Multiple Model Nonlinear Filtering for Low Signal Ground Target Applications

KEITH KASTELLA
CHRIS KREUCHER
General Dynamics

The design and implementation of a multiple model nonlinear filter (MMNLF) for ground target tracking using ground moving target indicator (GMTI) radar measurements is described. Like the well-known Interacting multiple model Kalman filter (IMMKF), the MMNLF is based on the theory of hybrid stochastic systems. However, since it models the probability distribution for the target in a region, rather than just the distribution's first and second moments, a nonlinear filter is able to capture more fine-grained detail of the target motion and requires fewer models than typical IMMKF implementations. This is illustrated here with a two-model MMNLF in which one motion model incorporates terrain constraints while the second is a nearly constant velocity (CV) model. Another feature of the MMNLF is that it enables incorporation of prethresholded measurements. To implement the filter, the target state conditional probability density is discretized on a set of moving grids and recursively updated with sensor measurements via Bayes' formula. The conditional density is time updated between sensor measurements using alternating direction implicit (ADI) finite difference methods, generalized for this hybrid application. In simulation testing against low signal-to-interference-plus-noise ratio (SINR) targets, the MMNLF is able to maintain track in situations where single model filters based on either of the component models or filters that use thresholded data fail. Potential applications of this work include detection and tracking of foliage-obscured moving targets.

Manuscript received February 8, 2004; revised August 23, 2004; released for publication January 12, 2005.

IEEE Log No. T-AES/41/2/849018.

Refereeing of this contribution was handled by C. Jauffret.

Authors' address: General Dynamics, PO Box 990, Ypsilanti, MI 48197, E-mail: (keith.kastella@gd-ais.com, christopher.kreucher@gd-ais.com).

I. INTRODUCTION

There is growing interest in ground target tracking, stimulated in large part by the success of the joint surveillance target attack radar system (JSTARS) [1] and similar ground moving target indicator (GMTI) radar systems [2]. Ground target tracking differs from airborne and maritime tracking in several important respects. First, since ground targets often have small scattering cross section, radar returns tend to have lower signal-to-interference-plus-noise ratio (SINR). Ground targets can be slow moving so that they are difficult to separate from the clutter background, especially when the GMTI sensor is airborne or space based. These problems are exacerbated by the long standoff range required for operations in hostile territory. Often ground targets are obscured by foliage, which requires radar systems that operate at low frequency, further degrading both range and cross range accuracy. Finally, the low SINR often requires long dwell times so that the target revisit rate is low (on the order of tens of seconds to minutes), which also makes tracking more difficult.

While the signal environment for ground targets is challenging, roads and terrain often provide motion constraints that are more rigid than those found in airborne and maritime tracking applications. These constraints constitute an additional information source that can be exploited to improve tracking performance if the effects on the target motion can be included in the motion models. For off-road targets, significant information can be gleaned, for instance, from the restriction of wheeled vehicles to certain types of soils or restrictions of vehicles from extreme slope values.

There has been significant work on incorporating terrain and road information into tracking filters that use variable structure interacting multiple model Kalman filters (VS-IMMKF) [7] and road constraints implemented as Gaussian sums [18]. In [7], a road network is modeled as a collection of one dimensional segments and vehicles are modeled with both on-road and off-road behaviors. The on-road vehicle motion is modeled using nearly constant velocity (CV) model spatially-varying nonisotropic plant noise. In this approach, when a vehicle's estimated position is close to a road segment, its on-road is constructed with more plant noise parallel to the road segment than perpendicular to it. However, in this approach there is no direct incorporation of dynamic vehicle inputs for preferred heading or speed and there is no "restoring force" in the model to constrain the vehicle to remain on the road segment. In [18], the influence of roads is incorporated as a pseudomeasurement each time the filter is measurement updated. This approach, similar to that employed by [19] to model land avoidance in maritime targets, is an effective practical solution but is somewhat unsatisfactory from a theoretical standpoint since the road information is

only incorporated at measurement update. This results in an unrealistic coupling between purely kinematic constraints and the measurement process.

Part of our motivation in exploring multiple model nonlinear filtering (MMNLF) is to include terrain information directly into the kinematics of the model. Our approach addresses this issue by incorporating the effect of terrain in a fine-grained kinematics model with explicit restoring terms. For example, when driving on a road, drivers provide steering and acceleration inputs to align the vehicle axis with the road and maintain a preferred speed. The use of a nonlinear filter allows us to incorporate this behavior directly into the filter model.

A number of US Army programs have worked extensively to characterize vehicle motion preferences in terms of a quantity referred to as "hospitability for maneuver" (HM) [10–12]. These preferences vary with position, weather, vehicle type (e.g. tracked versus nontracked), and mission. HM characterizes the ease with which a vehicle can traverse a particular area, and can be used to generate the preferred heading and velocity with which a vehicle will operate in an area. HM data is quite fine grained and can vary by an order of magnitude over spatial scales on the order of tens of meters, say the width of a road. This makes it difficult to capture using linearized techniques such as the extended Kalman filter, motivating, in part, our investigation here into nonlinear methods.

While HM is a potentially useful source of information to incorporate into an MMNLF structure, it has primarily been developed as a qualitative tool to aid human analysts. Another possible source of information is historic data collected by sensors observing an area prior to an engagement. We emulate that here by using recorded National Training Center (NTC) vehicle motion data. Another useful feature of NLF that we explore here is that it can be used to process nonthresholded data (i.e., envelope detected sensor pixel amplitudes) in a track-before-detect approach [5, 6, 19, 27, 28] that does not suffer from any SINR loss due to thresholding. This is likely to yield the most gain when tracking vehicles that are moving slowly or beneath a forest canopy where the SINR is particularly small. In such applications, nonlinear filter (NLF) methods [3, 14, 15, 16] may be particularly useful since the potential gain from improved target motion modeling increases as the SINR decreases.

With this motivating introduction, the primary focus of the work reported here is the development of an MMNLF combined with thresholded or nonthresholded data. We use an alternating direction implicit (ADI) finite difference method here to solve the nonlinear filtering equations. The details of the numerics are presented as an appendix, as the main thrust of this paper is the development of an MMNLF that incorporates HM and comparison with single model filters in the ground target tracking scenario. Particle filtering is an alternative approach to realizing the NLF which has been successfully applied in similar circumstances [24, 25, 26, 28]. See [23] for a nice survey of particle filtering methods, approaches, and results. Several researchers have compared the performance between NLFs (e.g. grid-based and particle-based) to standard Kalman filter approaches [32, 33, 34].

The paper is organized as follows. The mathematical formulation for the MMNLF is given in Section II. It includes a derivation of the two models combined in the MMNLF and the measurement models used for thresholded and nonthresholded data. Section III outlines how HM data are derived and how we use it to estimate vehicle motion parameters for the MMNLF. Vehicle motion model parameters can also be extracted directly from observed vehicle motion data. Section IV presents tracking results obtained using collected vehicle motion and simulated measurements. Summary and conclusions are presented in Section V. Appendix A presents numerical details of the method that we use to solve the multiple model Fokker-Planck equation (MMFPE) using an ADI schemes [8, 9, 13] that has been modified to accommodate multiple motion models. One nice feature of this ADI approach pointed out by [29] and [30] is that resulting finite difference scheme is equivalent to a Markov chain on a discrete state space. The resulting matrix operators are stochastic, conserving the total probability for the discrete system and improving its numerical stability.

II. MULTIPLE MODEL NONLINEAR FILTERING MATHEMATICAL FRAMEWORK

The objective of filtering is to estimate some kinematic state $\mathbf{x}_t$ (say position and velocity) of the target at time t, given a sequence of measurements $\mathbf{y}_k$ made at discrete times t_k (vectors and matrices are indicated by bold characters, scalars are italic). Like the IMMKF [20], MMNLF is based on a hybrid estimation approach. At each time t the target is in one of a finite number of kinematic models (or "modes") $m_t \in \{1,\ldots,r\}$. The target dynamics for each of the r models are described by the Ito stochastic differential equation

$$d\mathbf{x}_t = \mathbf{f}_{m_t}(\mathbf{x}_t)dt + \mathbf{G}_{m_t}(\mathbf{x}_t)d\beta_t, \qquad t \geq t_0 \tag{1}$$

where $\mathbf{x}_t$ and $\mathbf{f}_{m_t}$ are column n-vectors, $\mathbf{G}_{m_t}$ is an $n \times p$ matrix function and $\{\beta_t, t \geq t_0\}$ is a p-vector Brownian motion process with $E\{d\beta_t d\beta_t^T\} = \mathbf{Q}(t)dt$. Although not strictly necessary, we assume for simplicity that all models have the same n-dimensional state vector $\mathbf{x}_t$. The functions $\mathbf{f}_{m_t}$ and $\mathbf{G}_{m_t}$ can also contain explicit time dependence, but here we assume that they only depend on the kinematic state and the mode.

The mode evolves in time as a homogenous continuous-time Markov chain [21]. This means that we may define the probability

$$P\{m_t = m\} \tag{2}$$

and transition probabilities

$$\pi_{ij}(\tau) = P\{m_{i+\tau} = i \mid m_t = j\}. \tag{3}$$

(The fact that we restrict ourselves to homogenous chains makes this a function of τ only.) Transition rates are

$$\Lambda_{ij} = \left.\frac{\partial}{\partial \tau}\right|_{t\downarrow 0} \pi_{ij}(\tau). \tag{4}$$

The collection of measurements up to and including time t_k is denoted

$$Y_{t_k} = \{\mathbf{y}_t : t \le t_k\}. \tag{5}$$

The known measurement density $p(\mathbf{y} \mid \mathbf{x})$ is both time and mode independent.

To estimate the kinematic state of the target, the MMNLF recursively computes the hybrid conditional density that the target is in mode m and state $\mathbf{x}$ at time t, given some set of measurements $Y_{t'}$, $t' \le t$, which we denote $p_m(\mathbf{x}_t \mid Y_{t'})$. Each of the conditional probability density functions $p_m(\mathbf{x}_t \mid Y_t)$ is constructed recursively according to Bayes' formula:

$$p_m(\mathbf{x}_{t_k} \mid Y_{t_k}) = \frac{p(\mathbf{y}_{t_k} \mid \mathbf{x}_{t_k}) p_m(\mathbf{x}_{t_k} \mid Y_{t_{k-1}})}{\int d\mathbf{x}'_{t_k} \sum_{m'} p(\mathbf{y}_{t_k} \mid \mathbf{x}'_{t_k}) p_{m'}(\mathbf{x}'_{t_k} \mid Y_{t_{k-1}})}. \tag{6}$$

Equation (6) requires that we be able to evaluate the density for the state at time k given measurements only up to time $k-1$. Therefore, we must characterize the time evolution of the density during the time interval $t_k \le t \le t_{k+1}$, which is given by the MMFPE. Defining $\mathbf{p} = (p_1, \ldots, p_r)^T$, the MMFPE is

$$\frac{\partial \mathbf{p}}{\partial t}(\mathbf{x}_t \mid Y_{t_k}) = \mathbf{L}\mathbf{p}(\mathbf{x}_t \mid Y_{t_k}) + \mathbf{\Lambda}\mathbf{p}(\mathbf{x}_t \mid Y_{t_k}) \tag{7}$$

with initial condition $\mathbf{p}(\mathbf{x}_{t_k} \mid Y_{t_k})$ where

$$L_m p_m \equiv -\sum_{i=1}^{n} \frac{\partial(\mathbf{f}_{m,i} p_m)}{\partial \mathbf{x}_i} + \frac{1}{2} \sum_{i,j=1}^{n} \frac{\partial^2 (p_m (\mathbf{G}_m \mathbf{Q} \mathbf{G}_m^T)_{ij})}{\partial \mathbf{x}_i \partial \mathbf{x}_j} \tag{8}$$

$$\mathbf{L} = \begin{pmatrix} L_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & L_r \end{pmatrix} \tag{9}$$

$$\mathbf{\Lambda} = (\Lambda_{ij}). \tag{10}$$

(Equation (7) is solved numerically in the appendix.)

Given $p_m(\mathbf{x}_t \mid Y_{t_k})$, the marginal density for the target state is

$$p(\mathbf{x}_t \mid Y_{t_k}) = \sum_m p_m(\mathbf{x}_t \mid Y_{t_k}) \tag{11}$$

while the marginal for model m at time t is

$$p(m_t \mid Y_{t_k}) = \int p_m(\mathbf{x}_t \mid Y_{t_k}) d\mathbf{x}_t \tag{12}$$

The minimum mean square error state estimate $\hat{\mathbf{x}}_{t_k}$ is

$$\hat{\mathbf{x}}_{t_k} = \int d\mathbf{x}_{t_k} \mathbf{x}_{t_k} p(\mathbf{x}_{t_k} \mid Y_{t_k}) \tag{13}$$

with covariance

$$P_t = \int d\mathbf{x}_{t_k} (\hat{\mathbf{x}}_{t_k} - \mathbf{x}_{t_k})(\hat{\mathbf{x}}_{t_k} - \mathbf{x}_{t_k})^T p(\mathbf{x}_{t_k} \mid Y_{t_k}). \tag{14}$$

A. Target Motion Model with Vehicle Motion Preferences

For tracking ground targets, we use the 4-dimensional target state

$$\mathbf{x} = (x, y, \theta, v)^T \tag{15}$$

where x and y are the target's Cartesian location in the topocentric plane (meters) centered on the region of interest, θ is the target heading, and v is the target speed (m/s).

The novel feature that we introduce here is the incorporation of spatially varying preferred heading and speed. This models the fact that a driver will attempt to align the axis of his vehicle with that of the road and, depending on the terrain and road quality, there will be a preferred speed. When the vehicle deviates from these preferred values, the driver applies corrections and the vehicle responds. The time that it takes for the drive/vehicle system is characterized by a time constant, which may be spatially-varying.

Ito equations that couple the target dynamics to the spatially-varying vehicle motion preferences can be constructed using inhomogenous integrated Ornstein-Uhlenbeck (IIOU) models as follows. Given the position-dependant preferred velocity $v_0(x,y)$, and a mean time to speed corrections $\tau_v(x,y)$, the Ito equation for the target speed

$$dv = -\frac{1}{\tau_v(x,y)}(v - v_0(x,y))dt + \sqrt{q_v(x,y)}d\beta_v \tag{16}$$

where β_v is a scalar white Brownian motion process with power spectral density

$$E(d\beta_v^2) = dt \tag{17}$$

and

$$q_v(x,y) = \frac{2}{\tau_v(x,y)}\sigma_v^2 \tag{18}$$

with σ_v^2 is the variance of the speed deviation from its preferred value. $\tau_v(x,y)$ is sometimes referred to as the sojourn time. Similarly, given a spatially-varying

preferred heading $\theta(x,y)$ and a mean time to heading corrections $\tau_\theta(x,y)$, the Ito equation is

$$d\theta = -\frac{1}{\tau_\theta(x,y)}(\theta - \theta_0(x,y,\theta))dt + \sqrt{q_\theta(x,y)}d\beta_\theta. \tag{19}$$

We construct the model so that targets align themselves parallel or antiparallel to the preferred heading. This is accomplished by evaluating the heading modulo 180 deg. The preferred heading $\theta_0(x,y,\theta)$ is defined as

$$\theta_0(x,y,\theta) = \begin{cases} \varphi(x,y) & |\theta - \varphi(x,y)| < \pi/2 \\ \varphi(x,y) + \pi & \text{otherwise} \end{cases} \tag{20}$$

where $\varphi(x,y)$ is the data- or hospitability-derived preferred axis of motion. The spatially-varying function $\varphi(x,y)$ defines the preferred axis of travel at each point on the terrain. The heading process is

$$E(d\beta_\theta^2) = dt \tag{21}$$

and

$$q_\theta(x,y) = \frac{2}{\tau_\theta(x,y)}\sigma_\theta^2. \tag{22}$$

Defining $dx = v\cos(\theta)dt$ and $dy = v\sin(\theta)dt$, the Ito equations for all of the components are

$$\begin{aligned} dx &= v\cos(\theta)dt \\ dy &= v\sin(\theta)dt \\ d\theta &= -\frac{1}{\tau_\theta(x,y)}(\theta - \theta_0(x,y,\theta))dt + d\beta_\theta \\ dv &= -\frac{1}{\tau_v(x,y)}(v - v_0(x,y))dt + d\beta_v. \end{aligned} \tag{23}$$

Defining $\dot{x} = v\cos(\theta)$ and $\dot{y} = v\sin(\theta)$, the operator ((8)) for this model (hereafter called the IIOU model) is

$$\begin{aligned} L_{\mathrm{IIOU}}(p) = &-\dot{x}\frac{\partial p}{\partial x} - \dot{y}\frac{\partial p}{\partial y} \\ &+ \frac{1}{\tau_\theta(x,y)}\frac{\partial}{\partial\theta}((\theta - \theta_0(x,y,\theta))p) \\ &+ \frac{1}{\tau_v(x,y)}\frac{\partial}{\partial v}((v - v_0(x,y))p) \\ &+ \frac{1}{2}q_\theta(x,y)\frac{\partial^2 p}{\partial\theta^2} + \frac{1}{2}q_v(x,y)\frac{\partial^2 p}{\partial v^2}. \end{aligned} \tag{24}$$

This equation contains three separate contributors. The velocity advection terms are the one-way wave equations in x and y:

$$-\dot{x}\frac{\partial p}{\partial x} - \dot{y}\frac{\partial p}{\partial y}. \tag{25}$$

The heading and speed advection term enforces the constraint that the vehicle tends to move with velocity $v_0(x,y)$ and heading $\theta_0(x,y)$ (modulo π):

$$\frac{1}{\tau_\theta(x,y)}\frac{\partial}{\partial\theta}((\theta - \theta_0(x,y,\theta))p) + \frac{1}{\tau_v(x,y)}\frac{\partial}{\partial v}((v - v_0(x,y))p). \tag{26}$$

The heading and speed diffusion term enforces the spatially-varying tendency to diffuse away from preferred heading and velocity:

$$\frac{1}{2}q_\theta(x,y)\frac{\partial^2 p}{\partial\theta^2} + \frac{1}{2}q_v(x,y)\frac{\partial^2 p}{\partial v^2}. \tag{27}$$

The parameters of the model $(\theta_0, v_0, q_v, q_\theta, \tau_\theta, \tau_v)$ can all vary with position and vehicle type. The method of extracting them from HM data or recorded vehicle motion data is presented in Section III.

B. Target Motion Model that does not use Vehicle Motion Preferences

There are several instances in which using vehicle motion preferences in the NLF is not desirable. First, there are times/locations when HM or previous observations are not available. More often, there are locations where vehicle tracks conflict, indicating that there are either no paths through an area or multiple paths. In these instances, a constant velocity/constant heading model is preferred. In this case, we use the Ito equations

$$\begin{aligned} dx &= v\cos(\theta)dt \\ dy &= v\sin(\theta)dt \\ d\theta &= d\beta_\theta \\ dv &= d\beta_v. \end{aligned} \tag{28}$$

Defining $\dot{x} = v\cos(\theta)$ and $\dot{y} = v\sin(\theta)$ as before, the FPE for this model (which we refer to hereafter as the CV model) is

$$\begin{aligned} L_{\mathrm{CV}}(p) = &-\dot{x}\frac{\partial p}{\partial x} - \dot{y}\frac{\partial p}{\partial y} + \frac{1}{2}q_\theta(x,y)\frac{\partial^2 p}{\partial\theta^2} \\ &+ \frac{1}{2}q_v(x,y)\frac{\partial^2 p}{\partial v^2}. \end{aligned} \tag{29}$$

C. Measurement Models

We model the target measurements as square-law detected return amplitude on a uniform grid of size $M = N_x \times N_y$ (for simplicity, we ignore the usual Doppler estimate obtained as part of the GMTI measurement). The amplitude in pixel i for scan k is $y_{k,i}$ and the entire scan is $\mathbf{y}_k = \{y_{k,i} \mid i = 1,\ldots,M\}$. We assume that measurements in the different detection cells are independent, and that the measurements are uncorrelated over time. Let $i_{\mathbf{x}}$ denote the target-containing pixel, and let λ denote the SINR (here assumed known). For nonthresholded

measurements, we assume the probability distribution for the amplitude in pixel $i_{\mathbf{x}}$ is

$$p_1(y_{i_{\mathbf{x}}}) = \frac{1}{1+\lambda}\exp(-y_{i_{\mathbf{x}}}/(1+\lambda)). \tag{30}$$

The distribution in the empty cells is

$$p_0(y_{i_{\mathbf{x}}}) = \exp(-y_{i_{\mathbf{x}}}). \tag{31}$$

Now assume that state $\mathbf{x}_k$ maps into pixel i_x, then

$$p(\mathbf{y}_k \mid \mathbf{x}_k) = \prod_{i=1}^{M} p(y_{k,i} \mid \mathbf{x}_k) = p(y_{k,i_x} \mid \mathbf{x}_k) \prod_{\substack{i=1 \\ i \neq i_x}}^{M} p(y_{k,i} \mid \mathbf{x}_k) \tag{32}$$

which, via algebraic manipulation, can be further simplified to

$$p(\mathbf{y}_k \mid \mathbf{x}_k) = p_1(y_{k,i_x}) \prod_{\substack{i=1 \\ i \neq i_x}}^{M} p_0(y_{k,i}) = \frac{p_1(y_{k,i_x})}{p_0(y_{k,i_x})} \prod_{i=1}^{M} p_0(y_{k,i})$$

$$= \kappa \frac{p_1(y_{k,i_x})}{p_0(y_{k,i_x})} \tag{33}$$

where κ is a target-state independent constant that can be discarded in the Bayes' formula update.

We have chosen to use a Gaussian clutter plus noise model for simplicity and generality. The nonlinear filtering paradigm is sufficiently general that other models (e.g. Rayleigh clutter, or measurements that are correlated in space and/or time) could by used by adjusting the form of $p(\mathbf{y}_k \mid \mathbf{x})$. In our experiments of Section IV, sensor measurements are simulated according to the model $p(\mathbf{y}_k \mid \mathbf{x})$ given here. A primary benefit of nonlinear filtering is that prethresholded measurements such as those described by (30)–(33) can be directly incorporated into the filter. Note, however that the use of prethresholded data precludes the use of filters that approximate measurements as a nonlinear transformation of the target state with Gaussian additive noise such as the extended Kalman filter (EKF).

For thresholded measurements with a probability of detection P_{d} (again assumed known), we use the relationship

$$P_{\mathrm{fa}} = P_{\mathrm{d}}^{1+\lambda} \tag{34}$$

to define the false alarm probability P_{fa}. The probability distribution for the target containing cell becomes discrete (either 0 for no measurement or 1 for a measurement) and is written

$$p_1(y_{i_{\mathbf{x}}}) = \begin{cases} P_{\mathrm{d}} & y_{i_{\mathbf{x}}} = 1 \\ 1 - P_{\mathrm{d}} & y_{i_{\mathbf{x}}} = 0 \end{cases}. \tag{35}$$

The distribution on the nontarget containing cells is then

$$p_0(y_{i_{\mathbf{x}}}) = \begin{cases} P_{\mathrm{fa}} & y_{i_{\mathbf{x}}} = 1 \\ 1 - P_{\mathrm{fa}} & y_{i_{\mathbf{x}}} = 0 \end{cases}. \tag{36}$$

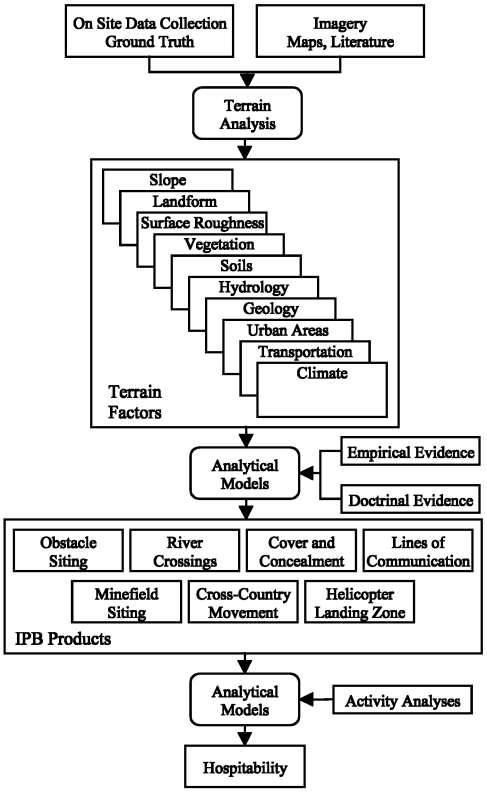


Fig. 1. Data flow in classic terrain analysis procedure.

Defining

$$\delta_i = \begin{cases} 1 & i = i_{\mathbf{x}} \\ 0 & i \neq i_{\mathbf{x}} \end{cases} \tag{37}$$

we write the density for the entire scan using the same algebraic manipulation as in the prethresholded case as

$$p(\mathbf{y}_k \mid \mathbf{x}_k) = \delta_1 p_1(y_{k,i_{\mathbf{x}}}) + (1 - \delta_i) p_0(y_{k,i_{\mathbf{x}}}). \tag{38}$$

III. EXTRACTING MOTION MODEL PARAMETERS

The motion model parameters defining the MMNLF can be based on HM data or extracted directly from observed vehicle motion data. As part of the preparation of battlefield intelligence, terrain analysis is performed to predict a number of effects that HM and similar quantities such as hospitability for emplacement can have on battle outcomes. In maneuver planning, existing lines of communication may be inappropriate, requiring cross-country movement.

Fig. 1 details the data flow for terrain analysis with the addition of an activity product. Terrain analysis begins with the raw terrain data extracted from imagery, maps, literature, or data collected during on-site inspections. These data are analyzed and reduced to a set of terrain factor products capturing important terrain features and classifications [10]. This portion of the analysis is usually automated and is described in more detail in publicly available

USGS and NIMA products. The next step in the terrain analysis procedure is to combine terrain factor products with empirical and doctrinal evidence to generate complex products capturing activities of military significance [11]. For example, cross-country movement incorporates terrain factors such as slope, landform, surface roughness, vegetation, and soils. These factors are combined with analytical models [12] which capture the capabilities of specific vehicles (usually empirically derived), to produce a maximum vehicle speed in a given area. The generation of activity-specific products is an advance over classical terrain analysis and intelligence preparation procedures. These products capture interactions between activity requirements such as a need for long-term hide site and constraints or opportunities provided by local terrain. For maneuver like activities, hospitability provides a measure of the support a given local area provides for a given activity. Hospitability has been successfully utilized in both DARPA and NIMA imagery exploitation systems, and has had particular success when applied to the identification of mobile missiles.

Tuning of the analytical models within this process is accomplished primarily in two ways: application of physical constraints, and the analysis of vehicle tracks. For example, the physical limitations of the M1A1 tank (e.g., maximum traversable slope, surface roughness versus speed effects) are directly incorporated into the analytical models. Vehicle track analysis provides exactly the terrain over which certain vehicles travel. While not excluding any terrain combination, it can be analyzed to provide preferences and as a test of the physical vehicle constraint derived model parameters.

1) *Deriving Vehicle Preferences from HM*: HM maps qualitatively represent the ease with which a vehicle can traverse a particular area. HM can be used as a spatially-varying measure of preferred target speed $v_0(x,y)$. One such scaling is shown in Fig. 2. In our experiments, we found that the best way to generate velocity preferences from HM is to derive a go/no-go map. All hospitability measures greater than a certain threshold imply that the vehicle moves quickly, and all velocities below the threshold imply the vehicles move slowly.

To develop a model of heading dynamics, we assume that targets prefer to follow regions of high HM. This corresponds to following the hospitability ridges seen in Fig. 3. The 180 deg ambiguity in θ is incorporated by computing angles modulo 180. Using HM, the time constants $\tau_v(x,y)$ and $\tau_\theta(x,y)$ and the diffusive components $q_v(x,y)$ and $q_\theta(x,y)$ will be chosen to be fixed values. Their specific values will be chosen based on problem domain knowledge or will be derived from a secondary source (such as vehicle observations, as outlined in the following section).

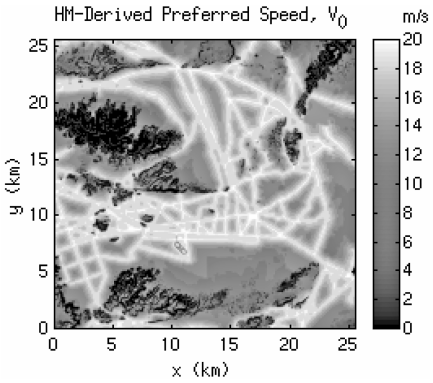


Fig. 2. Preferred velocities derived from HM map.

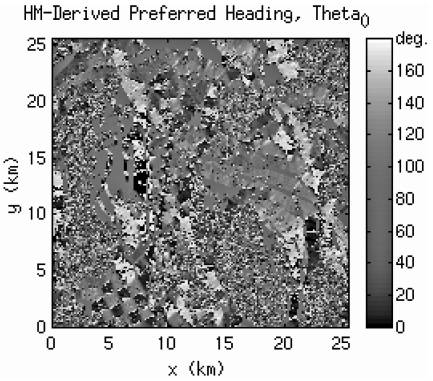


Fig. 3. Preferred headings derived from HM map.

2) *Deriving Vehicle Preferences from Observed Data*: More detailed IIOU parameters can be obtained by directly observing target motion in an area. In real applications, this could be derived from sensor observations of an area over time. Here, we emulate this by using truth data from GPS vehicle positions measurements that have been collected of the over time as part of battle training at NTC. This data can be used to directly estimate the IIOU parameters for a region of interest. This has the advantage that motion model parameters can be quantitatively inferred. The disadvantage of this approach is that, in wartime, vehicles will often need to be tracked in areas where they have not previously been observed. In such cases, the methods of HM terrain analysis described above will likely be more useful. To begin, the discrete time Ito equation for vehicle speed corresponding to (16) is

$$v^{k+1} = v^k - \frac{\Delta t}{\tau_v(x,t)}[v^k - v_0(x,y)] + \sqrt{q_v(x,y)\Delta t}\,w^{k+1} \tag{39}$$

where v^k is the speed at time k, $v_0(x,y)$ is the preferred speed at x,y, $\tau_v(x,y)$ is the time constant and $q_v(x,y)$ is the magnitude of the power spectral density and w^{k+1} is unit variance zero mean Gaussian noise. Note that we allow τ_v and q_v to vary with position here.

We estimate parameters $v_0(x,y)$, $\tau_0(x,y)$, and $q_v(x,y)$ using movement data consisting of vehicle positions (x_k,y_k) at times t_k for a collection of vehicles on training maneuvers at NTC as follows. For each vehicle, the speed at a point (x_k,y_k) is computed using forward differencing

$$v^k = \frac{\sqrt{(x^{k+1}-x^k)^2+(y^{k+1}-y^k)^2}}{(t^{k+1}-t^k)}. \tag{40}$$

The speed v^k is associated with the point (x_k,y_k) and the time t_k. The preferred vehicle speed at a point $v_0(x,y)$ can be estimated using a simple averaging technique:

$$\hat{v}_0(x_k,y_k) = \frac{v^{k-1}+v^k+v^{k+1}}{3}. \tag{41}$$

To obtain the time constant $\tau_v(x,y)$ define the residual difference between the true speed and the preferred speed as $\tilde{v}^k = v^k - v_0$. Then the residual obeys the Ito equation for a homogenous Ornstein-Uhlenbeck process,

$$\tilde{v}^{k+1} = \tilde{v}^k - \frac{\Delta t}{\tau_v(x,y)}\tilde{v}^k + \sqrt{q_v(x,y)\Delta t}\,w^{k+1}. \tag{42}$$

Evaluating expected values we find,

$$\langle \tilde{v}^{k+1}\tilde{v}^k\rangle = \langle \tilde{v}^k\tilde{v}^k\rangle - \left\langle \frac{\Delta t}{\tau_v(x,y)}\tilde{v}^k\tilde{v}^k \right\rangle + \left\langle \sqrt{q_v(x,y)\Delta t}\,w^{k+1}\tilde{v}^k \right\rangle \tag{43}$$

$$\langle \tilde{v}^{k+1}\tilde{v}^k\rangle = \left[1-\frac{\Delta t}{\tau_v(x,y)}\right]\langle \tilde{v}^k\tilde{v}^k\rangle \tag{44}$$

where $\langle\cdot\rangle$ denotes expectation over the ensemble. Since $\tilde{v}^k$ and w^{k+1} are uncorrelated and have mean 0, we may solve for $\tau_v(x,y)$ to obtain

$$\tau_v(x,y) = 2\Delta t\langle \tilde{v}^k\tilde{v}^k\rangle / \langle(\tilde{v}^{k+1}-\tilde{v}^k)^2\rangle. \tag{45}$$

This is approximated using a three-term sum

$$\hat{\tau}_v(x,y) \approx \frac{2\Delta t}{3}\frac{\sum_{i=k-1}^{i=k+1}(\tilde{v}^i)^2}{\sum_{i=k-1}^{i=k+1}(\tilde{v}^{i+1}-\tilde{v}^i)^2}. \tag{46}$$

Finally, to estimate $q_v(x,y)$ we note that (39) implies

$$q_v(x,y)(w^{k+1})^2 = \frac{1}{\Delta t}\left\{\left[\tilde{v}^{k+1}-\left(1-\frac{\Delta t}{\tau_v(x,y)}\right)\tilde{v}^k\right]\right\}^2. \tag{47}$$

Then, since w^{k+1} has unit variance,

$$q_v(x,y) = \frac{1}{\Delta t}\left\langle\left[\tilde{v}^{k+1}-\left(1-\frac{\Delta t}{\tau_v(x,y)}\right)\tilde{v}^k\right]^2\right\rangle \tag{48}$$

and

$$\hat{q}_v(x,y) = \frac{1}{3\Delta t}\sum_{i=k-1}^{i=k+1}\left[\tilde{v}^{i+1}-\left(1-\frac{1}{\tau_v(x,y)}\right)\tilde{v}^i\right]^2. \tag{49}$$

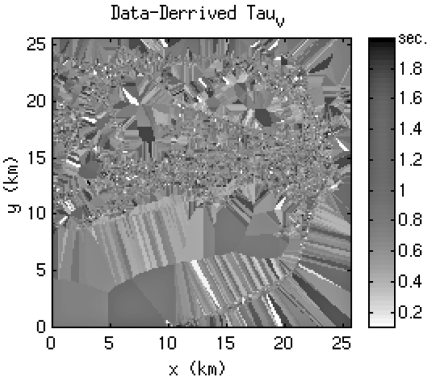


Fig. 4. Time to course corrections in velocity derived from vehicle observations.

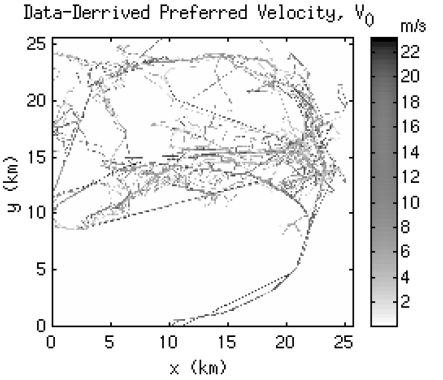


Fig. 5. Preferred velocities derived from vehicle observations.

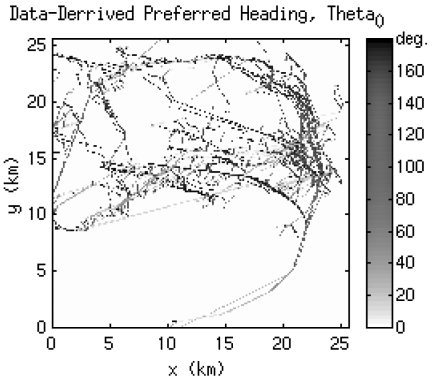


Fig. 6. Preferred headings derived from vehicle observations.

Similar reasoning can be applied to estimate the heading parameters $\theta_0(x,y)$, $\tau_\theta(x,y)$, and $q_\theta(x,y)$.

Using this methodology and a set of 100 vehicles observed in battlefield simulations, we can generate preferred headings and velocities on a grid as shown in Fig. 4–7.

IV. RESULTS

A set of data collected at the NTC was used to test the trackers. The data consisted of 616 vehicles performing battle simulations over a period of several weeks. Each of these vehicles had its position and

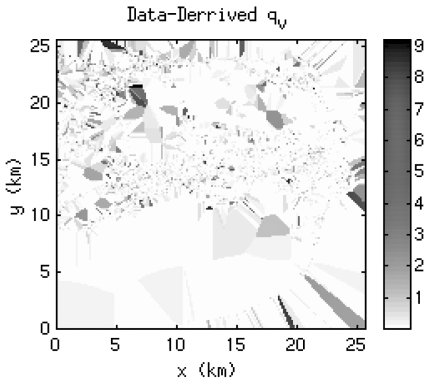


Fig. 7. Diffusive component in velocity derived from vehicle observations.

velocity monitored by GPS. The position and velocity estimates were interpolated to 1 s intervals.

Approximately 100 vehicles moving in a 25 kilometer × 25 kilometer area were extracted and used to create spatially-varying preferred vehicle headings and velocities. Fixed (not spatially-dependent) variances and time constants were calculated as averages of the 100 training vehicles

$$\tau_\theta = 1 \text{ s}, \qquad \tau_v = 1 \text{ s}, \qquad q_v = 1 \text{ m/s}, \qquad q_\theta = 3^\circ. \tag{50}$$

Five vehicles, identified as vehicles "3," "143," "355," "416," and "1432" were used to compare three different filtering models: a single-model filter using the CV model (the "CV filter"), a single-model filter using the IIOU model (the "IIOU filter"), and a multiple model filter that has both the CV and IIOU models (the "MMNLF"). These five vehicles were omitted from the training (generation of preferred motion parameters). Each of the test vehicles were chosen to illustrate a different scenario. Each vehicle was tracked for 300 time steps using each of the filters. A Monte Carlo test consisting of 100 different random measurement sequences was used to measure the relative performance of the filters. Further, the SINR was varied from 1 dB to 16 dB.

Measurements were simulated according to the model of Section IIC. The filters know the measurement model and its parameters (e.g. SINR) precisely. One interesting area for future study is the effect of model mismatch, which may often be the case in real world scenarios (i.e., where measurements are generated according to one model but the filter uses another model). In our application, we expect that once entering a model m (either the IIOU or the CV model), the target will continue to obey that model with high probability. Furthermore, transition probabilities, while small, will tend to favor the target entering the IIOU model. We use

$$\Lambda = \begin{bmatrix} -0.1 & 0.2 \\ 0.1 & -0.2 \end{bmatrix}. \tag{51}$$

The numerical FPE solver of Appendix A was implemented to solve (7). The grid size was $N_x = N_y = 20$ and $N_v = N_\theta = 6$ so the total number of grid cells in this implementation was $N = 14400$. The resolution of the grid was $\Delta x = \Delta y = 15$ m, $\Delta\theta = 30^\circ$, and $\Delta v = 3\frac{1}{3}$ m/s. Note that more finely spaced grid cells may lead to better numerical performance. However, this requirement for more grid cells would adversely effect the computation speed of the algorithm. Nonthresholded measurements are generated on a 1 s interval, which is the (interpolated) rate of reports given in the data. The spatial resolution of the sensor (i.e., the pixel size) was 15 m × 15 m (no Doppler measurement was assumed). At every time step, the grid is translated to remain centered on the estimated target location, heading, and speed, as discussed in the appendix. This translation results in some of the probability mass being lost from the grid. This is mitigated by the fact that the grid is large enough so that the probability mass at the edges is very low. The grid is renormalized to sum to 1 after each translation. A more sophisticated treatment of the issue of domain truncation is given in [22]. At initialization, the grid is centered on the true state of the target (position, velocity, and heading) and initialized to a uniform density over the extent of the grid.

Performance of the filters was measured in four ways: the position, velocity and heading RMS error, and the ability of the filter to keep the target in track. A target is deemed to be "lost" when the true position is off of the grid; this corresponds to the estimated target position being more than 150 m in error in both x and y components. We detail here the Monte Carlo testing results of each of the five test vehicles (see Table I).

Vehicle 3 moves with nearly constant heading and experiences 3 notable velocity changes, all of which are in accordance with motion preferences derived from the training vehicles (see Fig. 8, left). The heading matches the motion preferences very well. For these reasons, we expect the IIOU filter to outperform the CV filter. As shown in Fig. 8, all three filters are able to track the vehicle well at SINR above 5 dB. The IIOU filter and MMNLF are able to maintain good tracking at 4 dB and modest performance even at 1 dB. The position error of the IIOU filter and the MMNLF are nearly identical, while the CV filter exhibits higher error, especially at low SINR. The MMNLF and the IIOU filter are able to take advantage of the vehicle motion preferences, which aids in tracking when the measurements are poor (i.e. when the SINR is low).

Vehicle 143 moves with nearly constant heading and experiences velocity changes of small magnitude, which are not generally in accordance with the underlying motion preferences (see Fig. 9, left). Furthermore, the underlying velocity preference is a gradual acceleration and deceleration, which is not observed in the data. The heading

TABLE I
RMSE Results of Monte Carlo Testing on Five Test Vehicles

		CV			IIOU			MMNLF		
Veh.	SINR	Pos.	Head.	Vel.	Pos.	Head.	Vel.	Pos.	Head.	Vel.
	(dB)	(m)	(deg)	(m/s)	(m)	(deg)	(m/s)	(m)	(deg)	(m/s)
3	1	45.60	10.27	5.85	44.09	16.58	2.74	42.26	15.98	3.50
3	2	47.75	10.33	5.76	35.71	16.94	2.61	36.17	16.10	3.03
3	4	31.07	8.80	5.07	20.86	16.92	2.52	20.81	15.91	2.82
3	6	18.21	7.89	4.52	12.80	17.05	2.46	13.83	15.69	2.71
3	7	15.24	7.84	4.35	11.17	17.06	2.46	11.54	15.66	2.67
3	8	13.10	7.49	4.24	9.71	17.05	2.45	10.24	15.60	2.64
3	9	10.79	7.38	4.13	8.78	16.96	2.45	9.32	15.60	2.61
3	10	10.16	7.29	4.09	8.14	16.99	2.43	8.35	15.57	2.59
3	12	8.22	7.13	3.98	7.32	16.98	2.42	7.59	15.52	2.56
3	16	7.00	7.07	3.90	6.64	16.94	2.41	6.77	15.46	2.53
143	1	45.23	12.38	5.86	52.06	13.93	3.88	52.87	11.27	4.19
143	2	48.18	11.41	5.63	44.23	13.75	3.89	42.91	10.95	4.20
143	4	29.18	9.33	4.89	25.53	13.22	3.99	26.31	10.99	4.20
143	6	19.15	8.83	4.37	16.84	12.77	4.04	16.32	10.30	4.11
143	7	16.44	8.39	4.23	13.76	12.47	4.02	13.16	10.32	4.09
143	8	12.50	8.45	4.06	11.62	12.43	4.03	11.37	9.95	4.06
143	9	11.17	8.31	3.99	9.55	12.24	4.02	10.00	9.92	4.04
143	10	9.91	8.03	3.92	9.27	12.07	4.01	8.69	9.89	4.02
143	12	8.09	7.85	3.80	7.61	11.90	4.01	7.73	9.79	3.99
143	16	6.77	7.69	3.72	6.58	11.82	4.00	6.60	9.64	3.96
355	1	57.71	15.88	6.06	64.32	33.44	2.40	NaN	NaN	NaN
355	2	44.83	13.14	5.58	60.81	33.21	3.01	34.22	22.04	2.94
355	4	31.77	12.16	4.98	50.73	31.08	2.55	22.09	21.03	2.58
355	6	19.54	10.98	4.43	30.17	30.23	2.12	15.86	19.45	2.41
355	7	15.12	10.56	4.25	25.64	30.24	2.02	12.76	22.77	2.34
355	8	12.84	10.41	4.13	24.44	30.36	2.08	11.20	21.84	2.28
355	9	10.68	10.28	4.02	25.26	29.76	2.05	10.04	20.89	2.25
355	10	9.73	10.20	3.94	24.62	31.09	2.07	8.46	20.73	2.22
355	12	8.13	10.08	3.85	25.33	31.24	2.06	7.67	19.35	2.19
355	16	6.81	9.96	3.77	19.80	30.71	2.04	6.53	19.84	2.16
416	1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
416	2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
416	4	31.74	6.71	4.50	36.63	21.94	4.06	30.36	16.59	3.23
416	6	20.22	5.95	3.86	21.27	22.97	4.13	16.83	15.13	3.14
416	7	17.28	5.75	3.74	15.81	23.04	4.10	13.73	14.70	3.09
416	8	15.24	5.73	3.63	12.96	22.99	4.12	12.14	13.89	3.08
416	9	11.56	5.75	3.48	11.61	22.84	4.12	10.51	13.42	3.05
416	10	10.33	5.68	3.45	9.93	22.48	4.10	9.35	13.14	3.03
416	12	8.81	5.57	3.38	8.37	22.15	4.09	7.67	12.64	3.00
416	16	7.10	5.47	3.30	6.75	21.65	4.08	6.75	12.09	2.99
1432	1	49.16	16.89	5.07	NaN	NaN	NaN	48.95	18.19	2.07
1432	2	43.77	17.14	4.88	43.93	21.74	2.14	39.99	28.66	2.69
1432	4	30.13	15.21	4.32	32.60	36.25	3.18	26.21	26.76	2.85
1432	6	18.72	14.22	3.80	21.11	35.20	3.21	16.97	23.66	2.84
1432	7	14.26	13.77	3.62	17.15	34.51	3.23	13.59	22.21	2.83
1432	8	12.29	13.59	3.52	14.71	33.56	3.23	11.52	21.09	2.79
1432	9	10.93	13.53	3.45	11.81	32.90	3.25	10.31	20.87	2.80
1432	10	9.94	13.37	3.38	9.73	32.30	3.25	8.96	20.29	2.78
1432	12	8.67	13.30	3.29	7.66	31.37	3.25	7.76	19.52	2.76
1432	16	6.79	13.17	3.23	6.91	30.79	3.26	6.78	19.29	2.73

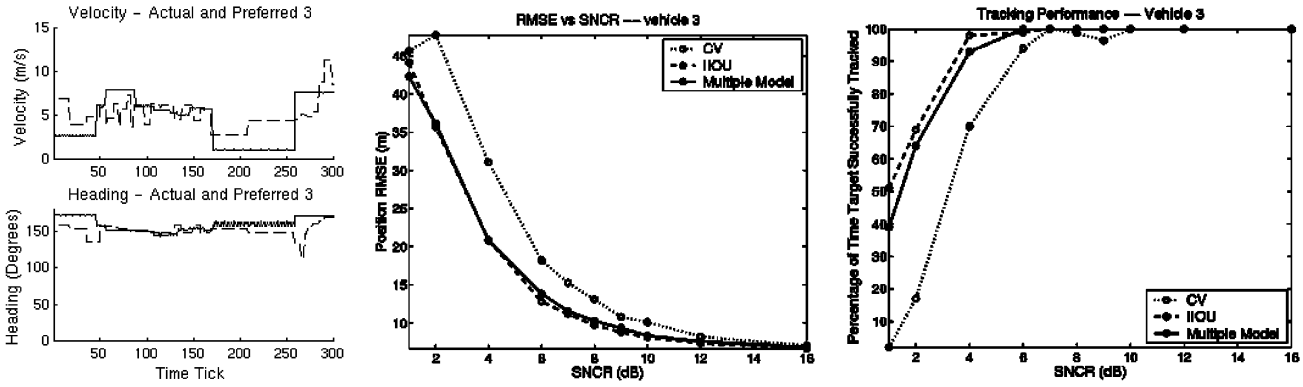


Fig. 8. Vehicle 3. (a) Preferred (dotted) and actual (solid) trajectories. (b) Position RMSE versus SINR. Error is shown averaged over only those trials where the target was successfully tracked. (c) Percentage of targets successfully tracked versus SINR.

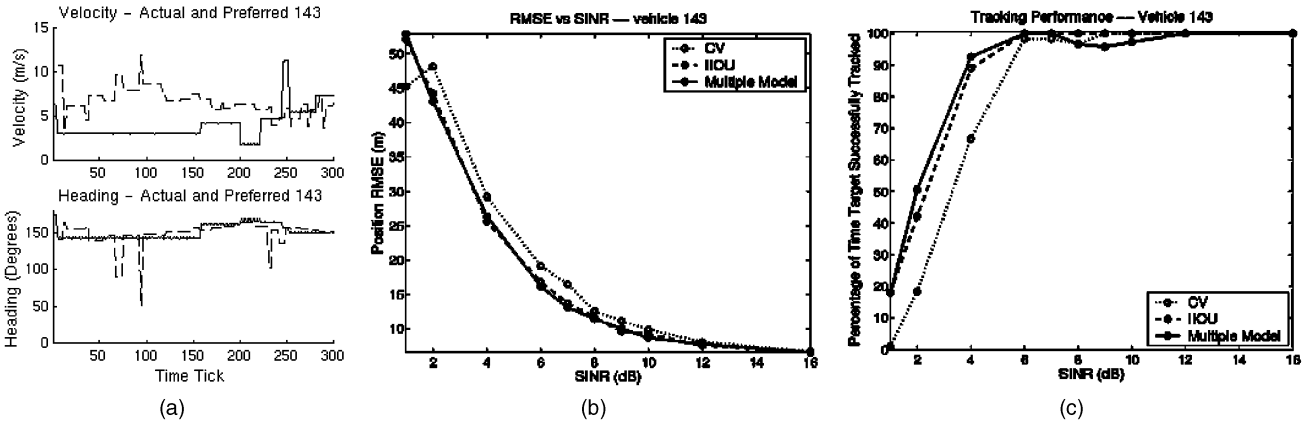


Fig. 9. Vehicle 143. (a) Preferred (dotted) and actual (solid) trajectories. (b) Position RMSE versus SINR. Error is shown averaged over only those trials where the target was successfully tracked. (c) Percentage of targets successfully tracked versus SINR.

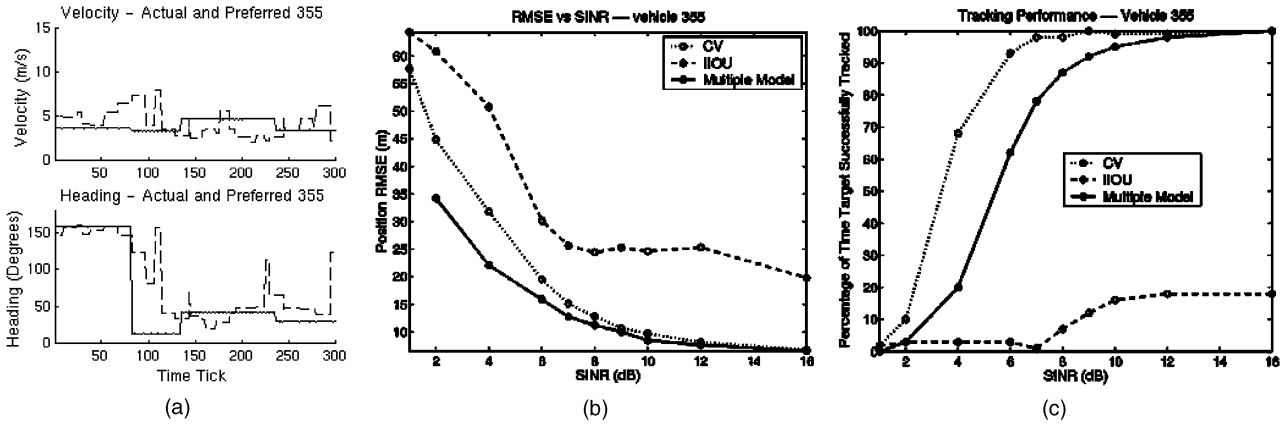


Fig. 10. Vehicle 355. (a) Preferred (dotted) and actual (solid) trajectories. (b) Position RMSE versus SINR. Error is shown averaged over only those trials where the target was successfully tracked. (c) Percentage of targets successfully tracked versus SINR.

matches the motion preferences very well. For these reasons, we expect the IIOU filter to perform worse than it did with vehicle 3. As shown in Fig. 9, all three filters are able to track the vehicle well at SNR above 5 dB SINR. The IIOU filter and MMNLF have 90% tracking ability at 4 dB SINR. Again the IIOU filter and MMNLF outperform the CV filter at low SINR due to their ability to rely on heading and velocity preferences.

Vehicle 355 has a slowly changing velocity and contains one very abrupt heading change (see Fig. 10, left). The heading change happens at a location at odds with the data derived preferred heading. For these reasons, we expect the IIOU filter to have difficulty. The CV filter, however, imposes no

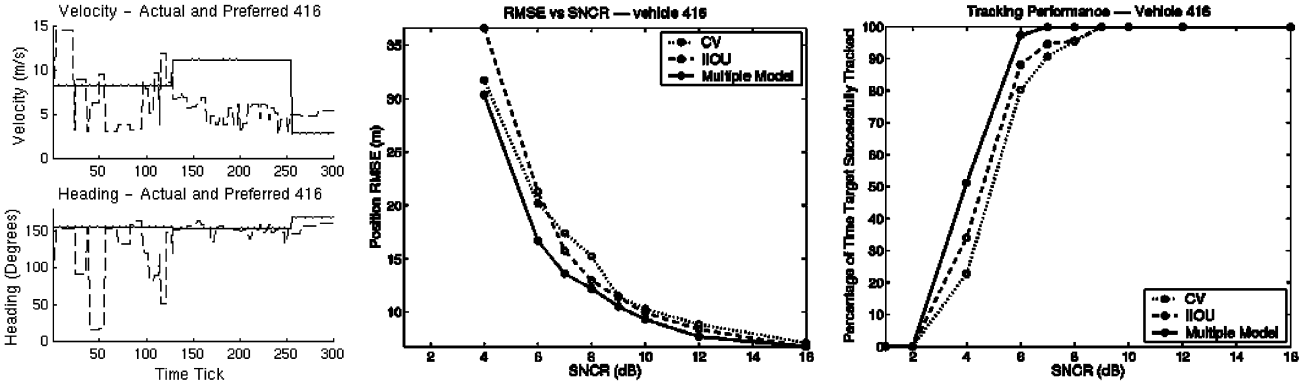


Fig. 11. Vehicle 416. (a) Preferred (dotted) and actual (solid) trajectories. (b) Position RMSE versus SINR. Error is shown averaged over only those trials where the target was successfully tracked. (c) Percentage of targets successfully tracked versus SINR.

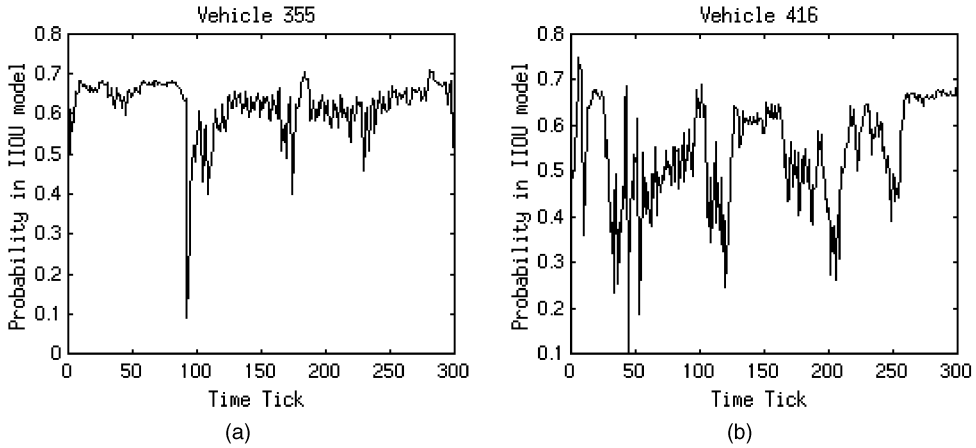


Fig. 12. Mode mixing for two of the trials. (a) Probability in IIOU model for one trial with vehicle 355. (b) Probability in IIOU model for one trial with vehicle 416.

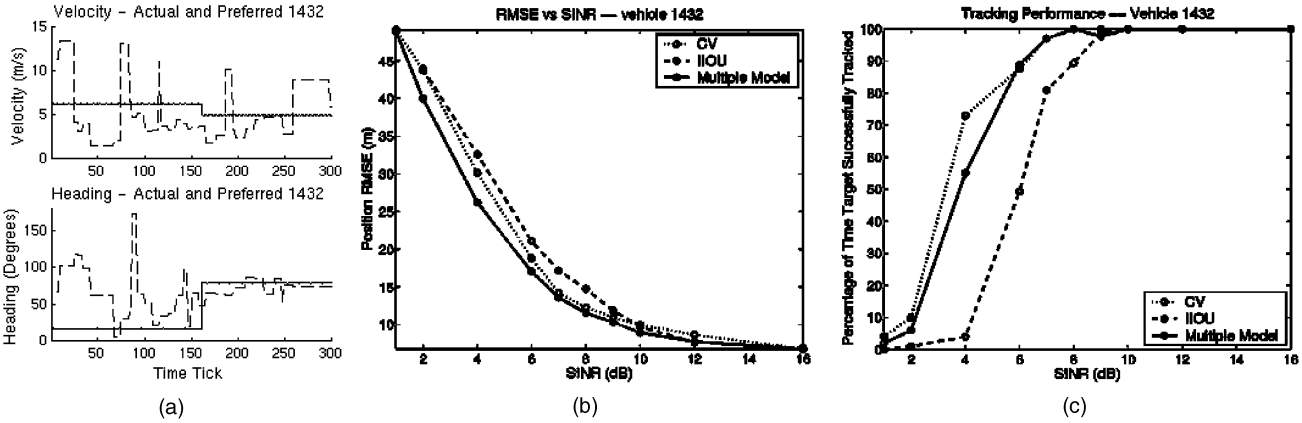


Fig. 13. Vehicle 1432. (a) Preferred (dotted) and actual (solid) trajectories. (b) Position RMSE versus SINR. Error is shown averaged over only those trials where target was successfully tracked. (c) Percentage of targets successfully tracked versus SINR.

preferred heading and may be able to overcome the very abrupt maneuver. The abrupt change in heading that is unanticipated by the IIOU filter (time step 80) causes it to consistently lose track. The IIOU filter, at best, only maintains track 20% of the time. The CV filter performs much better. The MMNLF, aided by the ability to switch into CV mode at the time of the unanticipated maneuver, performs much better than the IIOU filter as well.

Fig. 12(a) shows how the MMNLF is able to generally maintain the track in this case. At time step 80, where the abrupt change in heading occurs, probability rapidly flows out of the IIOU model and into the CV model. Note that, unlike conventional

multiple model filters, in this example both models (IIOU and CV) typically have validity. Therefore, under normal operating situations, the probability in each filter is dictated by the mixing matrix Λ.

Vehicle 416 has a slowly changing heading and contains 2 very abrupt velocity changes of several meters/second. With limited exceptions, the data-derived preferred motion accurately models the heading and for the most part underestimates the velocity (see Fig. 11, left). We expect the CV filter to have difficulty with the sudden velocity changes. We expect the IIOU filter to track acceptably, as the largest velocity change brings the vehicle into agreement with the preferred velocity. Fig. 11 shows the error curves for the three filters. All three filters perform reasonably well at SINR greater than 5 dB, with the MMNLF being the best of the three.

Fig. 12(b) illustrates how the MMNLF is able to succeed for this vehicle. It is able to quickly move into the CV model at those points where the preferred and actual trajectories differ (e.g. time step 50, and time step 120). It is also able to move into the IIOU model at the time that gives the CV filter significant trouble (time step 256, where the dramatic acceleration is encountered).

Vehicle 1432 has a very slowly changing velocity that is not in agreement with the data derived preferred velocity (see Fig. 13). The preferred and actual headings do not agree very well, particularly during the first 150 time steps. For these reasons, we expect the CV filter to perform as good or better than the IIOU filter. Fig. 13 shows that the CV filter and the MMNLF perform similarly. The IIOU filter performs poorly as it incorrectly enforces a preferred heading and velocity that the vehicle violates.

A study of the value of using nonthresholded measurements (as was done in the preceding experiments) is illustrated in Fig. 14. A comparison to the MMNLF using thresholded measurements, where the probability of detection was set at 0.5, shows a dramatic difference particularly at low SINR. Neither the MMNLF nor the thresholded MMNLF lost track in any of the trials. The thresholded MMNLF will give different results with different threshold settings. A parametric test of the thresholded MMNLF on vehicle 3 at 8 dB SINR (Fig. 15) shows that the optimal P_d is between 0.4 and 0.7.

As described earlier, HM can be used to generate the spatially-varying preferred headings and velocities. To illustrate the utility of HM, we compared the results of the multiple model filter using data-derived preferences (denoted MM-DD), and the multiple model filter using HM-derived preferences (denoted MM-HM) on test vehicle 3. This comparison is shown in Fig. 16. The HM was used to generate go and no-go regions. In areas of low hospitability, the preferred velocity was set to 1 m/s, and in areas of high hospitability the velocity was set to 4 m/s. The preferred heading was generated using the notion that vehicles tend to follow areas of high hospitability.

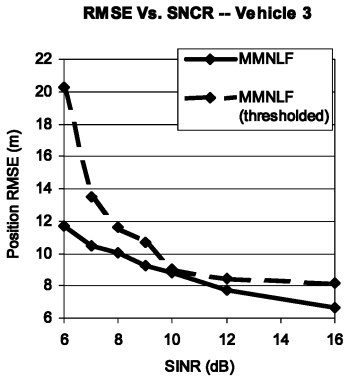


Fig. 14. Value of using nonthresholded measurements in the MMNLF.

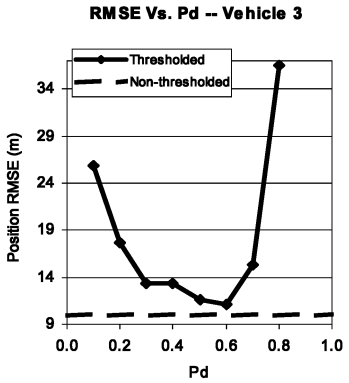


Fig. 15. Position RMSE versus detection threshold. Nonthresholded MMNLF results shown for reference.

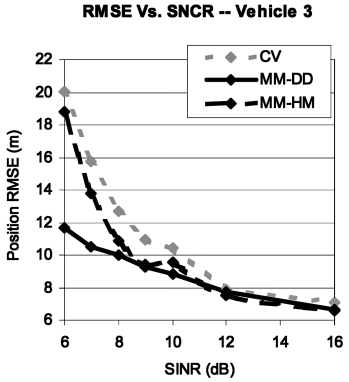


Fig. 16. Comparison of multiple model filters using data derived and HM-derived preferences.

The data-derived filter kept the vehicle in track 97% of the time (as compared to 93% for the CV model and 100% for the HM-derived filter). Both the data-derived and HM-derived multiple model filters show an advantage over a simple CV model, particularly at low SINR. The data-derived method, however, tracks with significantly lower error than the HM version.

V. CONCLUSION

This paper has described the design and implementation of MMNLF for ground target tracking using GMTI radar measurements. The filter equations are derived using general nonlinear filtering theory, and the partial differential equations are solved on a grid using an ADI scheme. An alternate approach to solving the nonlinear filtering equations is the particle filter approach [23]. In fact, as the state dimensionality increases (for example with multiple targets [24]) particle filter schemes may be the only tractable approach, as grid-based approaches such as those presented here have dimension that grows exponentially in the dimension of the state vector.

Experiments on real data have shown that the MMNLF implemented here performs better than either of the constituent individual filters. The performance was measured by the ability of the filter to keep the target on the grid and localize it with minimum error. Potential applications of this work include detection and tracking of foliage-obscured moving targets. In wartime or other realistic situations, this data may not be reliable for all areas of interest. Some areas will have incomplete or missing data, either due to lack of vehicle histories or conflicting vehicle histories. It is therefore necessary to allow the filter to operate without these control inputs in these situations. Here this is addressed by the second model, which is a nearly constant speed/constant heading model. Based on sensor measurements, the MMNLF adaptively weights the models and uses the one most appropriate for the current region.

ACKNOWLEDGMENTS

The authors wish to acknowledge and thank Stan Musick for helpful discussion regarding this work, and Michael Pagels for providing hospitability data and the necessary information for its use.

APPENDIX A. NUMERICAL SOLUTION OF MMFPE

The first step in solving the MMFPE, (7), is to sample the time interval $t_k \le t < t_{k+1}$ into L steps with resolution $\delta = (t_{k+1} - t_k)/L$. We denote the density at step l, $l = 1,\ldots,L$ as $\mathbf{p}^l \equiv \mathbf{p}(\mathbf{x}_{t_k+l\delta} \mid Y_{t_k}) = \mathbf{p}(\mathbf{x}, t_k + l\delta)$ (recall that the probability vector $\mathbf{p} = (p_1,\ldots,p_r)^T$ is composed of r model densities) . Taylor expanding in time, we have $\mathbf{p}^l = \mathbf{p}^{l+1} - \delta(\partial/\partial t)\mathbf{p}^{l+1} + O(\delta^2)$ where $O(\delta^2)$ indicates higher order terms in the series. Thus

$$\frac{\mathbf{p}^{l+1} - \mathbf{p}^l}{\delta} = (\mathbf{L} + \mathbf{\Lambda})\mathbf{p}^{l+1} + O(\delta^2). \tag{52}$$

Rearranging, we have

$$(\mathbf{1} - \delta(\mathbf{L} + \mathbf{\Lambda}))\mathbf{p}^{l+1} = \mathbf{p}^l + O(\delta^2) \tag{53}$$

where $\mathbf{1}$ is the $r \times r$ identity matrix. The matrix operator on the left-hand side can be factorized to $O(\delta^2)$ as

$$(\mathbf{1} - \delta(\mathbf{L} + \mathbf{\Delta})) = (\mathbf{1} - \delta\mathbf{L})(\mathbf{1} - \delta\mathbf{\Lambda}) + O(\delta^2). \tag{54}$$

Defining $(\mathbf{1} - \delta\mathbf{\Lambda})\mathbf{p}^{l+1} \equiv \tilde{\mathbf{p}}^{l+1}$, we see that the solution to (53) can be obtained by first solving

$$(\mathbf{1} - \delta\mathbf{L})\tilde{\mathbf{p}}^{l+1} = \mathbf{p}^l + O(\delta^2) \tag{55}$$

to obtain $\tilde{\mathbf{p}}^{l+1}$, then solving the matrix equation

$$(\mathbf{1} - \delta\mathbf{\Lambda})\mathbf{p}^{l+1} = \tilde{\mathbf{p}}^{l+1} \tag{56}$$

which simply requires a inversion of the $r \times r$ matrix $\mathbf{1} - \delta\mathbf{\Lambda}$.

To solve (55), note that it can be written as the set of r uncoupled scalar differential equations

$$(1 - \delta L)\tilde{p}_m^{l+1} = p_m^l + O(\delta^2), \qquad m = 1,\ldots,r \tag{57}$$

(1 is the scalar identity here) each of which can be solved using the conventional time-splitting method. This proceeds as follows. Each of the L_m operators can be decomposed as a sum,

$$L_m = \sum_i L_{m,i} \tag{58}$$

so that

$$(1 - \delta L_m)p_m^{l+1} = \prod_i (1 - \delta L_{m,i})p_m^{l+1} + O(\delta^2). \tag{59}$$

At this point, we discretize the spatial domain with resolution $\Delta\mathbf{x} = [\Delta x, \Delta y, \Delta\theta, \Delta v]$. On this grid, let g_m^l denote an $O(\Delta\mathbf{x}) + O(\delta^2)$ approximation to p_m^l and $L_{m,i\Delta\mathbf{x}}$ an $O(\Delta\mathbf{x})$ approximation to $L_{m,i}$, i.e.,

$$L_{m,i\Delta\mathbf{x}}g_m^l = L_{m,i}p_m^l + O(\Delta\mathbf{x}) + O(\delta^2). \tag{60}$$

Thus, if $\tilde{g}_m^{l+1}$ satisfies

$$\prod_i (1 - \delta L_{m,i\Delta\mathbf{x}})\tilde{g}_m^{l+1} = g_m^l \tag{61}$$

then

$$\tilde{g}_m^{l+1} = p_m^{l+1} + O(\Delta\mathbf{x}) + O(\delta^2). \tag{62}$$

To complete the solution, we must have explicit decompositions for the $L_{m,i\Delta\mathbf{x}}$. For the CV model (29), we choose 4 terms in the summation (58):

$$L_1 = -\dot{x}\frac{\partial}{\partial x} \tag{63}$$

$$L_2 = -\dot{y}\frac{\partial}{\partial y} \tag{64}$$

$$L_3 = \frac{1}{2}q_\theta(x,y)\frac{\partial^2}{\partial\theta^2} \tag{65}$$

$$L_4 = \frac{1}{2}q_v(x,y)\frac{\partial^2}{\partial v^2}. \tag{66}$$

The IIOU model (24) adds to these 4:

$$L_5 = \frac{1}{\tau_\theta(x,y)}\frac{\partial}{\partial\theta}(\theta-\theta_0(x,y,\theta)) \tag{67}$$

$$L_6 = \frac{1}{\tau_v(x,y)}\frac{\partial}{\partial v}(v-v_0(x,y)). \tag{68}$$

To discretize the L_i, we use the abbreviation $g_{x\pm1} \equiv (l\delta, x\pm\Delta x, y, \theta, v)$ (suppressing the mode index), with similar definitions for $g_{y\pm1}$, $g_{\theta\pm1}$, and $g_{v\pm1}$. Using upwind differencing for the advective terms in the CV model, we have

$$L_{1\Delta\mathbf{x}}g = \frac{-\dot{x}}{2\Delta x}\begin{cases} g_x - g_{x-1} & \dot{x}\geq 0 \\ g_{x+1}-g_x & \dot{x}<0 \end{cases} \tag{69}$$

$$L_{2\Delta\mathbf{x}}g = \frac{-\dot{y}}{2\Delta y}\begin{cases} g_y - g_{y-1} & \dot{y}\geq 0 \\ g_{y+1}-g_y & \dot{y}<0 \end{cases}. \tag{70}$$

Central differencing for the diffusive terms gives

$$L_{3\Delta\mathbf{x}}g = \frac{q_\theta(x,y)}{2\Delta\theta^2}(g_{\theta+1}-2g_\theta+g_{\theta-1}) \tag{71}$$

and

$$L_{4\Delta\mathbf{x}}g = \frac{q_v(x,y)}{2\Delta v^2}(g_{v+1}-2g_v+g_{v-1}) \tag{72}$$

and when using the IIOU-based model, we have for L_5 and L_6:

$$L_{5\Delta\mathbf{x}}g = \frac{-(\theta-\theta_0)}{2\Delta\theta}\begin{cases} g_\theta - g_{\theta-1} & \theta-\theta_0\geq 0 \\ g_{\theta+1}-g_\theta & \theta-\theta_0<0 \end{cases} \tag{73}$$

$$L_{6\Delta\mathbf{x}}g = \frac{-(v-v_0)}{2\Delta v}\begin{cases} g_v - g_{v-1} & v-v_0\geq 0 \\ g_{v+1}-g_v & v-v_0<0 \end{cases}. \tag{74}$$

We note that for each of the operators (69)–(74) is a tridiagonal system, i.e., of the form

$$a_j g_{j-1} + b_j g_j + c_j g_{j+1} = d_j, \qquad j = 1,\ldots,m-1 \tag{75}$$

with boundary conditions g_0 and g_m specified, g_j, $j=1,\ldots,m-1$ are unknowns and a_j, b_j, c_j and d_j are known (the boundary conditions are discussed in detail in the following subsection). This can be solved efficiently using Thomas's algorithm ([9]). The net complexity of this algorithm is $O(N)$. With this, the total complexity of the time-splitting scheme with Thomas' algorithm for solving the tridiagonal systems is $O(N)$.

To obtain the complete solution, let N_m denote the number of suboperators in the scalar system ($N_m = 4$ for CV and 6 for IIOU). Then for each scalar equation we define

$$(1-\delta L_{m,i\Delta\mathbf{x}})\tilde{g}_m^{k+i/N_m} = \tilde{g}_m^{k+(i-1)/N_m}, \qquad i=1,\ldots,N_m \tag{76}$$

$$\mathbf{g}^{k+1} = (\mathbf{1}-\delta\mathbf{\Lambda})^{-1}\tilde{\mathbf{g}}^{k+1} \tag{77}$$

with $\tilde{g}_m^k \equiv g_m^k$. The utility of this is that each of the operators $1-\Delta t L_{m,i\Delta\mathbf{x}}$ is a tridiagonal system so the (76) can be solved easily while (77) entails inversion of a low-dimensioned matrix. Thus, the overall complexity of the algorithm is linear in the number of grid nodes.

Boundary Conditions: To solve the discretized MMFPE numerically it must be restricted to a finite domain leading to an initial-boundary value problem. For each of the r kinematic models, the finite grid domain consists of the points $((i+i_0)\Delta x, (j+j_0)\Delta\dot{x}, (k+k_0)\Delta y, (l+l_0)\Delta\dot{y})^T$, $i=0,\ldots,N_x$, $j=0,\ldots,N_{\dot{x}}$, $k=0,\ldots,N_y$, $l=0,\ldots,N_{\dot{y}}$, where $i_0,\ldots,l_0$ are offsets used to translate the origin. For each model there are $(N_x+1)(N_{\dot{x}}+1)(N_y+1)(N_{\dot{y}}+1)$ grid nodes and $N=(N_x-1)(N_{\dot{x}}-1)(N_y-1)(N_{\dot{y}}-1)$ unknowns. Boundary conditions must be specified on this hyper-cube to determine the solution to the FPE uniquely. We assume that the target signal-to-noise ratio is sufficiently high that the target has been localized. Then the density will be concentrated in some small region and decay exponentially far from this region. We further assume that the grid is large enough that the density was small on its boundary. With this motivation we used a homogenous Dirichlet condition with the solution held at 0 on the boundary.

The FPE does not have second-order derivatives in all of its variables which means that it is a degenerate parabolic partial differential equation [31]. Qualitatively, since there are no x- and y-diffusion terms in the CV and IIOU FPE, the behavior on the x- and y-subspaces is characterized by the one-way wave equation. (For example $(\partial p/\partial t) = -\dot{x}(\partial p/\partial x)$ generates a wave that propagates solutions along the x-axis with velocity $\dot{x}$.) For a first-order operator of this sort the boundary condition is only specified on the incoming boundary as determined by the sign of $\dot{x}$. In this case if independent boundary conditions are specified on all of the faces then the solution is over determined. We avoided this problem by only specifying a physical boundary condition on the incoming x- and y-faces.

Even though the physical boundary condition is only defined for incoming x and y, the discretization scheme ((78) and (79)) requires that the solution also be specified on the out-going faces as well. This requires so-called numerical boundary conditions. A simple boundary condition to implement is to extrapolate the solution at the out-going faces [34]. For example, in the region with $\dot{x}>0$, the face with $i=N_x$ is an out-going face. On this face we specified $g_{N_x}^{k+1} = g_{N_x-1}^{k+1}$. On the other hand, when $\dot{x}<0$, the out-going face has $i=0$ so $g_0^{k+1} = g_1^{k+1}$. These numerical boundary conditions are incorporated into the Thomas tridiagonal solver without affecting its complexity.

Grid Translation: To reduce the size of the grid required to represent the target joint density,

the grid was translated after each measurement to approximately maintain the target's location near its center. After each measurement update the target position estimate $(\hat{x}_t, \hat{y}_t)$ was evaluated and the grid was shifted to center the grid near this estimate. This was achieved by placing the lower left corner of the spatial grid at (i_0, k_0) where

$$i_0 = [\hat{x}/\Delta x - N_x/2] \tag{78}$$

$$k_0 = [\hat{y}/\Delta y - N_y/2] \tag{79}$$

and $[x]$ denotes rounding to the nearest integer. This always translated the grid by an integral multiple of $(\Delta x, \Delta y)$. Grid nodes outside the intersection of the original and translated grids were set to 0.

Timing: This scheme was implemented in MatLab, running on an off-the-shelf 3 GHz Linux machine. For the simulations reported here, measurements arrived every 1 s. For the single model filters, the CV filter requires about 1 s of CPU to time and measurement update, and the IIOU filter requires about 1.3 s of CPU time. The MMNLF requires approximately 2 s of CPU time. It is anticipated that a C-based implementation may speed this up by as much as an order of magnitude. As mentioned earlier, an alternative approach to solving the nonlinear filtering equations is via particle methods [23]. In fact, using appropriate sampling schemes, in one study particle methods were shown to provide better computational performance than finite difference methods on a related problem [25].

REFERENCES

[1] Entzminger, J. N., Jr., Fowler, C. A., and Kenneally, W. J.
JointSTARS and GMTI: Past, present and future.
IEEE Transactions on Aerospace and Electronic Systems, **35**, 2 (Apr. 1999), 748–760.

[2] Gross, L. A., Guarino, R. A., and Holt, H. D., Jr.
AN/APY-6 real time surveillance and targeting radar development.
RTO SET Symposium on "High Resolution Radar Techniques," Granada, Spain, Mar. 22–24, 1999, RTO-MP-40, 31.1–31.6.

[3] Ho, Y. C., and Lee, R. C. K.
A Bayesian approach to problems in stochastic estimation and control.
IEEE Transactions on Automatic Control, **19** (Oct. 1964), 333–339.

[4] Jazwinski, A. H.
Stochastic Processes and Filtering Theory.
New York: Academic Press, 1970.

[5] Kastella, K., and Zatezalo, A.
Nonlinear filtering for low elevation targets in the presence of multipath propagation.
In *Proceedings of SPIE Aerosense '98*, Signal and Data Processing of Small Targets 1998, *Vol. 3373*, 452–459.

[6] Kastella, K.
Finite difference methods for nonlinear filtering and automatic target recognition.
In Y. Bar-Shalom and W. D. Blair (Eds.), *Multitarget/Multisensor Tracking: Applications and Advances Volume III*.
Norwood, MA: Artech House, 2000.

[7] Kirubarajan, T., Bar-Shalom, Y., Pattipati, K. R., and Kadar, I.
Ground target tracking with variable structure IMM estimator.
IEEE Transactions on Aerospace and Electronic Systems, **36**, 1 (Jan. 2000), 26–46.

[8] Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P.
Numerical Recipes in C (2nd ed.).
New York: Cambridge University Press, 1992.

[9] Strikwerda, J. C.
Finite Difference Schemes and Partial Differential Equations.
New York: Chapman & Hall, 1989.

[10] U.S. Army
U.S. Army Material Systems Analysis Activity, The Natick landform classification system.
Technical Report, 100, (Oct. 1974), Aberdeen Proving Ground, MD, U.S. Army.

[11] U.S. Army Corp of Engineers
Terrain analysis procedural guide for surface configuration.
Fort Belvoir, VA: Engineering Topographic Laboratories, 1984.

[12] U.S. Army Corp of Engineers
Stochastic vehicle mobility forecasts using the NATO reference mobility model.
Department of the Army, Washington, DC, GL-92-11, 1992.

[13] Zatezalo, A.
Tracking and detection for the target state model.
M.S. Thesis, University of Minnesota, Mar. 15, 1997.

[14] Kastella, K., Kreucher, C., and Pagels, M.
Nonlinear filtering for ground target applications.
Proceedings of SPIE Aerosense 2000, Signal and Data Processing of Small Targets 2000.

[15] Kreucher, C., and Kastella, K.
Multiple model nonlinear filtering for low signal ground target applications.
Proceedings of SPIE Aerosense 2001, Signal Processing, Sensor Fusion and Target Recognition 2001.

[16] Bar-Shalom, Y., and Blair, W.
Multitarget-Multisensor Tracking: Applications and Advances, Vol. III.
Norwood, MA: Artech House, 2000.

[17] Wax, N., et al.
Selected Papers on Noise and Stochastic Processes.
New York: Dover, 1954.

[18] Shea, P. J., Zadra, T., Klamer, D., Frangione, E., Brouillard, R., and Kastella, K.
Precision tracking of ground targets.
In *Proceedings of IEEE Aerospace 2000*, Big Sky, MT.

[19] Stone, L. D., Barlow, C. A., and Corwin, T. L.
Bayesian Multiple Target Tracking.
Norwood, MA: Artech House, 1999.

[20] Blom, H. A. P., and Bar-Shalom, Y.
The interacting multiple model algorithm for systems with Markovian switching coefficients.
IEEE Transactions on Automatic Control, **33** (Aug. 1988), 780–783.

[21] Papoulis, A.
Probability, Random Variables and Stochastic Processes.
New York: McGraw-Hill, 1984.

[22] Challa, S., and Bar-Shalom, Y.
Nonlinear filtering using Fokker Planck Kolmogorov probability density evolutions.
IEEE Transactions on Aerospace and Electronic Systems, **36** (Jan. 2000), 309–315.

[23] Doucet, A., de Freitas, N., and Gordon, N.
Sequential Monte Carlo Methods in Practice.
Berlin: Springer-Verlag, 2001.

[24] Kreucher, C., Kastella, K., and Hero, A.
Tracking multiple targets using a particle filter representation of the joint multitarget probability density.
Presented at SPIE International Symposium on Optical Science and Technology, San Diego, CA, 2003.

[25] Musick, S., Greenwald, J., Kreucher, C., and Kastella, K.
A comparison of particle method and finite difference nonlinear filters for low SNR target tracking.
Presented at The 4th International Conference on Information Fusion, Montreal, Quebec, Aug. 2001.

[26] McGinnity, S., and Irwin, G. W.
Manuevering target tracking using a multiple-model bootstrap filter.
In A. Doucet, N. de Freitas, and N. Gordon (Eds.), *Sequential Monte Carlo Methods in Practice*, New York: Springer-Verlag, 2001.

[27] Petrov, A., Rozovsky, B., and Tartakovsky, A.
Efficient nonlinear filtering methods for detection of dim targets by passive systems.
In X. R. Li, W. D. Blair, and T. Kirubarajan (Eds.), *Multitarget-Multisensor Tracking: Applications and Advances, Vol. IV*, Norwood, MA: Artech House, 2004.

[28] Boers, Y., and Driessen, H.
Particle filter based track before detect algorithms.
Presented at SPIE International Symposium on Optical Science and Technology, San Diego, CA, 2003.

[29] Kushner, H. J.
Probability Methods for Approximations in Stochastic Control and for Elliptic Equations.
New York: Academic Press, 1977.

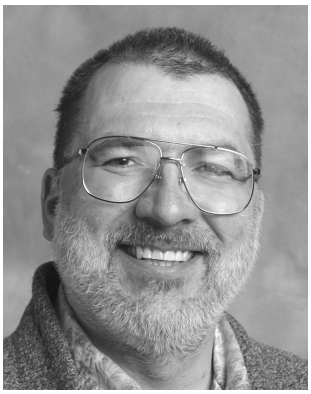
[30] Kushner, H. J., and Dupuis, P. G.
Numerical Methods for Stochastic Control Problems in Continuous Time.
New York: Springer-Verlag, 1991.

[31] Friedman, A.
Uniqueness for the Cauchy problem for degenerate parabolic equations.
Pacific Journal of Mathematics, **46**, 1 (1973).

[32] Arulampalam, M. S., Maskell, S., Gordon, N., and Clapp, T.
A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking.
IEEE Transactions on Signal Processing, **50**, 2 (2002).

[33] Gutmann, J-S., and Fox, D.
An experimental comparison of localization methods.
In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2002.

[34] Lin, X., Kirubarajan, T., Bar-Shalom, Y., and Maskell, S.
Comparison of EKF, pseudomeasurement, and particle filters for a bearing-only target tracking problem.
In *Proceedings of SPIE Signal and Data Processing of Small Targets 2002, Vol. 4728*, Aug. 2002, 240–250.



Keith Kastella received a B.A. in physics in 1980 from Reed College, Portland OR, and physics M.S. and Ph.D. degrees from the State University of New York at Stony Brook (SUNY-SB) in 1985 and 1988, respectively.

He is a chief scientist at General Dynamics Advanced Information Systems in Ann Arbor, MI. His current research interests are in data fusion, signal processing and novel applications of quantum entanglement in sensing and signal processing.

Chris Kreucher received the B.S. and M.S. degrees in electrical and computer engineering from the University of Michigan–Dearborn in 1996 and 1997, respectively.

Since 1997, he has been a researcher in the Multisensor Fusion and Signal Procssing Center at General Dynamics (formerly ERIM), located in Ann Arbor, MI. Since 2002, he has also been a Ph.D. student in the Department of Electrical Engineering and Computer Science at the University of Michigan. His current research interests include multitarget tracking and sensor management.