The Effect of Process Models on Short-term Prediction of Moving Objects for Autonomous Driving

Raj Madhavan and Craig Schlenoff

Abstract: We are developing a novel framework, PRIDE (PRediction In Dynamic Environments), to perform moving object prediction (MOP) for autonomous ground vehicles. The underlying concept is based upon a multi-resolutional, hierarchical approach which incorporates multiple prediction algorithms into a single, unifying framework. The lower levels of the framework utilize estimation-theoretic short-term predictions while the upper levels utilize a probabilistic prediction approach based on situation recognition with an underlying cost model. The estimation-theoretic short-term prediction is via an extended Kalman filter-based algorithm using sensor data to predict the future location of moving objects with an associated confidence measure. The proposed estimation-theoretic approach does not incorporate a priori knowledge such as road networks and traffic signage and assumes uninfluenced constant trajectory and is thus suited for short-term prediction in both on-road and off-road driving. In this article, we analyze the complementary role played by vehicle kinematic models in such short-term prediction of moving objects. In particular, the importance of vehicle process models and their effect on predicting the positions and orientations of moving objects for autonomous ground vehicle navigation are examined. We present results using field data obtained from different autonomous ground vehicles operating in outdoor environments.

Keywords: Autonomous ground vehicles, estimation theory, moving object prediction.

1. INTRODUCTION

Successful and purposive navigation of autonomous ground vehicles in unstructured and unknown environments demands the competency of the vehicle to predict, with an associated level of confidence, the future locations of moving objects that could interfere with its path. Examples of moving objects that the autonomous vehicle may encounter include other vehicles, people, or animals.

A survey of the open literature reveals that there has been relatively little research in moving object prediction (MOP) in realistic outdoor domains owing to the challenges it presents. There have been experiments performed with autonomous vehicles during on-road navigation. Perhaps the most successful has been that of Dickmanns [4,5] as part of the European Prometheus project in which the autonomous vehicle drove from Munich to Odense (over 1,600 km) in Germany at a maximum velocity of 180 km/hr. Although the vehicle was able to identify and track other moving vehicles in the environment, it could only make basic predictions of where those vehicles were expected to be at points in the future, considering the external vehicle's current velocity and acceleration. Firby [8] uses NaTs (navigation templates) as a symbolic representation of static and dynamic sensed obstacles to drive a robot's motors to respond quickly to moving objects. Gueting [9] extends database structures to allow for the representation of dynamic attributes (i.e., ones that change over time) and also extends the database's query language to allow for easier querying of the values of dynamic attributes. Singhal [22] introduces the concept of dynamic occupancy grids which allow each cell to have a state vector that contains information such as a probabilistic estimate of the entity's identity, location, and characteristics (such as velocity, acceleration) along with global probability distribution functions. Nagel [10] has performed research on moving object prediction during on-road driving based upon the concept of generally describable situations, fuzzy logic, and situation graph

Manuscript received January 6, 2005; revised July 30, 2005; accepted September 20, 2005. Recommended by Editorial Board member Kyong Su Yi under the direction of Editor Jae-Bok Song. Commercial equipment and materials are identified in this article in order to adequately specify certain procedures. Such identification does not imply recommendation or endorsement by the National Institute of Standards and Technology, nor does it imply that the materials or equipment identified are necessarily the best available for the purpose.

Raj Madhavan and Craig Schlenoff are with the Intelligent Systems Division, National Institute of Standards and Technology (NIST), Gaithersburg, MD 20899-8230, U.S.A. (e-mails: raj.madhavan@ieee.org, craig.schlenoff@nist.gov).

trees. However, based on the literature, Nagel has not tried to predict what the next actions of the moving object will be and has not assigned probabilities to those actions. RRTs (Rapidly-exploring Random Trees) is a popular approach for path planning problems that involve obstacles. They have been applied to a number of areas including collision-free control of virtual humans [12] and Mars exploration vehicles [23]. However, this approach does not take into account situation recognition.

Statistical methods for estimating obstacle locations using statistical features have been proposed by other researchers such as the Hidden Markov Models (HMMs) to predict obstacle motion [24], Poisson distribution to describe the probability of collision with obstacles [21], autoregressive models for one-step ahead prediction of moving obstacles [7] or probability of occupancy of cells in grid maps [17]. The principal disadvantages of these methods are that they are computationally intensive, thus precluding real-time implementations and perhaps most importantly have only been implemented for 2D polygonal environments.

We are using the 4D/RCS (Real-Time Control System) reference model architecture [2,3] as the basis on which to apply the representational approaches that are being developed in this effort. 4D/RCS was chosen due to its explicit and well-defined world modeling capabilities and interfaces, as well as its multi-resolution, hierarchical planning approach. Specifically, 4D/RCS allows for planning at multiple levels of abstraction, using different planning approaches as well as utilizing inherently different world model representation paradigms. By applying this architecture, we can ensure that the representations being developed for moving objects can accommodate different types of planners that have different representational requirements. Under 4D/RCS, the functional elements of an intelligent system can be broadly considered to include: behavior generation (task decomposition and control), sensory processing (filtering, detection, recognition, grouping), world modeling (store and retrieve knowledge and predict future states), and value judgment (compute cost, benefit, importance, and uncertainty). These are supported by a knowledge database, and a communication system that interconnects the functional elements and the knowledge database. This collection of modules and their interconnections make up a generic node in the 4D/RCS reference model architecture. A generic node (see Fig. 1) is defined as a part of the 4D/RCS system that processes sensory information, computes values, maintains a world model, generates predictions, formulates plans, and executes tasks. Each module in the node may have an operator interface.

4D/RCS has been actively researched and developed over the past thirty years and has been applied to many domains, including manufacturing machine tool control, combat air vehicles, unmanned undersea vehicles, nuclear submarines, coal mining, stamp distribution, mail sorting, and most recently autonomous vehicle control [1]. 4D/RCS also uses its underlying hierarchical structure as the architectural mechanisms to "chunk" and abstract systems into manageable layers of complexity. This structure greatly reduces the computational complexity of each agent, which allows the architecture to better lend itself to real-time application.

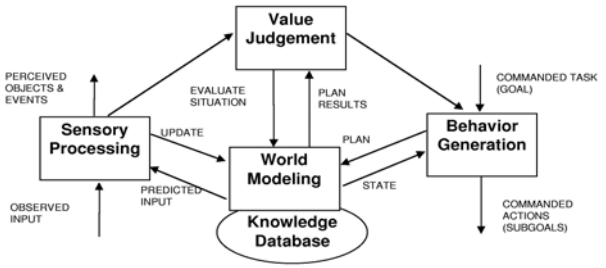


Fig. 1. A Real-time Control System (RCS) node.

The RCS architecture supports multiple behavior generation (BG) systems working cooperatively to compute a final plan for the autonomous system. The spatial and temporal resolution of the individual BG systems along with the amount of time allowed for each BG system to compute a solution are specified by the level of the architecture where it resides. In addition to multiple BG systems, multiple world models are supported, with each world model's content being tailored to the systems that it supports (in this case the BG system). As such, it is necessary for moving objects to be represented differently at the different levels of the architecture. To support this requirement, we have developed the PRIDE (PRediction In Dynamic Environments) framework. This framework supports the prediction of the future location of moving objects at various levels of resolution, thus providing prediction information at the frequency and level of abstraction necessary for planners at different levels within the hierarchy. The PRIDE framework is different than other related efforts in that it introduces a novel way to perform moving object prediction based upon a multi-resolutional, hierarchical approach which incorporates multiple prediction algorithms into a single, unifying framework. To date, two prediction approaches have been applied within this framework. The higher levels utilize a probabilistic prediction approach based upon situation recognition with an underlying cost model. At the lower levels, we utilize estimation-theoretic short-term predictions via an Extended Kalman Filter (EKF) algorithm using sensor data to predict the future location of moving objects with an associated confidence measure.

In this article, we concentrate on estimation-theoretic short-term predictions of moving objects by analyzing the role played by vehicle process models. In particular, the importance of vehicle kinematic models and their effect on predicting the positions and orientations (collectively referred to as the pose) of moving objects for autonomous ground vehicle navigation is examined using both simulated and real field data. The main contributions of the research reported in this article lie in the theoretical development of vehicle models of increasing complexity and the utility and verification of the developed models for predicting the pose of two kinematically different vehicles operating in outdoor environments.

The article is structured as follows: Section 2 details how short-term prediction of moving objects is performed within the PRIDE framework. Sections 3 and 4 describe the development of process models for a four-wheel-drive (4WD) vehicle and an Articulated Ground Vehicle (AGV) and discuss their effects in estimating the positions and orientations of the vehicles. Section 5 presents experimental and simulation results of the developed vehicle models followed by conclusions and further work in Section 6.

2. PREDICTION OF MOVING OBJECTS

The purpose of the PRIDE framework is to inform a planner about the probable location of moving objects in the environment so that the planner can make appropriate plans in dynamic environments. The MOP output is composed of a list of time steps in the future, external vehicle information (ID and type of the vehicle), all the possible future locations ($X_{position}$, $Y_{position}$) and probability information. Every predicted location has an associated probability to represent the probability that the vehicle will be at that location at a certain time in the future. Some of these predicted positions are not relevant due to a low probability, so a threshold can be applied to ignore those locations whose probabilities are under the threshold value.

Planners use the probability information from the PRIDE framework to determine the damage potential of occupying a location in space at a given time. Specifically, this damage potential will be based on the object the vehicle will encounter and the probability that the object will be there. For example, if the MOP algorithms determine that a HMMWV (High-Mobility Multipurpose Wheeled Vehicle), which we assume has a maximum damage potential of 200, has a 40% chance of occupying a point in space, then the planner may associate a damage potential (due to the presence of moving objects) of 80 (40% of 200) when determining the optimal path. With this information, a planner can produce appropriate plans in a dynamic environment.

As mentioned in Section 1, two prediction approaches have been applied within the PRIDE framework. At the higher levels of the framework, moving object prediction needs to occur at a relatively low frequency and a relatively greater level of inaccuracy is tolerable. At these levels, moving objects are identified as far as the sensors can detect, and a determination is made as to which objects should be classified as "objects of interest". In this context, an object of interest is an object that has a possibility of affecting our path in the time horizon in which we are planning. At this level, we use a moving object prediction approach based on situation recognition and probabilistic prediction algorithms to predict where we expect that object to be at various time steps into the future. Situation recognition is performed using spatio-temporal reasoning and pattern matching with an *a priori* database of situations that are expected to be seen in the environment. In these algorithms, we are typically looking at planning horizons on the order of tens of seconds into the future with one second plan steps. At this level, we are not looking to predict the exact location of the moving object. Instead, we are attempting to characterize the types of actions we expect the moving object to take and the approximate location the moving object would be in if it took that action. More information about this approach can be found in [19,20].

2.1. Short-term prediction of moving objects

At the lower levels of the framework, we utilize estimation theoretic short-term predictions via an extended Kalman filter-based algorithm using sensor data to predict the future location of moving objects with an associated confidence measure. The proposed short-term prediction approach does not incorporate *a priori* knowledge such as road networks and traffic signage and assumes uninfluenced constant trajectory and is thus suited for short-term prediction in both on-road and off-road driving.

Estimation-theoretic schemes using Kalman Filters (KFs) are well established recursive state estimation techniques where estimates of the states of a system are computed using the process and observation models [13]. The recursive nature of the algorithm utilizes the system's CPU more uniformly to provide estimates without the latency resulting from batch processing techniques. The (linear) KF is simply a recursive estimation algorithm that provides minimum mean squared estimates of the states of a linear system utilizing knowledge about the process and measurement dynamics, process and measurement noise statistics subject to Gaussian assumptions and initial condition information. When these assumptions are satisfied, the estimates provided by the Kalman filter are optimal. The extension of linear Kalman

filtering to a nonlinear system is termed extended Kalman filtering. The Extended Kalman Filter is a linear estimator for a nonlinear system obtained by linearization of the nonlinear state and observation equations. For any nonlinear system, the EKF is the best linear unbiased estimator with respect to minimum mean squared error criteria.

Within the PRIDE framework, short-term prediction of objects moving at variable speeds and at given look-ahead time instants are predicted using the EKF. Active research is exploring the integration of these two prediction approaches in a way that the predictions from one can help to reinforce or weaken the predictions of the other. The strength of using an EKF is that it provides a covariance matrix that is indicative of the uncertainty in the prediction. An EKF employs a process model to estimate the future location of the object of interest. Since the object classification module provides the type of moving object whose position and orientation needs to be predicted, we have envisaged a bank of EKFs for each type of classified object. In turn, this has the added advantage of cross-corroborating the object classification itself as the uncertainty in the EKF prediction will be an indicator of the quality of the prediction. The higher the uncertainty, the lower the confidence in the selection of the correct set of object models and thus consequent decreasing confidence of the object classification. Thus, our approach combines low-level (image segmentation/classifica-tion) and mid-level (recursive trajectory estimation) information to obtain the short-term prediction and combines it with the cross-corroboration to work symbiotically to effectively reduce the total uncertainty in predicting the positions and orientations of moving objects.

For short-term predictions, it was found that separate EKFs are necessary for different types of moving objects as opposed to a separate EKF for each individual moving object. In essence, a separate prediction equation is needed when the dynamics of the moving object significantly change. For example, multiple variations of tracked tanks could all use the same prediction equations since the kinematics of these tanks do not differ significantly. However, these equations could not be used for wheeled vehicles. Additionally, a generalized prediction equation was sufficient for near-term planning until the moving object could be classified. For example, in our simulated experiments, classification could be performed within 50 sensor observations. At a typical sensor rate of 30 Hz, this resulted in an initial classification within two seconds. Therefore, a generalized prediction equation was used for the first two seconds until the object was classified, at which point, the object-specific prediction equation was applied.

The next two sections examine the effect of process models on short-term prediction of moving objects. We develop the kinematic models of two vehicles: a four-wheel-drive (4WD) vehicle and an Articulated Ground Vehicle (AGV), by accounting for variables enabling the sufficient capture of vehicle motion that is key to short-term moving object prediction and thus central to successful autonomous navigation.

3. PROCESS MODELS FOR A FOUR-WHEEL-DRIVE VEHICLE

The model geometry of the 4WD vehicle is shown in Fig. 2. The vehicle states include the cartesian position (x_v, y_v) centered mid-way between the rear wheels and the orientation, ϕ_v of the vehicle. V is the measured velocity of the rear wheels, V' is the velocity of the front wheels along the steered angle γ and B is the vehicle wheel-base.

The continuous model of the vehicle with all quantities referenced to the center of the rear axle of the vehicle can be written as:

$$\dot{x}_v(t) = V(t)\cos\phi_v(t),$$
$$\dot{y}_v(t) = V(t)\sin\phi_v(t),$$
$$\dot{\phi}_v(t) = \left[\frac{1}{B}\right]V'(t)\sin\gamma(t).$$

Since

$$V'(t) = \frac{V(t)}{\cos\gamma(t)},$$

the model equations of the vehicle referenced to the center of the rear axle of the vehicle become:

$$\begin{aligned}\dot{x}_v(t) &= V(t)\cos\phi_v(t),\\ \dot{y}_v(t) &= V(t)\sin\phi_v(t),\\ \dot{\phi}_v(t) &= \left[\frac{1}{B}\right]V(t)\tan\gamma(t).\end{aligned} \quad (1)$$

3.1. Accounting for wheel slip

When a significant amount of wheel slip is present, the shaft velocity encoder measurements are no longer

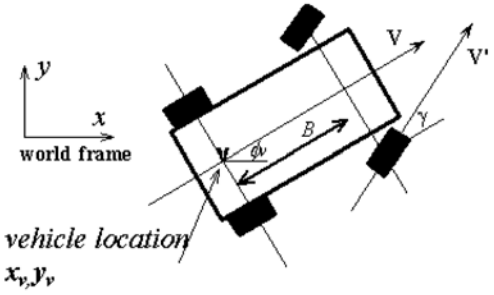


Fig. 2. Process model geometry for the 4WD vehicle.

representative of the vehicle speed but are indicative of the wheel speed. So a wheel radius state, r_v, is introduced in addition to the vehicle pose states. The wheel radius is modeled as a discrete additive disturbance rate error (a random walk) such that the error is the integral of white noise. Although in practice this variable may not evolve in a strictly Brownian manner, the Brownian model reflects the growth in uncertainty in its true value and the rate at which the true value is considered to vary. Wheel radius changes are a function of tire pressures, loading effects, temperature, wear and tear, and vehicle dynamics [6].

The velocity measured by the shaft encoder may be replaced by the product of the wheel radius and the wheel angular speed so that

$$V(t) = r_v(t)\omega(t).$$

(1) now becomes

$$\begin{aligned} \dot{x}_v(t) &= r_v(t)\omega(t)\cos\phi_v(t), \\ \dot{y}_v(t) &= r_v(t)\omega(t)\sin\phi_v(t), \\ \dot{\phi}_v(t) &= \left[\frac{1}{B}\right] r_v(t)\omega(t)\tan\gamma(t), \\ \dot{r}_v(t) &= 0. \end{aligned} \tag{2}$$

The control signals applied to the vehicle are:

$$\mathbf{u}_k = \left[\omega_k, \gamma_k\right],$$

where ω_k is the angular velocity and γ_k is the steering angle of the 4WD vehicle at time-instant k.

By integrating (2) using the Euler approximation and assuming that the control signals, ω and γ, are approximately constant over the sample period, the nominal discrete process model equations at time-instant k can be written as:

$$\begin{pmatrix} x_{v_k} \\ y_{v_k} \\ \phi_{v_k} \\ r_{v_k} \end{pmatrix} = \begin{pmatrix} x_{v_{k-1}} \\ y_{v_{k-1}} \\ \phi_{v_{k-1}} \\ r_{v_{k-1}} \end{pmatrix} + \Delta T \begin{pmatrix} \omega_k r_{v_{k-1}} \cos\phi_{v_{k-1}} \\ \omega_k r_{v_{k-1}} \sin\phi_{v_{k-1}} \\ \left[\frac{1}{B}\right] \omega_k r_{v_{k-1}} \tan\gamma_k \\ \delta r_{v_{k-1}} \end{pmatrix}, \tag{3}$$

where ΔT is the synchronous sampling interval between states at discrete-time instants, $(k-1)$ and k.

The errors due to the control inputs ω and γ are modeled as simple additive noise sources, $\delta\omega$ and $\delta\gamma$ about their respective means $\bar{\omega}$ and $\bar{\gamma}$ as below:

$$\begin{aligned} \omega_k &= \bar{\omega}_k + \delta\omega_k, \\ \gamma_k &= \bar{\gamma}_k + \delta\gamma_k. \end{aligned}$$

The error source vector due to both modeling errors and uncertainty in control is now defined as:

$$\delta\mathbf{w}_k = \left[\delta\omega_k, \delta\gamma_k, \delta r_{v_k}\right]^T,$$

where T denotes matrix transposition. The source errors $\delta\omega$, $\delta\gamma$ and δr_v are assumed to be zero-mean, uncorrelated Gaussian sequences with constant variances σ_ω^2, σ_γ^2 and $\sigma_{r_v}^2$, respectively.

3.2. Estimation cycle for the 4WD vehicle

To formulate a Kalman filter algorithm, process and observation (measurement) models are needed. In view of the availability of data at discrete instants of time from asynchronous sensors and implementation based on digital computers, a discrete time formulation of the continuous time vehicle kinematic models are necessary. In discrete time, only discrete sampling instants $t_0, t_1, \ldots$ are considered. The discrete time process model is usually derived by integrating the continuous time process model between two consecutive time steps.

A general discrete time process model can be expressed as

$$\mathbf{x}_k = \mathbf{f}\left(\mathbf{x}_{k-1}, \mathbf{u}_k, k\right) + \mathbf{w}_k, \tag{4}$$

where $\mathbf{f}\left(\cdot,\cdot,k\right)$ is a discrete function that maps the previous state and control inputs to the current state, $\mathbf{x}_k$ is the state at time instant k, $\mathbf{u}_k$ is a known control vector, and $\mathbf{w}_k$ is the discrete process noise.

Observations of the state $\mathbf{x}_k$ are made according to the observation model:

$$\mathbf{z}_k = \mathbf{h}\left(\mathbf{x}_k, k\right) + \mathbf{v}_k, \tag{5}$$

where $\mathbf{h}\left(\cdot,k\right)$ is the discrete function that maps the current state to observations.

The process noise, $\mathbf{w}_k$ and the measurement noise $\mathbf{v}_k$ are assumed to be Gaussian-distributed random variables of zero mean with covariances $\mathbf{Q}_k$ and $\mathbf{R}_k$, respectively, and are written as:

$$\begin{aligned} \mathbf{w}_k &\sim \mathrm{N}\left(0, \mathbf{Q}_k\right); \mathrm{E}\left[\mathbf{w}_k\right] = 0 \forall k, \\ \mathbf{v}_k &\sim \mathrm{N}\left(0, \mathbf{R}_k\right); \mathrm{E}\left[\mathbf{v}_k\right] = 0 \forall k, \end{aligned}$$

where $\mathrm{E}\left[\cdot\right]$ is the mathematical expectation operator. the notation $\mathbf{x} \sim \mathrm{N}\left(\mathbf{m}, \mathbf{P}\right)$ indicates that $\mathbf{x}$ is a Gaussian (normal) random vector with mean $\mathbf{m}$ and covariance $\mathbf{P}$.

In an autonomous vehicle navigation context, the prediction stage uses a model of the motion of the vehicle (a process model having the form described in

(4)) to predict the vehicle position, $\hat{\mathbf{x}}_{(k|k-1)}$, at instant k given the information available until and including instant $(k-1)$. The state prediction function $\mathbf{f}(\cdot)$ is defined by (4) assuming zero process and control noise. The prediction of state is therefore obtained by simply substituting the previous state and current control inputs into the state transition equation with no noise. Taking expected values of (3) conditioned on the first $(k-1)$ observations, the state prediction becomes:

$$\begin{pmatrix} x_{v_{(k|k-1)}} \\ y_{v_{(k|k-1)}} \\ \phi_{v_{(k|k-1)}} \\ r_{v_{(k|k-1)}} \end{pmatrix} = \begin{pmatrix} x_{v_{(k-1|k-1)}} \\ y_{v_{(k-1|k-1)}} \\ \phi_{v_{(k-1|k-1)}} \\ r_{v_{(k-1|k-1)}} \end{pmatrix} + \Delta T \begin{pmatrix} \omega_k r_{v_{(k-1|k-1)}} \cos\phi_{v_{(k-1|k-1)}} \\ \omega_k r_{v_{(k-1|k-1)}} \sin\phi_{v_{(k-1|k-1)}} \\ \left[\frac{1}{B}\right]\omega_k r_{v_{(k-1|k-1)}} \tan\gamma_k \\ 0 \end{pmatrix}.$$

The prediction covariance can now be computed using:

$$\mathbf{P}_{(k|k-1)} = \nabla\mathbf{f}_{\mathbf{x}_{v_k}} \mathbf{P}_{(k-1|k-1)} \nabla\mathbf{f}_{\mathbf{x}_{v_k}}^T + \nabla\mathbf{f}_{\mathbf{w}_k} \Sigma_c \nabla\mathbf{f}_{\mathbf{w}_k}^T, \quad (6)$$

where $\nabla\mathbf{f}_{\mathbf{x}_{v_k}}$ represents the Jacobian evaluated with respect to the states, $\nabla\mathbf{f}_{\mathbf{w}_k}$ is the Jacobian with respect to the error sources and Σ_c is the noise strength matrix given by:

$$\nabla\mathbf{f}_{\mathbf{x}_{v_k}} = \begin{bmatrix} 1 & 0 & -\Delta T\omega_k r_{v_{k-1}} \sin\phi_{v_{k-1}} & \Delta T\omega_k \cos\phi_{v_{k-1}} \\ 0 & 1 & \Delta T\omega_k r_{v_{k-1}} \cos\phi_{v_{k-1}} & \Delta T\omega_k \sin\phi_{v_{k-1}} \\ 0 & 0 & 1 & \Delta T\left[\frac{1}{B}\right]\omega_k \tan\gamma_k \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\nabla\mathbf{f}_{\mathbf{w}_k} = \Delta T \begin{bmatrix} r_{v_{k-1}} \cos\phi_{v_{k-1}} & 0 & 0 \\ r_{v_{k-1}} \sin\phi_{v_{k-1}} & 0 & 0 \\ \left[\frac{1}{B}\right] r_{v_{k-1}} \tan\gamma_k & \left[\frac{1}{B}\right]\omega_k r_{v_{k-1}}\left(1+\tan^2\gamma_k\right) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\Sigma_c = \begin{bmatrix} \sigma_\omega^2 & 0 & 0 \\ 0 & \sigma_\gamma^2 & 0 \\ 0 & 0 & \sigma_{r_v}^2 \end{bmatrix}.$$

When an exteroceptive sensor observation becomes available, the states of the EKF (comprising the moving object's position and orientation) are to be updated so that during the next cycle the prediction starts from a reasonably known position. This is important since the EKF computes the future position based on the current position. If the prediction starts from a wrong position, the cumulative errors will result in an estimate that will be far from the truth.

Once the state and covariance predictions are available, the next step is to compute a predicted observation and a corresponding innovation for updating the predicted state. Expanding (5) as a Taylor series about the predicted state $\hat{\mathbf{x}}_{(k|k-1)}$

$$\hat{\mathbf{z}}_k = \mathbf{h}\left(\hat{\mathbf{x}}_{(k|k-1)}, \mathbf{u}_k, k\right) + \nabla\mathbf{h}_{\mathbf{x}_k}\left[\hat{\mathbf{x}}_{(k|k-1)} - \mathbf{x}_k\right] + O\left(\left[\hat{\mathbf{x}}_{(k|k-1)} - \mathbf{x}_k\right]^2\right) + \mathbf{v}_k,$$

where $\nabla\mathbf{h}_{\mathbf{x}_k}$ is the Jacobian evaluated at $\mathbf{x}_k = \hat{\mathbf{x}}_{(k|k-1)}$.

The predicted observation $\hat{\mathbf{z}}_{(k|k-1)}$ is found by using the nonlinear relation described in (5) and taking expectations conditioned on the first $(k-1)$ observations by considering only terms up to the first order and neglecting higher order terms such that

$$\hat{\mathbf{z}}_{(k|k-1)} \triangleq \mathbf{h}(\hat{\mathbf{x}}_{(k|k-1)}).$$

The difference between the actual observation and the predicted observation at time step k is termed the innovation and is written as: $\nu_k = \mathbf{z}_k - \hat{\mathbf{z}}_{(k|k-1)}$. The innovation covariance is found by squaring the estimated observation error and taking expectations conditioned on the first $(k-1)$ measurements

$$\mathbf{S}_k = \nabla\mathbf{h}_{\mathbf{x}_k} \mathbf{P}_{(k|k-1)} \nabla\mathbf{h}_{\mathbf{x}_k}^T + \mathbf{R}_k.$$

The observations that arrive are accepted only if the observation falls inside the normalized innovation validation gate:

$$\nu_k^T \mathbf{S}_k^{-1} \nu_k \le \varepsilon_\gamma,$$

where ν_k is the innovation defined as the difference between the actual and predicted positions. The value of ε_γ can be chosen from the fact that the normalized innovation sequence is a χ^2 random variable with m degrees of freedom (m being the dimension of the observation) [16].

Once a validated observation is available, the update of the estimate equal to the weighted sum of the observation and the prediction can be computed as:

$$\hat{\mathbf{x}}_{(k|k)} = \hat{\mathbf{x}}_{(k|k-1)} + \mathbf{W}_k \nu_k,$$

where $\mathbf{W}_k$ is the Kalman gain matrix determined by the relative confidence in vehicle prediction and observation and determines the influence of the innovation on the updated estimate.

The error in the updated estimate is

$$\tilde{\mathbf{x}}_{(k|k)} = \mathbf{x}_k - \hat{\mathbf{x}}_{(k|k)}$$
$$= \mathbf{x}_k - \left[\hat{\mathbf{x}}_{(k|k-1)} + \mathbf{W}_k \nu_k\right] = \tilde{\mathbf{x}}_{(k|k-1)} - \mathbf{W}_k \nu_k .$$

The covariance update is

$$\mathbf{P}_{(k|k)} = \mathrm{E}\left[\tilde{\mathbf{x}}_{(k|k)} \tilde{\mathbf{x}}_{(k|k)}^T\right] = \mathbf{P}_{(k|k-1)} - \mathbf{W}_k \mathbf{S}_k \mathbf{W}_k^T ,$$

where the Kalman gain matrix is given by

$$\mathbf{W}_k = \mathbf{P}_{(k|k-1)} \nabla \mathbf{h}_{\mathbf{x}_k}^T \mathbf{S}_k^{-1} .$$

3.3. Accounting for wheel slip and skid

To account for both slip and skid of the vehicle that might be present, two states V_s and γ_s can be added to the process model. The errors in slip and skid of the vehicle are modeled as discrete random walks such that they are the integral of white noise and are found to provide better performance.

The control signals applied to the vehicle in this case are $\mathbf{u}_k = \left[V_k, \gamma_k\right]$ where V_k and γ_k are the speed and steering angle of the 4WD vehicle, respectively. The nominal discrete process model equations at time instant k can now be written as:

$$\begin{pmatrix} x_{v_k} \\ y_{v_k} \\ \phi_{v_k} \\ V_{s_k} \\ \gamma_{s_k} \end{pmatrix} = \begin{pmatrix} x_{v_{k-1}} \\ y_{v_{k-1}} \\ \phi_{v_{k-1}} \\ V_{s_{k-1}} \\ \gamma_{s_{k-1}} \end{pmatrix} + \Delta T \begin{pmatrix} \left(V_k + V_{s_{k-1}}\right)\cos\phi_{v_{k-1}} \\ \left(V_k + V_{s_{k-1}}\right)\sin\phi_{v_{k-1}} \\ \frac{1}{B}\left(V_k + V_{s_{k-1}}\right)\tan\left(\gamma_k + \gamma_{s_{k-1}}\right) \\ \delta V_{s_{k-1}} \\ \delta\gamma_{s_{k-1}} \end{pmatrix}.$$

The errors due to the control inputs V and γ are modeled as simple additive noise sources, δV and $\delta\gamma$ about their respective means $\bar{V}$ and $\bar{\gamma}$ as below:

$$V_k = \bar{V}_k + \delta V_k ,$$
$$\gamma_k = \bar{\gamma}_k + \delta\gamma_k .$$

The error source vector is defined as:

$$\delta\mathbf{w}_k = \left[\delta V_k, \delta\gamma_k, \delta V_{s_k}, \delta\gamma_{s_k}\right]^T .$$

The source errors δV, $\delta\gamma$, δV_s and $\delta\gamma_s$ are assumed to be zero-mean, uncorrelated Gaussian sequences with constant variances σ_V^2, σ_γ^2, $\sigma_{V_s}^2$ and $\sigma_{\gamma_s}^2$, respectively.

Additionally, the vehicle position estimates (x_v, y_v) are correlated with the vehicle orientation (ϕ_v) estimates; errors as a result of the orientation cause errors in the vehicle position estimates. When the steering rate is high, the orientation error is also at its highest and the position subsequently becomes highly correlated with the orientation estimates. The addition of the Σ_s matrix helps to decorrelate the effects of orientation on the position estimates of the vehicle. The value of Σ_s was chosen such that the control signals $\left[V_k, \gamma_k\right]$ were weighted more heavily than the slip and skid of the vehicle, V_s and γ_s. In addition, since the vehicle slips more than skidding due to the inherent kinematics, V_s was weighted more heavily than γ_s. These values were obtained based on data from several field trials. Σ_s is added to (6) for computing the predicted covariance in this case.

The Jacobians and the stabilizing noise matrix are as given below:

$$\nabla\mathbf{f}_{\mathbf{x}_{v_k}} =$$
$$\begin{bmatrix} 1 & 0 & -\Delta T k_1 \sin\phi_{v_{k-1}} & \Delta T\cos\phi_{v_{k-1}} & 0 \\ 0 & 1 & \Delta T k_1 \cos\phi_{v_{k-1}} & \Delta T\sin\phi_{v_{k-1}} & 0 \\ 0 & 0 & 1 & \Delta T\frac{1}{B}\tan k_2 & \Delta T\frac{k_1}{B}(1+\tan^2 k_2) \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

$$\nabla\mathbf{f}_{\mathbf{w}_k} = \begin{bmatrix} \Delta T\cos\phi_{v_{k-1}} & 0 & 0 & 0 \\ \Delta T\sin\phi_{v_{k-1}} & 0 & 0 & 0 \\ \Delta T\frac{1}{B}\tan k_2 & \Delta T\frac{k_1}{B}(1+\tan^2 k_2) & 0 & 0 \\ 0 & 0 & \Delta T & 0 \\ 0 & 0 & 0 & \Delta T \end{bmatrix},$$

$$\Sigma_c = \begin{bmatrix} \sigma_V^2 & 0 & 0 & 0 \\ 0 & \sigma_\gamma^2 & 0 & 0 \\ 0 & 0 & \sigma_{V_s}^2 & 0 \\ 0 & 0 & 0 & \sigma_{\gamma_s}^2 \end{bmatrix},$$

$$\Sigma_s = \begin{bmatrix} \frac{\sigma_V^2}{10^2} & 0 & 0 & 0 \\ 0 & \frac{\sigma_\gamma^2}{10^2} & 0 & 0 \\ 0 & 0 & \frac{\sigma_{V_s}^2}{10^3} & 0 \\ 0 & 0 & 0 & \frac{\left(\sigma_V^2+\sigma_\gamma^2+\sigma_{V_s}^2+\sigma_{\gamma_s}^2\right)}{10^7} \end{bmatrix},$$

where $k_1 \triangleq V_k + V_{s_{k-1}}$, $k_2 \triangleq \gamma_k + \gamma_{s_{k-1}}$.

4. PROCESS MODEL FOR AN ARTICULATED GROUND VEHICLE

Articulated vehicles are widely used both in commercial and military applications. Examples of these vehicles include the Load-Haul-Dump (LHD) trucks used in the mining industry and the Small Unit Support Vehicle (SUSV) used by the military. The AGV described in this article has a front and a rear body which can rotate relative to each other and the front and rear wheel sets are fixed to remain parallel with the body of the vehicle. It is mounted on rubber tires and propelled by four-wheel drive. Steering is achieved by driving the articulation joint (located mid-way between the front and rear axles) through hinge pins connecting the two main frames. Combined with power steering, this feature provides high maneuverability and a small turning radius. Both the front and rear wheel sets are driven at the same speed through a single transmission. Bidirectional operation with the same number of speeds forward and reverse allows loads to be hauled in either direction with the same efficiency.

In general, the front and rear axles of the vehicle cannot be assumed to be equidistant from the articulation joint [18]. Here, the different lengths, L_1 and L_2 (see Fig. 3) are explicitly taken into account. The state vector for the AGV contains the pose of the vehicle (x_v, y_v, ϕ_v), the wheel radius r_v, the rear and front slip-angles (α_v, β_v). An inexpensive fiber optic gyro was employed in the AGV trials. These trials were of longer duration than the 4WD trials and thus the accompanying gyro drift was much more than that experienced during the 4WD trials. Accordingly a shaping state, ss_v was added to the state vector to compensate for the gyro drift. In the absence of this shaping state, the error due to the drift will corrupt the vehicle pose estimate. The addition of the shaping state ensures that the low frequency errors in the gyro are estimated by the EKF.

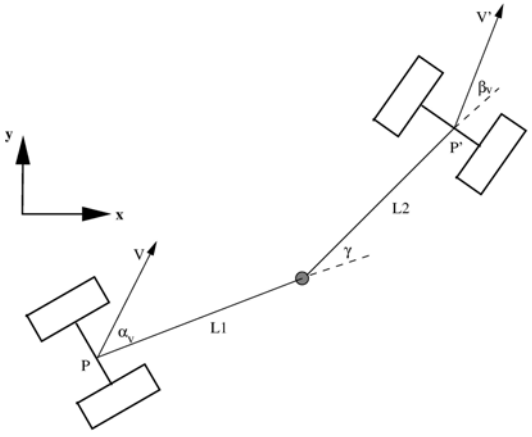


Fig. 3. Kinematic representation of the AGV.

The nominal continuous-time model of the AGV can be written as:

$$\begin{aligned}
&\dot{x}_v(t) = \omega(t) r_v(t) \cos\left[\alpha_v(t) + \phi_v(t)\right], \\
&\dot{y}_v(t) = \omega(t) r_v(t) \sin\left[\alpha_v(t) + \phi_v(t)\right], \\
&\dot{\phi}_v(t) = \frac{\omega(t) r_v(t) \sin\left[\beta_v(t) - \alpha_v(t) + \gamma(t)\right] - \dot{\gamma}(t) L_2 \cos\beta_v(t)}{L_2 \cos\beta_v(t) + L_1 \cos\left[\beta_v(t) + \gamma(t)\right]}, \\
&\dot{r}_v(t) = 0, \quad \dot{\alpha}_v(t) = 0, \quad \dot{\beta}_v(t) = 0, \quad ss_v(t) = 0.
\end{aligned} \tag{7}$$

The control vector is given by:

$$\mathbf{u}_k = \left[\omega_k, \gamma_k, \dot{\gamma}_k\right]^T,$$

where ω_k is the angular velocity, γ_k is the articulation angle of the AGV and $\dot{\gamma}_k$ is the computed derivative of γ_k at time-instant k.

Similar to the 4WD vehicle, a discrete representation of the continuous process model in (7) was obtained by a first-order Euler approximation as the control signals ω, γ and $\dot{\gamma}$ and the sensor observations are sampled at discrete instants of time. The errors in wheel radius, slip angles and the low frequency gyro errors are difficult to model accurately. However, a compromise that has been found to work well is to model the errors in these variables as random walks such that the error in each of the variables is the integral of white noise.

The nominal process model at the discrete-time instant k is given by:

$$\begin{pmatrix} x_{v_k} \\ y_{v_k} \\ \phi_{v_k} \\ r_{v_k} \\ \alpha_{v_k} \\ \beta_{v_k} \\ ss_{v_k} \end{pmatrix} = \begin{pmatrix} x_{v_{k-1}} \\ y_{v_{k-1}} \\ \phi_{v_{k-1}} \\ r_{v_{k-1}} \\ \alpha_{v_{k-1}} \\ \beta_{v_{k-1}} \\ ss_{v_{k-1}} \end{pmatrix} + \Delta T \begin{pmatrix} \omega_k r_{v_{k-1}} \cos\left(\alpha_{v_{k-1}} + \phi_{v_{k-1}}\right) \\ \omega_k r_{v_{k-1}} \sin\left(\alpha_{v_{k-1}} + \phi_{v_{k-1}}\right) \\ \dfrac{\omega_k r_{v_{k-1}} \sin\left(\beta_{v_{k-1}} - \alpha_{v_{k-1}} + \gamma_k\right) - \dot{\gamma}_k L_2 \cos\beta_{v_{k-1}}}{L_2 \cos\beta_{v_{k-1}} + L_1 \cos\left(\beta_{v_{k-1}} + \gamma_k\right)} \\ \delta r_{v_{k-1}} \\ \delta\alpha_{v_{k-1}} \\ \delta\beta_{v_{k-1}} \\ \delta ss_{v_{k-1}} \end{pmatrix}. \tag{8}$$

The primary sources of error are due to ω, γ, $\dot{\gamma}$, α_v, β_v and ss_v since the errors in these variables feed directly through to the states. The errors in control inputs are modeled as simple additive noise sources $\delta\omega$, $\delta\gamma$ and $\delta\dot{\gamma}$ about their respective means $\bar{\omega}$, $\bar{\gamma}$ and $\dot{\bar{\gamma}}$ such that

$$\begin{aligned}\omega_k &= \bar{\omega}_k + \delta\omega_k,\\ \gamma_k &= \bar{\gamma}_k + \delta\gamma_k,\\ \dot{\gamma}_k &= \dot{\bar{\gamma}}_k + \delta\dot{\gamma}_k.\end{aligned}$$

The error source vector is defined as:

$$\delta\mathbf{w}_k = \left[\delta\omega_k, \delta\gamma_k, \delta\dot{\gamma}_k, \delta r_{v_k}, \delta\alpha_{v_k}, \delta\beta_{v_k}, \delta ss_{v_k}\right]^T.$$

The source errors $\delta\omega$, $\delta\gamma$, $\delta\dot{\gamma}$, δr_v, $\delta\alpha_v$, $\delta\beta_v$ and δss_v are assumed to be zero-mean, uncorrelated Gaussian sequences with constant variances σ_ω^2, σ_γ^2, $\sigma_{\dot{\gamma}}^2$, $\sigma_{r_v}^2$, $\sigma_{\alpha_v}^2$, $\sigma_{\beta_v}^2$ and $\sigma_{ss_v}^2$, respectively.

4.1. Estimation cycle for the AGV

Taking expected values of (8) conditioned on the first $(k-1)$ observations, the state prediction becomes:

$$\begin{pmatrix} x_{v_{(k|k-1)}}\\ y_{v_{(k|k-1)}}\\ \phi_{v_{(k|k-1)}}\\ r_{v_{(k|k-1)}}\\ \alpha_{v_{(k|k-1)}}\\ \beta_{v_{(k|k-1)}}\\ ss_{v_{(k|k-1)}} \end{pmatrix} = \begin{pmatrix} x_{v_{(k-1|k-1)}}\\ y_{v_{(k-1|k-1)}}\\ \phi_{v_{(k-1|k-1)}}\\ r_{v_{(k-1|k-1)}}\\ \alpha_{v_{(k-1|k-1)}}\\ \beta_{v_{(k-1|k-1)}}\\ ss_{v_{(k-1|k-1)}} \end{pmatrix} + \Delta T \begin{pmatrix} \omega_k r_{v_{(k-1|k-1)}} \cos\left(\alpha_{v_{(k-1|k-1)}} + \phi_{v_{(k-1|k-1)}}\right)\\ \omega_k r_{v_{(k-1|k-1)}} \sin\left(\alpha_{v_{(k-1|k-1)}} + \phi_{v_{(k-1|k-1)}}\right)\\ \dfrac{\omega_k r_{v_{(k-1|k-1)}} \sin\left(\beta_{v_{(k-1|k-1)}} - \alpha_{v_{(k-1|k-1)}} + \gamma_k\right) - \dot{\gamma}_k L_2 \cos\beta_{v_{(k-1|k-1)}}}{L_2 \cos\beta_{v_{(k-1|k-1)}} + L_1 \cos\left(\beta_{v_{(k-1|k-1)}} + \gamma_k\right)}\\ 0\\ 0\\ 0\\ 0 \end{pmatrix}.$$

As before, the predicted covariance requires the computation of $\nabla\mathbf{f}_{\mathbf{x}_{v_k}}$, $\nabla\mathbf{f}_{\mathbf{w}_k}$ and Σ_c. $\nabla\mathbf{f}_{\mathbf{x}_{v_k}}$ represents the Jacobian evaluated with respect to the states, $\nabla\mathbf{f}_{\mathbf{w}_k}$ represents the Jacobian with respect to the error sources and Σ_c represents the noise strength matrix and are given as below:

$$\nabla\mathbf{f}_{\mathbf{x}_{v_k}} = \begin{bmatrix} 1 & 0 & f_{13} & f_{14} & f_{15} & 0 & 0\\ 0 & 1 & f_{23} & f_{24} & f_{25} & 0 & 0\\ 0 & 0 & 1 & f_{34} & f_{35} & f_{36} & 0\\ 0 & 0 & 0 & 1 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 1 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

$$\nabla\mathbf{f}_{\mathbf{w}_k} = \begin{bmatrix} g_{11} & 0 & 0 & 0 & 0 & 0 & 0\\ g_{21} & 0 & 0 & 0 & 0 & 0 & 0\\ g_{31} & g_{32} & g_{33} & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & \Delta T & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & \Delta T & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & \Delta T & 0\\ 0 & 0 & 0 & 0 & 0 & 0 & \Delta T \end{bmatrix},$$

$$\Sigma_c = \begin{bmatrix} \sigma_\omega^2 & 0 & 0 & 0 & 0 & 0 & 0\\ 0 & \sigma_\gamma^2 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & \sigma_{\dot{\gamma}}^2 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & \sigma_{r_v}^2 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & \sigma_{\alpha_v}^2 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & \sigma_{\beta_v}^2 & 0\\ 0 & 0 & 0 & 0 & 0 & 0 & \sigma_{ss_v}^2 \end{bmatrix}$$

and

$$\begin{aligned} f_{13} &= -\Delta T\left[\omega_k r_{v_{k-1}} \sin(\alpha_{v_{k-1}} + \phi_{v_{k-1}})\right],\\ f_{14} &= \Delta T\left[\omega_k \cos(\alpha_{v_{k-1}} + \phi_{v_{k-1}})\right],\\ f_{15} &= -\Delta T\left[\omega_k r_{v_{k-1}} \sin(\alpha_{v_{k-1}} + \phi_{v_{k-1}})\right],\\ f_{23} &= \Delta T\left[\omega_k r_{v_{k-1}} \cos(\alpha_{v_{k-1}} + \phi_{v_{k-1}})\right],\\ f_{24} &= \Delta T\left[\omega_k \sin(\alpha_{v_{k-1}} + \phi_{v_{k-1}})\right],\\ f_{25} &= \Delta T\left[\omega_k r_{v_{k-1}} \cos(\alpha_{v_{k-1}} + \phi_{v_{k-1}})\right],\\ f_{34} &= \Delta T\left[\frac{\omega_k \sin(\beta_{v_{k-1}} - \alpha_{v_{k-1}} + \gamma_k)}{L_2 \cos\beta_{v_{k-1}} + L_1 \cos(\beta_{v_{k-1}} + \gamma_k)}\right],\\ f_{35} &= -\Delta T\left[\frac{\omega_k r_{v_{k-1}} \cos(\beta_{v_{k-1}} - \alpha_{v_{k-1}} + \gamma_k)}{L_2 \cos\beta_{v_{k-1}} + L_1 \cos(\beta_{v_{k-1}} + \gamma_k)}\right]. \end{aligned}$$

$$f_{36} = \Delta T \left[\begin{array}{c} \frac{\omega_k r_{v_{k-1}} \cos(\beta_{v_{k-1}} - \alpha_{v_{k-1}} + \gamma_k) - \dot{\gamma}_k L_2 \sin\beta_{v_{k-1}}}{L_2 \cos\beta_{v_{k-1}} + L_1 \cos(\beta_{v_{k-1}} + \gamma_k)} \\ + \frac{\left[\omega_k r_{v_{k-1}} \sin(\beta_{v_{k-1}} - \alpha_{v_{k-1}} + \gamma_k) - \dot{\gamma}_k L_2 \sin\beta_{v_{k-1}}\right]\left[L_2 \sin\beta_{v_{k-1}} + L_1 \sin(\beta_{v_{k-1}} + \gamma_k)\right]}{\left[L_2 \cos\beta_{v_{k-1}} + L_1 \cos(\beta_{v_{k-1}} + \gamma_k)\right]^2} \end{array} \right],$$

$$g_{11} = \Delta T \left[r_{v_{k-1}} \cos(\alpha_{v_{k-1}} + \phi_{v_{k-1}}) \right],$$

$$g_{21} = \Delta T \left[r_{v_{k-1}} \sin(\alpha_{v_{k-1}} + \phi_{v_{k-1}}) \right],$$

$$g_{31} = \Delta T \left[\frac{r_{v_{k-1}} \sin(\beta_{v_{k-1}} - \alpha_{v_{k-1}} + \gamma_k)}{L_2 \cos\beta_{v_{k-1}} + L_1 \cos(\beta_{v_{k-1}} + \gamma_k)} \right],$$

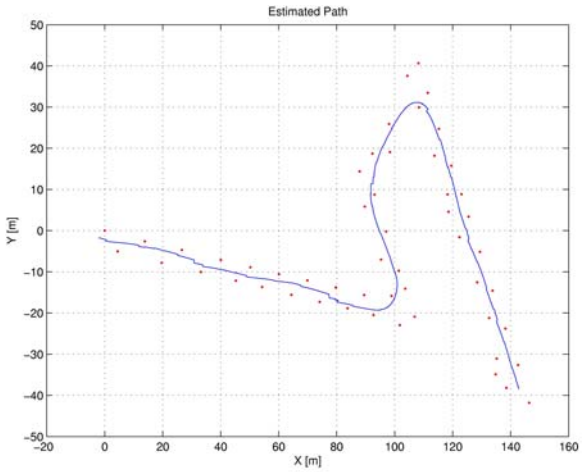
$$g_{32} = \Delta T \left[\begin{array}{c} \frac{\omega_k r_{v_{k-1}} \cos(\beta_{v_{k-1}} - \alpha_{v_{k-1}} + \gamma_k)}{L_2 \cos\beta_{v_{k-1}} + L_1 \cos(\beta_{v_{k-1}} + \gamma_k)} \\ + \frac{\left[\omega_k r_{v_{k-1}} \sin(\beta_{v_{k-1}} - \alpha_{v_{k-1}} + \gamma_k) - \dot{\gamma}_k L_2 \cos\beta_{v_{k-1}}\right]\left[L_1 \sin(\beta_{v_{k-1}} + \gamma_k)\right]}{\left[L_2 \cos\beta_{v_{k-1}} + L_1 \cos(\beta_{v_{k-1}} + \gamma_k)\right]^2} \end{array} \right],$$

$$g_{33} = -\Delta T \left[\frac{\cos\beta_{v_{k-1}}}{L_2 \cos\beta_{v_{k-1}} + L_1 \cos(\beta_{v_{k-1}} + \gamma_k)} \right].$$

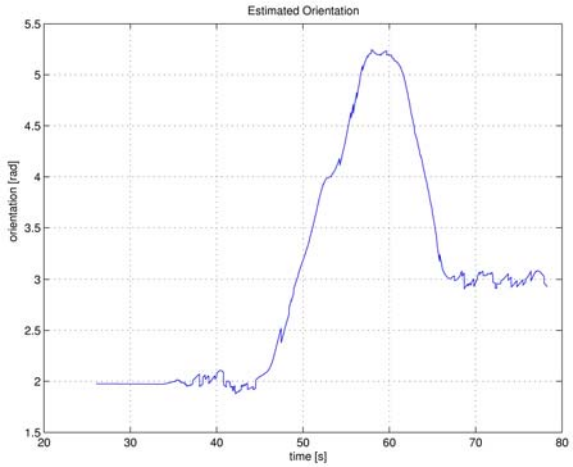
5. EXPERIMENTAL RESULTS AND DISCUSSION

To validate the process models, we present results using both simulated and real field data. The simulation data were obtained using a Distributed Interactive Simulation (DIS) platform namely One Semi-Automated Force (OneSAF), a training and research system developed by the U.S. Army's Program Executive Office for Simulation, Training, and Instrumentation Command. Typically, OneSAF models are employed to represent individual soldiers or vehicles and their coordination into orderly-moving squads and platoons; but, their tactical actions as units are planned and executed by a human controller. OneSAF uses state transition constructs inspired by finite state machines (FSMs) to represent the behavior and functionality of a process for a pre-defined number of states [11]. By querying OneSAF, we can retrieve an object's location at any given time. The real field data were logged as the 4WD vehicle and the AGV traversed unstructured and undulating outdoor terrain. This vehicle was equipped with a bearing-only laser and wheel and steering encoders. External corrections were provided using observations from landmarks in the environments that were detected by the bearing-only laser. For additional details, see [14,15].

Figs. 4(a) and 4(b) show the estimated path and orientation of the 4WD vehicle, respectively, using the process model that accounts for both vehicle slip and skid, where the dots represent the artificial landmark locations. The exclusion of the slip and skid variables in addition to the stabilizing noise matrix would result in the estimated vehicle path drifting and the error growing without bounds. This is easy to understand as the estimated vehicle position at a given instant depends on the previous estimate which makes it



(a)

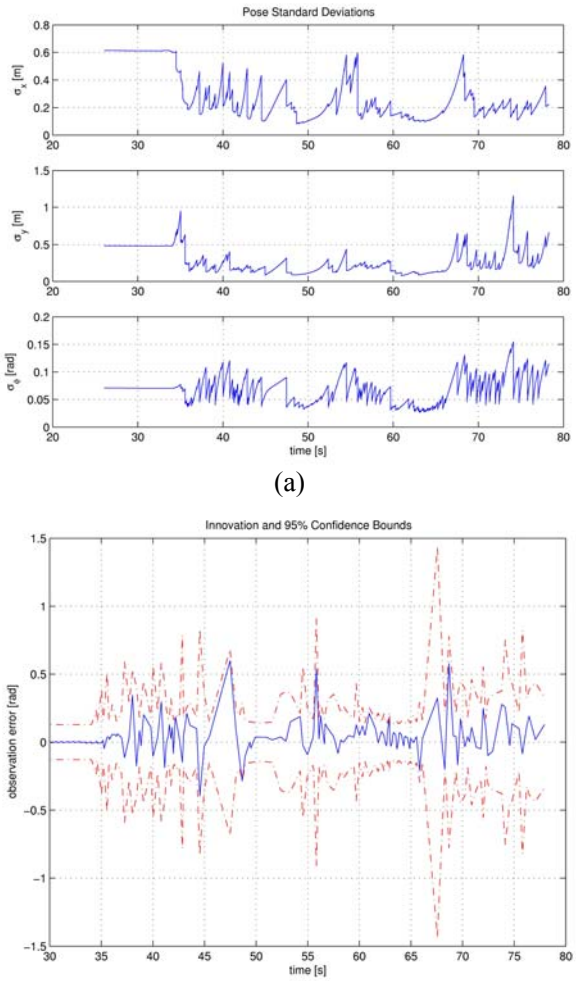


(b)

Fig. 4. (a) and (b) show the estimated path and the orientation of the 4WD vehicle, respectively. In (a), the starting location of the vehicle is at $(142.76, -38.31)$. The direction of travel is from right to left.

difficult to eliminate errors associated with the previous cycle due to sensor inaccuracies, the assumption that the heading remains constant over the sampling interval, wheel slippage and quantization effects. As a consequence, the vehicle pose (position and especially the orientation) would become less and less certain and the errors associated with the pose grow without bound.

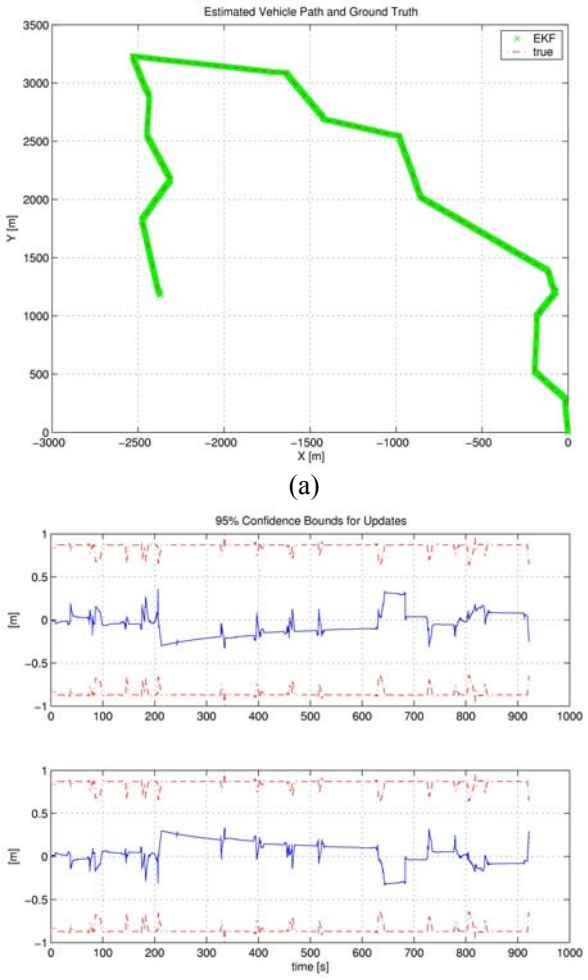
In Fig. 5(a), the periodic rise and fall of the pose standard deviations can be seen. The decrease in the standard deviations is due to certain landmarks coming into view and being detected reliably. The increase in standard deviations is due to the vehicle moving away from the landmarks and its position being estimated based on the prediction alone. When the landmarks provide no aiding information towards estimating the pose of the vehicle, the standard deviation is at a maximum. Thus the algorithm continually corrects the diverging dead-reckoning



(a)

(b)

Fig. 5. The pose standard deviations are shown in (a) and the innovation sequence is shown in (b) with 95% confidence bounds (dotted line).



(a)

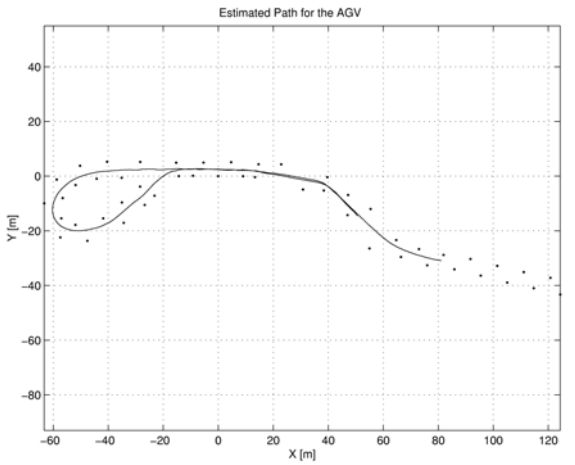
(b)

Fig. 6. (a), and (b) show the estimated path, the errors in position between the estimated path and the ground truth with 95% confidence bounds (dotted line), respectively. The traverse is approximately 7 km long and takes about 15 minutes of simulation time to travel at a speed of 8 m/s. The updates were generated at a rate of 15 Hz.

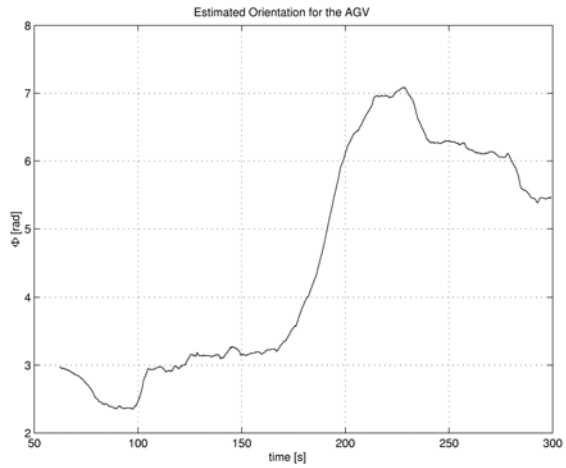
estimates based on external sensing information provided by the landmarks (also see Fig. 4(a)).

We found the use of innovation gating critical for the data association problems encountered in the research described in this article. A difficult decision in such data association problems is that, if more than one observation falls within the allowed thresholds, which one(s) should be selected? Possible strategies include selecting the closest one, selecting the average of all that fall within the threshold or not selecting any at all. Since a wrongly matched observation can have catastrophic effects on the pose estimate and subsequently on vehicle control, if more than one association was made, such observations were discarded to avoid any ambiguity. If no landmarks are found to satisfy the above two conditions then that observation was also discarded. If only one landmark passes both of the above tests, then the updates are performed with that observation. The bearing-only laser innovation sequence along with the 95% (2σ) bounds is shown in Fig. 5(b). Due to the observation validation procedure, the innovations are clearly bounded. The rise and fall pattern is also observed with the innovation, standard deviation of the laser.

Fig. 6 depicts the results obtained using data from the simulation platform. Here, we are able to generate *ground truth* to compare with the estimated position estimates generated by the vehicle model shown in (3). The position of the 4WD vehicle is predicted until the errors exceed a predefined threshold. Once the errors are above a given threshold, an update is deemed to be performed by utilizing the observations from OneSAF. We check the validity of the observation by testing if it falls within the normalized innovation gate. A validated observation is then used to update the states of the EKF and the estimation cycle continues as in the 4WD vehicle case.



(a)

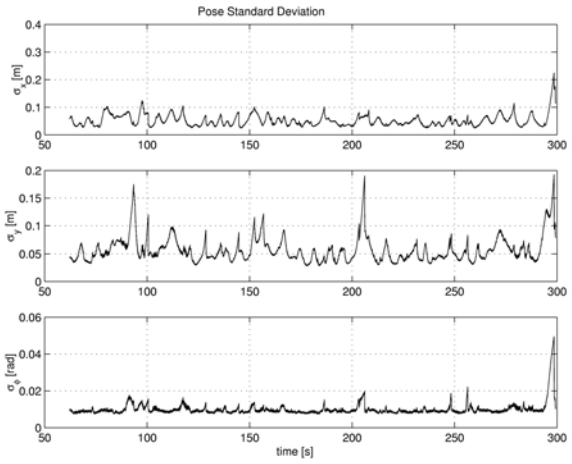


(b)

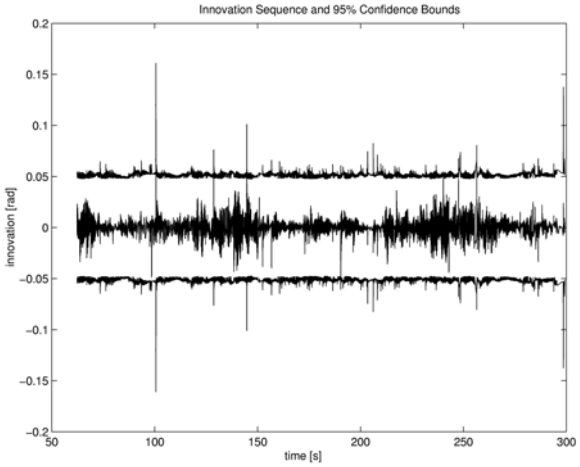
Fig. 7. The estimated path (a) and orientation (b) of the AGV. The dots represent the surveyed artificial landmark locations. The traversed path runs from right to left and is in the anti-clockwise direction around the loop at the left end of the figure. The starting location is at $(81.1, -30.9)$ in (a).

Figs. 6(a) and 6(b) show the estimated path and the position errors with the 2σ observation confidence bounds, respectively. It can be seen that the estimated path and the ground truth agree well. As the estimated path and the corresponding ground truth are very close, extra effort is required on the part of the reader to distinguish between the two. As before, the validated innovation sequence falls within the prescribed 95% (2σ) bounds indicative of consistent vehicle state estimates.

Figs. 7(a) and 7(b) show the estimated path and orientation for the AGV. The direction of vehicle travel is from right to left and is in the anti-clockwise direction around the loop at the left extreme of Fig. 7(a). The dots represent the artificial landmark



(a)

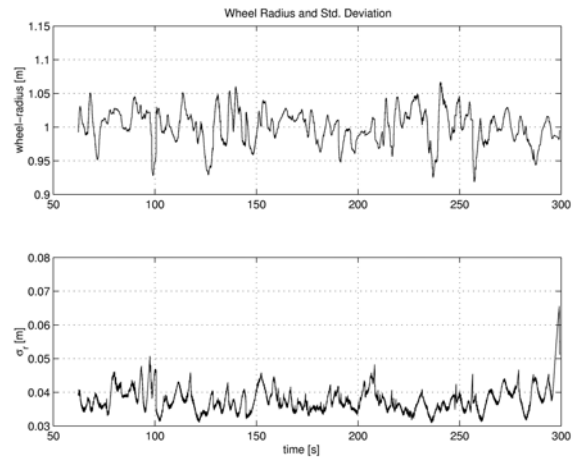


(b)

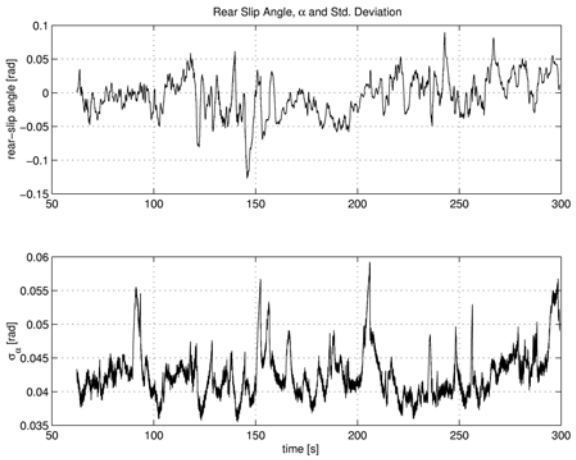
Fig. 8. The estimated standard deviations of the AGV pose are shown in (a). The laser innovation sequence and the 2σ confidence bounds are shown in (b).

locations. Figs. 8(a) and 8(b) show the estimated standard deviation of the vehicle pose and the 95% confidence bounds for the innovation sequence. As in the 4WD case, the rise and fall of the standard deviations for both the pose and the innovation sequence are indicative of the fact that these deviations rise when there are no landmarks reliably detected or landmarks are far away from the vehicle and the fall in the standard deviations are indicative of information contribution from landmarks. It can also be seen from Fig. 8(b) that the validated innovation sequence is clearly within the 2σ bounds indicative of a well-tuned filter thus resulting in consistent vehicle state estimates.

Figs. 9(a), 9(b), and 10(a) show the wheel radius, the rear-slip angle, the front-slip angle and their standard deviations, respectively. Due to the correlation between the articulation angle and the slip angles, the vehicle will slip much more during a

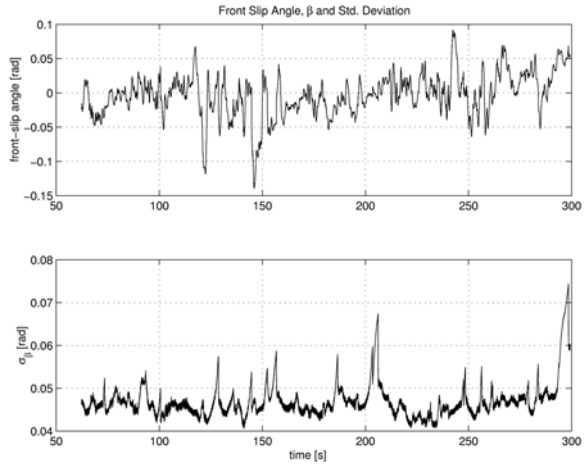


(a)

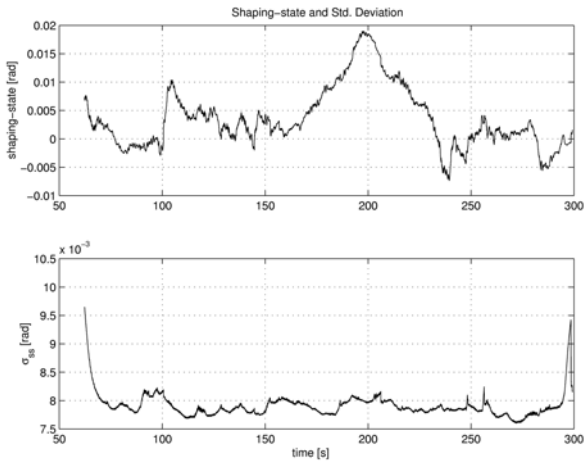


(b)

Fig. 9. The estimated wheel radius and its standard deviation for the AGV are shown in (a). The rear slip angle and its standard deviation are shown in (b).



(a)



(b)

Fig. 10. The estimated front slip angle and its standard deviation for the AGV are shown in (a). The shaping state and its standard deviation are shown in (b).

cornering maneuver in agreement with intuition. It can also be seen that the rear of the vehicle slips more than the front of the vehicle. The wheel radius also decreases during the cornering of the vehicle since the change in the forward velocity is interpreted as change in the wheel radius of the vehicle by the EKF. Fig. 10(b) shows the shaping state for the gyro-drift and its standard deviation.

6. CONCLUSIONS AND FURTHER WORK

The effect of vehicle process models on the short-term prediction of moving objects within a hierarchical, multi-resolutional framework for autonomous driving was the main theme of this article. The framework currently employs two different prediction methodologies that lend themselves best to the constraints imposed by the planning horizon and replanning rates of the planners at different levels of the control hierarchy. An estimation-theoretic approach is used at the lower levels of the hierarchy that require a fast replanning rate and where constraints on the environment do not greatly affect the predicted location of the moving object. A situation-based probabilistic prediction approach is used at the higher levels of the control hierarchy that require slower replanning rates and where constraints on the environment greatly affect the probabilities of where the moving object will be in the future.

The importance of vehicle process models and their effect on predicting the positions and orientations of two different vehicles were examined. The main contributions of the research reported in this article are the theoretical development of vehicle models of increasing complexity and the utility and verification of the developed models for predicting the pose of autonomous ground vehicles. Experimental results were presented using both simulated and real data for

a 4WD vehicle and an articulated ground vehicle operating in outdoor environments. The results clearly showed the need for sufficiently adequate process models and their importance in short-term prediction.

As we move forward with the PRIDE framework, many issues remain to be addressed. For the short-term EKF based approach, we need to build additional kinematic and dynamic models corresponding to different types of vehicles we perceive in the environment. Such models will allow for more accurate predictions that are specific to the types of vehicles we encounter. For the situation-based probabilistic approach, we need to encode additional situations (and pertinent actions when encountering those situations), and a more elaborate cost model.

The short-term predictions will also need to be quantitatively compared to the longer-term predictions being determined at the higher levels of the architecture. In addition, the short-term predictions may be used as inputs to the longer-term prediction to better refine the estimates. We also plan to investigate the applicability of other prediction algorithms such as particle filters within the framework.

REFERENCES

[1] J. Albus, "The NIST real-time control system (RCS): An application survey," *Proc. of the American Association for Artificial Intelligence (AAAI) Spring Symposium Series*, 1995.

[2] J. Albus and 33 coauthors, "4D/RCS version 2.0: A reference model architecture for unmanned vehicle systems," *Technical Report NISTIR 6910*, National Institute of Standards and Technology, Gaitherburg, MD 20899, U.S.A., 2002.

[3] J. Albus and A. Meystel, *Engineering of Mind*, John Wiley & Sons, Inc., 2001.

[4] E. Dickmanns, "An expectation-based multi-focal saccadic (EMS) vision system for vehicle guidance," *Proc. of the International Symposium on Robotics Research*, 1999.

[5] E. Dickmanns, "The development of machine vision for road vehicles in the last decade," *Proc. of the International Symposium on Intelligent Vehicles*, pp. 644-651, 2002.

[6] H. Durrant-Whyte, "An autonomous guided vehicle for cargo handling applications," *The International Journal of Robotics Research*, vol. 15, no. 5, pp. 407-440, October 1996.

[7] A. Elnager and K. Gupta, "Motion prediction of moving objects based on autoregressive model," *IEEE Trans. on Systems, Man and Cybernetics-Part A: Systems and Humans*, vol. 28, no. 6, pp. 803-810, 1998.

[8] J. Firby, "Architecture, representation, and integration: An example from robot navigation," *Proc. of the AAAI Fall Symposium Series Workshop on the Control of the Physical World by Intelligent Agents*, 1994.

[9] R. Gueting, "A foundation for representing and querying moving objects," *ACM Transactions on Database Systems*, vol. 25, no. 1, pp. 1-42, 2000.

[10] H. Haag and H.-H. Nagel, "Incremental recognition of traffic situations from video image sequences," *Image and Vision Computing*, vol. 18, pp. 137-153, 2000.

[11] A. Henninger and R. Madhavan, "Empirical comparison of predictive models for mobile agents," *Robotics and Autonomous Systems: Special Issue on Knowledge Engineering and Ontologies for Autonomous Systems 2004 AAAI Spring Symposium*, vol. 49, no. 1-2. pp. 91-103, November 2004.

[12] M. Kallman and M. Mataric, "Motion planning using dynamic roadmaps," *Proc. of the IEEE International Conference on Robotics and Automation*, April 2004.

[13] R. Kalman, "A new approach to linear filtering and prediction problems," *Transactions of the ASME-Journal of Basic Engineering*, vol. 82, no. Series D, pp. 35-45, 1960.

[14] R. Madhavan and H. Durrant-Whyte, "Natural landmark-based autonomous navigation using curvature scale space," *Robotics and Autonomous Systems*, vol. 46, no. 2, pp. 79-95, February 2004.

[15] R. Madhavan and H. Durrant-Whyte, "Terrain aided localization of autonomous ground vehicles," *Journal of Automation in Construction (Invited)*, vol. 13, no. 1, pp. 83-100, January 2004.

[16] P. Maybeck, *Stochastic Models, Estimation, and Control Vol. 1*, Academic Press, New York, June 1979.

[17] H. Moravec, "Sensor fusion in certainty grids for mobile robots," *AI Magazine*, vol. 9, no. 2, pp. 61-74, 1988.

[18] S. Scheding, G. Dissanayake, E. Nebot, and H. Durrant-Whyte, "An experiment in autonomous navigation of an underground mining vehicle," *IEEE Trans. on Robotics and Automation*, vol. 15, no. 1, pp. 85-95, February 1999.

[19] C. Schlenoff, R. Madhavan, and S. Balakirsky, "Representing dynamic environments for autonomous navigation," *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 644-649, October 2003.

[20] C. Schlenoff, R. Madhavan, and T. Barbera, "A hierarchical, multi-resolutional moving object prediction approach for autonomous on-road driving," *Proc. of the IEEE International Conference on Robotics and Automation*, pp. 1956-1961, April 2004.

[21] R. Sharma, "Locally efficient path planning in an uncertain, dynamic environment using a

probability model," *IEEE Trans. on Robotics and Automation*, vol. 8, no. 1, pp. 105-110, 1992.
[22] A. Singhal, "Issues in autonomous mobile robot navigation," Master's thesis, Dept. of Computer Science, University of Rochester, 1997.
[23] B. Williams and P. Kim, "Model-based reactive programming of cooperative vehicles for mars exploration," *Proc. of the International Symposium on Artificial Intelligence*, *Robotics and Automation in Space*, 2001.
[24] Q. Zhu, "Hidden Markov model for dynamic obstacle avoidance of mobile robot navigation," *IEEE Trans on Robotics and Automation*, vol. 7, no. 3, pp. 390-396, 1991.

Raj Madhavan received the Bachelor of Engineering degree (Electrical and Electronics) from the College of Engineering, Anna University, India and the Master of Engineering (Research) degree from the Department of Engineering, The Australian National University, Australia. He received his Ph.D. degree (Field Robotics) from the School of Aerospace, Mechanical and Mechatronic Engineering, The University of Sydney, Australia. Since 2001, he has been a research associate with the Oak Ridge National Laboratory (ORNL) and is currently a guest researcher with the Intelligent Systems Division of the National Institute of Standards and Technology (NIST). Dr. Madhavan is listed in Madison Who's Who, Strathmore Who's Who, The Contemporary Who's Who of Professionals, America's Registry of Outstanding Professionals, and United Who's Who. He is also an elected full member of the scientific research society, Sigma Xi and the Washington Academy of Sciences and a member of IEEE, AAAI, AUVSI, and AIME. He was the Postgraduate winner of the 1998 IEEE Region 10 (Asia and the Pacific) Student Paper Competition and the Graduate Division winner of the 1998 Society for Mining, Metallurgy and Exploration (SME) Outstanding Student Paper Contest. His current research interests include autonomous vehicle navigation in complex unstructured environments, performance metrics for intelligent systems, distributed heterogeneous sensing, systems and control theory. Dr. Madhavan has published more than 75 papers in referred journals, book chapters, conference publications, technical reports, and is currently editing a book "Intelligent Vehicle Systems: A 4D/RCS Approach" (with E. Messina and J. Albus).

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 committees of numerous knowledge representation and ontology related conferences and workshops, and has published over 60 papers in related areas.