

Efficient Particle Filtering for Road-Constrained Target Tracking

YANG CHENG

TARUNRAJ SINGH
University at Buffalo

The variable-structure multiple-model particle filtering approach for state estimation of road-constrained targets is addressed. The multiple models are designed to account for target maneuvers including “move-stop-move” and motion ambiguity at an intersection; the time-varying active model sets are adaptively selected based on target state and local terrain condition. The hybrid state space is partitioned into the mode subspace and the target subspace. The mode state is estimated based on random sampling; the target state as well as the relevant likelihood function associated with a mode sample sequence is approximated as Gaussian distribution, of which the conditional mean and covariance are deterministically computed using a nonlinear Kalman filter which accounts for road constraints in its update. The importance function for the sampling of the mode state approximates the optimal importance function under the same Gaussian assumption of the target state.

Manuscript received September 26, 2005; revised April 21 and August 15, 2006; released for publication September 13, 2006.

IEEE Log No. T-AES/43/4/914424.

Refereeing of this contribution was handled by W. Koch.

Authors' address: Dept. of Mechanical and Aerospace Engineering, University at Buffalo, Amherst, NY 14260, E-mail: (cheng3@eng.buffalo.edu, tsingh@buffalo.edu).

0018-9251/07/\$25.00 © 2007 IEEE

I. INTRODUCTION

A notable characteristic of ground target tracking is that prior nonstandard information such as target speed constraints, road networks, and so forth can be exploited in the tracker to reduce the uncertainty of target motion and provide better estimates of the target state [1]. A tracker that ignores or is unable to make use of this additional source of information can only attain limited performance. In the cases of low signal-to-noise ratio, the incorporation of such constraint information is essential to successful tracking.

Multiple-model estimation is widely used in the tracking community to tackle motion and environment uncertainty. The interacting multiple-model estimator [2] based on (extended) Kalman filtering is one of the best known multiple-model estimators. Recent applications of multiple-model estimators to ground target tracking were presented in [1], [3]–[11]. The multi-model structure in ground target tracking is a result of dynamics motion uncertainties, varying terrain conditions, road segment constraints, and road intersections. The underlying filtering techniques of the multiple-model estimators include the nonlinear filter that solves for the probability distribution using an alternating direction implicit finite difference method [3], the (extended) Kalman filter [4–8], and the particle filter [1, 8–11].

Multiple-model estimation falls into the category of nonlinear filtering even if every single model is a linear system with Gaussian noise. A sufficient statistic of the hybrid state distribution with a fixed dimension is thus impossible. Moreover, the complexity of the optimal multiple-model estimator increases exponentially with time [1]. Both the interacting multiple-model estimator and the multiple-model particle filter are suboptimal nonlinear filtering algorithms that maintain constant complexity and computational expense. The former maintains a constant number (i.e., the number of models) of Kalman filters while the latter maintains a constant number of (the most likely) particle trajectories. Such suboptimality is inevitable in practical applications.

Kirubarajan and Bar-Shalom noted that for ground target tracking a fixed-structure multiple-model estimator has to consist of a large number of models, owing to the many possible motion modes and various road constraints, which is not only computationally undesirable but also potentially results in highly degraded estimates (due to the excessive “competition” among the many models) [4]. In order to overcome this problem, they proposed an adaptive or variable-structure interacting multiple-model estimator for ground target tracking [4]. The basic idea is that the active model set varies in an adaptive manner and thus only a small number of active models need to be maintained at each time.

Another variable-structure interacting multiple-model estimator improving state estimation through the use of road information was proposed in [7]. A feature of the estimator in [7] is that the road network is modeled as having both length and width. A variety of motion models were considered in [7], including the stopped model, the stationary moving model, the constant velocity model, and the maneuver model. In [8], the predicted probability distribution of the road-constrained target is approximated as a Gaussian sum on road segments, and roads with different characteristics are treated within the interacting multiple-model scheme.

Following the same idea of the variable-structure interacting multiple-model estimator, a variable-structure multiple-model particle filter was proposed for ground target tracking in [1]. Particle filters comprise a very broad class of suboptimal nonlinear filters based on sequential Monte Carlo simulations in which the distributions are approximated by weighted particles (random samples) that are generated using pseudorandom number generators. The specific particle filter employed in [1] is known as the bootstrap filter. Simulation results showed that the particle filtering based approach has remarkably better error performance. The reason for the superiority of this particle filtering based approach to the interacting multiple-model estimator, as noted in [1], is that with particles or random samples the simulation-based particle filter is able to incorporate more accurate dynamics models and estimate non-Gaussian distributions (e.g., an on-road target at an intersection) more accurately than the Kalman-filtering-based interacting multiple-model estimator. It was also pointed out in [1] that while the target is off-road the performance of the two filters is comparable because the posterior distribution in the off-road region, under no explicit hard constraints, tends to be nearly Gaussian. The superiority of multiple-model bootstrap filter to the interacting multiple-model estimator within the fixed structure multiple-model framework was demonstrated in [12]. The applications of the bootstrap filter in road-constrained target tracking were also proposed in [8], [10], [11].

One of the major concerns of the application of particle filtering to target tracking is its efficiency and computational expense. The convergence rate of ideal Monte Carlo sampling is of the order of $\mathcal{O}(1/\sqrt{N})$, where N is the number of particles [12]. The actual convergence rate of a particle filter depends on the choice of the importance function (also known as the proposal distribution) and the size of the state space. The bootstrap filter for multiple-model target tracking, as given in [1], [8], [10]–[12], is simple to implement but is also known for its inefficiency. A large number of particles and therefore high computational expense

are usually required in order to attain certain accuracy and robustness.

The main objective of this paper is to derive an efficient particle filter for road-constrained target tracking. The basic idea is to reduce the whole sampling space of the multiple-model system to the mode subspace by marginalization over the target subspace and to choose better importance function for mode state sampling. The design of such an efficient particle filter is based on optimal particle filtering for ideal jump Markov linear Gaussian systems, as given in [13], [14]. Of particular interest is that for jump Markov linear Gaussian systems, the target state conditioned on a mode sample sequence (as well as the observation time history) is exactly Gaussian and therefore can be analytically determined using a standard Kalman filter; the mode state is the only part that needs to be estimated using particle filtering; because the mode state only takes on a finite number of discrete values, the optimal importance function for the mode state based on marginalization can be used. Stated in another way, for jump Markov linear Gaussian systems, the variance reduction techniques such as Rao-Blackwellization and optimal importance function can be applied.

For variable-structure multiple-model-based road-constrained target tracking, the assumption of jump Markov linear Gaussian system is not exactly met because of the nonlinearity of the observation model, the use of variable-structure multiple models for motion uncertainty and road ambiguity, and the road constraints in target motion. (Strictly speaking, the target mode is not an exact jump Markov process; the transition of the target mode may depend on the target position and velocity as well.) Our proposed efficient particle filter is based on the Gaussian approximation about the conditional target state and likelihood function associated with each mode sample sequence. The mean and covariance of the conditional target state and the likelihood function is computed using a nonlinear Kalman filter known as the unscented filter [15]. This particle filter is a combination of the sampling approximation of the mode subspace and the analytic approximation of the target subspace, and may outperform those particle filters that are solely based on sampling approximation, when the Gaussian approximation holds to a satisfactory degree. Were the Gaussian approximation of the target state severely violated, a more general framework purely based on sampling approximation [16] would be desired. As an aside, we note that when applied to nonlinear models, the interacting multiple-model estimator makes the same Gaussian approximation about the target states. Furthermore, the interacting multiple-model estimator approximates a Gaussian mixture with a single Gaussian distribution in the merging step. An application of a similar efficient particle filter

was proposed in [9] to solve the problem of ground target tracking in [4]. An analogous particle filter with similar algorithmic structure and Gaussian approximation was successfully applied in a very different context of multiple target data association [17].

The ground moving target indicator (GMTI) radar can provide detections on moving ground targets over a large region and has become an extremely useful sensor for surveillance of ground targets [1]. A special problem with GMTI tracking of ground targets is that when a target's radial velocity (along the line of sight from the sensor) falls below the minimum detectable velocity, the target cannot be detected by the sensor [5]. Note that the above statement is only approximately true. A more realistic GMTI sensor model characterized by the minimum detectable velocity was discussed in detail in [18], which includes a quantitative treatment of the probability of detection as a continuous function of the target kinematic state and relative target-sensor geometry. In [8], [18], the dependence of the probability of detection on the target state is used as a fictitious measurement in the update of the target state. An evasive target can use the so-called "move-stop-move" strategy, in which it deliberately stops or moves at a very low speed for some time before accelerating again, to avoid detection by the sensor [5]. When the target is actually stopped, prediction with a constant velocity model or maneuver model leads to large errors. The inference of target stop from the event of lack of detection can at best be done in a probabilistic manner because multiple causes exist for the lack of detection. For example, in addition to target stop, the target may not be detected due to the less-than-one probability of detection. In [5], a simple strategy to tracking the move-stop-move maneuver by augmenting the mode set with a stopped target model was proposed. More complex strategy that takes data association into account was presented in [6]. Assuming that the correct data association is known, we apply the strategy of [5] in the context of particle filtering.

The organization of the paper proceeds as follows. First, the dynamics and observation models for road-constrained target are presented. Then the particle filtering theory for jump Markov systems is reviewed. Finally, an efficient particle filter for road-constrained target tracking is derived and compared with the bootstrap filter within the variable structure multiple-model framework.

II. DYNAMICS AND OBSERVATION MODELS

A. Target Dynamics Model

A road is assumed to be represented in the road database by organized endpoints connected with

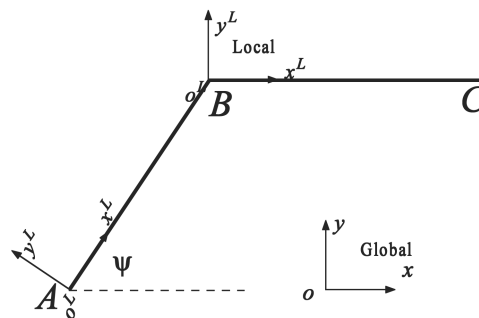


Fig. 1. Global and local coordinate systems.

straight line segments. Also assumed is that the knowledge of the road network in terms of the endpoint positions and the directions and lengths of the road segments is available to the tracker. For sake of simplicity, no road errors are considered in this study. The road information incorporated in the dynamics models imposes constraints on position and moving directions of on-road targets and thus greatly reduces motion uncertainty.

Although the full position and velocity of a road-constrained target or ground moving target is three dimensional, the fact that the target can only move on the terrain surface clearly imposes constraints in the components of the three-dimensional position and velocity. Given the two-dimensional target position on the horizontal plane and the three-dimensional positions of the adjacent points provided by the road database, the three-dimensional position of the target can be constructed using interpolation techniques. The vertical component of the target velocity, though oftentimes insignificant, can also be derived from the horizontal components and the local slope information. Hence, we only need to consider the two-dimensional motion of the target in filter design. The road-constrained target is modeled as a point mass moving in a road network on the horizontal plane. Its acceleration is modeled as a stochastic process.

1) Along- and Perpendicular-to-Road Propagation:

Two kinds of coordinate systems are employed in order to describe the target motion on the horizontal plane, as shown in Fig. 1. The global coordinate system is predetermined. The local coordinate systems are road-segment based. Each local coordinate system is associated with a road segment. The origin (or the reference point) of a local coordinate system is an endpoint of the corresponding segment, and the x- and y-axes of it are aligned along and perpendicular to the road, respectively. Since the global coordinates of the endpoints of the road segments are immediately available from the road database, the relation between global and local coordinates in terms of translation and rotation can be derived with ease. For example, for the local coordinate system fixed on road segment \overrightarrow{AB} of Fig. 1, the two-dimensional rotation matrix

$C(\psi)$ relating the global and local coordinate systems is defined by

$$C(\psi) = \begin{bmatrix} \cos \psi & -\sin \psi \\ \sin \psi & \cos \psi \end{bmatrix} \quad (1)$$

with ψ being the angle between the road direction and the x-axis of the global coordinate system and satisfying

$$\tan \psi = \frac{y_B - y_A}{x_B - x_A} \quad (2)$$

where x_A , x_B , y_A , and y_B are the global coordinates of the two endpoints of the road segment.

A natural decomposition of the road-constrained target motion is the motion along the road and the motion perpendicular to the road, with the dominant motion being the former. The transverse motion (perpendicular to the road) is comparatively insignificant. Roughly speaking, the road-constrained target has only one degree of freedom because of the road constraints. The target is free to cruise, accelerate, or stop along the road, whereas the controlled motion perpendicular to the road is tightly constrained within the boundaries of the road or lanes. Because of the heterogeneous nature of the motion in the two directions, the road-constrained target motion is best described in local coordinate systems. The motion along the road and the motion perpendicular to the road can be respectively modeled as

$$\ddot{x}^L(t) = w_x^L(t) \quad (3)$$

and

$$\ddot{y}^L(t) + 2\xi\omega_n\dot{y}^L(t) + \omega_n^2y^L(t) = w_y^L(t) \quad (4)$$

where x^L and y^L are the local coordinates along and perpendicular to the road, respectively. The noise $w_x^L \sim \mathcal{N}(w_x^L; 0, \sigma_{w_x}^2)$ and $w_y^L \sim \mathcal{N}(w_y^L; 0, \sigma_{w_y}^2)$ are assumed to be Gaussian and white. The notation $\mathcal{N}(x; \hat{x}, \sigma^2)$ denotes a Gaussian distribution of x with mean \hat{x} and variance σ^2 . Large noise levels correspond to high maneuvers. When the damping factor ξ and the frequency ω_n vanish, (4) reduces to

$$\ddot{y}^L(t) = w_y^L(t). \quad (5)$$

The reduced model is critically stable. However, the propagated variance of $y^L(t)$, σ_y^2 , as well as σ_x^2 increases with time. The idea of (4) is to better restrict the motion perpendicular to the road by system dynamics. The two stable poles given by reasonable choice of ω_n and ξ ensure that the off-road displacement and variance owing to initialization, measurement update, and driving process noise is always well bounded. The stationary value of σ_y^2 is a function of ξ , ω_n and the process noise level $\sigma_{w_y}^2$. From the requirement that when there is no measurement update, the stationary standard deviation

σ_y should be of the order of meters (the width of road), the values of $\sigma_{w_y}^2$ and ξ can be determined accordingly.

Note that different second-order models are used for the two orthogonal directions. In a more general setting, the models may even be of different orders. Higher order models for x^L may be used, for example, when the target is deemed highly maneuverable. On the other hand, reduced order models for y^L may be used, for example, when the off-road motion is considered much less significant, the observation accuracy is modest, or the road map data are imprecise. In the limiting case, simply setting $y^L = 0$ for all the time would be sufficient.

Because sensor observations are only available at discrete time instants, discrete-time dynamics models are derived. The continuous-time model given by (3) and (4) is a linear system with Gaussian noise and thus its discrete-time counterpart can be determined analytically. In the conversion we use the approximation that the process noise is piecewise constant over a sampling interval, denoted by T . The resultant discrete-time dynamics model for \mathbf{x}_k^L is given by

$$\mathbf{x}_k^L = \Phi^L \mathbf{x}_{k-1}^L + G^L \mathbf{w}_{k-1}^L \quad (6)$$

with the state vector consisting of the position and velocity components given by

$$\mathbf{x}_k^L = \begin{bmatrix} \mathbf{p}_k^L \\ \mathbf{v}_k^L \end{bmatrix} = \begin{bmatrix} x_k^L \\ y_k^L \\ \dot{x}_k^L \\ \dot{y}_k^L \end{bmatrix} \quad (7)$$

and

$$\Phi^L = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & \phi_{22} & 0 & \phi_{24} \\ 0 & 0 & 1 & 0 \\ 0 & \phi_{42} & 0 & \phi_{44} \end{bmatrix} \quad (8)$$

and

$$G^L = \begin{bmatrix} T^2/2 & 0 \\ 0 & g_{22} \\ T & 0 \\ 0 & g_{42} \end{bmatrix}. \quad (9)$$

The constants ϕ_{22} , ϕ_{24} , ϕ_{42} , ϕ_{44} , g_{22} , and g_{42} are functions of ξ and ω_n and are computed off-line. When $\xi = \omega_n = 0$, that is, the same model is used for both orthogonal directions, the aforementioned matrices become

$$\Phi^L = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (10)$$

and

$$G^L = \begin{bmatrix} T^2/2 & 0 \\ 0 & T^2/2 \\ T & 0 \\ 0 & T \end{bmatrix}. \quad (11)$$

The covariance matrix Q of the anisotropic process noise \mathbf{w}_k^L is defined by

$$Q = \begin{bmatrix} q_{11} & 0 \\ 0 & q_{22} \end{bmatrix}. \quad (12)$$

Roughly speaking, $q_{11} = \sigma_{w_x}^2 T$ and $q_{22} = \sigma_{w_y}^2 T$. The parameter q_{11} is of the order of the square of the maximum acceleration of a dynamics model, while q_{22} is chosen such that the stationary variance of y_k^L is well within the square of the road width. Further tuning may be needed.

Define the state error covariance matrix $P_k^L \triangleq E\{(\mathbf{x}_k^L - \hat{\mathbf{x}}_k^L)(\mathbf{x}_k^L - \hat{\mathbf{x}}_k^L)^T\}$ where E denotes the conditional expectation operator and $\hat{\mathbf{x}}_k^L$ the conditional mean. The dynamics model for the state error covariance matrix P_k^L is given by

$$P_{k+1}^L = \Phi^L P_k^L (\Phi^L)^T + G^L Q (G^L)^T \quad (13)$$

where the superscript T denotes matrix transpose. As an aside, the stationary covariance matrix P_{y_∞} for the direction perpendicular to the road satisfies (ignoring possible correlation between x^L and y^L directions)

$$P_{y_\infty} = \begin{bmatrix} \phi_{22} & \phi_{24} \\ \phi_{42} & \phi_{44} \end{bmatrix} P_{y_\infty} \begin{bmatrix} \phi_{22} & \phi_{42} \\ \phi_{24} & \phi_{44} \end{bmatrix} + q_{22} \begin{bmatrix} g_{22} \\ g_{24} \end{bmatrix} [g_{22} \ g_{24}]. \quad (14)$$

This is used to determine from P_{y_∞} the parameters for the motion perpendicular to the road. The solutions are not unique though.

2) *Local versus Global Coordinates*: To convert between the local coordinates \mathbf{x}_k^L and the global coordinates $\mathbf{x}_k \equiv [\mathbf{p}_k^T, \mathbf{v}_k^T]^T$, the following relations are used:

$$\mathbf{p}_k - \mathbf{p}_k^O = C(\psi_k) \mathbf{p}_k^L \quad (15)$$

$$\mathbf{v}_k = C(\psi_k) \mathbf{v}_k^L \quad (16)$$

and

$$\mathbf{p}_k^L = C^T(\psi_k)(\mathbf{p}_k - \mathbf{p}_k^O) \quad (17)$$

$$\mathbf{v}_k^L = C^T(\psi_k) \mathbf{v}_k \quad (18)$$

where \mathbf{p}_k^O is the global coordinates of the reference endpoint of the current road segment and ψ_k corresponds to the angle between the current associated road segment and the x-axis of the global coordinate system. The relation between the covariances in global coordinates and local coordinates is given by

$$P_k = C(\psi_k) P_k^L C^T(\psi_k) \quad (19)$$

$$P_k^L = C^T(\psi_k) P_k C(\psi_k). \quad (20)$$

The conversion between local and global coordinates can only be carried out after road association is resolved. The road association problem is discussed later in the section.

3) *Multiple Models*: In order to account for the motion uncertainty of the target, multiple models are used for the motion along the road. Only one model is used for the motion perpendicular to the road because the off-road motion uncertainty is tightly controlled and a single model for that direction is adequate.

Three modes of the target are considered: cruise, maneuver, and stop [4]. They are all described using the same form, given by (6). The difference among the motion modes is only quantitative. Large variance q_{11} is used in the maneuver mode; the variance q_{11} in the cruise mode is assigned a small value. For the stopped mode, the velocity is set to zero. In the following, we assume the motion mode state $r = 1$ corresponds to the cruise mode, $r = 2$ corresponds to the maneuver mode, and $r = 3$ corresponds to the stopped mode. According to the idea of the variable-structure multiple-model approach in [5], the cruise and the maneuver modes are active all the time, but the stopped mode is active only when there is no detection. Thus, the stopped mode is added to the active mode set when the target is no longer detected and removed after the target is detected again.

Now assumptions of the mode transition have to be made. The transition of the motion mode r is assumed to occur only at sensor sampling instants and is governed by the transition probability matrix \mathcal{P} , whose elements are defined by

$$p_{ij} = p(r_k = j \mid r_{k-1} = i) \quad (21)$$

where p_{ij} satisfy $\sum_{j=1}^{\mathcal{S}} p_{ij} = 1$, with \mathcal{S} the number of active modes (also the number of columns of the matrix \mathcal{P}). The active mode set may be $\{1, 2\}$ or $\{1, 2, 3\}$. Hence, four transition matrices are needed. For sake of simplicity, they are assumed to be constant. These matrices are calculated based on the sojourn time of the modes [5].

4) *Road Association*: The multiple-model propagation based on (6) only determines how far the target travels along the road (measured from a reference endpoint) and perpendicular to the road for a given sampling period. This process is independent of the road database. It is the road association process that yields the position and velocity of target relative to the road network. The objective of road association is to determine which road segment the target is on. Road association is nontrivial when the target track is not exactly on any road segment and multiple adjacent road segments exist in the neighborhood of the target. Unique road association solution may not exist in difficult cases. Road association is first run at the initialization time, and then performed after every dynamical propagation and measurement update. For

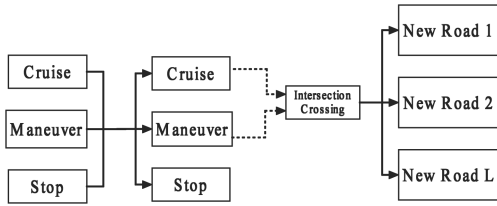


Fig. 2. Along-road propagation.

sake of simplicity, only the two-dimensional position estimate of the target (no three-dimensional position, no velocity) is used in road association.

The initial road association is complex but the basic idea of it is quite simple: given the target position, find all admissible road segments in the neighborhood and then associate the closest road segment(s) to the target. Here the target is treated as a point instead of an uncertainty ellipse and the vector from the target to the road segment is given by \mathbf{d} . The standard distance is then given by $\|\mathbf{d}\|$. The covariance information about the position may also be used to form a weighted distance $\sqrt{\mathbf{d}^T P_d^{-1} \mathbf{d}}$, where $\|\cdot\|$ denotes the vector norm and P_d the covariance matrix of \mathbf{d} . The orientation of the error uncertainty characterized by P_d can preferentially weigh one direction against another.

The road association following dynamical propagation or measurement update is relatively easy. Suppose the target is correctly associated with a road segment \overrightarrow{AB} , and after propagation the along-road coordinate of the target (with respect to endpoint A) is x^L . Then the target is said to remain on the road segment after propagation if $0 \leq x^L \leq |\overrightarrow{AB}|$; it is said to leave the segment via the endpoint A if $x^L < 0$ and via the endpoint B if $x^L > |\overrightarrow{AB}|$. When the target leaves the road segment, the new segment or segments (when A or B is an intersection) it enters is determined by examining the adjoining segment or segments. For a new road segment, the new coordinate x^L is adjusted to $|x^L|$ or $x^L - |\overrightarrow{AB}|$, depending on whether the target leaves the previous segment via A or via B . The adjustment of x^L will be repeated until the target is located within a new road segment. More than one road segment may be passed, especially when the road segments of the database are of small length or high resolution. Because of the road ambiguity at an intersection, the target will end up with more than one destination point after passing the intersection. The along-road propagation incorporating motion uncertainty and road ambiguity is illustrated in Fig. 2 (assuming only one intersection is crossed). Clearly, owing to multiple hypotheses, a single target trajectory is split into several after each sampling period.

The measurement update of the Kalman filter employed in the efficient particle filter corrects the

target position and velocity estimates and may change the associated road segment as well. Because the covariance components for transverse direction are small, the correction in that direction is small too. The along-road correction, however, may be large, for example, as a result of high intensity of the process noise, and may force the target off the current road segment. Therefore, a similar road association procedure based on the along-road correction needs to be performed after the measurement update. More details of the measurement update of the Kalman filter are given in a later section.

The road association result is stored as an auxiliary pointer p_k pointing to the road segment the target is on at time t_k . All the information about the current road segment, such as its endpoints, directions, and neighbors, is indexed in the road database via the pointer p_k . When there is only one new segment to enter, the pointer is updated without ambiguity. However, at an intersection where more than one road meets, all the road hypotheses have to be considered. If no prior knowledge about the route or destination of the target is available, then it is reasonable to assign identical probability to each hypothesis. Suppose the number of roads to enter is L , the probability to enter any road segment is $1/L$. The ambiguity can only be eliminated after new observations are available.

B. Observation Model

A simple GMTI sensor model is used throughout the article. The observations are assumed to be obtained from a GMTI sensor with known position \mathbf{p}_k^S (in terms of global coordinates). It is assumed the data association problem is correctly solved and only the valid tracks are processed. The observation model of a detected target is given by [1]

$$\mathbf{y}_k = \mathbf{h}(\mathbf{x}'_k) + \boldsymbol{\nu}_k \quad (22)$$

$$\mathbf{h}(\mathbf{x}'_k) = \begin{bmatrix} \rho_k \\ \theta_k \\ \dot{\rho}_k \end{bmatrix} = \begin{bmatrix} \sqrt{(x'_k)^2 + (y'_k)^2} \\ \text{ATAN2}(y'_k, x'_k) \\ \frac{x'_k \dot{x}'_k + y'_k \dot{y}'_k}{\sqrt{(x'_k)^2 + (y'_k)^2}} \end{bmatrix} \quad (23)$$

where $\mathbf{x}'_k = \mathbf{x}_k - [(\mathbf{p}_k^S)^T \mathbf{0}_{1 \times 2}]^T$ is the relative position from the sensor to the target, and ρ_k , θ_k , and $\dot{\rho}_k$ denote range, azimuth, and Doppler, respectively. Note that \mathbf{x}_k is the global coordinates of the target state. The function ATAN2 is the four-quadrant arc tangent function. For sake of simplicity, the observation noise is assumed to be Gaussian and white: $\boldsymbol{\nu}_k \sim \mathcal{N}(\boldsymbol{\nu}_k; \mathbf{0}_{3 \times 1}, R)$. For the above model, the likelihood function $p(\mathbf{y}_k | \mathbf{x}_k) = p(\mathbf{y}_k | \mathbf{x}'_k) = \mathcal{N}(\mathbf{y}_k; \mathbf{h}(\mathbf{x}'_k), R)$. Also note that the GMTI observations are functions of the two-dimensional position and velocity only. The

vertical position and velocity components cannot be inferred from the above measurements directly.

A problem associated with the GMTI radar is that the target will not be detected when it is stopped or its radial velocity is smaller than the minimum detectable velocity. (The GMTI radar alone would have difficulty tracking a very slow, undetectable target for a long time.) Another important cause for the lack of detection is that the probability of detection P_D of the sensor is less than one. That is, even if the target moves at a velocity above the minimum detectable velocity, it is possible that the target is not detected due to the imperfectness of the detection process. While the first cause is deterministic, the second occurs randomly.

In addition to the stopped mode, there are other deterministic causes of lack of detection. When a single GMTI sensor is the sole information source and the target moves perpendicular to the line of sight (LOS) from the sensor at the sampling instants, the motion cannot be detected due to the relative geometry. The target is also impossible to be detected when it is hidden from the field of view of the sensor, for example, the target entered a tunnel and stays there. The moving target not being detected due to small radial velocity or terrain obscuration will be referred to as singular cases.

The event of no detection can be regarded as a generalized observation in the sense that the mode of the target can be inferred in a probabilistic manner even if the target is not detected. A generalized likelihood function incorporating the detection event is then used [5]. The probability associated with the detection event is summarized as follows

$$\begin{aligned} \Pr(\text{detected} \mid \text{stopped}) &= 0 \\ \Pr(\text{detected} \mid \text{moving normally}) &= P_D \\ \Pr(\text{detected} \mid \text{moving perpendicular to LOS}) &= 0 \\ \Pr(\text{detected} \mid \text{hidden from the sensor}) &= 0 \end{aligned} \quad (24)$$

and

$$\begin{aligned} \Pr(\text{undetected} \mid \text{stopped}) &= 1 \\ \Pr(\text{undetected} \mid \text{moving normally}) &= 1 - P_D \\ \Pr(\text{undetected} \mid \text{moving perpendicular to LOS}) &= 1 \\ \Pr(\text{undetected} \mid \text{hidden from the sensor}) &= 1. \end{aligned} \quad (25)$$

According to this model, the probability of detection is either 0 or P_D (assumed to be constant in the simulations). In [8], [18], a refined model of the probability of detection P_D is given:

$$P_D = p_d \left[1 - \exp \left(-\log 2 \left(\frac{n_c}{v_m} \right)^2 \right) \right] \quad (26)$$

where p_d includes the directivity pattern and propagation effects, n_c is the difference of the radial velocities of the target and the surrounding mainlobe clutter relative to the sensor, and v_m is the minimum detectable velocity. Here P_D is a continuous function in $[0, p_d]$. When $n_c \gg v_m$, $P_D = p_d$; when $|n_c| < v_m$, $P_D < p_d/2$; when $|n_c| = 0$, $P_D = 0$. This model is different from the former model mainly in the low radial velocity region. In the particle filters of this paper, P_D is only used to update the associated weights of the particles. Thus, the difference due to different models of probability of detection is quantitative. In the analysis below and the simulations, the simpler model given by (24) and (25) is used.

Of most interest is how to infer from “no detections” whether the target is stopped. Under our assumptions, the target cannot be stopped if detected by the sensor. When lack of detection is observed for n consecutive sampling intervals, it is very unlikely that the target is in “normal” motion because $(1 - P_D)^n$, the probability of a “normally” moving target not being detected for n consecutive intervals, is very small (assuming P_D is close to one). In contrast with that, the probability that the target is stopped is $1^n = 1$. The difference between $(1 - P_D)^n$ and 1 helps distinguish between nondetection due to less-than-one probability of detection and nondetection due to target being stopped. In the singular cases, however, the stopped mode is not distinguishable from the moving modes in general.

The first singular case can be eliminated if more than one GMTI sensor or other sensors are available. In practice, the probability of the occurrence of this case is not high because of the motion of the GMTI sensor [5]. When the target enters a long tunnel, however, there is no way to predict whether it will be stopped or keep moving on. The only thing the tracker can do is wait until the target reappears. When multiple targets are involved in this singular case, the previous tracks are likely to get lost. Very limited information about the target motion is contained in the events of no detection alone. When the singular cases cannot be eliminated and both moving and stopped mode hypotheses have to be maintained, the ambiguity about the target mode dramatically increases and the inference becomes much more inaccurate.

III. PARTICLE FILTERS FOR JUMP MARKOV SYSTEMS

A general jump Markov system can be described by the following state-space model:

$$\mathbf{x}_k \sim p(\mathbf{x}_k \mid \mathbf{x}_{k-1}, r_k) : \quad \mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}, r_k, \mathbf{w}_{k-1}) \quad (27)$$

$$\mathbf{y}_k \sim p(\mathbf{y}_k \mid \mathbf{x}_k, r_k) : \quad \mathbf{y}_k = \mathbf{h}(\mathbf{x}_k, r_k, \mathbf{v}_k) \quad (28)$$

$$r_k \sim p(r_k \mid r_{k-1}) \quad (29)$$

where the hybrid state vector consists of the base state \mathbf{x}_k and the mode state r_k . The base state vector \mathbf{x}_k and the observation vector \mathbf{y}_k are continuous-valued; the mode state r_k is discrete-valued and the discrete values are denoted by $\{1, 2, \dots, S\}$. The dynamics function $\mathbf{f}(\cdot)$ and the observation function $\mathbf{h}(\cdot)$ may be nonlinear. The process noise \mathbf{w}_k and the observation noise ν_k are assumed to be white noise. When \mathbf{f} and \mathbf{h} are linear and $\mathbf{w}_k(r_k) \sim \mathcal{N}(\mathbf{w}_k(r_k); \mathbf{0}, Q(r_k))$, $\nu_k(r_k) \sim \mathcal{N}(\nu_k(r_k); 0, R(r_k))$ as well as the initial base state $\mathbf{x}_0 \sim \mathcal{N}(\mathbf{x}_0; \hat{\mathbf{x}}_0, P_0)$ are Gaussian noise, the general jump Markov system reduces to the jump Markov linear Gaussian system, given by

$$\mathbf{x}_k = \Phi(r_k)\mathbf{x}_{k-1} + G(r_k)\mathbf{w}_{k-1}(r_k) \quad (30)$$

$$\mathbf{y}_k = H(r_k)\mathbf{x}_k + \nu_k(r_k) \quad (31)$$

$$r_k \sim p(r_k | r_{k-1}). \quad (32)$$

The Markov transition model $r_k \sim p(r_k | r_{k-1})$ may be generalized to $r_k \sim p(r_k | \mathbf{R}_{k-1}, \mathbf{Y}_{k-1})$, where $\mathbf{R}_k \equiv \{r_1, \dots, r_k\}$ and $\mathbf{Y}_k \equiv \{\mathbf{y}_1, \dots, \mathbf{y}_k\}$.

The objective of particle filtering for jump Markov systems is to estimate recursively the joint posterior distribution $p(\mathbf{X}_k, \mathbf{R}_k | \mathbf{Y}_k)$ or $p(\mathbf{x}_k, r_k | \mathbf{Y}_k)$ with $\mathbf{X}_k \equiv \{\mathbf{x}_1, \dots, \mathbf{x}_k\}$. Of practical interest is the marginal distribution $p(\mathbf{x}_k | \mathbf{Y}_k)$, obtained from $p(\mathbf{x}_k, r_k | \mathbf{Y}_k)$ by standard marginalization. With N weighted particles $\{r_k^{(i)}, \mathbf{x}_k^{(i)}, w_k^{(i)}\}_{i=1}^N$, they are approximated as [14]

$$p(\mathbf{x}_k, r_k | \mathbf{Y}_k) \approx \sum_{i=1}^N w_k^{(i)} \delta_{\mathbf{x}_k^{(i)}, r_k^{(i)}}(d\mathbf{x}_k, r_k) \quad (33)$$

and

$$p(\mathbf{x}_k | \mathbf{Y}_k) \approx \sum_{i=1}^N w_k^{(i)} \delta_{\mathbf{x}_k^{(i)}}(d\mathbf{x}_k) \quad (34)$$

where δ is the point-mass delta function.

The particles and their associated weights are recursively updated by 1) sampling $r_k^{(i)}$ and $\mathbf{x}_k^{(i)}$ from a certain importance function $q(\mathbf{x}_k, r_k | \mathbf{X}_{k-1}^{(i)}, \mathbf{R}_{k-1}^{(i)}, \mathbf{Y}_k)$ and 2) updating the importance weights according to

$$w_k^{(i)} \propto w_{k-1}^{(i)} \frac{p(\mathbf{y}_k | \mathbf{x}_k^{(i)}, r_k^{(i)})p(\mathbf{x}_k^{(i)} | \mathbf{x}_{k-1}^{(i)}, r_k^{(i)})p(r_k^{(i)} | r_{k-1}^{(i)})}{q(\mathbf{x}_k^{(i)}, r_k^{(i)} | \mathbf{X}_{k-1}^{(i)}, \mathbf{R}_{k-1}^{(i)}, \mathbf{Y}_k)}. \quad (35)$$

A resampling step that eliminates particles with low importance weights and multiplies particles with high importance weights should be added in the filter cycle in order to suppress the degeneracy problem [1].

The bootstrap filter for jump Markov systems corresponds to the special choice of the importance function

$$q(\mathbf{x}_k, r_k | \mathbf{X}_{k-1}^{(i)}, \mathbf{R}_{k-1}^{(i)}, \mathbf{Y}_k) = p(\mathbf{x}_k | \mathbf{x}_{k-1}^{(i)}, r_k^{(i)})p(r_k | r_{k-1}^{(i)}). \quad (36)$$

The importance weights are then updated using

$$w_k^{(i)} \propto w_{k-1}^{(i)} p(\mathbf{y}_k | \mathbf{x}_k^{(i)}, r_k^{(i)}). \quad (37)$$

The bootstrap filter makes few assumptions about the state-space model and employs little of the structure of jump Markov systems. The sampling scheme as shown above is simple to implement but can be inefficient. The efficient particle filter for general jump Markov systems presented in [16] greatly improves the efficiency of sampling $r_k^{(i)}$ and $\mathbf{x}_k^{(i)}$ by using an importance function that makes better use of the Markov structure. That is, because r_k can only take on a finite number of values and therefore the mode transition from $r_{k-1}^{(i)}$ to r_k can only have a finite number of possibilities, the marginalization over the mode subspace as required by a better importance function that can be implemented with summation.

When the system is a jump Markov linear Gaussian one, the benign structure makes it possible to design more efficient particle filter based on Rao-Blackwellization. The idea of Rao-Blackwellized particle filtering is to reduce the sampling space as much as possible by analytic marginalization. For jump Markov linear Gaussian systems, the technique is based on a partition of the joint distribution, given by [14]:

$$p(\mathbf{X}_k, \mathbf{R}_k | \mathbf{Y}_k) = p(\mathbf{X}_k | \mathbf{R}_k, \mathbf{Y}_k)p(\mathbf{R}_k | \mathbf{Y}_k). \quad (38)$$

Because $p(\mathbf{X}_k | \mathbf{R}_k, \mathbf{Y}_k)$ is exactly Gaussian in this case, it can be sufficiently represented by its mean and covariance. (This statement does not hold any more when r_k is dependent on \mathbf{x}_{k-1} .) Given \mathbf{R}_k and \mathbf{Y}_k , the conditional mean and covariance can be determined in closed form using Kalman filtering. Thus, only the mode distribution $p(\mathbf{R}_k | \mathbf{Y}_k)$ needs to be estimated using particle filtering. The marginal distribution $p(\mathbf{R}_k | \mathbf{Y}_k)$ satisfies a recursion [14]:

$$\begin{aligned} p(\mathbf{R}_k | \mathbf{Y}_k) &= \frac{p(\mathbf{y}_k | \mathbf{Y}_{k-1}, \mathbf{R}_k)p(\mathbf{R}_k | \mathbf{Y}_{k-1})}{p(\mathbf{y}_k | \mathbf{Y}_{k-1})} \\ &= \frac{p(\mathbf{y}_k | \mathbf{Y}_{k-1}, \mathbf{R}_k)p(r_k | r_{k-1})}{p(\mathbf{y}_k | \mathbf{Y}_{k-1})} p(\mathbf{R}_{k-1} | \mathbf{Y}_{k-1}) \end{aligned} \quad (39)$$

where $p(\mathbf{R}_k | \mathbf{Y}_{k-1}) = p(r_k | r_{k-1})p(\mathbf{R}_{k-1} | \mathbf{Y}_{k-1})$ is used. The density $p(\mathbf{y}_k | \mathbf{Y}_{k-1}, \mathbf{R}_k)$ can be regarded as the ‘‘likelihood’’ of a mode state sequence and can be analytically determined. Based on the recursion, a particle filter scheme for $p(r_k | \mathbf{Y}_k)$ can be designed, in which the filtering distribution $p(r_k | \mathbf{Y}_k)$ is approximated with the weighted particles $\{r_k^{(i)}, w_k^{(i)}\}_{i=1}^N$ as

$$p(r_k | \mathbf{Y}_k) \approx \sum_{i=1}^N w_k^{(i)} \delta_{r_k^{(i)}}(r_k) \quad (40)$$

and the filtering distribution $p(\mathbf{x}_k | \mathbf{Y}_k)$ is then approximated as a Gaussian mixture

$$\begin{aligned} p(\mathbf{x}_k | \mathbf{Y}_k) &\approx \sum_{i=1}^N w_k^{(i)} p(\mathbf{x}_k | \mathbf{R}_k^{(i)}, \mathbf{Y}_k) \\ &= \sum_{i=1}^N w_k^{(i)} \mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_k^{(i)}, P_k^{(i)}) \end{aligned} \quad (41)$$

where $p(\mathbf{x}_k | \mathbf{R}_k^{(i)}, \mathbf{Y}_k)$, the posterior distribution of \mathbf{x}_k for an individual mode sequence $\mathbf{R}_k^{(i)}$, is exactly Gaussian and the associated mean $\hat{\mathbf{x}}_k^{(i)}$ and covariance $P_k^{(i)}$ are determined using Kalman filtering. Since the distribution $p(\mathbf{x}_k | \mathbf{R}_k^{(i)}, \mathbf{Y}_k)$ is exactly determined (equivalent to using an infinite number of samples of \mathbf{x}_k), the approximation of the filter only comes from the particle representation of $p(r_k | \mathbf{Y}_k)$.

For jump Markov linear Gaussian systems, samples of r_k can be drawn from the optimal importance function, given by [14]

$$p(r_k | \mathbf{R}_{k-1}, \mathbf{Y}_k) = \frac{p(\mathbf{y}_k | r_k, \mathbf{R}_{k-1}, \mathbf{Y}_{k-1}) p(r_k | r_{k-1})}{p(\mathbf{y}_k | \mathbf{R}_{k-1}, \mathbf{Y}_{k-1})}. \quad (42)$$

Then the associated normalized importance weights are [14]

$$\begin{aligned} w_k^{(i)} &\propto w_{k-1}^{(i)} \frac{p(\mathbf{y}_k | r_k^{(i)}, \mathbf{R}_{k-1}^{(i)}, \mathbf{Y}_{k-1}) p(r_k^{(i)} | r_{k-1}^{(i)})}{p(r_k^{(i)} | \mathbf{R}_{k-1}^{(i)}, \mathbf{Y}_k)} \\ &= w_{k-1}^{(i)} p(\mathbf{y}_k | \mathbf{R}_{k-1}^{(i)}, \mathbf{Y}_{k-1}) \end{aligned} \quad (43)$$

where

$$p(\mathbf{y}_k | \mathbf{R}_{k-1}^{(i)}, \mathbf{Y}_{k-1}) = \sum_{j=1}^S p(\mathbf{y}_k | j, \mathbf{R}_{k-1}^{(i)}, \mathbf{Y}_{k-1}) p(j | r_{k-1}^{(i)}). \quad (44)$$

The running variable j in the above equation refers to $r_k = j$. In the summation,

$$\begin{aligned} p(\mathbf{y}_k | j, \mathbf{R}_{k-1}^{(i)}, \mathbf{Y}_{k-1}) \\ = \mathcal{N}(\mathbf{y}_k; H(j)\hat{\mathbf{x}}_k^{-(i,j)}, H(j)P_k^{-(i,j)}H^T(j) + R(j)) \end{aligned} \quad (45)$$

with $\hat{\mathbf{x}}_k^{-(i,j)}$ and $P_k^{-(i,j)}$ being the mean and covariance of the prediction $p(\mathbf{x}_k | j, \mathbf{R}_{k-1}^{(i)}, \mathbf{Y}_{k-1})$, respectively.

Because the importance weights $p(\mathbf{y}_k | \mathbf{R}_{k-1}^{(i)}, \mathbf{Y}_{k-1})$ do not depend on r_k , it is possible to select the fittest particle trajectories $\mathbf{R}_{k-1}^{(i)}$ based on \mathbf{y}_k before samples of $r_k^{(i)}$ are drawn.

Because random samples are used to represent $p(r_k | \mathbf{Y}_k)$ while exact means and covariances are used to represent $p(\mathbf{x}_k | \mathbf{R}_k, \mathbf{Y}_k)$, the full representation for the distribution of the hybrid state is composed of $\{r_k^{(i)}, \hat{\mathbf{x}}_k^{(i)}, P_k^{(i)}, w_k^{(i)}\}_{i=1}^N$, where $\hat{\mathbf{x}}_k^{(i)}$ and $P_k^{(i)}$ are deterministically updated given $r_k^{(i)}$.

IV. EFFICIENT PARTICLE FILTER FOR ROAD-CONSTRAINED TARGET TRACKING

In this section, two variable-structure multiple-model particle filters for road-constrained target tracking are presented. The bootstrap filter serves as the baseline algorithm and the proposed efficient particle filter is compared with it. Both filters are designed based on the same assumptions and models.

A. Bootstrap Filter

The joint distribution of \mathbf{x}_k and r_k is approximated by weighted particles $\{r_k^{(i)}, \mathbf{x}_k^{(i)}, w_k^{(i)}\}_{i=1}^N$. The pointers $p_k^{(i)}$ to the present road segment are also assigned to particles. Thus, the full particle representation is given by $\{r_k^{(i)}, \mathbf{x}_k^{(i)}, p_k^{(i)}, w_k^{(i)}\}_{i=1}^N$. The outline of a filter cycle of the bootstrap filter for road-constrained target tracking is given in Table I. The mean of the target state is given by

$$\hat{\mathbf{x}}_k = \sum_{i=1}^N w_k^{(i)} \mathbf{x}_k^{(i)}. \quad (46)$$

It provides limited information about the position and velocity of the target when the particles are associated with road segments of different directions.

B. Efficient Particle Filter

1) *Particle Filtering Algorithm:* The efficient particle filter for road-constrained target tracking is designed based on the optimal particle filtering theory for jump Markov linear Gaussian systems. Analytic approximation is made for the target state distribution:

$$p(\mathbf{x}_k | \mathbf{R}_k^{(i)}, \mathbf{Y}_k) \approx \mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_k^{(i)}, P_k^{(i)}) \quad (47)$$

where the mean $\hat{\mathbf{x}}_k^{(i)}$ and covariance $P_k^{(i)}$ are updated using unscented Kalman filtering. For nonlinear filtering problems, when the parameters of the unscented transformation are appropriately tuned, the unscented Kalman filter can yield better estimation results than the extended Kalman filter. The details of the unscented Kalman filter can be found in [15]. Approximation is needed mainly because the observation model is nonlinear and hence the conditional distribution $p(\mathbf{x}_k | \mathbf{R}_k^{(i)}, \mathbf{Y}_k)$ is not strictly Gaussian. The road constraints may also make the distribution non-Gaussian. The likelihood $p(\mathbf{y}_k | r_k, \mathbf{R}_{k-1}, \mathbf{Y}_{k-1})$ used for recursive sampling of r_k is also calculated based on Gaussian approximation. That is,

$$\begin{aligned} p(\mathbf{y}_k | r_k, \mathbf{R}_{k-1}^{(i)}, \mathbf{Y}_{k-1}) &\approx p(\mathbf{y}_k | r_k, \hat{\mathbf{x}}_{k-1}^{(i)}, P_{k-1}^{(i)}) \\ &\approx \mathcal{N}(\mathbf{y}_k; \hat{\mathbf{y}}_k^{(i)}, P_{y_k}^{(i)}). \end{aligned} \quad (48)$$

Given r_k , $\hat{\mathbf{x}}_{k-1}^{(i)}$, and $P_{k-1}^{(i)}$, the mean $\hat{\mathbf{y}}_k^{(i)}$ and covariance $P_{y_k}^{(i)}$ of \mathbf{y}_k are estimated using standard unscented transformation.

TABLE I
Bootstrap Filter for Road-Constrained Target Tracking

For $i = 1, \dots, N$,

determine the active motion mode set for i th particle at t_k

determine the transition probability matrix $\mathcal{P}_k^{(i)}$

sample

$$r_k^{(i)} \sim P(r_k | r_{k-1}^{(i)})$$

propagate $\mathbf{x}_{k-1}^{(i)}$ through the model specified by mode $r_k^{(i)}$ and road segment $p_{k-1}^{(i)}$ to generate $\mathbf{x}_k^{*(i,l)}$, where $l = 1, \dots, L^{(i)}$, with $L^{(i)}$ the number of feasible road segments at time t_k . (if the target does not cross an intersection, $L^{(i)} = 1$; if the target crosses an intersection, $L^{(i)} > 1$.)

if $L^{(i)} > 1$, draw $\mathbf{x}_k^{(i)}$ from $\mathbf{x}_k^{*(i,l)}$ randomly (the probability of $\mathbf{x}_k^{*(i,l)}$ is $1/L^{(i)}$).

determine the road pointer $p_k^{(i)}$ according to $\mathbf{x}_k^{(i)}$.

For $i = 1, \dots, N$,

evaluate the likelihood $\Lambda_k^{(i)}$:

if the target is not detected

if $r_k^{(i)} = 3$, $\Lambda_k^{(i)} = 1$

if $r_k^{(i)} = 1$ or 2 AND the target is hidden or moves perpendicular to the line of sight, $\Lambda_k^{(i)} = 1$

if $r_k^{(i)} = 1$ or 2 AND the target is in normal motion, $\Lambda_k^{(i)} = 1 - P_D$

if the target is detected

if $r_k^{(i)} = 3$ OR the target is hidden or moves perpendicular to the line of sight, $\Lambda_k^{(i)} = 0$

if $r_k^{(i)} = 1$ or 2, $\Lambda_k^{(i)} = p(\mathbf{y}_k | \mathbf{x}_k^{(i)}, r_k^{(i)})$

evaluate and normalize the importance weights

$$w_k^{(i)} \propto w_{k-1}^{(i)} \Lambda_k^{(i)}$$

Resampling: multiply/discard particles $\{r_k^{(i)}, \mathbf{x}_k^{(i)}, p_k^{(i)}, w_k^{(i)}\}_{i=1}^N$ with respect to high/low importance weights $w_k^{(i)}$ to obtain N new particles $\{r_k^{(i)}, \mathbf{x}_k^{(i)}, p_k^{(i)}, w_k^{(i)}\}_{i=1}^N$ with equal weights.

The full particle representation is given by $\{r_k^{(i)}, \hat{\mathbf{x}}_k^{(i)}, p_k^{(i)}, p_k^{(i)}, w_k^{(i)}\}_{i=1}^N$, where $\hat{\mathbf{x}}_k^{(i)}$ and $p_k^{(i)}$ are deterministically updated given $r_k^{(i)}$ and $p_k^{(i)}$. The outline of a filter cycle of the efficient particle filter for road-constrained target tracking is given in Table II. The mean of the target state is given by

$$\hat{\mathbf{x}}_k = \sum_{i=1}^N w_k^{(i)} \hat{\mathbf{x}}_k^{(i)}. \quad (49)$$

It provides limited information about the position and velocity of the target when the particles are associated with road segments of different directions.

2) *Kalman Measurement Update*: The efficient particle filter for road-constrained target tracking is an efficient particle filter for motion mode estimation coupled with multiple Kalman filters (unscented filters) for conditional target state estimation, with each Kalman filter corresponding to a mode sample sequence. In the Kalman filter, the time update starts with the propagation of the mean and covariance in local coordinates, followed by road association. Because the dynamics model is linear, the time update given by (6) and (13) is identical to that of the linear Kalman filter. No unscented transformation is needed. Unscented transformation is used in measurement update to deal with the nonlinearity in the observation model. Before measurement update is performed, the mean and covariance in local coordinates are

converted into global coordinates in order to apply the Kalman measurement update formulae. It is in the local-to-global conversion that road association is required. The Kalman measurement update based on the unscented transformation is essentially an approximation to linear minimum mean-square error estimation [2]. The approximation here comes from the nonlinearity of the observation model.

To put the measurement update as an optimization problem, the cost function the Kalman measurement update attempts to minimize may be written as [2]

$$J_k(\mathbf{x}_k) = \frac{1}{2}(\mathbf{x}_k - \hat{\mathbf{x}}_k^-)^T (P_k^-)^{-1} (\mathbf{x}_k - \hat{\mathbf{x}}_k^-) + \frac{1}{2}(\mathbf{y}_k - \mathbf{h}(\mathbf{x}_k))^T R_k^{-1} (\mathbf{y}_k - \mathbf{h}(\mathbf{x}_k)) \quad (50)$$

where all quantities are expressed with respect to the same global coordinate system. The Kalman measurement update works well in many cases but may be problematic when the consecutive road segments change directions sharply (not necessarily at an intersection), because the update does not automatically satisfy the road constraints.

A more reasonable measurement update that takes road constraints into consideration is formulated as an unconstrained minimization problem that minimizes an alternative cost function, given by

$$J_k(\mathbf{x}_k, \mathbf{x}_k^L) = \frac{1}{2}(\mathbf{x}_k^L - \hat{\mathbf{x}}_k^{L-})^T (P_k^{L-})^{-1} (\mathbf{x}_k^L - \hat{\mathbf{x}}_k^{L-}) + \frac{1}{2}(\mathbf{y}_k - \mathbf{h}(\mathbf{x}_k))^T R_k^{-1} (\mathbf{y}_k - \mathbf{h}(\mathbf{x}_k)) \quad (51)$$

TABLE II
Efficient Particle Filter for Road-Constrained Target Tracking

For $i = 1, \dots, N$,
determine the active motion mode set for i th particle at t_k
determine the transition probability matrix $\mathcal{P}_k^{(i)}$
For $j = 1, \dots, S$, where S is the number of active motion modes (hypotheses),
propagate $\hat{\mathbf{x}}_{k-1}^{(i)}$ and $P_{k-1}^{(i)}$ through the model specified by mode j and road segment $p_{k-1}^{(i)}$ to generate $\hat{\mathbf{x}}_k^{*(i,j,l)}$ and $P_k^{*(i,j,l)}$, where
 $l = 1, \dots, L^{(i,j)}$ (if the target does not cross an intersection, $L^{(i,j)} = 1$; if the target crosses an intersection, $L^{(i,j)} > 1$); determine the
road pointer $p_k^{*(i,j,l)}$ according to $\hat{\mathbf{x}}_k^{*(i,j,l)}$ and $P_k^{*(i,j,l)}$
For $j = 1, \dots, S$ and for $l = 1, \dots, L^{(i,j)}$, evaluate the likelihood $\Lambda_k^{(i,j,l)}$:
if the target is not detected
if $j = 3$, $\Lambda_k^{(i,j,l)} = 1$
if $j = 1$ or 2 AND the target is hidden or moves perpendicular to the line of sight, $\Lambda_k^{(i,j,l)} = 1$
if $j = 1$ or 2 AND the target is in normal motion, $\Lambda_k^{(i,j,l)} = 1 - P_D$
if the target is detected
if $j = 3$ OR the target is hidden or moves perpendicular to the line of sight, $\Lambda_k^{(i,j,l)} = 0$
if $j = 1$ or 2 , compute $\Lambda_k^{(i,j,l)} = \hat{p}(\mathbf{y}_k | \hat{\mathbf{x}}_k^{*(i,j,l)}, P_k^{*(i,j,l)})$ based on Gaussian approximation
For $i = 1, \dots, N$, compute

$$w_k^{(i)} \propto w_{k-1}^{(i)} \sum_{j=1}^S \sum_{l=1}^{L^{(i,j)}} \Lambda_k^{(i,j,l)} p(j | r_{k-1}^{(i)}) / L^{(i,j)}$$

Resampling: multiply/discard the particle set $\{r_{k-1}^{(i)}, \hat{\mathbf{x}}_k^{*(i,j,l)}, P_k^{*(i,j,l)}, p_k^{*(i,j,l)}, \Lambda_k^{(i,j,l)}\}_{i=1}^N$ with respect to high/low importance weights $w_k^{(i)}$ to obtain N new $\{r_{k-1}^{(i)}, \hat{\mathbf{x}}_k^{*(i,j,l)}, P_k^{*(i,j,l)}, p_k^{*(i,j,l)}, \Lambda_k^{(i,j,l)}\}_{i=1}^N$ with equal weights.

For $i = 1, \dots, N$, sample

$$(r_k^{(i)}, l^{(i)}) \sim \hat{p}(\mathbf{y}_k | \hat{\mathbf{x}}_k^{*(i,r_k,l)}, P_k^{*(i,r_k,l)}) p(r_k | r_{k-1}^{(i)}) / L^{(i,j)}$$

where $\hat{p}(\mathbf{y}_k | \hat{\mathbf{x}}_k^{*(i,j,l)}, P_k^{*(i,j,l)}) = \Lambda_k^{(i,j,l)}$; set

$$\hat{\mathbf{x}}_k^{-(i)} = \hat{\mathbf{x}}_k^{*(i,r_k,l^{(i)})}, \quad P_k^{-(i)} = P_k^{*(i,r_k,l^{(i)})}, \quad p_k^{(i)} = p_k^{*(i,r_k,l^{(i)})}$$

For $i = 1, \dots, N$, update $\hat{\mathbf{x}}_k^{(i)}$, $P_k^{(i)}$ from $\hat{\mathbf{x}}_k^{-(i)}$, $P_k^{-(i)}$ based on Gaussian approximation, and update $p_k^{(i)}$ according to $\hat{\mathbf{x}}_k^{(i)}$ and $P_k^{*(i,j,l)}$

where \mathbf{x}_k^L and \mathbf{x}_k are the local and global representations of the same target state vector. The conversion between them is available given correct road association.

The difference $(\mathbf{x}_k^L - \hat{\mathbf{x}}_k^{L-})$ is not defined as the usual straight line vector pointing from $\hat{\mathbf{x}}_k^{L-}$ to \mathbf{x}_k^L . The position part of the difference is a curved line consisting of perpendicular-to-road and along-road segments and determined from the road-based route connecting the two points. The velocity part is given by the difference between the along-road and transverse velocities. (We assume that the route is unique.) The relations $\|\mathbf{x}_k^L - \hat{\mathbf{x}}_k^{L-}\| = \|\mathbf{x}_k - \hat{\mathbf{x}}_k^-\|$ and $(\mathbf{x}_k^L - \hat{\mathbf{x}}_k^{L-})^T (P_k^{L-})^{-1} (\mathbf{x}_k^L - \hat{\mathbf{x}}_k^{L-}) = (\mathbf{x}_k - \hat{\mathbf{x}}_k^-)^T (P_k^-)^{-1} \cdot (\mathbf{x}_k - \hat{\mathbf{x}}_k^-)$ hold when \mathbf{x}_k (or \mathbf{x}_k^L) is associated with the same road as the prediction $\hat{\mathbf{x}}_k^-$ (or $\hat{\mathbf{x}}_k^{L-}$).

No explicit inequality constraints are included in the above minimization problem, although the target is "road constrained." With those inequality constraints, a complicated programming problem has to be solved. Explicit road constraints in terms of inequality constraints are not required in general, as long as the prediction $\hat{\mathbf{x}}_k^L$ and P_k^{L-} is a faithful representation of a road-constrained target, i.e., the position estimate is on or very close to the road,

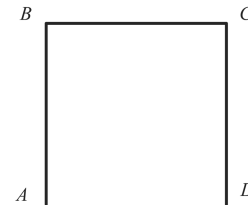


Fig. 3. One-dimensional measurement update example.

the velocity estimate is almost parallel to the road direction, and the off-road variance is small. Tight control provided by the prediction covariance over corrections in the off-road direction ensures that the main correction to the predicted position and velocity is along the road. The minimization problem is complex because the road as well as the target route may include road segments with very different directions and thus cannot be treated as a straight line.

Take the road ABCD in Fig. 3 as a simplified example, in which the global coordinates of A, B, C, and D are $(0, -l)$, $(0, 0)$, $(l, 0)$, and $(l, -l)$, respectively, where l is comparable to the prediction and measurement covariances. For sake of simplicity, a one-dimensional, linear estimation problem is considered. Suppose the target is always exactly on

the road and the observation is the target Cartesian position converted from the range and azimuth measurements. The Doppler measurement is not used. When the transverse displacement is small compared with the along-road displacement and the covariance constraint for that direction is tight, the reduced problem is a reasonable approximation. Now suppose the target prediction in local coordinates $(\hat{x}, \hat{\dot{x}})$ is associated with road segment \overrightarrow{AB} , and the global coordinates of the observation are (x_m, y_m) . Obviously, the observation does not obey any road constraints. The prediction covariance matrix in local coordinates and the measurement covariance matrix in global coordinates are given by

$$\begin{bmatrix} p_{11} & p_{12} \\ p_{12} & p_{22} \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} r_{11} & r_{12} \\ r_{12} & r_{22} \end{bmatrix}$$

respectively. We have the following.

1) The cost function (51) for \overrightarrow{AB} is given by

$$\begin{aligned} J_{AB} = & \frac{1}{2} [x - \hat{x} \quad \dot{x} - \hat{\dot{x}}] \begin{bmatrix} p_{11} & p_{12} \\ p_{12} & p_{22} \end{bmatrix}^{-1} \begin{bmatrix} x - \hat{x} \\ \dot{x} - \hat{\dot{x}} \end{bmatrix} \\ & + \frac{1}{2} [-x_m \quad x - y_m - l] \begin{bmatrix} r_{11} & r_{12} \\ r_{12} & r_{22} \end{bmatrix}^{-1} \begin{bmatrix} -x_m \\ x - y_m - l \end{bmatrix} \end{aligned} \quad (52)$$

where x is measured from A and \dot{x} is along \overrightarrow{AB} .

2) For \overrightarrow{BC} the cost function is given by

$$\begin{aligned} J_{BC} = & \frac{1}{2} [x - \hat{x} + l \quad \dot{x} - \hat{\dot{x}}] \begin{bmatrix} p_{11} & p_{12} \\ p_{12} & p_{22} \end{bmatrix}^{-1} \begin{bmatrix} x - \hat{x} + l \\ \dot{x} - \hat{\dot{x}} \end{bmatrix} \\ & + \frac{1}{2} [x - x_m \quad -y_m] \begin{bmatrix} r_{11} & r_{12} \\ r_{12} & r_{22} \end{bmatrix}^{-1} \begin{bmatrix} x - x_m \\ -y_m \end{bmatrix} \end{aligned} \quad (53)$$

where x is measured from B and \dot{x} is along \overrightarrow{BC} .

3) For \overrightarrow{CD} the cost function is given by

$$\begin{aligned} J_{CD} = & \frac{1}{2} [x - \hat{x} + 2l \quad \dot{x} - \hat{\dot{x}}] \begin{bmatrix} p_{11} & p_{12} \\ p_{12} & p_{22} \end{bmatrix}^{-1} \begin{bmatrix} x - \hat{x} + 2l \\ \dot{x} - \hat{\dot{x}} \end{bmatrix} \\ & + \frac{1}{2} [l - x_m \quad x - y_m] \begin{bmatrix} r_{11} & r_{12} \\ r_{12} & r_{22} \end{bmatrix}^{-1} \begin{bmatrix} l - x_m \\ x - y_m \end{bmatrix} \end{aligned} \quad (54)$$

where x is measured from C and \dot{x} is along \overrightarrow{CD} .

In the second and third cost functions, the first terms contributed by the prediction are written as if the predicted target is on the extended line of \overrightarrow{BC} and \overrightarrow{CD} , respectively. The Kalman measurement update formulae provide “local” solutions to the above minimization problems. The globally optimal update is one of the three local solutions.

1) For AB :

$$\hat{x}^+ = (r^* \hat{x} + p_{11} x_m^*) / (r^* + p_{11}) \quad (55)$$

$$\hat{\dot{x}}^+ = \hat{\dot{x}} + (\hat{x}^+ - \hat{x}) p_{12} / p_{11} \quad (56)$$

$$J_{AB}^{\min} = (\hat{x} - x_m^*)^2 / (r^* + p_{11}) \quad (57)$$

$$x_m^* = y_m + l - r_{12} / r_{11} x_m \quad (58)$$

$$r^* = r_{22} - r_{12}^2 / r_{11}. \quad (59)$$

2) For BC :

$$\hat{x}^+ = [r^* (\hat{x} - l) + p_{11} x_m^*] / (r^* + p_{11}) \quad (60)$$

$$\hat{\dot{x}}^+ = \hat{\dot{x}} + (\hat{x}^+ - \hat{x}) p_{12} / p_{11} \quad (61)$$

$$J_{BC}^{\min} = (l - \hat{x} + x_m^*)^2 / (r^* + p_{11}) \quad (62)$$

$$x_m^* = x_m - r_{12} / r_{22} y_m \quad (63)$$

$$r^* = r_{11} - r_{12}^2 / r_{22}. \quad (64)$$

3) For CD :

$$\hat{x}^+ = [r^* (\hat{x} - 2l) + p_{11} x_m^*] / (r^* + p_{11}) \quad (65)$$

$$\hat{\dot{x}}^+ = \hat{\dot{x}} + (\hat{x}^+ - \hat{x}) p_{12} / p_{11} \quad (66)$$

$$J_{CD}^{\min} = (2l - \hat{x} + x_m^*)^2 / (r^* + p_{11}) \quad (67)$$

$$x_m^* = y_m - r_{12} / r_{11} (x_m - l) \quad (68)$$

$$r^* = r_{22} - r_{12}^2 / r_{11}. \quad (69)$$

For this example, the following comments are made.

1) Although expected to be in the neighborhood of the prediction, the globally optimal update is not always associated with the same road segment the prediction is associated with. The global minimum of the cost function is determined by comparing the local minima for \overrightarrow{AB} , \overrightarrow{BC} , and \overrightarrow{CD} . The multiple local minima are a direct result of the road constraints, not of nonlinearity of the observation model (in this example, the observation model is exactly linear). If the three road segments are almost parallel to one another, then there is only one minimum, which is given by the Kalman measurement update.

2) Because a road segment is of finite length, the unconstrained local minimum for a road segment provided by the Kalman measurement update may be beyond the endpoints. A local minimum outside the road segment cannot be used as it violates the road constraints. Because the cost function of this linear example monotonically increases from the unconstrained local minimum to the points on the road, the on-road minimum is located at the closer endpoint to the unconstrained minimum. (If the Doppler is also used, which is nonlinear in x or \dot{x} , then the statement may not strictly hold. The endpoint, however, can still serve as a reasonable approximation to the local minimum.)

3) The search for the global minimum of the cost function involves all possible road segments

in the neighborhood (in the example, three). The procedure can be complex. If one is satisfied with a local minimum that obeys the road constraints, the measurement update can be much simplified. The outline of the approximate scheme is as follows. Suppose the state update will be associated with the same road segment as the prediction, and use the standard Kalman measurement update to compute the state update. Then based on the along-road position component, check if the result is really within the road segment. If the update is within the road segment, then it is qualified as a local minimum. If not, choose the closer endpoint as the interim minimum, select the adjoining road segment, and rewrite the prediction mean and covariance for the new road segment. Again, compute the local minimum using the Kalman measurement update and check if the solution is admissible. The procedure is repeated until a local minimum satisfying the road constraints (which can be an endpoint) is obtained.

For the original two-dimensional measurement update problem, a similar scheme for global or local minimum can be used.

V. SIMULATION RESULTS

The main objective of this section is to demonstrate the feasibility and efficiency of the proposed efficient particle filter. The main simulation parameters are given as follows: the sampling interval is 5 s; the standard deviations of the range, azimuth, and Doppler are 20 m, 0.01 rad, and 1 m/s, respectively; the probability of detection is $P_D = 0.85$; the minimum detectable velocity is 2 m/s. For sake of simplicity, the GMTI sensor is assumed to be stationary at $[-1000 \text{ m}, 1000 \text{ m}]$, denoted by the circle in Fig. 4. A road network used for simulation is illustrated in Fig. 4. The route of the target is from segment \overrightarrow{AB} to segment \overrightarrow{DE} via \overrightarrow{BC} and \overrightarrow{CD} . The points B, C, and D are intersections. The modes of target motion alternate between acceleration, cruise, deceleration, and stop. Only one target is considered. If the correct data association is known, the tracking of multiple targets reduces to the separate tracking of individual targets. The equivalent target positions in Cartesian coordinates directly converted from the range ρ_k and azimuth θ_k observations of a typical run are plotted in Fig. 4 as the dots in the neighborhood of the road. At positions where the velocity of the target is smaller than the minimum detectable velocity, the target is not detected for deterministic reasons. The target may still be undetected at other points due to $P_D < 1$.

The results of a typical run of the efficient particle filters with 50 particles are presented in Figs. 5 and 6. The filter shows good tracking ability. The rms error of the estimated position over this run is about 20 m.

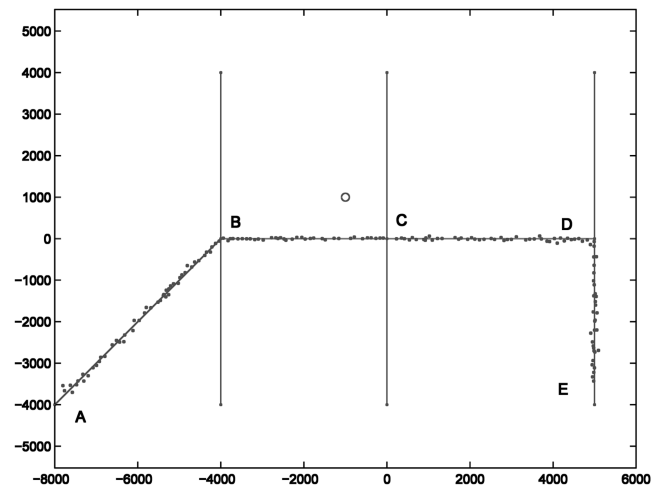


Fig. 4. Road network.

The rms velocity error is about 2.4 m/s. In Fig. 5, the position errors (relative to the true positions) of the position observations and the position estimates given by the particle filter are also compared. Among others, it can be seen that in the position observations the errors perpendicular to the road are quite large, with the peak value around 180 m. For this reason, the position estimates in the off-road direction are tightly constrained from tracking the off-road observations too closely. In our particle filters, such constraint is imposed with dynamics models that are strictly stable in the direction perpendicular to the road, for example, with $\xi = 1$ and $\omega_n = 0.2$, so that the particle cloud is restricted well within the road. The spikes in the position and velocity errors are unlikely to be avoided because a tracker only works well in the mean sense. Here are some typical cases.

- 1) The filters are usually tuned so that a target stop can be quickly detected from lack of detection. However, if during a number of (e.g., 3) consecutive sampling intervals the target is accelerating from the stopped mode but no observations are available due to $P_D < 1$, the tracker tends not to respond and the delayed response gives rise to spikes.
- 2) If at a sampling point the target is crossing the intersection but no observations are available or the observations are close to the intersection, the tracker cannot effectively eliminate the road ambiguity around the intersection until more postcrossing observation data are processed. But before that it has to maintain all the hypotheses of the road segments around the intersection, which again leads to spikes in estimation errors. The spike around time $t = 590$ s corresponds to this case.
- 3) Observation outliers and the singular cases can also lead to imprecise judgment on target maneuvers.

Now the performance of the efficient particle filter with 50 particles is compared with that of the bootstrap filter with 1000 particles. The estimation

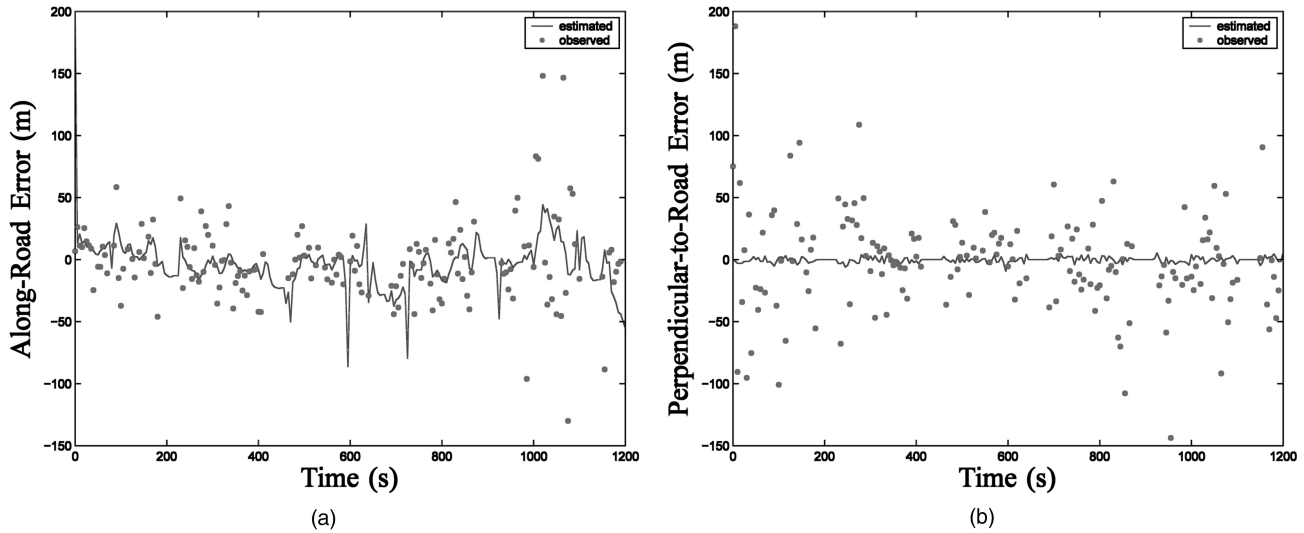


Fig. 5. Time history of position errors.

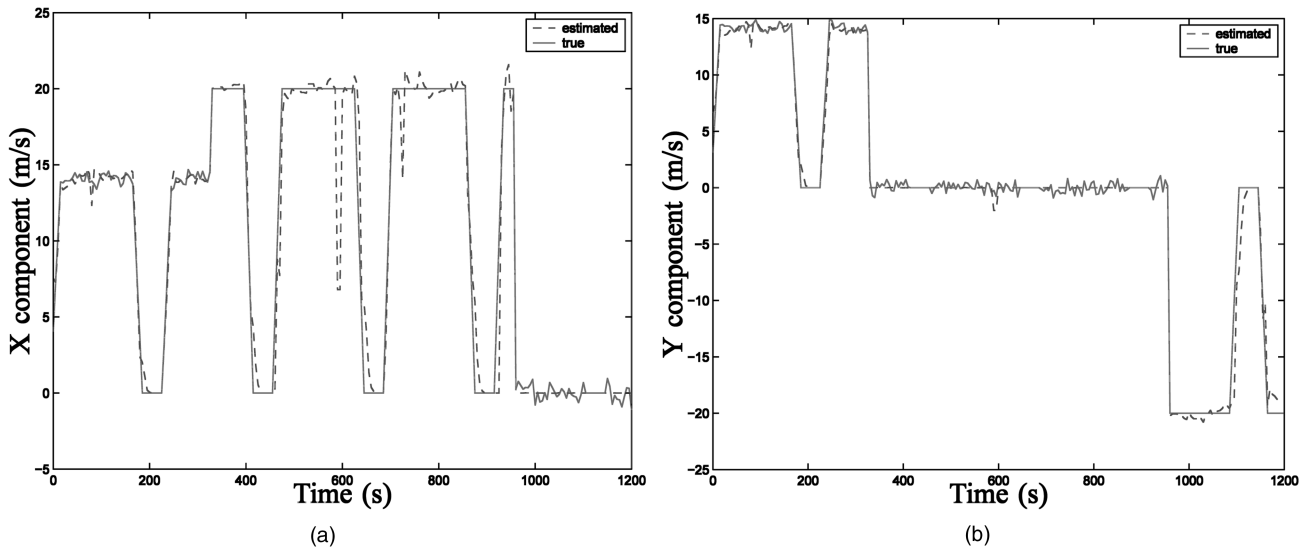


Fig. 6. Time history of true and estimated velocities.

accuracy is measured by rms errors over 50 runs. The MATLAB function CPUTIME that “returns the CPU time in seconds that has been used by the MATLAB process since MATLAB started” is used to roughly measure the computational expense of the algorithms. The rms errors of the efficient particle filter with 50 particles are 18.4 m and 1.91 m/s. The rms errors of the bootstrap filter with 1000 particles are 19.5 m and 1.96 m/s. The efficient particle filter outperforms the bootstrap filter in position estimation in 36 runs, in velocity estimation in 32 runs, and in both in 25 runs. The average CPUTIME for the efficient particle filter is about 42 s and that for the bootstrap filter is about 150 s. The average CPUTIME taken by the bootstrap filter with 1000 particles is about four times longer than that taken by the efficient particle filter with 50 particles. From the above comparisons, a modest conclusion can be drawn that the efficient particle

filter with 50 particles can achieve performance similar to that of the bootstrap filter with 1000 particles with much less computational expense.

VI. CONCLUSIONS

For road-constrained targets, the incorporation of road information into the dynamics models can greatly reduce the target motion uncertainty. A variable-structure multiple-model framework is used to address target maneuvers along the road. The proposed efficient particle filter is an approximation to the optimal particle filter for jump Markov linear Gaussian systems. The main approximation of the filter is the Gaussian assumption about the conditional target state distribution given a mode sequence and observations. The efficient particle filter with 50 particles yields satisfactory simulation results.

Compared with the standard bootstrap filter, the proposed efficient particle filter involves much less computation for similar accuracy and robustness. Future work should investigate the data association and filtering problems simultaneously.

REFERENCES

- [1] Ristic, B., Arulampalam, S., and Gordon, N.
Beyond the Kalman Filter: Particle Filters for Tracking Applications.
Boston, MA: Artech House, 2004.
- [2] Bar-Shalom, Y., Li, X. R., and Kirubarajan, T.
Estimation with Applications to Tracking and Navigation: Theory, Algorithms and Software.
New York: Wiley, 2001.
- [3] Kreucher, C., and Kastella, K.
Multiple model nonlinear filtering for low signal ground target applications.
IEEE Transactions on Aerospace and Electronic Systems, **41**, 2 (Apr. 2005), 549–564.
- [4] 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.
- [5] Kirubarajan, T., and Bar-Shalom, Y.
Tracking evasive move-stop-move targets with a GMTI radar using a VS-IMM estimator.
IEEE Transactions on Aerospace and Electronic Systems, **39**, 4 (Oct. 2003), 1452–1457.
- [6] Lin, L., Bar-Shalom, Y., and Kirubarajan, T.
New assignment-based data association for tracking move-stop-move targets.
IEEE Transactions on Aerospace and Electronic Systems, **40**, 2 (Apr. 2004), 714–725.
- [7] Shea, P. J., Zadra, T., Klamer, D., Frangione, E., and Brouillard, R.
Improved state estimation through use of roads in ground tracking.
In O. E. Drummond (Ed.), *Proceedings of Signal and Data Processing of Small Targets*, vol. 4048, SPIE, 2000, 321–332.
- [8] Ulmke, M.
Improved GMTI-tracking using roadmaps and topographic information.
In O. E. Drummond (Ed.), *Proceedings of Signal and Data Processing of Small Targets*, vol. 5204, SPIE, 2003, 143–154.
- [9] Payne, O., and Marrs, A.
An unscented particle filter for GMTI tracking.
In *2004 IEEE Aerospace Conference Proceedings*, vol. 3, 2004, 1869–1875.
- [10] Agate, C., and Sullivan, K. J.
Particle filtering algorithm for tracking multiple road-constrained targets.
In *Proceedings of SPIE Conference on Signal Processing, Sensor Fusion, and Target Recognition XII*, vol. 5096, Apr. 2003, 256–266.
- [11] Agate, C. S., and Sullivan, K. J.
Road-constrained target tracking and identification using a particle filter.
In O. E. Drummond (Ed.), *Proceedings of Signal and Data Processing of Small Targets*, vol. 5204, 2003, 532–543.
- [12] Doucet, A., de Freitas, N., and Gordon, N.
Sequential Monte Carlo Methods in Practice.
New York: Springer-Verlag, 2001.
- [13] Doucet, A., Gordon, N. J., and Krishnamurthy, V.
Particle filters for state estimation of jump Markov linear systems.
IEEE Transactions on Signal Processing, **49**, 3 (Mar. 2001), 613–624.
- [14] de Freitas, N., Dearden, R., Hutter, F., Morales-Menéndez, R., Mutch, J., and Poole, D.
Diagnosis by a waiter and a Mars explorer.
In *Proceedings of the IEEE*, **92**, 3 (Mar. 2004), 455–468.
- [15] Julier, S. J., and Uhlmann, J. K.
Unscented filtering and nonlinear estimation.
In *Proceedings of the IEEE*, **92** (Mar. 2004), 401–422.
- [16] Andrieu, C., Davy, M., and Doucet, A.
Efficient particle filtering for jump Markov systems. Application to time-varying autoregressions.
IEEE Transactions on Signal Processing, **51**, 7 (July 2003), 1762–1770.
- [17] Särkkä, S., Vehtari, A., and Lampinen, J.
Rao-Blackwellized Monte Carlo data association for multiple target tracking.
In *Proceedings of FUSION 2004*, The 7th International Conference on Information Fusion, Stockholm, June 2004.
- [18] Koch, W.
Ground target tracking with stap-radar: Selected tracking aspects.
In R. Klemm (Ed.), *Applications of Space-Time Adaptive Processing*, London: Institution of Electrical Engineers, 2004.



Yang Cheng received his B.E., M.E., and Ph.D. degrees in aerospace engineering from Harbin Institute of Technology, Harbin, China, in 1997, 1999, and 2003, respectively.

He is currently a research assistant professor of the Department of Mechanical and Aerospace Engineering, University at Buffalo, State University of New York. His research interests include target tracking, data fusion, spacecraft attitude estimation, relative navigation, sensor calibration, and data assimilation.



Tarunraj Singh received his B.E., M.E., and Ph.D. degrees in mechanical engineering from Bangalore University, Indian Institute of Science, and the University of Waterloo, respectively.

He was a Postdoctoral Fellow in the Aerospace Engineering Department of Texas A & M University prior to starting his tenure at the University at Buffalo in 1993, where he is currently a professor in the Department of Mechanical and Aerospace Engineering. He was a von Humboldt Fellow and spent his sabbatical at the Technische Universität Darmstadt in Germany and at the IBM Almaden Research center in 2000–2001. He was a NASA Summer Faculty Fellow at the Goddard Space Flight Center in 2003. His research is supported by the National Science Foundation, AFOSR, NSA, Office of Naval Research and various industries including MOOG Inc., Praxair and Delphi Thermal Systems. His research interests are in robust vibration control, optimal control, nonlinear estimation and intelligent transportation.

Dr. Singh has published over 100 refereed journal and conference papers and has presented over 30 invited seminars at various universities and research laboratories.