, each learned something that could provide insight for future efforts. Group one observed that both a combination of both a priori and real-time information are required to perform object identification. They also saw value in designing a hierarchical decision tree for object identification that was based upon robot sensor data. Tying together the physical hardware with the knowledge representation structure would help to ensure a tighter coupling between the two and more efficient processing.

To group two's dismay, they recognized that a tight coupling exists between the knowledge representation formalisms, the reasoning algorithms, and the system design choices. In examining the knowledge representation options for specific tasks they found those options coupled to the reasoning algorithms that would be used in accomplishing the tasks. In addition, system design choices, distributed versus central control for example, affected choices of reasoning algorithms. Also, outside entities, such as suppliers or standards bodies, would often dictate interface representations and constrain design choices. These couplings suggest the value of making early decisions in the system design in order to focus the representation issues.

Group three identified the need for context when performing object classification. A newspaper could be trash in some contexts (it is crumpled up on the floor with no one around it) or may not be in other circumstances (it is folded up neatly on the suitcase of a person who is sitting right next to it). As such, a level of situation awareness is necessary to identify trash, above and beyond just the recognition of the object.

Overall, the breakout groups were a success in that they brought colleagues with diverse backgrounds together to jointly attack a problem that they were all interested in; knowledge representation in autonomous systems. The different backgrounds allowed the groups to perceive the problem from different perspectives, allowing them to explore the issue of knowledge representation taking into account different needs and expectations, as opposed to designing an approach that only satisfies a single perspective.

Stephen Balakirsky received the PhD degree from the University of Bremen, Germany in 2003. He is currently a researcher in the Intelligent Systems Division of the National Institute of Standards and Technology. His research interests include planning systems, knowledge representations, world modeling, and architectures for autonomous systems.

Elena Messina is Knowledge Systems Group Leader In the Intelligent Systems Division at NIST. She has over 20 years experience in robotics and in software engineering for complex systems.

Craig Schlenoff received his Bachelors degree in mechanical engineering from the University of Maryland, College Park and his Masters degree in mechanical engineering from Rensselaer Polytechnic Institute. He is a researcher in the Intelligent Systems Division at the National Institute of Standards and Technology. His research interests include knowledge representation, ontologies, and process specification, primarily applied to autonomous systems and manufacturing. He has recently served as the program manager for the Process Engineering Program at NIST as well as the Director of Ontologies and Domain Knowledge at VerticalNet, Inc. He has served on the organizing and program committee of numerous knowledge representation and ontology related conferences and workshops, and has published over 35 papers in related areas.

Scott D.G. Smith received the BSE degree in computer science from the University of Pennsylvania, Philadelphia, in 1989. He is now an Associate Technical Fellow at The Boeing Company, where he has worked for 14 years. He has lead numerous projects involving research and application of neural networks and autonomy technologies.

Mike Uschold is a research scientist at Boeing Phantom Works, the advanced research and development organization of The Boeing Company. His interests center around the field concerned with the development and application of ontologies. This includes the emerging Semantic Web, semantic integration, knowledge management, and more recently, in the area of world modeling for autonomous vehicle navigation. For over two decades, Mike has been involved in a wide range of activities in these areas, including research, applications and teaching. Dr Uschold is on the industrial advisory boards of various projects and initiatives related to the Semantic Web and other knowledge technologies. He is very active in organizing and participating in workshops and conferences on these topics. He received his BS in mathematics and physics at Canisis College in Buffalo, N.Y in 1977, a Masters in computer science from Rutgers University in 1982, and a PhD in Artificial Intelligence from The University of Edinburgh in 1991. Before arriving at the Boeing Company in 1997, Dr Uschold was a senior member of technical staff in the Artificial Intelligence Applications Institute (AIAI) at the University of Edinburgh. He has also been a lecturer and a research associate at the Department of AI at the University of Edinburgh.