



Multitarget Localization on Road Networks with Hidden Markov Rao–Blackwellized Particle Filters

Nisar Ahmed*

University of Colorado Boulder, Colorado 80309

David Casbeer†

U.S. Air Force Research Laboratory, Wright–Patterson Air Force Base, Ohio 45433

Yongcan Cao‡

University of Texas at San Antonio, San Antonio, Texas 78249

and

Derek Kingston†

U.S. Air Force Research Laboratory, Wright–Patterson Air Force Base, Ohio 45433

DOI: 10.2514/1.1010539

This paper considers the problem of tracking multiple moving targets on a road network with sparse, highly localized, unattended ground sensor data that are subject to clutter and missed detections. Hidden Markov models for single-target localization with unattended ground sensor data are first derived for road networks, under the assumption of perfect data association. These hidden Markov models are then used to solve the data association problem in the presence of clutter and missed detections for multitarget tracking using a Rao–Blackwellized particle filter. The proposed hidden Markov model tracking approach permits easy generation of accurate probabilistic models from a priori road network structure information, and it naturally enables sparse computationally efficient handling of multimodal target state uncertainties using both positive and negative unattended ground sensor information. The Rao–Blackwellized particle filter provides a fully Bayesian solution to the data association problem, enabling exploration of the association hypotheses space that leverages the computational advantages of exact hidden Markov model inference for multimodal state estimation. Numerical simulations demonstrate the effectiveness of the hidden Markov model/Rao–Blackwellized particle filter on challenging multitarget tracking scenarios with high false-alarm and missed detection rates.

I. Introduction

UNMANNED aerial vehicles (UAVs) and unattended ground sensors (UGSs) have become vital tools for enhancing situational awareness in a variety of applications. Large-scale integrated surveillance, target acquisition, and reconnaissance missions form an important family of applications for networked UAVs and UGSs in which timely information about adversarial activities, uncertain environmental conditions, and friendly asset status are essential to support coordinated decision making in complex dynamic environments. With expanded deployment of autonomous UAV and UGS networks, robustness and performance limitations arising from practical constraints must be tackled with formal mathematical insights about uncertain dynamical systems.

Recent work has focused on control and sensing strategies for the UAV–UGS road reconnaissance (U2R2) problem [1], in which a small autonomous UAV must use a set of stationary UGSs deployed along a large uncontrolled road network to detect and localize potentially hostile dynamic intruders (Fig. 1). The UAV must autonomously plan and execute a tour of the UGSs to gather information that will allow it to then take a limited number of high-resolution images of intruders at their most likely locations. Although the UAV can communicate wirelessly with each UGS to extract and fuse signals from the ground, it cannot process high-resolution images on board and must deliver these to a base station for analysis. Due to payload restrictions and size, weight, and power constraints, the UAV cannot carry any other onboard sensors to observe the road network. The battery-powered UGSs can monitor a variety of signals (seismic, Doppler, magnetic, and infrared) but only locally store data over short time windows following trigger events. Because the UGSs have restricted communication ranges and lines of sight (due to concealment), they are unable to send data directly to each other or to the base station. The UAV must therefore act as a mobile data fusion center for UGS nodes in communication range.

The U2R2 problem has four operational phases that are handled autonomously by the UAV/UGS system. In the patrol phase, the UAV repeatedly visits and interrogates UGS nodes to gather evidence for possible intruder target detections. Upon confirmed detection, the UAV moves into the isolation phase to localize and capture informative images of the intruder targets along the road network. In the delivery and response phases, the UAV sends data back to the base station for further analysis and target confirmation, and it receives further target monitoring instructions from human supervisors. It is assumed here that a known number of targets has been detected and that the UAV is in the isolation phase for all targets.

The UAV must infer the locations of all dynamic targets solely on the basis of UGS data, even though these only cover an extremely limited portion of the road network; are obtained at different partially overlapping time intervals; and will generally be collected out of order relative to actual visitation times by targets. The UGS data available to the UAV are therefore sparse, asynchronous, and disordered. There are three challenging issues at the core of the underlying data fusion problem for multitarget localization:

1) The first problem is dynamic target modeling. How can stochastic target motion on a road network be accurately modeled and estimated online from sparse UGS data by a small UAV with limited computing abilities?

Received 13 February 2017; revision received 22 July 2017; accepted for publication 15 August 2017; published online 26 September 2017. Copyright © 2017 by the American Institute of Aeronautics and Astronautics, Inc. All rights reserved. All requests for copying and permission to reprint should be submitted to CCC at www.copyright.com; employ the ISSN 2327-3097 (online) to initiate your request. See also AIAA Rights and Permissions www.aiaa.org/randp.

*Assistant Professor, Smead Aerospace Engineering Sciences. Member AIAA.

†Senior Research Engineer, Aerospace Vehicle Systems. Member AIAA.

‡Assistant Professor, Electrical and Computer Engineering. Member AIAA.

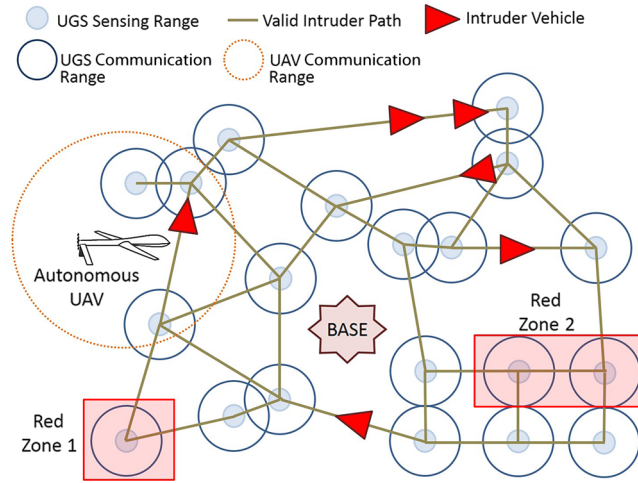


Fig. 1 Illustration of the U2R2 problem.

2) The second problem is data association. How can UGS data be associated and fused among multiple target tracks in the presence of clutter (false alarms) and missed detections by the UGS?

3) The third problem is disordered data fusion. How can out-of-order and asynchronous UGS data be fused online for multiple target tracks?

Most previous work on UAV-enabled road target localization assumes that the UAV is equipped with onboard sensors that provide dense data streams for persistent target monitoring (e.g., radar, IR, and video). As such, problem 3 has largely been overlooked, whereas the analogs of problems 1 and 2 are often handled, respectively, by conventional recursive “predict/update” tracking algorithms and data association techniques. The present work focuses on problems 1 and 2, and it provides a firm theoretical foundation to derive principled tractable solutions to problem 3.

This paper derives novel methods for dynamic target modeling and multitarget data association in the U2R2 problem. It is shown that dynamic Bayesian estimation algorithms based on discrete time hidden Markov models (HMMs) provide highly practical and theoretically sound solutions to both issues. The highly structured nature of HMMs allows multistep target state prediction and filtering operations with multiple UGSs to be carried out very quickly through sparse matrix operations. These properties allow the challenging multitarget data association problem to be solved efficiently via the Rao–Blackwellized particle filter (RBPF). Numerical simulations on challenging U2R2 tracking scenarios support two major conclusions:

1) HMMs provide a very good approximation of target dynamics for the U2R2 problem.

2) The newly proposed HMM/RBPF procedure generally produces accurate and reliable multitarget state estimates with sparse UGS data, even in the presence of heavy clutter and multiple missed detections.

This work provides several significant contributions on top of previous work presented by the authors in [2], namely, 1) an improved and more accurate HMM/UGS positive and negative sensor data model, which allows for much more accurate target localization in the perfect and imperfect data association cases; 2) a formal treatment and derivation of an RBPF-based solution to the multitarget tracking problem for a known number of targets; and 3) thorough numerical simulation studies to assess the efficacy of the proposed HMM and HMM/RBPF approaches for localizing multiple targets simultaneously in different road networks and sensing conditions (with variable clutter and missed detection rates).

The remainder of this paper is organized as follows. Section II formally defines the U2R2 tracking problem and reviews related prior work. Section III shows how probabilistic discrete time Markov chain models can be automatically identified to provide good approximations of continuous stochastic target motion on a known road network. Section IV considers the fusion of negative and positive UGS data with unknown target labels for target state estimation via the single-target HMM Bayes filter and multitarget HMM/Rao–Blackwellized (HMM/RBPF). Section V presents the numerical simulation results and discussion for several simulated U2R2 scenarios with varying road network geometries and clutter conditions. Section VI concludes the paper.

II. Problem Formulation

This section formally defines the road network target state dynamics and UGS observation models for the U2R2 problem. These models are oriented toward previous work on UAV control and planning described in [1,3], which only required knowledge of the two-dimensional target position and heading on a known map. Additional information pertaining to target velocity, size, type, lane position, etc.; or environmental features (e.g., terrain features, road conditions, etc.) is not considered. In the following, $p(\cdot)$ describes a probability density function (PDF) and $P(\cdot)$ describes a probability mass function.

A. Road Network Dynamics Model and Single-Target State Specification

Figure 2 shows a road network for the U2R2 problem modeled as a graph $\mathcal{G}_R = (\mathcal{N}, \mathcal{E})$ consisting of a finite set of nodes \mathcal{N} and edges \mathcal{E} . Each edge $r \in \mathcal{E}$, $i = 1, \dots, N_e$ represents one of N_e roads in the network. Each road edge r defines a single direction of travel for a single target between the start and nodes $n_s(r)$, $n_e(r) \in \mathcal{N}$, which represent intersections. The state of a single target on edge r is given by its position x and velocity v , which evolve according to a stochastic discrete time state transition function f_r :

$$\begin{aligned} [x_{i+1}, v_{i+1}]^T &= f_r(x_i, v_i, \epsilon_i) \\ \epsilon_i &\sim p_r(\epsilon_i) \end{aligned} \quad (1)$$

where ϵ_i represents discrete time process noise on edge r drawn according to probability distribution $p_r(\epsilon_i)$. The simplest motion model along r consists of one-dimensional (1-D) stochastic kinematics, where $v_i = \epsilon_i$,

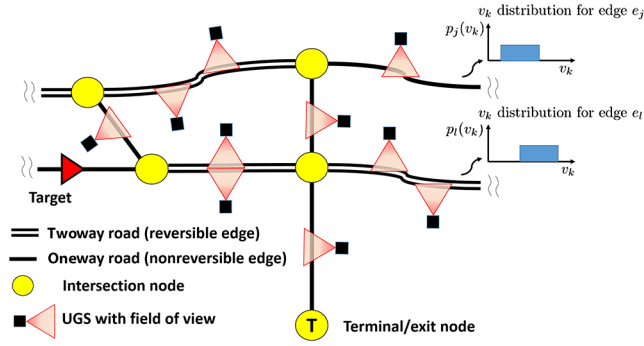


Fig. 2 Graph model segment for a portion of a large road network.

$$\begin{aligned} x_{i+1} &= x_i + v_i \Delta t, \\ v_i &\sim p_i(v_i) \end{aligned} \quad (2)$$

where $x_i \in [0, 1]$ is the (normalized) 1-D position of the target along edge r at discrete time step $i = 0, 1, 2, \dots$ (where $x_i = 0$ indicates the start of the edge), v_i is the (normalized) discrete time random velocity drawn from probability distribution $p_i(v_i)$, and Δt is the discrete time step. It is assumed here that $p_r(v_i) = \mathcal{U}(v_{\min,r}, v_{\max,r})$; i.e., a uniform distribution defined by upper- and lower-bound velocities $0 \leq v_{\min,r} \leq v_{\max,r}$, although other velocity distributions based on truncated normals, exponentials, etc., can also be considered.

If the target is on edge r , it is assumed to move in a single direction with a randomly changing but bounded velocity: it cannot perform U turns and change travel direction in the middle of a road. To model the target's possible motion along the same road in the other direction [i.e., after possibly performing a U turn at end node $n_e(r)$], another parallel "reverse" edge r' can be added with the same velocity bounds but with the implied direction of travel reversed by interchanging the start and end nodes. Thus, the discrete time state for a single target is fully described by the hybrid state vector $\xi(i) = [e_i, x_i]^T$, where the discrete part $e_i \in \mathcal{E}$ indicates which edge the target is on and $x_i \in [0, 1]$ indicates the target's continuous location along e_i .

All edges are connected by intersection nodes $n_h \in \mathcal{N}$, $h = 1, \dots, N_n$, which define the discrete time conditional probabilities of 1) transitioning from the end of one road edge to the start of a new road edge, i.e., $P(e_{i+1} = j | e_i = i, x_i = 1)$ (where $j \neq i$); 2) performing a U turn along the same road edge, i.e., $P(e_{i+1} = r | e_i = i, x_i = 1)$; or 3) terminating at the node and "exiting" the road network, i.e., $P(e_{i+1} = T | e_i = i, x_i = 1)$, where $e_{i+1} = T$ denotes the terminal escape/exit state for which $P(e_{i+1} = T | e_i = T) = 1$ and $P(e_{i+1} = r | e_i = T) = 0$ for all $r \neq T$. Note that, if $\mathcal{E}(n_h)$ denotes the set of edges connected to node n_h , then

$$\sum_{(j) \in \mathcal{E}(n_h)} P(e_{i+1} = j | e_i = r, x_i = 1) = 1$$

for all $r \in \mathcal{E}(n_h)$.

B. UGS Data

UGSs are fixed at N_u locations $\sigma(u) = [e^u, x^u]^T$ along the road network, where e^u and x^u are the edge and location along the edge for sensor $u \in \{1, \dots, N_u\}$. Because targets are measured briefly and infrequently, UGS data are stored locally and reported in the form of event logs. Therefore, a large amount of negative information is implicitly recorded in between sparse positive measurement events. However, the UGSs are also subject to missed detections (false negatives) and false alarms (false positives), which are indistinguishable from true negatives or true positives in the recorded data.

In practical applications, a UGS can record a wide variety of data, including target position and velocity information. For ease of derivation, it is assumed here that each UGS either explicitly logs a detection event or a nondetection event at time i ; that is, each sensor u records whether or not a target has passed by the vicinity of $\sigma(u)$ as the random variable:

$$o_i^u \in \{0 = \text{No Detection}, 1 = \text{Detect}\}$$

The statistical characteristics of these data are characterized by the sensor's false-alarm density function $\rho^u(e^u, x^u)$ and probability of detection function $d^u(e^u, x^u)$ over some nominal field of view of $f(u) = [e^u, x^{u,-}, x^{u,+}]^T$ along the road network adjacent to $\sigma(u)$, where $x^{u,-}$ and $x^{u,+}$ denote the lower and upper extents of sensor coverage along road e^u (see Fig. 2). It is assumed that the fields of view for all UGSs are well separated and nonoverlapping; each UGS senses only its local edge, although each edge can have multiple UGSs. Both $\rho^u(e^u, x^u)$ and $d^u(e^u, x^u)$ can be integrated over $f(u)$ to obtain the false-alarm probability P_{FA}^u and average detection rate P_D^u for sensor u . These values can be used to formally define the UGS likelihood function $P(o_i^u | \xi)$, which will be discussed later in Sec. IV. Let $O_i = [o_i^1, \dots, o_i^{N_u}]^T$ be the stacked vector of UGS observations at i , which has the joint likelihood function

$$P(O_i | \xi_k) = \prod_{u=1}^{N_u} P(o_i^u | \xi_k) \quad (3)$$

where each o_i^u is conditionally independent of all others, given ξ_k . Note that both $P(o_i^u | \xi_k)$ and $P(O_i | \xi_k)$ only assume the presence of a single target for now. The extension of these likelihoods to account for target identities in the multitarget case is considered in Sec. IV.

C. Bayesian Data Fusion and Target State Estimation

Given an initial state PDF of $p(\xi_0) = p([e_0, x_0]^T)$ for a single target and a sequence of associated UGS measurements of $O_{1:k} = \{O_1, \dots, O_k\}$ up to time k , it is desired to fuse all information to update statistical beliefs about the target's state at any particular point in time i ; i.e., for times

$0 \leq i < k$ (smoothing or retrodiction), at time $i = k$ (filtering), or for future times $i > k$ (prediction). The updated beliefs can then be used to derive Bayes's optimal target state estimates according to an appropriate loss function, e.g., the mean squared error, infinity norm, etc. [4]. The beliefs can be computed recursively online via Bayesian inference.

For filtering (the focus of this work), beliefs are updated recursively through a series of time prediction updates based on the dynamic road network model:

$$p(\xi_i | O_{1:i-1}) = p([e_i, x_i]^T | O_{1:i-1}) = \int_{\xi_{i-1}} p(\xi_i | \xi_{i-1}) p(\xi_{i-1} | O_{1:i-1}) d\xi_{i-1} \quad (4)$$

and Bayesian measurement updates based on UGS data models, which give the posterior conditional PDF $p(\xi_i | O_{1:i})$ at time i :

$$p(\xi_i | O_{1:i}) = \frac{p(\xi_i | O_{1:i-1}) P(O_i | \xi_i)}{\int_{\xi_i} p(\xi_i | O_{1:i-1}) P(O_i | \xi_i) d\xi_i} \quad (5)$$

where $p(\xi_{i-1} | O_{1:i-1}) = p(\xi_0)$ for $i = 1$. The conditional PDF $p(\xi_i | \xi_{i-1})$ is the state transition probability kernel from ξ_{i-1} to ξ_i for all $\xi_i \in \mathcal{E} \times [0, 1]$; this is defined directly by the dynamics model [Eq. (2)] for a particular road network. The abused operator

$$\int_{\xi_i} (\cdot) d\xi_i$$

marginalizes over the hybrid state, i.e., summation over e_i and integration over x_i . The observation likelihood $P(O_i | \xi_i)$ for the Bayes update is defined by Eq. (3), which assumes the ideal case where all UGS measurements are available together at fusion time. This can occur in practice if UGS data are reported synchronously for online fusion (e.g., if the UAV and all UGSs are always in communication range, or if data relays are present) or if data are postprocessed together for delayed/offline fusion after asynchronous data collection. Asynchronous and out-of-order fusion is also possible [5–10], but it is beyond the scope of this work and not considered here.

For pure forward prediction with a T -step lookahead, the state PDF at time $i + T > i$ is computed as follows:

$$p(\xi_{i+T} | O_{1:i}) = \int_{\xi_{i+T-1}} \int_{\xi_{i+T-2}} \cdots \int_{\xi_i} \prod_{k=1}^T p(\xi_{k+i} | \xi_{k+i-1}) p(\xi_i | O_{1:i}) d\xi_i \cdots d\xi_{i+T-2} d\xi_{i+T-1} \quad (6)$$

Similarly, the state smoothing/retrodiction PDF of $p(\xi_{i-T} | O_{1:i})$ for $i - T < i$ is found via recursive backward corrections to the Bayes filter PDFs between times i and $i - T$ so that past state beliefs are updated by future information.

Because

$$p(\xi_i | O_{1:i}) = P(e_i | O_{1:i}) p(x_i | e_i, O_{1:i})$$

describes a hybrid (mixed discrete and continuous) state vector, Bayesian inference operations for state PDF updates are intractable. In particular, Eq. (4) leads to an exponentially growing sum of complicated truncated PDFs because each transition kernel must account for the possibility that $e_i = j \neq T$ switches to $e_i = j' \neq j$ any time the probability of $x_i = 1$ is strictly greater than zero (i.e., whenever it is possible the target has reached an intersection node in the network). Given the sparse and intermittent nature of the UGS observations, the PDFs required for inference quickly become multimodal, and thus highly non-Gaussian [3]. Furthermore, measurement updates require Bayesian inference on continuous x_i states using discrete “detect”/“no-detect” observations, which yield analytically intractable hybrid inference calculations [11].

These challenges directly extend to the general case of multitarget state inference, where multiple PDFs over states ξ_i^t are updated for $t \in \{1, \dots, N_t\}$ and N_t is the number of (known) targets on the road network. Because the UGSs cannot identify individual targets and are vulnerable to false alarms (clutter) as well as missed detections, the multitarget case introduces the challenging problem of data association, i.e., assignment of each $o_i^t \in O_i$ to a single element of $\{1, \dots, N_t, \emptyset\}$, where \emptyset represents a false alarm. As discussed in Sec. IV, different target tracks cannot be simply updated in isolation from one another due to the subtle statistical dependencies induced by uncertain data-to-target association labels (which may persist for a long time across multiple predict/update cycles due to complex dynamics and the sparsity of UGS data).

D. Related Work

Conventional approaches to multitarget state estimation with unknown data association are characterized by two main features: a data-to-target assignment algorithm, and an algorithm for single-target state estimation under presumed data-to-target associations. (This leaves out more recent nonparametric approaches, such as finite random set techniques, which bypass the data association problem altogether [12].)

For the problem of single-target state filtering with known data association, the Kalman filter (KF) and its variants could theoretically be adapted to the U2R2 setting for UGS-only fusion [13,14]. However, the KF and related methods are only effective if targets always remain within sensor view; otherwise, targets must be “well-behaved” dynamically, such that target measurements occur frequently enough to counteract prediction errors from target motion and model uncertainty. In general, the possible set of motions for targets along a road network leads to highly nonlinear/hybrid dynamics and non-Gaussian prediction errors, which make KF methods unviable for long-term tracking of out-of-sight targets in the U2R2 problem. For instance, if an unsighted target can travel along any road at a fourway intersection, then the hypotheses of its likely predicted future locations will be poorly modeled by a KF that only considers a single-target state mean and covariance. Semiparametric KF methods such as the Gauss sum filter [15], or nonparametric approaches such as the histogram filter [3] and particle filter [16,17], cope with this by maintaining multiple-target state hypotheses for a single track. However, these methods are computationally expensive for large road networks where multiple hypotheses must be spawned frequently. Furthermore, these methods require heuristics to prevent hypothesis explosion/depletion, which can compromise accuracy and make it computationally expensive to implement data association at the same time.

Regarding the problem of unknown data association, standard widely used methods such as probabilistic data association (PDA) [18] and multiple hypothesis tracking (MHT) [19] are most useful in scenarios where sensors persistently provide dense positive returns at each time step. Significant complications arise for the UGS data considered here, as many of the practical heuristics required by these techniques (e.g., measurement gating) break down for highly sparse positive returns and localized fields of view with wide coverage gaps (i.e., low state observability). These heuristics are also primarily designed to contend with Gaussian/unimodal target track distributions, and are thus invalid for

state PDFs that tend to be non-Gaussian and highly multimodal (even when data associations are known) [20]. Reference [21] developed a method for adapting PDA to the particle filter to track targets from a mobile robot using highly nonlinear dynamics and sensor models. However, in addition to requiring a large number of samples to ensure robustness in complex tracking situations, this approach still crucially relies on having dense measurement returns to balance out per-measurement information loss that results from probabilistic hedging in data-to-target assignments. MHT avoids such information losses by making explicit data-to-target label assignments following accumulation of evidence along different likely branches of the target assignment hypothesis tree. This requires a brute force search over all possible data-to-target assignments, which grows combinatorially over multiple time steps. As such, MHT search and pruning techniques lean quite heavily on assumptions of dense positive measurement returns and Gaussian target state PDFs.

Another related limitation of conventional PDA and MHT methods is that they do not handle negative information fusion, which generally leads to non-Gaussian/multimodal state PDFs [22,23]. Because targets are expected to be out of view from UGSs most of the time in the U2R2 problem, negative information is actually predominant in each UGS's data stream. Negative information does not require solving the data association problem (a true no-detection measurement is informative of all targets) and does not immediately lead to precise target state information in the same way as positive detection measurements. Nevertheless, negative information is crucial for pruning inconsistent target state beliefs on large complex road networks [3], and thus provides valuable information for solving the data association problem.

III. Markov Model for Single-Target Dynamics

A hidden Markov model is defined by two dependent discrete time stochastic processes: 1) a (first-order time invariant) Markov chain over a finite number of discrete latent states ζ , and 2) an observation process that produces random measurements o that are conditionally dependent on ζ [24]. This work models a target's dynamics on the road network by a Markov chain where ζ represents the unknown target position and direction of motion. Given a set of conditional transition probabilities between different values for ζ , the Markov chain model efficiently updates and predicts the probability of a target's location at any (past or future) point in time starting from an initial probability distribution over ζ . Furthermore, efficient Bayesian inference algorithms for HMMs exist to recursively update the ζ distribution, given new UGS data o collected by the UAV. This section describes how the graph-based road network stochastic dynamics model from Sec. II can be automatically converted into a time-invariant Markov chain, and it discusses some useful features of the Markov chain representation. HMM algorithms for fusing negative and positive UGS data with known data association are then discussed in Sec. IV.

A. Derivation of Markov Chain Model of Target Location from Road Network Specification

The latent target state ζ is obtained from spatial discretization of x_k at time k along each edge r . Suppose r is discretized into b_r cells so that the total number of cells in the road network is

$$\bar{b} = \sum_{c=1}^{N_e} b_c$$

This allows each possible discrete edge/road pair of the discretized state vector $\xi_k = [e_k, x_k]^T$ to be treated as a unique discrete coordinate point along the road network so that

$$\xi_k = [e_k, x_k]^T \mapsto \zeta'_k \in \{1, 2, \dots, \bar{b}\} \quad (7)$$

Suppose also that each intersection node n_h is counted as a separate discrete position cell. If the set of possible discretized edge/road position states ζ'_k are augmented to include each n_h node as "temporary position" states located between all possible edge-to-edge transitions, then the total number of discrete spatial coordinates (cells) is $d = N_n + \bar{b}$. The full discretized state transformation thus becomes $\zeta_k \in \{1, 2, \dots, d\}$. Under this discretization scheme, Eqs. (4) and (5) become

$$P(\zeta_k | O_{1:k-1}) = \sum_{\zeta_{k-1}} P(\zeta_i | \zeta_{i-1}) P(\zeta_{i-1} | O_{1:k-1}) \quad (8)$$

$$P(\zeta_k | O_{1:k}) \propto P(\zeta_k | O_{1:k-1}) P(O_k | \zeta_k) \quad (9)$$

where $P(\zeta_k | O_{1:0}) = P(\zeta_0)$ and all intractable integrals have now been replaced by a tractable set of recursive summations over discrete state cell realizations.

The continuous-to-discrete state mapping $\xi_k \mapsto \zeta_k$ thus permits modeling of discrete state transition probabilities in discrete time. These are naturally formulated as a Markov chain, which is specified by an initial probability distribution $P(\zeta_0)$ and transition kernel $P(\zeta_i | \zeta_{i-1})$ for $i = 1, 2, \dots, k, \dots$ [24]. Although $P(\zeta_0)$ is easily derived by integrating $p(\xi_0)$, it is less obvious how to obtain $P(\zeta_i | \zeta_{i-1})$ from the node transition probabilities and the edge-dependent stochastic position transition kernels $p(\xi_i | \xi_{i-1})$. Indeed, d^2 conditional transition probabilities $P(\zeta_i | \zeta_{i-1})$ must be specified to describe how the target can move between any pair of d states from $i-1$ to i . This is resolved by noting that the following conditional independence relationship holds from the Markov assumption on the target position dynamics:

$$p(\xi_i | \xi_{i-1}) = p([e_i, x_i]^T | [e_{i-1}, x_{i-1}]^T) = P(e_i | e_{i-1}, x_{i-1}) p(x_i | x_{i-1}, e_{i-1}) \quad (10)$$

The conditional probabilities $P(e_i | e_{i-1}, x_{i-1} = 1)$ are already defined for transitions from the end of an edge to another edge via the intersection nodes (see Sec. II.A). For transitions from one edge r onto itself before reaching the end of edge r , $P(e_i = r | e_{i-1} = r, x_{i-1} < 1) = 1$. Hence, it remains to define cell-to-cell transitions $p(x_i | x_{i-1}, e_{i-1} = r)$ for two cases.

In the first case, if the target transitions from an intersection node onto a new edge, this can be defined such that the target always lands on the first cell of the new edge at the next time step (with probability equal to the probability of turning onto that new edge).

In the second case, if $e_i = e_{i-1}$, then $p(x_i | x_{i-1}, e_{i-1})$ should be derived directly through integration of the stochastic dynamics model along the road edge; see Eq. (2). In this case, before discretization of x_{i-1} , it follows that (for discretization interval ΔT between time steps i and $i-1$)

$$p(x_i|x_{i-1}, e_{i-1}) = \mathcal{U}(x_{i-1} + v_{\min}(e_{i-1})\Delta T, x_{i-1} + v_{\max}(e_{i-1})\Delta T) = U(x_{i,\min}, x_{i,\max}) \quad (11)$$

i.e., the predicted position at x_i is uniformly distributed given the precise position x_{i-1} along edge $e_{i-1} = e_i$; see Fig. 3a. If the grid used to discretize x_i and x_{i-1} along the same edge is constant in time, then $P(x_{i-1} \in c_j|e_{i-1})$ can be approximated by a uniform distribution that “smears out” the PDF $p(x_{i-1}|e_{i-1})$ within cell c_j , as shown in Fig. 3a. Then, if $p(x_{i-1}|e_{i-1}) \approx U(c_j^-, c_j^+)$ (where c_j^- and c_j^+ give the upper and lower bounds for c_j), the following joint PDF is obtained:

$$p(x_i, x_{i-1}, e_{i-1}) = p(x_{i-1}|e_{i-1})p(x_i|x_{i-1}, e_{i-1}) = U(c_j^-, c_j^+)U(x_{i,\min}, x_{i,\max}) \quad (12)$$

The discrete state transition probabilities follow from integration of this PDF with respect to all possible x_i and x_i values given e_{i-1} because

$$\begin{aligned} P(\zeta_i = c_l|\zeta_{i-1} = c_j) &= P(x_i \in c_l|x_{i-1} \in c_j, e_{i-1}) \\ &\propto P(x_i \in c_l, x_{i-1} \in c_j|e_{i-1}) \approx \int_{c_j^-}^{c_j^+} p(x_{i-1}|e_{i-1}) \int_{c_l^-}^{c_l^+} p(x_i|x_{i-1}, e_{i-1}) dx_i dx_{i-1} \end{aligned}$$

Hence, $P(\zeta_i = c_l, \zeta_{i-1} = c_j)$ is approximately

$$P(\zeta_i = c_l, \zeta_{i-1} = c_j) \approx \int_{c_j^-}^{c_j^+} \mathcal{U}(c_j^-, c_j^+) \int_{c_l^-}^{c_l^+} \mathcal{U}(x_{i,\min}, x_{i,\max}) dx_i dx_{i-1} = \int_{c_j^-}^{c_j^+} q(x_{i-1}) dx_{i-1} \quad (13)$$

where c_l and c_j are the discretized cells for x_i and x_{i-1} along the edge $r = e_{i-1} = e_i$, and

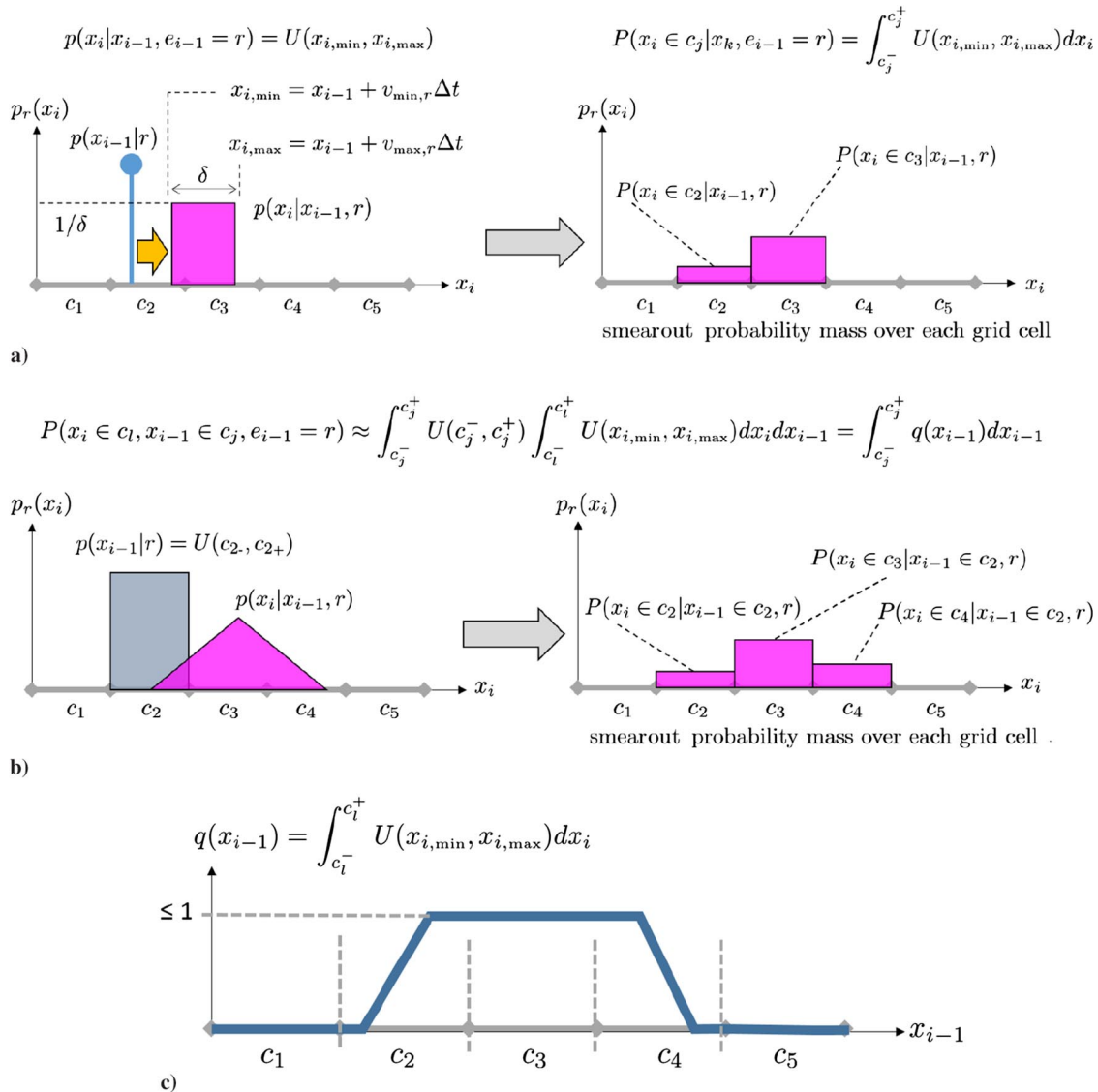


Fig. 3 Representations of a) updated location probabilities with uniform velocity distribution; b) convolution integral [Eq. (13)]; and c) function integrated in convolution for transition probabilities.

$$q(x_{i-1}) = \int_{c_i^-}^{c_i^+} \mathcal{U}(x_{i,\min}, x_{i,\max}) dx_i \quad (14)$$

Figure 3b shows that Eq. (13) is a convolution integral representing the one-step probability transition function for a target starting anywhere in cell c_j that moves to any point in cell c_i along edge r . Applying this to all possible cell transitions from c_j at time $i-1$ to c_i at i , the desired set of discretized transition probabilities $P(\zeta_i = c_i | \zeta_{i-1} = c_j)$ along r can be derived easily from $p(x_i | x_{i-1}, e_{i-1} = r)$. Note that $P(\zeta_i | \zeta_{i-1}) = 0$ for all $\zeta_i < \zeta_{i-1}$ along the same edge because the target is only assumed to move forward along the edge. Figure 3c shows the function $q(x_i)$, which is integrated with respect to x_{i-1} in the convolution process to obtain the cell transition probability $P(\zeta_i = c_i | \zeta_{i-1} = c_j) \equiv a_{ij}$ along edge r , where

$$P(\zeta_i = c_i | \zeta_{i-1} = c_j) = \frac{P(\zeta_i = c_i, \zeta_{i-1} = c_j)}{P(\zeta_{i-1} = c_j)} = \frac{P(x_i \in c_i, x_{i-1} \in c_j)}{P(x_{i-1} \in c_j)} = a_{ij}$$

and

$$P(\zeta_{i-1} = c_j) = \sum_{c_l} P(\zeta_i = c_l, \zeta_{i-1} = c_j)$$

The grid size along edge r is selected such that $|c_j| = |c_l| = v_{\max}(r)\Delta T$, which ensures that the grid cells are not too large relative to $v_{\min}(r)$ and $v_{\max}(r)$. Alternatively, it is also reasonable to select ΔT such that

$$\Delta T \frac{v_{\max}(r) - v_{\min}(r)}{2} = \Delta c$$

so that the probability of holding a target in a cell and the probability of pushing it out of a cell are symmetric; this is useful for deriving alternative multitarget Markov models for the U2R2 problem [25].

Combining the interedge transition probabilities from the node specifications with the intraedge position transition probabilities derived via Eq. (13), a total of d^2 transition probabilities are recovered. Fortunately, the vast majority of these probabilities is zero; in fact, the only nonzero probabilities are those transitions corresponding to physically feasible target state transitions, namely, 1) between neighboring states along a given edge (given by the preceding convolution integral); 2) from the end of one edge to a node (with probability 1); 3) from a node to an edge (with a prespecified probability of making the turn onto a new outgoing edge or performing a U turn back onto the old incoming edge); or 4) to a terminating node (with a prespecified termination probability).

If $p(x_{i-1} | e_{i-1} = r)$ and $p(x_i | e_i = r)$ are well approximated by uniform PDFs over $[c_j^-, c_j^+]$ and $[c_i^-, c_i^+]$ for all i and r (which can be ensured by choosing a suitable grid size $|c_j| = |c_l|$), then probability calculations for ζ_i at any i can be obtained in a compact recursive form. Let $\pi_i \equiv P(\zeta_i)$ be a $d \times 1$ vector of probabilities over the discretized state ζ such that $0 \leq \pi_i(z) \leq 1$ for index variable $z \in \{1, \dots, d\}$, where $\pi_i(z) = P(\zeta_i = z)$ and

$$\sum_{z=1}^d \pi_i(z) = 1$$

Furthermore, define the $d \times d$ column stochastic matrix A to be the matrix for which the $(m, n)^{\text{th}}$ entry defines the transition probability $P(\zeta_i = m | \zeta_{i-1} = n) = a_{mn}$, such that each column n satisfies

$$\sum_{m=1}^d a_{mn} = 1$$

Then, the one-step prediction update for Eq. (8) is equal to

$$\pi_i = A\pi_{i-1} \quad (15)$$

where A is a highly sparse matrix. Extending this prediction update to the general form of Eq. (8),

$$\pi_i = A^i \pi_0 = \Phi(i, 0) \pi_0 \quad (16)$$

where the general Markov state transition matrix (STM) is $\Phi(i, 0) \equiv A^i$. Therefore, probabilistic predictions of the target location can be readily approximated via fast highly sparse d -dimensional vector-matrix operations: even if d is on the order of hundreds or thousands, the required operations for stochastic state prediction can be carried out for real-time online operation with modern linear algebra libraries.

1. Markov Chain Analysis of Stochastic Target Motion Dynamics

Figures 4a and 4c shows two benchmark road networks used to study the U2R2 problem, which each encode velocity distributions and intersection turn probabilities for targets along edges and nodes according to Eq. (2). Figures 4b and 4d show the sparsity pattern of the resulting Markov transition matrix A for this network obtained via the proposed convolution approximation: all nonzero elements are found exclusively on either the main diagonal, the first superdiagonals, or a handful of offdiagonal elements. This model clearly captures the “highly local” nature of the stochastic target dynamics that stems from constrained forward motion along each edge (corresponding to the nonzero main band diagonal entries) as well as more “global” aspects related to turn/termination events at nodes (corresponding to the nonzero offdiagonal elements). It is interesting to note that the nearly diagonal block matrix entries are sub-Markov chains with the so-called left-to-right transition property, which encodes the constraint of “oneway travel” along a particular edge with random (forward) velocities (such Markov chains are also called “Bakis models” in the signal processing literature [24]). For the road networks considered here, the mean velocities along each edge are constant; therefore, the values on these band diagonal entries consist of only a handful of unique numbers that are completely determined by the resolution of the spatial and temporal discretizations (e.g., for both examples in Fig. 4, the band diagonals for each subblock of A contain only two to three

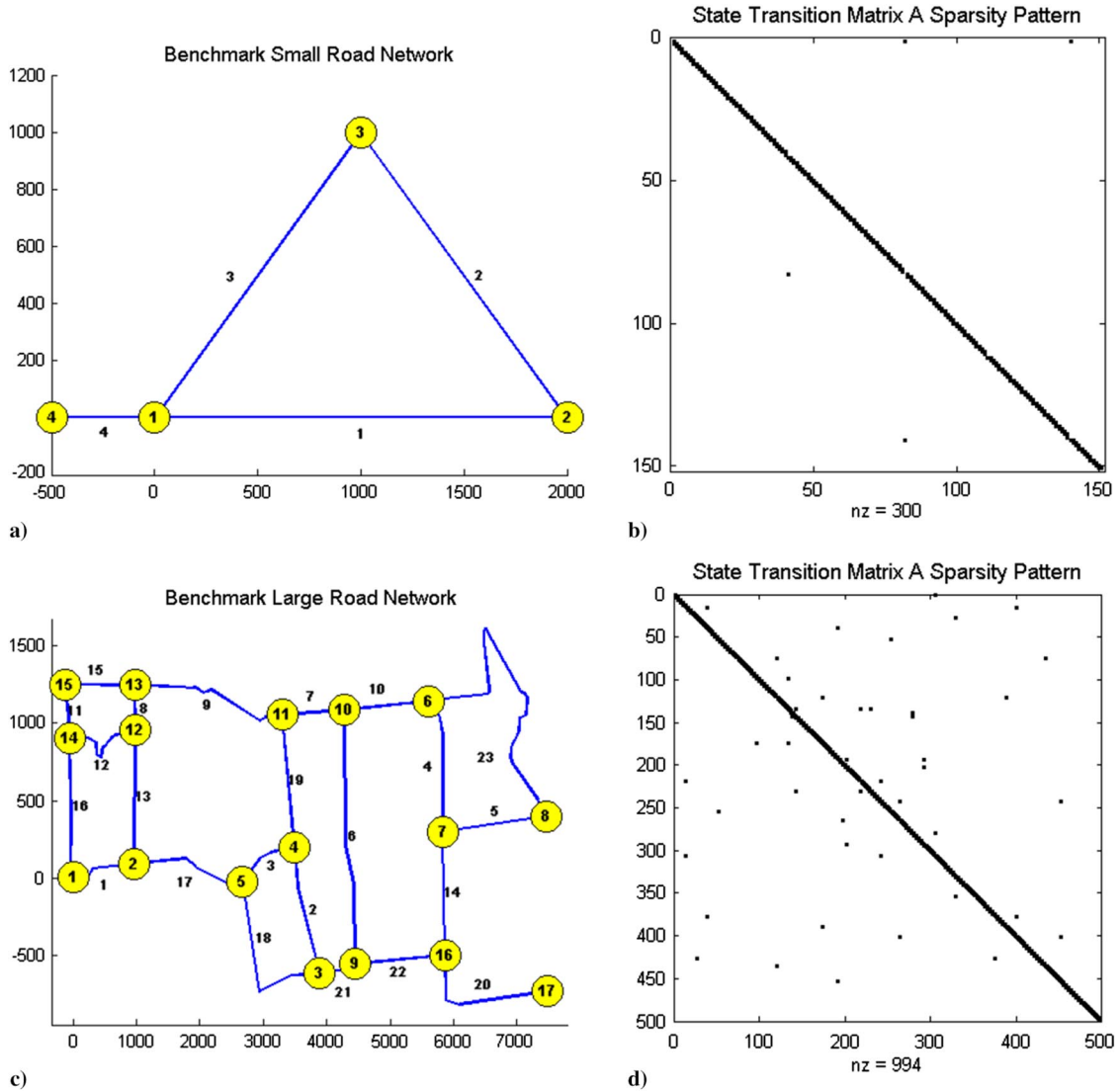


Fig. 4 Representations of a–b) small road network and STM; and c–d) large road network.

unique numbers corresponding to conditional transition probabilities). These highly sparse and constrained structural properties make it quite trivial to add, subtract, or modify road/edges and intersection nodes in an existing road network model.

Equation (16) shows that the discretized target state probability can be predicted an arbitrary number of time steps s into the future using straightforward matrix power calculations, starting from any initial distribution $p(\zeta_k)$. In particular, the conditional probability of reaching any state $\zeta_{k+s} = m$ in s steps starting from $\zeta_k = n$ is given by the (m, n) entry of the STM $\Phi(k + s, k) = \Phi(s, 0)A^s$. This provides a simple direct way to compute target state reachability probabilities, which in turn could be used in intelligent target interception strategies for UAV planning and control. For instance, such information could be used to rule out or to minimize UAV flybys over portions of the road network where certain entries of A^s are guaranteed to fall below a certain threshold for at least s time steps in the future. In general, the fact that A is sparse and A^s can be quickly computed (e.g., via precalculated eigenvalue/eigenvector decomposition) makes the discrete Markov chain model particularly attractive for closed-loop planning/control when compared to more computationally expensive continuous non-Gaussian state prediction models (e.g., the interactive multiple model Kalman Filter or particle filter).

Equation (16) can also be used to derive the Markov chain's fundamental matrix [26]:

$$N = \sum_{i=0}^{\infty} Q^i = (I - Q)^{-1} \quad (17)$$

where Q is the submatrix of A corresponding to entries for only those transitions between all nonterminating states, and the (m, n) entry of N gives the expected number of times the target visits state m , starting from state n . Several other important quantities can be computed from N , such as the variance in the number of times the target visits m starting from n , as well as the expected number of time steps to reach any terminating node starting from any nonterminating state. If A is invertible, then powers of A^{-1} can also be used for probabilistic retrodiction, i.e., to determine where the target could have arrived from.

2. Caveats

Appropriate space and time discretization parameters are needed to generate a suitably accurate Markov chain approximation of the target's stochastic motion model. The state transition probabilities from the convolution approximation tend toward one as the spatial interval increases

(leading to inaccurate predictions of extremely slow target motion along edges), whereas the probabilities tend to “bleed” out of cells faster as the time discretization interval increases (leading to inaccurate predictions of fast target motion along edges). Although small spatial and time discretization intervals provide better prediction accuracy, they also lead to a significant increase in the size of d and A , making predictions via Eq. (16) more costly (although A itself remains sparse). These issues can be avoided by choosing an arbitrary discretization time interval ΔT and constraining spatial intervals to only be as large as the maximum expected target travel time along any given edge within a single ΔT step.

It is also worth noting that a target's stochastic residency time in state ζ is exponentially distributed according to the simple Markov model [27]. However, exact marginalization of highly variable target velocities from the road network motion model would actually imply that the discretized space/time dynamics obey a semi-Markov process, which in turn means that target state residency times should actually be distributed according to mixtures of exponentials. As such, the distributions for edge/road traversal times for each edge in the network are not completely characterized by single mean and variance values but rather by upper- and lower-bounding collections of traversal time statistics. Such processes are best described by variable-duration Markov models, in which the expected traversal time through any ζ state is modeled by an additional “countdown” random variable [24]. Compared to the simpler time-invariant Markov chains derived here, variable-duration Markov models could lead to even higher-fidelity target location probability predictions at arbitrary space/time scales, but they are significantly harder to analyze and implement.

B. Pure Prediction Accuracy of the Markov Model Versus Monte Carlo Simulation

The state transition probabilities and predicted target state probabilities obtained via the Markov chain approximation were validated against “ground-truth” Monte Carlo particle simulations for both the small and large road networks shown in Fig. 4, with respective discretization parameters ($\Delta T = 4.167$ s, $\Delta c = 50$ m) and ($\Delta T = 5.7692$ s, $\Delta c = 75$ m) respectively. Figures 5a–5d and 5i–5l show ground-truth results for the predicted target state probabilities obtained with 10,000 particles at two different future time steps for the small and large networks starting from the initial belief state $\pi_0(1) = 1$ and $\pi_0(2:d) = 0$. The ground-truth distributions show the relative frequency of obtaining continuous state particles in the discretized cell states used for the HMM at the indicated time steps (using the same ΔT discretization) if all particles are propagated from the initial time and position through the continuous-time stochastic motion model for each edge given by Eq. (2) and edge transition probabilities specified for the intersection nodes. Figures 5e–5h and 5m–5p show the predicted state probabilities obtained by the HMM

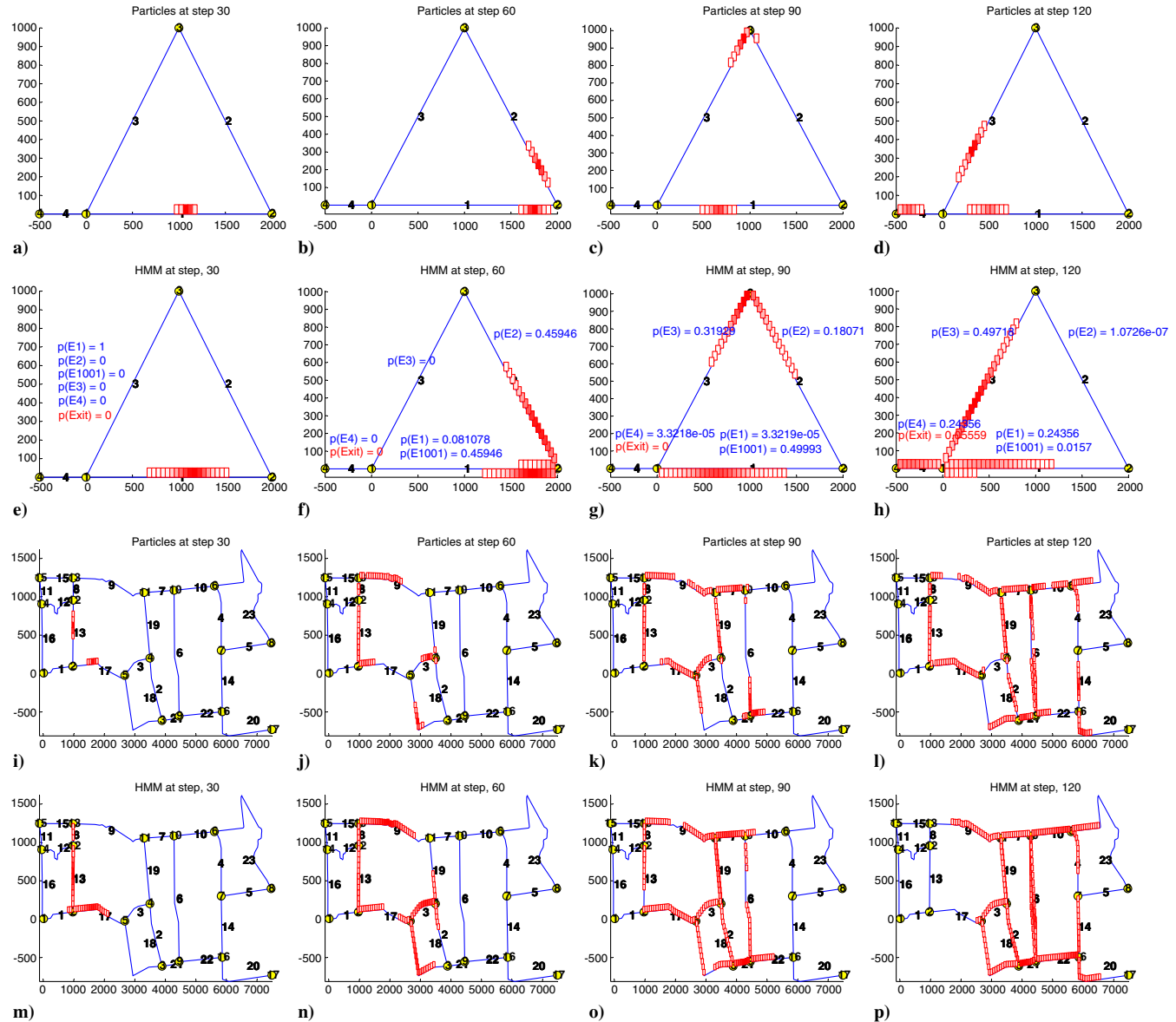


Fig. 5 Predicted probabilities: a–d) and i–l) Monte Carlo; e–h) and m–p) HMM (darker cells more probable).

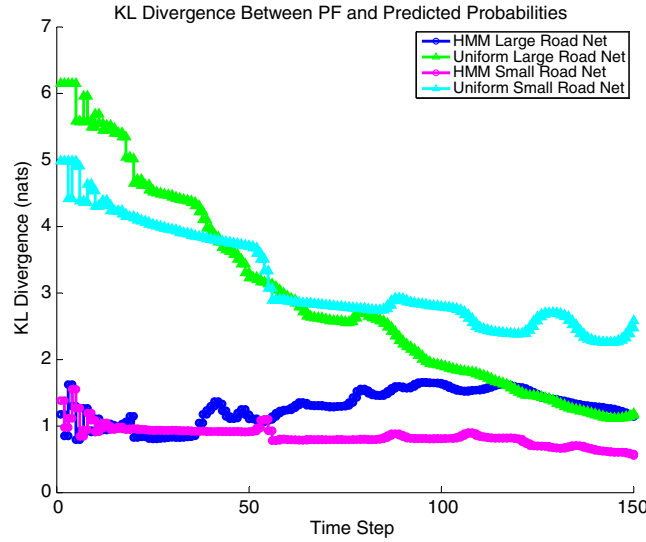


Fig. 6 KL divergences between ground-truth particle and approximate predicted target state distributions for HMM and baseline static uniform distribution (PF, particle filter).

prediction equation [Eq. (16)] for the same two scenarios and time steps using the corresponding STMs shown in Figs. 4b and 4d. Figure 6 shows the Kullback–Leibler (KL) divergence (KLD) profile between the particle-predicted ground-truth distribution ρ_k and HMM-predicted approximate distributions π_k for 150 consecutive forward time steps k , where the KLD for time k is given by

$$\text{KL}[\rho_k \|\pi_k] = \sum_{z=1}^d \rho_k(z) \log \frac{\rho_k(z)}{\pi_k(z)} \quad (18)$$

Because $\text{KL}[\rho_k \|\pi_k] \geq 0$ and $\text{KL}[\rho_k \|\pi_k] = 0$ if, and only if, $\pi_k = \rho_k$, the KLD provides an informal notion of “distance” between the predicted distributions π_k and ρ_k : the smaller Eq. (18) is, the harder it becomes to tell the two distributions apart [27]. The KLD provides a general notion of statistical consistency for approximations to highly non-Gaussian/multimodal distributions because Eq. (18) grows larger when $\pi_k(z) < \rho_k(z)$ for z where $\rho_k(z) > 0$, i.e., if the approximation π_k incorrectly diminishes the support of the “exact” distribution ρ_k , thus indicating inconsistency due to excessive information loss. For reference, also shown are the KLDs between the particle ground-truth distribution and a simple baseline static uniform distribution for π_k (i.e., which always has the same probability in every possible discretized state for each road network).

These results show that the predicted state distributions obtained with the computationally cheaper HMM are in good agreement with the predictions made by the more computationally expensive Monte Carlo sampling procedure; indeed, only minor discrepancies can be found between the two sets of predictions in each case for areas with low probability mass. As Fig. 5 shows, these discrepancies can be attributed to the fact that a small amount of probability mass tends to bleed out from the front and back tails of the predicted HMM distributions, although the bulk of the probability mass still tends to concentrate around the actual support of the “true” predicted state distribution. These tail errors are a direct consequence of the key assumption used to derive the recursive Markov chain prediction update, i.e., that $p(x_k)$ and $p(x_{k+1})$ are always approximately uniform over each discrete cell because this causes small portions of “redistributed” probability mass to either repeatedly lag slightly behind or jump slightly ahead of the true support for the predicted distribution. Nevertheless, as shown in Fig. 6, this leads to relatively small bounded HMM prediction errors, even as the target passes through several highly connected intersections to yield multimodal state distributions. Furthermore, the HMM predictions stay much closer to the ground-truth distributions than the naive static uniform probability models, which are significantly worse in the beginning but tend to lower KLDs, because the true distributions tend to become more uniform (although not exactly so). The KLD values can be understood further using an insight from [28]: the KLD can be related to the expected number of samples from the π_k distribution required to tell ρ_k and π_k apart in a Bayesian log-likelihood hypothesis test set with a given threshold τ , where the expected number of samples is $\tau/\text{KL}[\rho_k \|\pi_k]$. Thus, if $\tau = \log(100)$, KLDs of 1 and 1.5 (as obtained on average for the HMMs) give expected sample sizes of 4.61 and 3.07, respectively; whereas KLDs of 3 and 5 yield 1.53 and 0.77 samples. That is, as the KLD gets larger, fewer random samples from π_k are needed on average to distinguish ρ_k and π_k in simulated hypothesis tests; the average sample size grows larger as the KLD drops and the distributions become harder to separate.

IV. Bayesian HMM Data Fusion for Multitarget State Estimation

The fusion of UGS data with Markov model dynamics predictions is now considered for multitarget localization in the presence of false alarms and clutter. Hidden Markov model Bayes filters for each target $t = 1, \dots, N_t$ are used to fuse together positive/detection UGS data with known target associations and negative/no-detection UGS observations at each discrete time step, producing a recursively updated set of probabilities over each ζ_t . It is assumed that all observations are fused synchronously at each time step at a single centralized fusion center, where the number of targets of interest N_t is known. The basic HMM filter is presented first in the context of single-target localization with perfectly known positive UGS data associations. The HMM Rao–Blackwellized particle filter is then derived to solve the unknown multitarget data association problem by directly exploiting the fast closed-form update of the basic HMM filter for a sampled set of association hypotheses.

A. General Probabilistic Model

Figure 7 shows the probabilistic graphical model (PGM) that encodes the conditional independence assumptions for the U2R2 multitarget tracking problem. Here, the (known) number of targets to track is N_t and the total number of discrete UGSs available is N_u . The state of each target $t \in \{1, \dots, N_t\}$ at discrete time i is ζ_t^i . Positive (detection) and negative (no detection) measurements from UGS $u \in \{1, \dots, N_u\}$ at time i are denoted by $o_t^i = 1$ and $o_t^i = 0$, respectively. The PGM shows the UGS data split into two groups at each time step: a set of N_t^+ positive

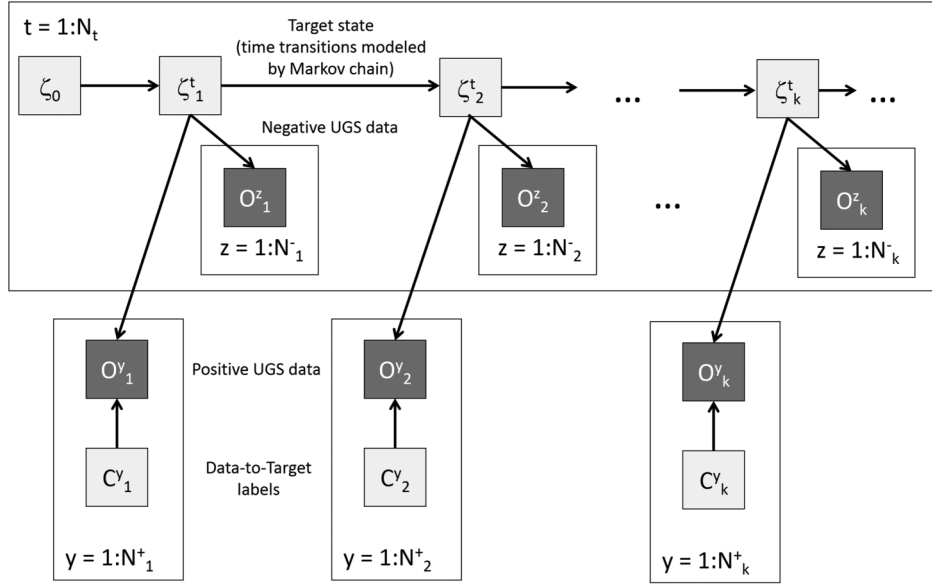


Fig. 7 Probabilistic graphical plate model of multitarget tracking problem with positive/negative UGS data. Dark shaded nodes denote observed random variables.

measurements for which the UGS are indexed by $y \in \mathcal{Y}_i$; and a set of N_i^- negative measurements for which the UGS are indexed by $z \in \mathcal{Z}_i$, where $N_u = N_i^+ + N_i^-$. An observation at UGS u depends on the state of all N_i targets at time i . For each UGS of $y \in \mathcal{Y}_i$, the positive measurement o_i^y also depends on the unknown discrete random variable c_i^y that encodes the identity of the target being observed, where $c_i^y \in \{\emptyset, 1, 2, \dots, N_i\}$ and \emptyset denotes the null target (false alarm/clutter). For each UGS of $z \in \mathcal{Z}_i$, the negative measurement o_i^z is copied and assigned to all targets.

The latent target states ζ_i^t evolve according to the Markov chain model derived earlier. A hidden Markov model is obtained for each target once positive data labels c_i^y and likelihood functions are specified for the UGS data, which can then be fused via recursive Bayesian inference. Let $l(u)$ be the Markov model discretization of $\sigma(u) = [e^u, x^u]^T$, which is the location of the UGS u according to the original road network graph model. The likelihood for any (positive or negative) observation o_i^u as a function of the HMM states $\mathcal{X}_i = \{\zeta_i^1, \dots, \zeta_i^{N_i}\}$ is

$$P(o_i^u | \mathcal{X}, c_i^u) = \prod_{t=1}^{N_i} P(o_i^u | \zeta_i^t, c_i^u) \quad (19)$$

where

$$P(o_i^u | \zeta_i^t, c_i^u) = [\delta(l(u), \zeta_i^t) P_{D,u}]^{\delta(c_i^u, t) \times o_i^u} \times [(1 - P_{D,u})(1 - P_{FA,u})]^{\delta(l(u), \zeta_i^t) \times (1 - o_i^u)} \quad (20)$$

For UGSs u that are indexed by $y \in \mathcal{Y}_i$ at time i , positive observations of $o_i^y = 1$ depend on all target states as well as c_i^y in order to account for the association uncertainty. The likelihood is thus split into N_i factors:

$$P(o_i^y = 1 | \zeta_i^1, \dots, \zeta_i^{N_i}, c_i^y) = \prod_{t=1}^{N_i} P(o_i^y = 1 | \zeta_i^t, c_i^y) \quad (21)$$

where the following factors are defined for each target t via Eq. (20):

$$P(o_i^y = 1 | \zeta_i^t, c_i^y) = [\delta(l(y), \zeta_i^t) P_{D,y}]^{\delta(c_i^y, t)} = \begin{cases} P_{D,y}, & \text{if } l(y) = \zeta_i^t \text{ and } c_i^y = t \\ 0, & \text{if } l(y) \neq \zeta_i^t \text{ and } c_i^y = t \\ 1 \forall \zeta_i^t, & \text{if } c_i^y \neq t \end{cases} \quad (22)$$

The first two cases correspond to the assignment of o_i^y to target t ; these place the target exactly at the UGS location $l(u)$ at time i . The third case corresponds to assignment of o_i^y to either another target or clutter; hence, no information is gained about target t .

For UGSs u that are indexed by $z \in \mathcal{Z}_i$ at time i , a negative observation of $o_i^z = 0$ gives the following likelihoods via Eq. (20) that are independent of c_i^z :

$$P(o_i^z = 0 | \zeta_i^t, c_i^z) = P(o_i^z = 0 | \zeta_i^t) = [(1 - P_{D,z})(1 - P_{FA,z})]^{\delta(l(z), \zeta_i^t)} = \begin{cases} (1 - P_{D,z})(1 - P_{FA,z}), & \text{if } l(z) = \zeta_i^t \\ 1, & \text{if } l(z) \neq \zeta_i^t \end{cases} \quad (23)$$

where $\delta(\cdot, \cdot)$ is the Kronecker delta function. The $l(z) = \zeta_i^t$ case accounts for the possibility of a missed detection: that is, if neither target nor clutter is detected by UGS z at time i . The $l(z) \neq \zeta_i^t$ case increases the likelihood that the target is anywhere except $l(z)$ at time i because each UGS has a limited sensing field with $P_{FA,z} < 1$ and $P_{D,z} \leq 1$.

B. HMM Bayes Filter with Known Data Associations

Consider a single target t with state ζ_i^t to which all positive measurements are known to be associated, i.e., $c_i^y = t$ and clutter is completely ignored. For convenience, denote the set of all positive UGS measurements at time i associated with t as \mathcal{Y}_i^t , where $\mathcal{Y}_i^t \subseteq \mathcal{Y}_i$, and let \mathcal{Z}_i be the set of all negative UGS measurements at i , as before. The Bayes filter recursions are given by the following updates, starting from measurements taken at time $i = 1$ with some $P(\zeta_0^t)$ initial distribution:

Dynamics prediction update:

$$P(\zeta_i^t | \mathcal{Y}_{1:i-1}^t, \mathcal{Z}_{1:i-1}) = \sum_{\zeta_{i-1}^t} P(\zeta_i^t | \zeta_{i-1}^t) P(\zeta_{i-1}^t | \mathcal{Y}_{1:i-1}^t, \mathcal{Z}_{1:i-1}) \quad (24)$$

Negative measurement updates:

$$P(\zeta_i^t | \mathcal{Y}_{1:i-1}^t, \mathcal{Z}_{1:i}) = \kappa_{\mathcal{Z}}^{-1}(i) P(\mathcal{Z}_i | \zeta_i^t) P(\zeta_i^t | \mathcal{Y}_{1:i-1}^t, \mathcal{Z}_{1:i-1}) \quad (25)$$

Positive measurement updates:

$$P(\zeta_i^t | \mathcal{Y}_{1:i}^t, \mathcal{Z}_{1:i}) = \kappa_{\mathcal{Y}}^{-1}(i) P(\mathcal{Y}_i | \zeta_i^t) P(\zeta_i^t | \mathcal{Y}_{1:i-1}^t, \mathcal{Z}_{1:i}) \quad (26)$$

where Eq. (24) corresponds to Eq. (15), and where

$$\kappa_{\mathcal{Z}}(i) = P(\mathcal{Z}_i | \mathcal{Y}_{1:i-1}^t, \mathcal{Z}_{1:i}) \sum_{\zeta_i^t} P(\mathcal{Z}_i | \zeta_i^t) P(\zeta_i^t | \mathcal{Y}_{1:i-1}^t, \mathcal{Z}_{1:i-1}) \quad (27)$$

$$P(\mathcal{Z}_i | \zeta_i^t) = \prod_n P(o_i^z = 0 | \zeta_i^t) \quad (28)$$

$$\kappa_{\mathcal{Y}}^t(i) = P(\mathcal{Y}_i^t | \mathcal{Y}_{1:i-1}^t, \mathcal{Z}_{1:i}) = \sum_{\zeta_i^t} P(\mathcal{Y}_i^t | \zeta_i^t) P(\zeta_i^t | \mathcal{Y}_{1:i-1}^t, \mathcal{Z}_{1:i}) \quad (29)$$

$$P(\mathcal{Y}_i^t | \zeta_i^t) = \prod_u P(o_i^y = 1 | \zeta_i^t, c_i^y = t) \quad (30)$$

It will be convenient to later represent the positive measurement updates sequentially and identify their corresponding marginal measurement likelihoods. In this case, with $c_i^y = t$ for all N_i^+ elements of $y \in \mathcal{Y}_i^t$:

$$P(\zeta_i^t | o_i^1, \mathcal{Y}_{1:i-1}^t, \mathcal{Z}_{1:i}) = \kappa_{o^1}^{-1}(i) P(o_i^1 | \zeta_i^t) P(\zeta_i^t | \mathcal{Y}_{1:i-1}^t, \mathcal{Z}_{1:i}) \quad (31)$$

$$\kappa_{o^1}(i) = P(o_i^1 | \mathcal{Y}_{1:i-1}^t, \mathcal{Z}_{1:i}) = \sum_{\zeta_i^t} P(o_i^1 | \zeta_i^t) P(\zeta_i^t | \mathcal{Y}_{1:i-1}^t, \mathcal{Z}_{1:i}) \quad (32)$$

⋮

$$P(\zeta_i^t | o_i^{1:N_i^+}, \mathcal{Y}_{1:i-1}^t, \mathcal{Z}_{1:i}) = \kappa_{o^{N_i^+}}^{-1}(i) P(o_i^{N_i^+} | \zeta_i^t) P(\zeta_i^t | o_i^{1:N_i^+-1}, \mathcal{Y}_{1:i-1}^t, \mathcal{Z}_{1:i}) \quad (33)$$

$$\kappa_{o^{N_i^+}}(i) = P(o_i^{N_i^+} | o_i^{1:N_i^+-1}, \mathcal{Y}_{1:i-1}^t, \mathcal{Z}_{1:i}) = \sum_{\zeta_i^t} P(o_i^{N_i^+} | \zeta_i^t) P(\zeta_i^t | o_i^{1:N_i^+-1}, \mathcal{Y}_{1:i-1}^t, \mathcal{Z}_{1:i}) \quad (34)$$

$$\kappa_{\mathcal{Y}^t}(i) = \prod_{y=1}^{N_i^+} \kappa_{o^y}(i) \quad (35)$$

Note that these filter equations implicitly account for possible missed detections through the UGS observation likelihoods. When there is more than one target, a single HMM filter is run for each target, using all negative data for all targets and positive measurements that correspond to each target according to known data associations.

C. Rao-Blackwellized HMM Particle Filtering for Unknown Data Association

Now, the problem of HMM Bayes filtering with unknown positive data associations is addressed. In this case, each c_i^y must be inferred along with each ζ_i^t . Let \mathcal{C}_i be the set of all associations at time i for all positive measurements in $\mathcal{Y}_i = \{\mathcal{Y}_i^1, \dots, \mathcal{Y}_i^{N_i}\}$. Because \mathcal{C}_i is unknown, the partition of measurements in \mathcal{Y} into the sets \mathcal{Y}_i^t is also unknown. For $N_t = 1$ with clutter present, one seeks the joint posterior distribution:

$$P(\zeta_i^t, \mathcal{C}_i | \mathcal{Y}_{1:i}, \mathcal{Z}_{1:i}, \mathcal{C}_{1:i-1}) = P(\zeta_i^t | \mathcal{C}_i, \mathcal{Y}_{1:i}, \mathcal{Z}_{1:i}, \mathcal{C}_{1:i-1}) P(\mathcal{C}_i | \mathcal{Y}_{1:i}, \mathcal{Z}_{1:i}, \mathcal{C}_{1:i-1}) \quad (36)$$

More generally, for $N_t > 1$ targets with clutter present, the following joint posterior is sought

$$P(\zeta_i^1, \dots, \zeta_i^{N_t}, C_i | \mathcal{Y}_{1:i}, \mathcal{Z}_{1:i}, C_{1:i-1}) = P(C_i | \mathcal{Y}_{1:i}, \mathcal{Z}_{1:i}, C_{1:i-1}) \prod_{t=1}^{N_t} P(\zeta_i^t | C_i, \mathcal{Y}_{1:i}, \mathcal{Z}_{1:i}, C_{1:i-1}) \quad (37)$$

where the factorization follows from conditional independence of target tracks once each c_i^y in C_i is given. Even if N_i^+ and N_t are small, the number of possible association histories up to time k , i.e.,

$$C_{1:k} \equiv \{c_1^{1:N_1^+}, \dots, c_k^{1:N_k^+}\}$$

explodes quickly. This makes explicit enumeration and evaluation of all possible association hypotheses intractable.

This directly motivates the use of Rao–Blackwellized particle filtering to approximate Eq. (37). Rather than explore the full (feasible) hypothesis space or collapse all hypotheses via marginalization, the RBPF uses Monte Carlo importance sampling to explore only the statistically most likely association hypotheses C_i and retains crisp state estimates for each sampled hypothesis by running a full bank of HMM filters for each particle. By sampling the hypothesis space according to HMM track set likelihoods (which automatically account for negative UGS information), highly multimodal multitarget track probabilities are handled naturally without introducing heuristic gating functions.

The RBPF samples each c_i^y in C_i according to $P(C_i | \mathcal{Y}_{1:i}, \mathcal{Z}_{1:i}, C_{1:i-1})$, and then it uses the resulting set of associated measurements within each sample to exactly compute $P(\zeta_i^t | C_i, \mathcal{Y}_{1:i}, \mathcal{Z}_{1:i}, C_{1:i-1})$ for each target t via analytically tractable HMM filter measurement updates. C_i is sampled by using the fact that

$$P(C_i | \mathcal{Y}_{1:i}, \mathcal{Z}_{1:i}, C_{1:i-1}) = P(c_i^1 | o_i^1, \mathcal{Y}_{1:i-1}, \mathcal{Z}_{1:i}, C_{1:i-1}) \prod_{y=2}^{N_t} P(c_i^y | o_i^y, c_i^{1:y-1}, o_i^{1:y-1}, \mathcal{Y}_{1:i-1}, \mathcal{Z}_{1:i}, C_{1:i-1}) \quad (38)$$

so samples can be drawn sequentially from the conditional distributions

$$P(c_i^y | o_i^y, c_i^{1:y-1}, o_i^{1:y-1}, \mathcal{Y}_{1:i-1}, \mathcal{Z}_{1:i}, C_{1:i-1})$$

for $y = 1, \dots, N_i^+$. In the context of recursive filtering, a single complete sequence of sampled realizations up to time k for

$$C_{1:i} = \{c_1^{1:N_1^+}, \dots, c_k^{1:N_k^+}\}$$

is referred to as one particle. Each particle in the RBPF is associated with a bank of N_t recursive HMM filters, which are used to update state information for each target based on the sampled association hypotheses and update hypothesis weights (probabilities) for each particle.

The critical step is then to draw association samples from $P(C_i | \mathcal{Y}_{1:i}, \mathcal{Z}_{1:i}, C_{1:i-1})$. Särkkä et al. [29] presented an optimal importance sampling (OIS) approach to RBPF data association for linear-Gaussian tracking. Their approach exploited the fact that linear-Gaussian filter likelihoods can be exactly computed in closed form for given association histories, which allows for sequential sampling of association hypotheses through sequences of OIS distributions. In our case, because HMM filter likelihoods can also be exactly computed for each particle, samples for association histories can be drawn according to OIS distributions for each

$$P(c_i^y | o_i^y, c_i^{1:y-1}, o_i^{1:y-1}, \mathcal{Y}_{1:i-1}, \mathcal{Z}_{1:i}, C_{1:i-1})$$

which factorize $P(C_i | \mathcal{Y}_{1:i}, \mathcal{Z}_{1:i}, C_{1:i-1})$. Our adaptation of the OIS algorithm to HMM filter tracking has two significantly distinct features from the linear-Gaussian version:

1) The HMM naturally allows for handling of multimodal track distributions in the data association problem, and therefore obviates the need for the heuristic observation gating functions used in [29].

2) Our version accounts for crucial negative information in the sparse sensing environment by conditioning on no-detection UGS data in $\mathcal{Z}_{1:i}$. To emphasize this distinction, we denote our version of the RBPF as the HMM/RBPF.

The steps for deriving and implementing the HMM/RBPF for known N_t are now presented. Assume that N_p particles are available at time $i-1$, with each representing a different possible positive measurement association sequence $C_{1:i-1}$ for a bank of N_t HMM filters, which have also all fused negative information $\mathcal{Z}_{1:i-1}$. The normalized importance weight for each particle is $w_{i-1}^s \in [0, 1]$, where

$$\sum_{s=1}^{N_p} w_{i-1}^s = 1 \quad (39)$$

For each particle $s = 1, \dots, N_p$, we first perform the usual Bayes's filter prediction and negative measurement update steps for all N_t target filters up to time i :

$$P^s(\zeta_i^t | C_{1:i-1}^s, \mathcal{Y}_{1:i-1}, \mathcal{Z}_{1:i-1}) = \sum_{\zeta_{i-1}^t} P(\zeta_i^t | \zeta_{i-1}^t) P^s(\zeta_{i-1}^t | C_{1:i-1}^s, \mathcal{Y}_{1:i-1}, \mathcal{Z}_{1:i-1}) \quad (40)$$

$$P^s(\zeta_i^t | C_{1:i-1}^s, \mathcal{Y}_{1:i-1}, \mathcal{Z}_{1:i}) = [\kappa_{\mathcal{Z}}^s(i)]^{-1} P(\mathcal{Z}_i | \zeta_i^t) P^s(\zeta_i^t | C_{1:i-1}^s, \mathcal{Y}_{1:i-1}, \mathcal{Z}_{1:i-1}) \quad (41)$$

$$\kappa_{\mathcal{Z}}^s(i) = \sum_{\zeta_i^t} P(\mathcal{Z}_i | \zeta_i^t) P^s(\zeta_i^t | C_{1:i-1}^s, \mathcal{Y}_{1:i-1}, \mathcal{Z}_{1:i-1}) \quad (42)$$

Next, for each s , we assign association labels \mathcal{C}_i to \mathcal{Y}_i and perform the corresponding Bayes fusion updates. Specifically, we use OIS to assign target association labels $c_i^{s,y} \in \{1, \dots, N_t, \emptyset\}$ sequentially to each $o_i^y \in \mathcal{Y}_i$, where $y = 1, \dots, N_t^+$ (UGS identification). For each s and y , $c_i^{s,y}$ is sampled from the optimal importance density:

$$\pi_j^{s,y,i} = \frac{\hat{\pi}_j^{s,y,i}}{\sum_{j' \in \{1, \dots, N_t, \emptyset\}} \hat{\pi}_{j'}^{s,y,i}} = \frac{\hat{\pi}_j^{s,y,i}}{\eta^{s,y,i}} \quad (43)$$

where we have

$$\hat{\pi}_j^{s,y,i} = P(c_i^{s,y} = j | o_i^y, \mathcal{C}_{1:i-1}^s, \mathcal{Y}_{1:i-1}, \mathcal{Z}_{1:i}) \quad (44)$$

This importance sampling step is the same as sampling from the true label posterior up to an unknown normalizing constant. Hence, the importance density is “optimal” in the sense that it minimizes the variance of the importance sampling weights (described in the following) [30].

Applying Bayes’s rule, we get

$$\hat{\pi}_j^{s,y,i} \propto P(o_i^y | c_i^{s,y} = j, \mathcal{C}_{1:i-1}^s, \mathcal{Y}_{1:i-1}, \mathcal{Z}_{1:i}) \cdot P(c_i^{s,y} = j) \quad (45)$$

Here, $P(c_i^{s,y} = j)$ reflects the prior probability of UGSs u detecting a target ($j \in \{1, \dots, N_t\}$) or false alarm ($j = \emptyset$) before conditioning on any other information. Assuming uniform probability of truly detecting any target, then

$$P(c_i^{s,y} \in \{1, \dots, N_t\}) = \frac{1}{N_t} [1 - P(c_i^{s,y} = \emptyset)] \quad (46)$$

$$P(c_i^{s,y} = \emptyset) = P_{FA} \quad (47)$$

The likelihood $P(o_i^y | c_i^{s,y} = j, \mathcal{C}_{1:i-1}^s, \mathcal{Y}_{1:i-1}, \mathcal{Z}_{1:i})$ gives the posterior conditional HMM filter likelihood in particle s that o_i^y belongs to target j . This is computed through the usual HMM filter measurement update equations [i.e., Eq. (34)] for each possible assignment of j in each particle s :

$$P(o_i^y | c_i^{s,y} = \emptyset, \mathcal{C}_{1:i-1}^s, \mathcal{Y}_{1:i-1}, \mathcal{Z}_{1:i}) = 1 \quad (48)$$

$$P(o_i^y | c_i^{s,y} = 1, \mathcal{C}_{1:i-1}^s, \mathcal{Y}_{1:i-1}, \mathcal{Z}_{1:i}) = \sum_{\zeta_i^1} P(o_i^y | \zeta_i^1) P^s(\zeta_i^1 | c_i^{s,y} = 1, \mathcal{C}_{1:i-1}^s, o_i^{1:y-1}, \mathcal{Y}_{1:i-1}, \mathcal{Z}_{1:i}) = \kappa_{o_i^y}^{s,1} \quad (49)$$

$$P(o_i^y | c_i^{s,y} = 2, \mathcal{C}_{1:i-1}^s, \mathcal{Y}_{1:i-1}, \mathcal{Z}_{1:i}) = \sum_{\zeta_i^2} P(o_i^y | \zeta_i^2) P^s(\zeta_i^2 | c_i^{s,y} = 2, \mathcal{C}_{1:i-1}^s, o_i^{1:y-1}, \mathcal{Y}_{1:i-1}, \mathcal{Z}_{1:i}) = \kappa_{o_i^y}^{s,2} \quad (50)$$

⋮

$$P(o_i^y | c_i^{s,y} = N_t, \mathcal{C}_{1:i-1}^s, \mathcal{Y}_{1:i-1}, \mathcal{Z}_{1:i}) = \sum_{\zeta_i^{N_t}} P(o_i^y | \zeta_i^{N_t}) P^s(\zeta_i^{N_t} | c_i^{s,y} = N_t, \mathcal{C}_{1:i-1}^s, o_i^{1:y-1}, \mathcal{Y}_{1:i-1}, \mathcal{Z}_{1:i}) = \kappa_{o_i^y}^{s,N_t} \quad (51)$$

After sampling, if $c_i^{s,y} = j \in \{1, \dots, N_t\}$, particle s updates target j ’s posterior distribution with o_i^y :

$$P^s(\zeta_i^j | c_i^{s,y} = j, \mathcal{C}_{1:i-1}^s, o_i^{1:y}, \mathcal{Y}_{1:i-1}, \mathcal{Z}_{1:i}) = \left[\kappa_{o_i^y}^{s,j} \right]^{-1} P(o_i^y | \zeta_i^j, c_i^{s,y} = j) P^s(\zeta_i^j | \mathcal{C}_{1:i-1}^s, o_i^{1:y-1}, \mathcal{Y}_{1:i-1}, \mathcal{Z}_{1:i}) \quad (52)$$

Otherwise, if $c_i^{s,y} = \emptyset$, target j ’s posterior does not change in particle s :

$$P^s(\zeta_i^j | c_i^{s,y} = \emptyset, \mathcal{C}_{1:i-1}^s, o_i^{1:y}, \mathcal{Y}_{1:i-1}, \mathcal{Z}_{1:i}) = P^s(\zeta_i^j | \mathcal{C}_{1:i-1}^s, o_i^{1:y-1}, \mathcal{Y}_{1:i-1}, \mathcal{Z}_{1:i}) \quad (53)$$

This result is then used to sample the next association label $c_i^{s,y+1}$ for particle s via Eq. (43) so that the preceding expressions [Eqs. (44–53)] also become conditioned on $c_i^{s,1:y-1}$. After sampling and updating target posteriors for all N_t^+ positive measurements, particle s updates its (unnormalized) importance weight according to the combined negative and positive data likelihood:

$$w_i^s \propto w_{i-1}^s \times \kappa_{\mathcal{Z}}^s(i) \prod_{u=1}^{N_u} \frac{P(o_i^u | c_i^{s,y} = j, c_i^{s,1:y-1}, \mathcal{C}_{1:i-1}^s, \mathcal{Y}_{1:i-1}, \mathcal{Z}_{1:i}) \cdot P(c_i^{s,y} = j)}{\pi_j^{s,y,i}} \quad (54)$$

Simplifying this last expression gives

$$\begin{aligned}
w_i^s &\propto w_{i-1}^s \times \kappa_Z^s(i) \times \prod_{u=1}^{N_u} \eta^{s,y,i} \\
&\propto w_{i-1}^s \times \kappa_Z^s(i) \prod_{u=1}^{N_u} \left[\sum_{j' \in \{1, \dots, N_i, \emptyset\}} \hat{\pi}_{j'}^{s,y,i} \right]
\end{aligned} \tag{55}$$

which gives a proper set of particle weights upon normalization.

It is important to emphasize that each particle represents an entire history of sampled associations. Because a finite number of particles N_s will not be able to explore every possible branch of the exponentially growing association hypothesis tree, the RBPF samples will eventually be “depleted.” That is, only a very small number of particles will actually carry any significant probability mass via nonzero weights w_i^s , whereas all others will drop to negligibly small or zero weights. This phenomenon causes large estimation variance, but it can be mitigated with particle resampling. This allows the most likely particles to remain in the overall posterior approximation while pruning highly unlikely particles. Surviving resampled particles are cloned, and their weights are reset to uniform; whereas un-resampled particles are deleted entirely (i.e., their entire sampled association hypothesis histories are wiped out). Here, we adopt the standard multinomial particle resampling scheme, and we sample only if the effective sample size N_{eff} for the particles (i.e., the portion of N_s particles actually contributed to the overall estimate) falls below some predefined threshold, where [30]

$$N_{\text{eff}} = \frac{N_s}{1 + \text{cv}(w_i)}, \quad \text{cv}(w_i) = \frac{\text{var}(w_i)}{(\text{mean}(w_i))^2} \tag{56}$$

and $\text{cv}(w_i)$ is the coefficient of variation of the unnormalized particle weights.

Remark 1: Regarding RBPF versus other association methods, standard methods such as multi-hypothesis tracking (MHT) and joint probabilistic data association (JPDA) attempt to cope with multitarget association hypothesis explosions by heuristically gating measurements with probabilistic thresholds to form a feasible set of hypotheses for C_i . MHT then uses sequential likelihood ratio tests to further constrain C_i as new data are obtained. In contrast, JPDA subsequently marginalizes over all feasible hypotheses in C_i to produce weighted average track estimates at each time. MHT advantageously retains “crisp” target state estimates for each realization of C_i , i.e., without “diluting” target track information as JPDA does. On the other hand, JPDA does not maintain separate tracks for highly unlikely realizations of C_i , and it can thus use problem statistics to improve computational parsimony and efficiency over MHT (especially when P_{FA} is high). The RBPF strikes a balance between MHT’s ability to retain crisp multihypothesis track estimates and JPDA’s statistical parsimony. Furthermore, the RBPF easily accommodates the multimodal posteriors produced by the HMM Bayes filter without resorting to gating heuristics, whereas conventional MHT and JPDA methods rely strongly on such heuristics and on assumptions of unimodal/Gaussian tracking distributions.

Remark 2: Regarding forward–backward smoothing and Viterbi trajectory solutions per particle, a particularly nice feature of the closed-form HMM Bayes updates for known/sampled data associations is that Bayes’s smoother distributions and most likely state trajectories for each target are obtained easily by, respectively, using the forward–backward and Viterbi algorithms on any particle. The Bayes smoother solutions provide marginal posterior distributions of each target’s state at each time step for a given particle, i.e., conditioned on all assigned data in a given sequence. Viterbi solutions provide the single most likely sequence of states visited by targets according to a given particle, i.e., again conditioned on all assigned data in a given sequence. The likelihood of the particle describing a particular data assignment sequence does not change as a result of either Bayes smoothing or Viterbi inference, and the forward Bayes HMM filter updates for the RBPF remain the same as previously stated. It is noted that the HMM Bayes filter described here gives the “forward” pass of the forward–backward algorithm that would be needed for any particle; extra information needs to be stored to compute the subsequent “backward” passes for Bayes’ smoothing, as well as the Viterbi algorithm. The reader is referred to [27,31] for complete details.

Remark 3: Regarding adaptation for unknown N_t , as discussed in [29], the OIS step of the RBPF can be modified to handle an arbitrary unknown number of targets N_t by augmenting the probabilistic target dynamics models with spatial target birth/death processes. For the HMM Bayes filter, these birth/death processes must be defined over ζ_i^t space instead of continuous x_i^t space. Target birth/death process models could, for instance, be obtained via discretization of continuous Poisson processes defined on the road network. These would allow each particle to adaptively grow/shrink the possible target label space by sampling new tracks that best fit “unexplained” UGS data and terminating old or spurious tracks that are no longer supported by data. The hidden Markov dynamics model for each target would remain the same for each target, but the particle likelihood computations presented here would have to be modified to account for target birth/death possibilities, as well as false alarms and missed detections. A detailed treatment of the unknown N_t problem is beyond the scope of this paper (as this would involve nontrivial specification and analysis of target birth/death processes), and it will be examined in future work.

However, it is worth emphasizing that the known N_t solution strategy presented here (although somewhat limiting) would actually provide most of the basis to the unknown N_t problem via such a modified OIS procedure. It is thus important to establish the known N_t case first to provide a solid foundation for the more general and complex unknown N_t case. On a more practical level, the fixed N_t assumption can also still be very useful in many applications. For instance, many widely used JPDA tracking systems are based on the known N_t assumption, and they typically use heuristic outer-loop logic to add/delete tracks in the full data association filter as time moves forward [18]. Similar heuristics could be adapted and implemented for the HMM/RBPF to dynamically adjust N_t . The main tradeoff of using such heuristics is that they would not provide any sort of guarantees on performance, despite being potentially easier/cheaper to implement than a formal full RBPF solution that accounts for uncertainty in N_t . More recently, [25] presented a potential “middle-ground” modeling approach that somewhat relaxes the fixed N_t assumption by defining a Markov target counting process for an arbitrary upper bound on N_t using the single-target HMM derived here. This approach interestingly circumvents the need to perform data association altogether and naturally accommodates variable target numbers below the upper bound, although it is only able to resolve and track “hot spot” target clusters rather than individual targets. These and other alternatives for dealing with unknown N_t will also be examined in future work.

D. Extracting and Assessing Point State Estimates with the HMM/RBPF

In the multitarget HMM/RBPF, filtering beliefs are a mix of nonparametric weighted samples for discrete data association variables and HMM beliefs for discretized target states. Because these beliefs are in general multimodal, and it is assumed that target state estimation can be decoupled from the U2R2 control/planning problem, it is assumed here that crisp point estimates for each target’s state can be obtained from the maximum a posteriori (MAP) state of each target’s HMM Bayes filtering distribution at each time step. In addition to their simple intuitive appeal, MAP estimates can be interpreted as the implicit minimizers of an expected infinity norm error loss function for Bayes’s optimal point estimation [4].

Unlike other estimators, such as the conditional mean (the expected mean squared error loss minimizer), a MAP approach ensures that resulting state estimates sensibly reside in regions of highest posterior probability in the Bayes filter.

However, MAP estimation also raises three subtle issues for the HMM/RBPF. First, the Bayes posterior may not always have a unique MAP value, i.e., different states could be equally likely. Depending on the application, this can be practically resolved in many ways, e.g., randomly select among the alternatives.

Second, at least two different kinds of MAP estimates can be extracted from the HMM/RBPF at each time step. The first MAP estimator considered here, referred to as the “MAP particle” (MAPP), looks for the most likely particle (i.e., most likely data assignments) up to a given time step i and then uses the corresponding MAP values of each target’s HMM state filter (for that particle) for each time step up to i :

$$\hat{C}_i^{\text{MAPP}} = \arg \max P(C_i | \mathcal{Y}_{1:i}, \mathcal{Z}_{1:i}, C_{1:i-1}) \approx \arg \max \sum_{s=1}^{N_p} w_i^s \delta(C_i, C_i^s) \quad (57)$$

$$\hat{\zeta}_i^{t, \text{MAPP}} = \arg \max P(\zeta_i^t | \hat{C}_i^{\text{MAPP}}, \mathcal{Y}_{1:i}, \mathcal{Z}_{1:i}, C_{1:i-1}), \quad t \in \{1, \dots, N_t\} \quad (58)$$

The second MAP filter, called the “marginal MAP” (MMAP), first computes the weighted average of each target’s HMM state beliefs at each time step across all association hypothesis particles, and then it assigns the MAP state estimate based on this average:

$$\hat{\zeta}_i^{t, \text{MMAP}} = \arg \max P(\zeta_i^t | \mathcal{Y}_{1:i}, \mathcal{Z}_{1:i}) = \arg \max_{C_{1:i}} \sum P(C_{1:i} | \mathcal{Y}_{1:i}, \mathcal{Z}_{1:i}) P(\zeta_i^t | C_{1:i}, \mathcal{Y}_{1:i}, \mathcal{Z}_{1:i}) \quad (59)$$

$$\approx \arg \max_{C_{1:i}} \sum_{s=1}^{N_p} w_i^s \delta(C_{1:i}, C_{1:i}^s) P(\zeta_i^t | C_{1:i}, \mathcal{Y}_{1:i}, \mathcal{Z}_{1:i}) = \arg \max_{s=1}^{N_p} w_i^s P(\zeta_i^t | C_{1:i}^s, \mathcal{Y}_{1:i}, \mathcal{Z}_{1:i}) \quad (60)$$

Because data association variables are marginalized out, no “commitment” to a particular hypothesis is required to estimate target states. However, information from all hypotheses still contributes to the final estimates via the particle weights.

Finally, regardless of which point estimation strategy is used, it remains to determine how estimation errors for discrete HMM states should be properly assessed. To account for road network constraints in state error calculations, the undirected road network graph distance between two states (i.e., absolute number of hops between discretized cells for each HMM state) can be used to compute “along-the-road distance” errors. This provides a more conservative target state error metric than Euclidean distances between geospatial state coordinates on the road network, which ignore road constraints and target heading. The MAP estimation performance for each target is thus measured here by the graph distance error (GDE), where the GDE is defined as the shortest along-the-road measure of displacement (number of consecutive cells) between the true target location state according to the Markov model and the MAP state estimate. The GDE implicitly accounts for errors in target orientation along twoway roads because these are modeled by two distinct edges with different cells and HMM states.

V. Numerical Simulation Results

This section presents simulated multitarget localization results for the proposed HMM Bayes filter on two different road network models by considering both known and unknown data association scenarios. In these simulations, the ground-truth motion for each target follows a discrete time hybrid stochastic dynamics model for each road network, as described in Sec. II.A. The UGSs in each network are all assumed to have the same coverage range (one cell corresponding to the UGS location) as well as identical P_{FA} and P_D values.

The simulated road networks and UGS placements are the same as those shown in Fig. 8. In the simple cyclical “small” road network (road network 1) in Fig. 8a, each target circulates the road network indefinitely, and there is a single twoway road at the bottom of the network (highlighted in red). The inner pair of UGSs along the twoway edge cover only the right-to-left direction, whereas the outer pair cover only the left-to-right direction. In the “large” road network (road network 2) in Fig. 8b, each target can terminate at one of two possible end nodes (colored orange). The road segments in this network are longer, more connected, and each covered by only a single UGS in the middle of the segment (with the exception of edge 8, which has no UGS at all).

Monte Carlo simulations of true target motions, as well as false-alarm and missed detection events, were recorded for five targets moving simultaneously and independently in each road network for a range of eight typical combined P_D and P_{FA} values: $P_D \in \{1, 0.95\}$ and

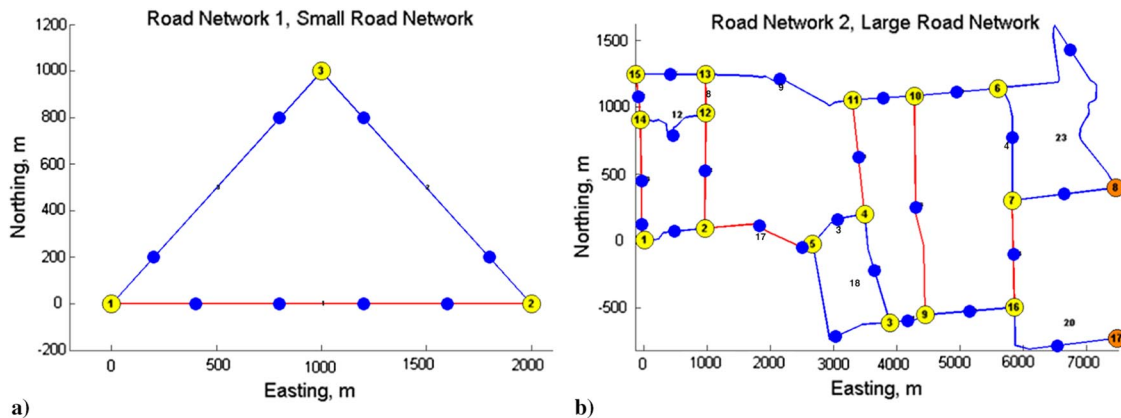


Fig. 8 Simulation study networks: a) road network 1; and b) road network 2. Twoway roads shown as red edges; terminating nodes shown in orange; and UGSs indicated by blue dots.

$P_{FA} \in \{0, 0.05, 0.10, 0.15\}$. Ten independent simulations were recorded at each combined setting using the same set of initial target locations. The resulting negative and positive UGS data for each road network were then processed by a set of baseline “oracle” HMM filters (one for each target, with knowledge of the true target association labels for each positive UGS detection) and the HMM/RBPF (which had no knowledge of the true target association labels). Each HMM/RBPF simulation was run with four different sample sizes: $N_s \in [100, 200, 500, 1000]$. All simulations were performed in MATLAB on a Windows 7 desktop (Intel Core i5, with 8 GB of RAM).

The underlying Markov models for the oracle HMM and HMM/RBPF filters were defined using discretization parameters ($\Delta T = 4.167$ s, $\Delta c = 50$ m) for road network 1 and ($\Delta T = 5.769$ s, $\Delta c = 75$ m) for road network 2, resulting in 140 and 499 states, respectively. The simulations for road networks 1 and 2 were run for 450 time steps and 150 time steps, respectively (where the latter was the average time for all five targets to reach a terminating node).

A. Oracle HMM Filtering Results (Baseline)

The combined use of perfect data association and negative information fusion provides a high baseline for assessing target localization performance via the HMM Bayes filter approximation. In this case, because there is only one filter per target, the MAP state computed by the oracle HMM filter for a given target at each time step is treated as that target’s state estimate for that time step. Figures 9a–9d show the graph distance error cumulative distribution functions (CDFs) on both road networks for both P_D levels (the false-alarm rate does not affect these results because the oracle HMM filters screen these out perfectly). The CDFs indicate the relative contribution of different GDE levels to each target track over all simulations for a particular sensing configuration (such that perfect localization performance places all probability mass at zero error). These results show that, with perfect data association and with fusion of negative information, the HMM filter can accurately localize each target to within three cells of their actual positions approximately 98% of the time. As expected, some slightly larger errors are more likely to appear as P_D is lowered, although these errors rarely exceed 10 cells. The results for road network 2 show that the oracle HMM filters can localize each target to within three cells of their true positions about 85% of the time; GDEs of up to 11 cells account for about 95% of the errors. On average, the oracle HMM only required about 6 ms to run through all data for each target in road networks 1 and 2.

The performance discrepancy for both road networks is explained by the fact that road network 1 is smaller, has more simply connected cyclic/nonterminating road segments, and has better UGS coverage. These properties make each target more persistently observable, and thus limit uncertainty growth in the HMM tracks between positive measurement updates. In contrast, road network 2 covers a larger area with longer individual road segments, has more branching connections, and has (at most) only a single UGS in the middle of edge/road segment. These factors greatly reduce target observability and allow HMM track uncertainty to grow more rapidly between positive measurement updates. As shown

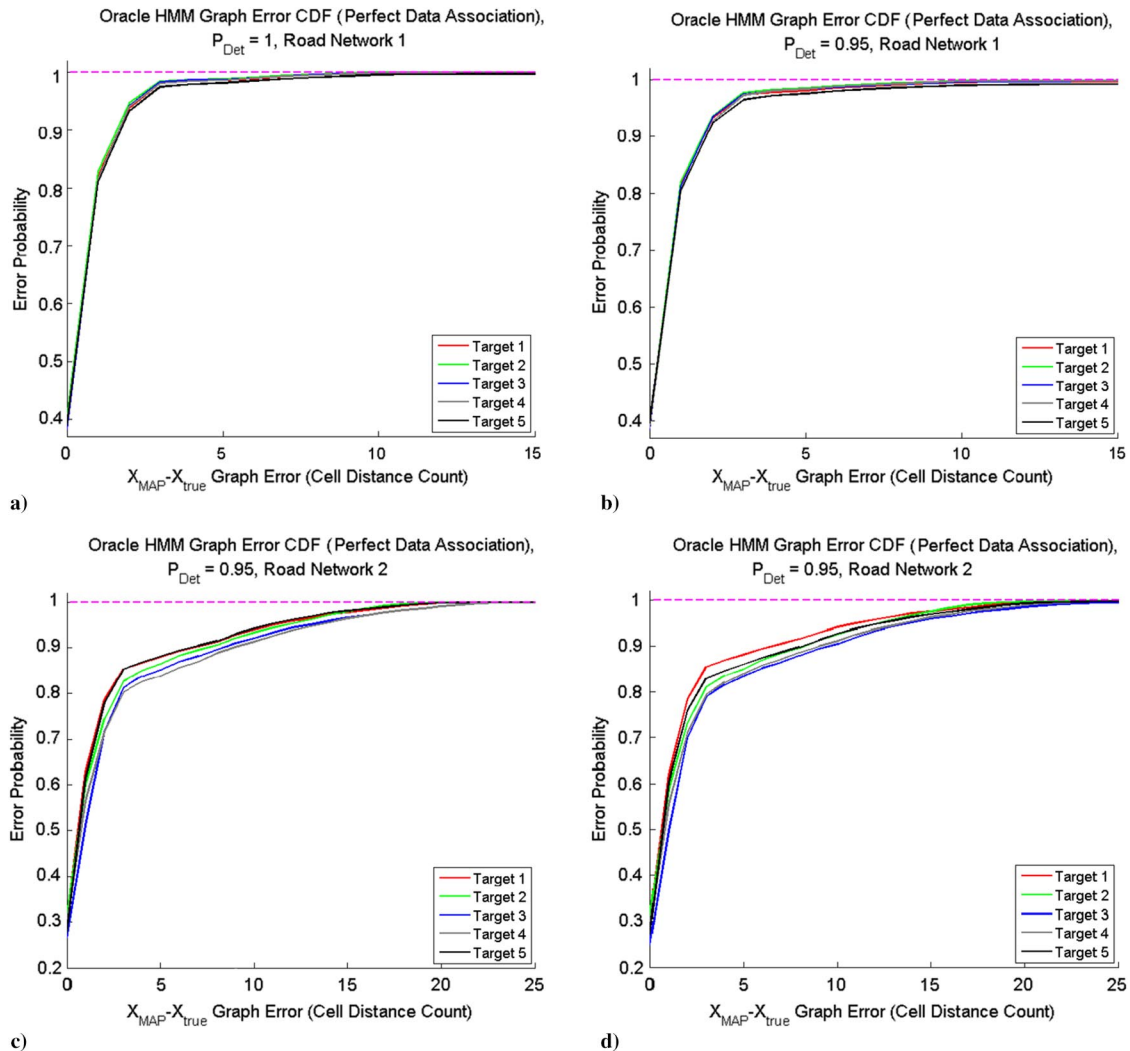


Fig. 9 Oracle HMM filter error CDFs: a–b) small road network 1; c–d) large road network 2.

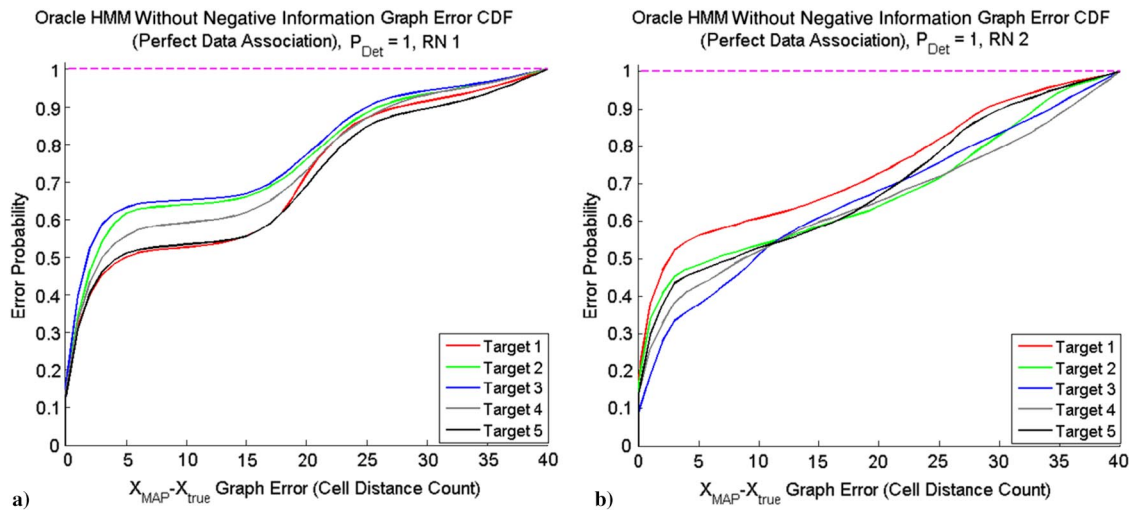


Fig. 10 Oracle HMM filter error CDFs without negative information: a–b) network 1; c–d) network 2 (RN, road network).

earlier in Figs. 5i–5p, the probability mass for each target's predicted locations tends to spread out more quickly and uniformly across road network 2 if positive measurements are not taken for a long period of time, making MAP state estimates less reliable in the interim. However, because the majority of the estimation errors fall within three cells, the HMM filters still remain consistent and can reliably recover in such sparse sensing situations. These results underscore the inherent difficulty of achieving precision target localization: even when data association is perfectly known, the inherent sparsity of the positive data limits the usefulness of point estimates, and thus necessitates full state distribution tracking.

It is also interesting to consider the extent to which negative vs positive UGS data contribute to the best achievable accuracy of the HMM filter. The oracle HMM filter's performance was therefore also simulated without negative UGS data fusion at each time step, i.e., such that only positive UGS data are fused when available. Figure 10 shows the resulting error CDFs with $P_D = 1$ (results for $P_D = 0.95$ are similar). These show significantly worse MAP localization performance as compared to the full oracle HMM filter using both negative and positive UGS data, with substantially larger proportions (40–50%) of GDEs of 10 cells or more in both road networks. These results demonstrate the importance of fusing both negative and positive UGS data whenever possible. In particular, negative information is needed to prune unlikely prediction hypotheses and contain the spread of probability mass across the road network between positive data fusion events. This in turn helps combat “drift” in MAP state estimates that accumulate across successive pure state prediction updates, due to motion model errors and process uncertainty.

B. HMM/RBPF Results

The results for the HMM/RBPF are presented here for the $P_D = 1$ case only, as the results for $P_D = 0.95$ are not substantially different (aside from the fact that they show somewhat worse MAP localization performance, as expected). Furthermore, all results are shown in terms of the MAP particle point state estimate because the results obtained for the marginal MAP state estimate are quite similar.

Figure 11 shows the GDE CDFs with $P_{FA} = 0$ for $N_s = 100$ and $N_s = 1000$; these simulations contain no clutter measurements, although there is still the possibility that target labels can get confused. Localization performance consistently improves for three of the five targets on road network 1 as the number of particles increases; as expected, this allows the HMM/RBPF to explore more association hypotheses and increases the probability that it will identify the correct association hypotheses for positive UGS data over time, thereby yielding a localization performance that is comparable to the high baseline oracle HMM filters for those targets. However, two of the five targets (targets 1 and 5) are frequently confused for one another in road network 1, leading to a noticeably higher proportion of large errors. Closer inspection reveals that these two targets tended to remain in close proximity to each other, and thus could not always be easily distinguished, as their HMM state distributions tended to heavily overlap. Such target label switching occurs less frequently for road network 2, although there are also significantly fewer positive UGS data events, leading to increased uncertainty and error in each target's MAP state estimates. As N_s increases, localization performance becomes more consistent across all targets and comparable to the high baseline oracle HMM filtering results.

Figures 12 and 13 show the GDE CDFs for simulations where $P_{FA} > 0$ for an illustrative set of P_{FA} and N_s values on both road networks. These results show an interesting interaction between two expected trends: increasing P_{FA} tends to increase the probability of obtaining large errors (because clutter can be incorrectly categorized as true target data, and vice versa), whereas increasing N_s tends to improve the probability of assigning correct association hypotheses and improve the consistency of localization performance across all targets. The selection of N_s is even more crucial when $P_{FA} > 0$ because an increase in P_{FA} also implicitly increases the number of positive UGS data received (i.e., more false alarms create more data to process). This in turn can drastically increase the size of the hypothesis space to be explored for all five targets, even when the number of true positive target measurements remains the same. In these scenarios, the effective true signal content can be expressed in terms of the number of true positives versus the total number of positive measurements (true positives and false alarms). For the P_{FA} values considered, the signal content changes drastically for road network 1, going from average values of 100% ($P_{FA} = 0$, i.e., zero false returns in 115 positive returns on average) to 42% ($P_{FA} = 0.05$, or 88 false returns in 151 positive returns) to 27% ($P_{FA} = 0.10$, 226 false returns in 310 positive returns) to 20% ($P_{FA} = 0.15$, or 370 false returns in 463 positive returns). Given a fixed number of particles N_s , the presence of clutter makes it even more difficult for the HMM/RBPF to correctly “untangle” confused target labels via each particle's sampled association history while also simultaneously screening out false alarms from new/old data. In other words, N_s needs to grow much more rapidly to keep up with the additional hypotheses that need to be explored as P_{FA} increases.

Figure 14 shows that the HMM/RBPF behaves more conservatively (rather than optimistically) as the clutter rate increases. These plots show the average data label confusion matrices taken from the most likely particle at the end of $N_s = 1000$ simulations for both road networks (the numbers along the main diagonal show the average probability of correct association). As P_{FA} increases, the most likely particle in the HMM/RBPF increasingly tends to incorrectly label true positive data events as false alarms. At the same time, the HMM/RBPF also does consistently

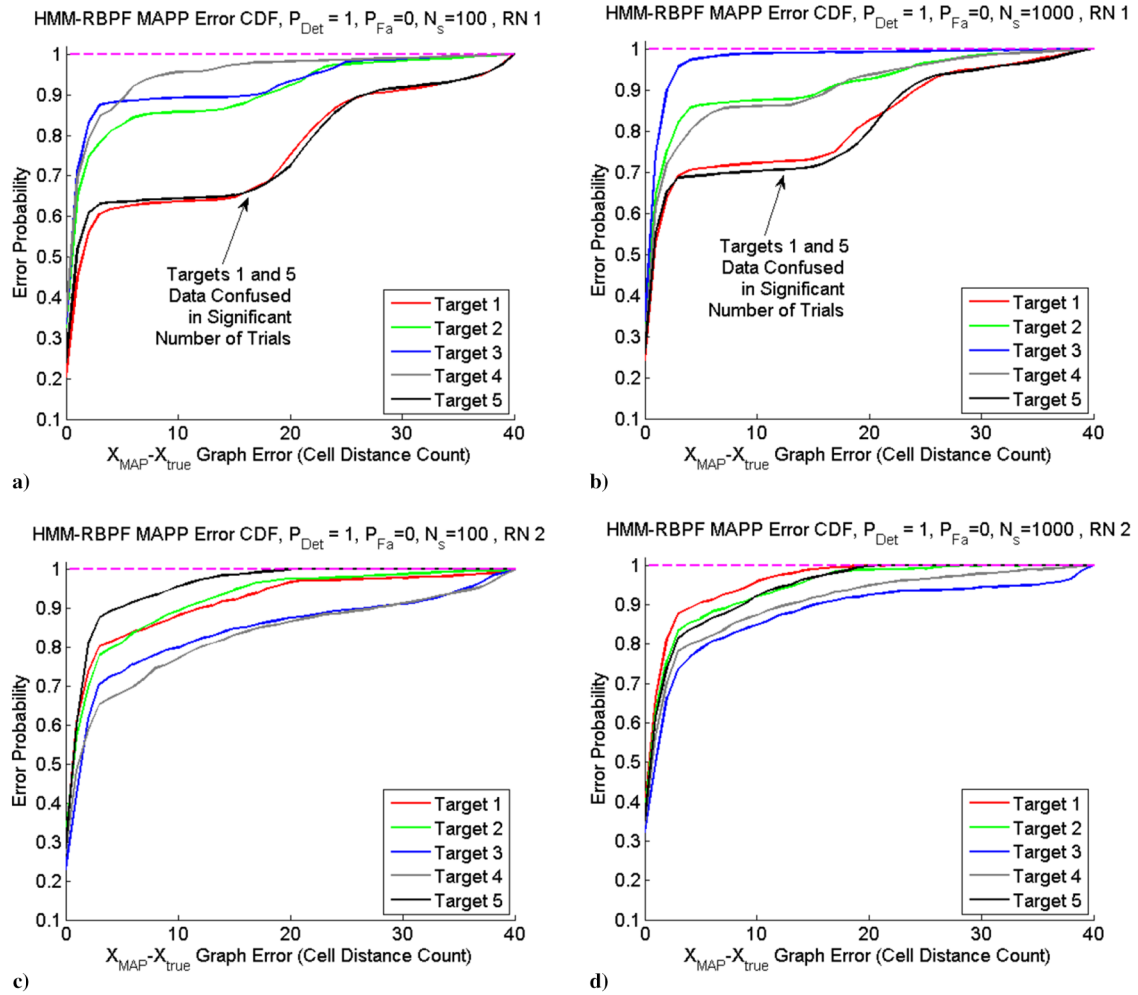


Fig. 11 RBPF errors for $P_{FA} = 0$ and different sample sizes N_s : a–b) network 1; c–d) network 2.

well at correctly filtering out false alarms, regardless of P_{FA} . These results indicate that the HMM/RBPF of each target's state distribution is more spread out across the road network in high-clutter cases. This thin multimodal spreading of probability mass in turn decreases the likelihood that any given positive measurement belongs to a particular target, compared to the prior false-alarm probability. Hence, the HMM/RBPF will only attempt to assign positive target labels when there is relatively little uncertainty in the target's state. As shown by the earlier results in Fig. 10, this is achieved in instances where sparse positive UGS data can be augmented by an adequate amount of negative UGS data, which helps to correct errors in predicted MAP state estimates and (more important) weed out poor positive data association hypotheses. However, as the expected clutter rate increases, the expected number of useful negative measurements also decreases because true no-detection events that can eliminate spurious modes of a multimodal target distribution are instead replaced by false alarms.

It is also worth considering whether the HMM/RBPF suffers from particle depletion effects at higher P_{FA} values, i.e., whether there is ever a catastrophic drop in the effective sample size (ESS) due to inefficient sampling of the data association hypothesis space. Figure 15 shows that this is not the case because the ESS in a typical high P_{FA} simulation always stays near or above the resampling threshold of $0.5 * N_s$. This means that the HMM/RBPF always maintains a "healthy" collection of particles over the data association hypothesis space with statistically consistent information about each target, even if N_s is not always large enough to cover the entire hypothesis space for high P_{FA} scenarios. Note that the ESS dropoffs between resampling events are also expected due to accumulation of negative information, which alters the particle weights as per Eq. (55).

C. Discussion

The HMM filter provides a complete yet compact probabilistic description of each target's state given all available UGS information. When the number of targets and data associations are known, the HMM filter yields accurate MAP localization results with relatively sparse sets of positive UGS observations. When data associations are unknown but the number of targets is known, the HMM/RBPF provides fairly precise MAP target localization performance in low-to-moderate false-alarm environments, even with a reasonably small sample size N_s . In higher false-alarm environments, the MAP localization accuracy degrades and the HMM/RBPF requires significantly larger N_s for MAP localization to be as precise and reliable as possible. Note that alternative joint particle and state MAP estimation strategies (e.g., application of the Viterbi algorithm to infer joint target states for the most likely particle at each time step) might possibly produce better localization performance than either MAPP or MMAP state estimation in certain cases. However, such approaches are not explored here because an online Viterbi implementation would require repeatedly solving a dynamic programming problem that becomes more computationally expensive over time.

Although highly precise localization is beneficial to the U2R2 isolation phase described in Sec. I, it is not strictly required (and indeed impossible to ensure according to Fig. 9). Rather, the main requirement for target localization is that the most likely portions of the road-network-containing targets be identified and prioritized for planning UAV flyby imaging and interception tasks. Specifically, it should be possible

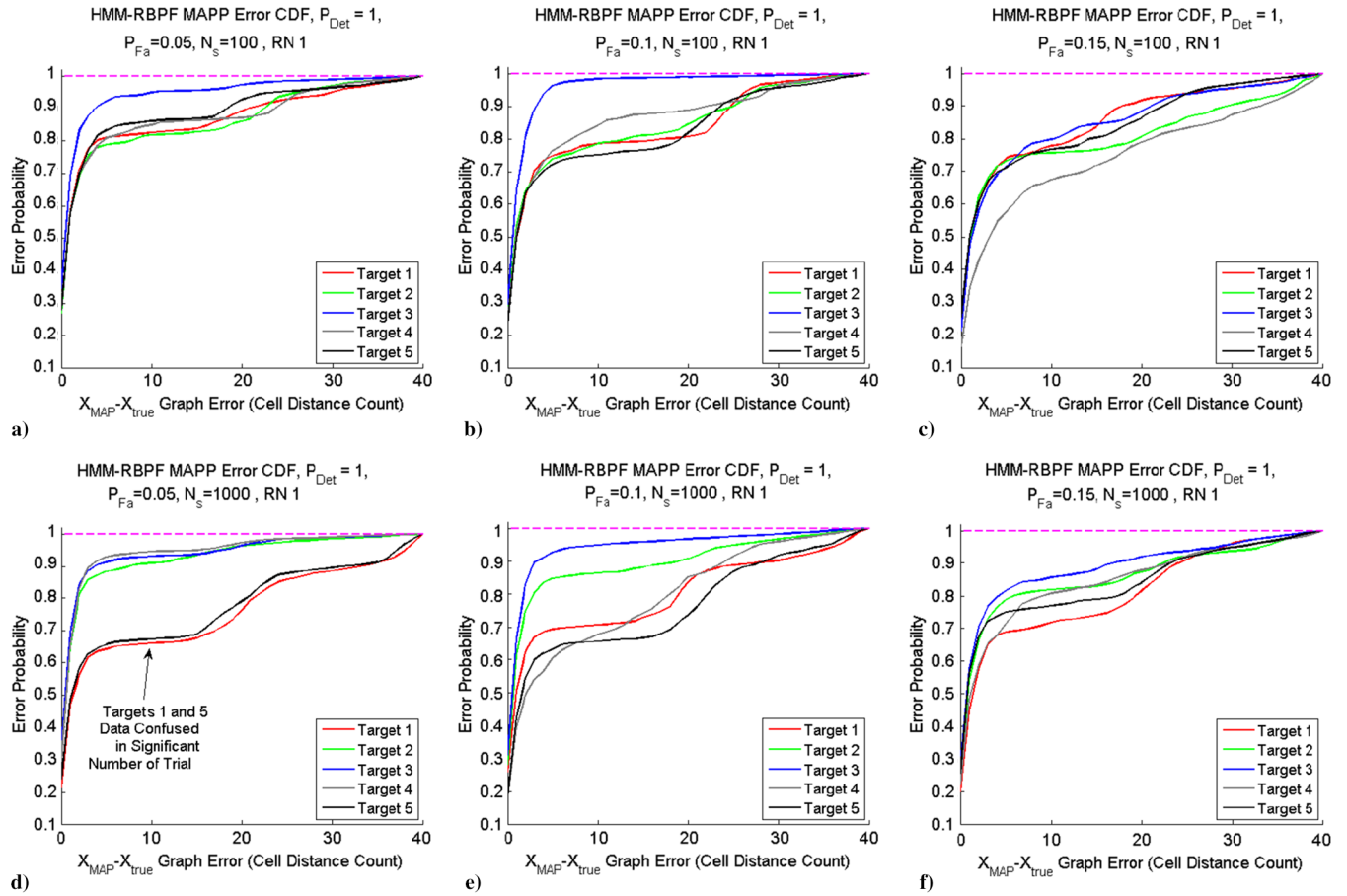


Fig. 12 RBPF errors on network 1 for $P_{FA} > 0$ vs N_s : a-c) $N_s = 100$; and d-f) $N_s = 1000$.

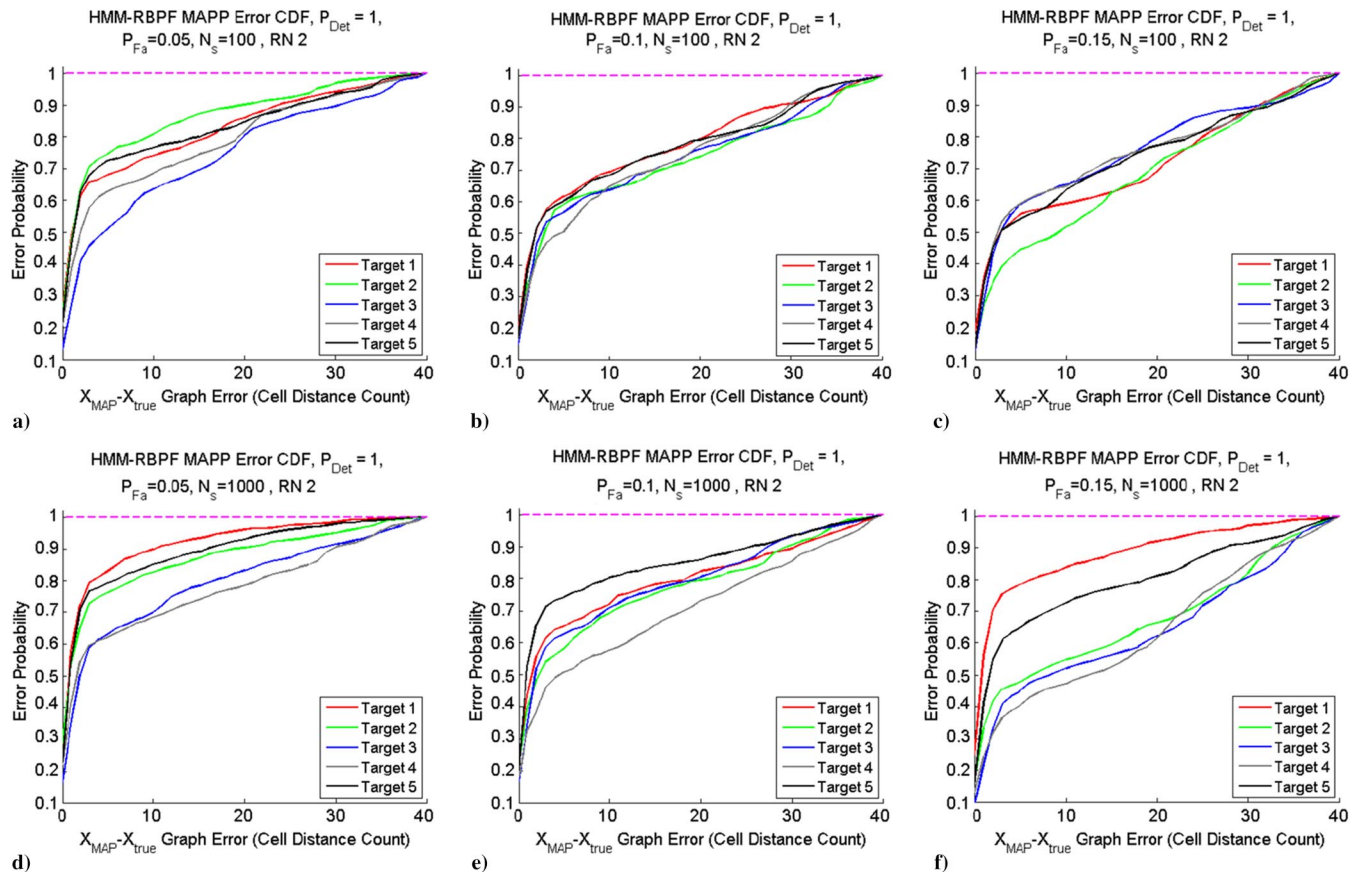


Fig. 13 RBPF errors on network 2 for $P_{FA} > 0$ vs N_s : a-c) $N_s = 100$; and d-f) $N_s = 1000$.

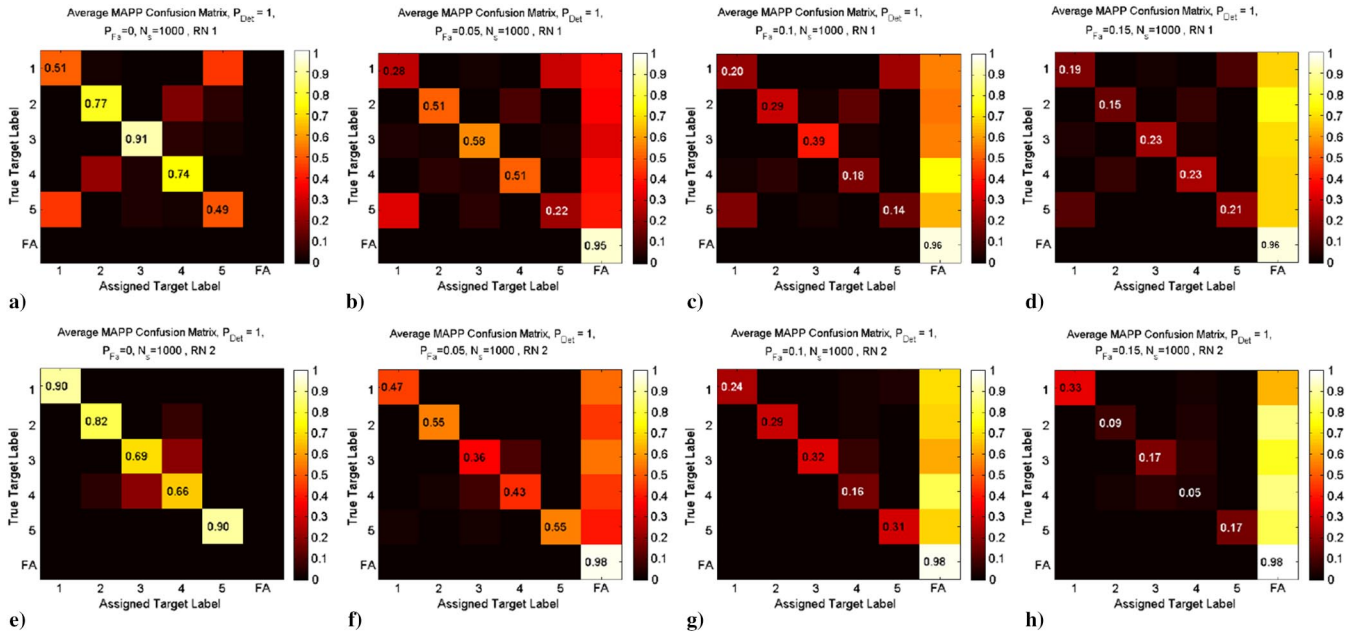


Fig. 14 Positive UGS data confusion matrices for MAPP particle for various P_{FA} : a-d) network 1; e-h) network 2.

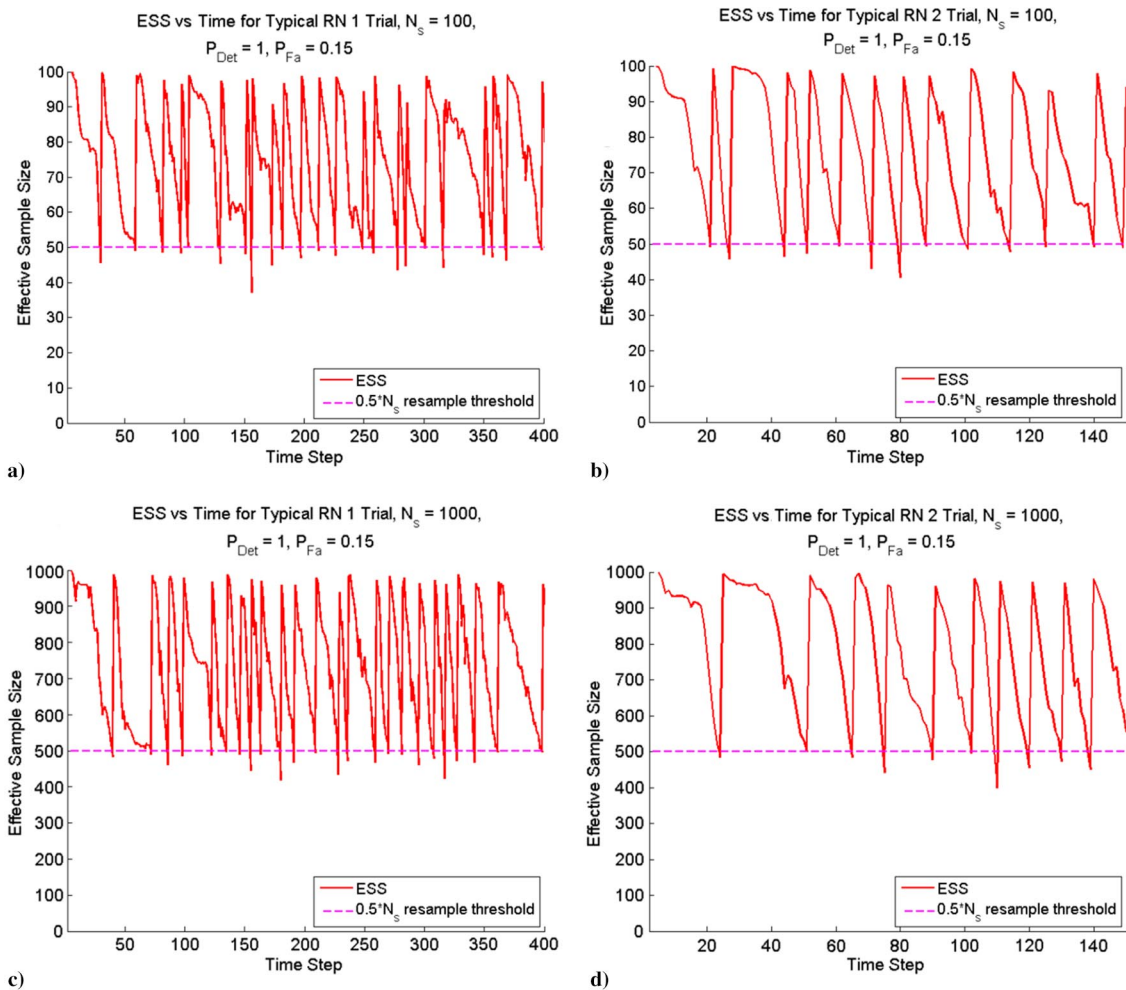


Fig. 15 Typical RBPF ESS results for networks 1 and 2 ($P_D = 1, P_{FA} = 0.15$): a-b) $N_s = 100$; c-d) $N_s = 1000$.

(at a minimum) to identify the correct road segments (edges) that targets are on, the correct direction of travel for each target, and which road segments they are likely to travel to next. The simulation results for the known and unknown data association cases indicate that the HMM filter and HMM/RBPF provide theoretically viable solutions for determining these, even in highly cluttered sensing scenarios. In such cases, the importance of maintaining accurate probability distributions over target states is paramount.

However, the computational requirements for the HMM/RBPF deserve extra consideration because these depend on several coupled problem variables and lead to subtle cost/benefit tradeoffs in different application contexts. In general, the HMM/RBPF runs N_s oracle HMM filters for each target; this processing can be easily parallelized across particles and targets simultaneously. Several steps for the HMM filter calculations can also be optimized across particles and targets to reduce redundant calculations, as well as exploit STM and sensor model sparsity for state prediction and Bayes measurement updates, respectively. With the sampling time ΔT on the order of a few seconds, it is generally feasible to run the HMM/RBPF online and in real time, although the upper range of N_s and/or N_t will be highly dependent on specific hardware implementations (e.g., onboard flight hardware will be more restricted than a ground base station computer).

The two other major computational factors to consider are the cost of performing particle resampling (which depends on N_s and the ESS threshold for resampling) and the maximum number of association hypotheses spawned at any given time step (which depends on the number of targets N_t , the number of UGS N_u , and P_{FA}). Both processes are coupled by the arrival of new UGS observations because N_s must be tuned to adequately cover the most likely portions of the current association hypothesis space, whereas the ESS threshold must be tuned so that unlikely association hypotheses are discarded at an appropriate rate via resampling. In settings with mild clutter, i.e., $P_{FA} < 0.1$, a relatively small N_s of 100–200 offers performance that is comparable to that obtained with $N_s = 1000$ for a wide range of ESS thresholds. When P_{FA} gets larger, N_s must necessarily be larger and the ESS threshold must also be selected more carefully because the long-term best association hypotheses require a longer time to build up evidence from new UGS observations and achieve high likelihood. If resampling occurs too often in between positive UGS observations, the HMM/RBPF could discard particles for the best hypotheses too soon as a result.

This latter issue also points to an interesting fundamental limitation of the HMM/RBPF: because it only runs forward in time as a statistical filter, it cannot revisit/revise sampled hypotheses and target states from the past in light of new UGS data. However, if it were possible to revisit previous estimates over a sufficiently large time window, the effects of “greedy” hypothesis pruning could be mitigated. In high P_{FA} scenarios, it may therefore be desirable to resample entire block windows of target association labels rather than just single-target association labels at a time. Although not explored here, retrodictive smoothing-and-resampling approaches have been shown to improve the performance of the RBPF for highly nonlinear/non-Gaussian state estimation problems, although at extra computational cost [10]. Retrodictive smoothing and sampling could theoretically be implemented within the HMM/RBPF by exploiting HMM forward–backward smoothing methods to each set of target states. Such methods could also directly extend the HMM/RBPF to the more general and open problem of fusing asynchronous/late UGS data.

Remark 4: Regarding comparisons to other approximate filtering approaches, finally, although there are no established baselines for the U2R2 problem in its most general form considered here, it is worth mentioning how the HMM/RBPF compares to some alternative Bayes filtering approximations for each target in the aforementioned perfect and imperfect data association scenarios. The huge variety of Bayes filter approximations precludes an exhaustive comparison, but two closely related candidate approaches can be considered: the standard bootstrap particle filter (BPF) [32], based on the continuous dynamics model prediction and post hoc probability discretization described at the end of Sec. III; and an adaptation of the histogram filter (HF) developed by [3], which was originally developed for a single mobile UAV sensor rather than a network of static UGSs. The results for the BPF and HF formulations are not shown here because it was found that (in either the perfect or imperfect data association cases) the BPF and HF were much more expensive to implement as compared to the HMM-based approach, but they provided little/no benefit in terms of target state estimation performance.

For the perfect association case, the BPF and HF provide similar estimation performance to the oracle HMM. This is unsurprising because positive measurement events collapse target state posterior distributions to a single cell in all methods. Interestingly, the BPF and HF avoid the probability “bleeding effect” that arises from HMM state discretization (noted in Sec. III); this makes “dead reckoning” for target state predictions more accurate than in the HMM filter over extended stretches of time that are free of positive UGS returns (although this is mitigated by negative UGS returns). However, in the HF, the dynamics update step for cell probabilities in each road edge must be performed on a per cell basis, using a discrete approximation of Eq. (4) at every time step for each target; this is costly even for the relatively simple linear stochastic kinematics model assumed here. Likewise, the BPF incurs a high cost because each target requires its own set of dynamic particles, which must be individually updated via the continuous stochastic dynamics model. The BPF is also subject to extra variance from the effects of resampling each target’s particle set: a process that also significantly increases the BPF’s computational cost.

Imperfect data association only exacerbates these issues for the BPF and HF. The BPF becomes very expensive to implement because each particle must not only sample values for data association histories but all (continuous) target states as well. This increase in sampling dimensionality leads to a growth in target state estimation error variance. The HMM/RBPF avoids this effect via Rao–Blackwellization [30]. The BPF thus requires many more particles than the HMM/RBPF to achieve a given level of estimation error performance, especially as the false-alarm probability P_{FA} increases. Although HFs for each target could in theory also be used in the RBPF in place of HMM filters, the required HF computations remain more expensive.

VI. Conclusions

Motivated by the unmanned aerial vehicle (UAV)–unattended ground sensor (UGS) road reconnaissance problem, this paper considered the problem of tracking multiple dynamic targets on a road network with sparse highly localized UGS data subject to clutter and missed detections. Three novel contributions were made. First, a discrete time Markov chain model for single-target probabilistic state modeling was derived using approximate spatial discretization of the multimodal and highly non-Gaussian target state probability density function on the constrained road network. Second, the Markov chain model was then used to develop a hidden Markov model for single-target localization with UGS data, under the assumption of perfectly known data association. Finally, the hidden Markov model (HMM) was extended to the HMM Rao–Blackwellized particle filter to solve the corresponding multitarget tracking data association problem in the presence of clutter and missed detections.

The proposed Markov modeling approach permits easy generation of accurate probabilistic models from a priori road network structure information. The HMM approach also naturally enables fast sparse calculations of complex target state probability distributions for online prediction, filtering, and other state estimation operations, using both positive and negative UGS information. The Rao–Blackwellized particle filter (RBPF) allows the HMM to be easily extended to the more general problem of tracking a known number of targets in clutter, without the need for gating heuristics. The resulting HMM/RBPF provides a fully Bayesian solution to the data association problem, enabling an online exploration of the association hypotheses space that leverages the computational advantages of exact HMM inference for efficient state estimation and data label inference. This leads to simpler implementation and better scalability than purely parametric or nonparametric filtering methods: neither of which have been extended to the data association problem for this domain.

Numerical simulations demonstrated the perfect-association oracle HMM and HMM/RBPF’s effectiveness on challenging multitarget tracking scenarios. With data associations perfectly known, the oracle HMM provided highly accurate maximum a posteriori (MAP) localization using both negative and positive UGS data, but it provided substantially worse performance using positive data only. This underscored the importance of leveraging negative information in this problem domain. The HMM/RBPF provided highly accurate MAP localization performance in mild

false-alarm rate environments with sparse detection data and small particle sample sizes. MAP localization performance degraded at high false-alarm rates due to the HMM/RBPF's inability to easily distinguish true measurements from false alarms with limited sample sizes on short resampling timescales. However, false alarms were almost always correctly labeled and rejected by the HMM/RBPF's most likely particle in both low and high false-alarm rate cases. The proposed techniques permit online and real-time implementation for UAV-based road network reconnaissance and survey applications, such as the isolation phase of the U2R2 problem.

Acknowledgments

The work of Nisar Ahmed was supported in part by an American Society for Engineering Education Air Force Summer Faculty Fellowship. The work of Yongcan Cao was supported in part by the U.S. Air Force Office of Scientific Research under award number FA9550-16-1-0504.

References

- [1] Kingston, D., "Intruder Tracking Using UAV Teams and Ground Sensor Networks," *German Aviation and Aerospace Conference 2012 (DLRK 2012)*, German Soc. for Aeronautics and Astronautics (DLGR), 2012.
- [2] Ahmed, N., Casbeer, D., Cao, Y., and Kingston, D., "Bayesian Hidden Markov Models for UAV-Enabled Target Localization on Road Networks with Soft-Hard Data," *SPIE Defense + Security*, International Society for Optics and Photonics, SPIE Paper 94640Q, Bellingham, WA, 2015.
- [3] Niedfeldt, P., Kingston, D., and Beard, R., "Vehicle State Estimation within a Road Network Using a Bayesian Filter," *American Control Conference 2011 (ACC 2011)*, IEEE Publ., Piscataway, NJ, 2011, pp. 4910–4915.
- [4] Bar-Shalom, Y., Li, X., and Kirubarajan, T., *Estimation with Applications to Navigation and Tracking*, Wiley, New York, 2001, pp. 89–115.
- [5] Challa, S., Evans, R. J., and Wang, X., "A Bayesian Solution and its Approximations to Out-of-Sequence Measurement Problems," *Information Fusion*, Vol. 4, No. 3, Sept. 2003, pp. 185–199.
doi:10.1016/S1566-2535(03)00037-X
- [6] Maskell, S., Everitt, R., Wright, R., and Briers, M., "Multi-Target Out-of-Sequence Data Association," *7th International Conference on Information Fusion (FUSION 2004)*, IEEE Publ., Piscataway, NJ, 2004, pp. 1–8.
- [7] Chan, S., and Paffenroth, R., "Out-of-Sequence Measurement Updates for Multi-Hypothesis Tracking Algorithms," *SPIE Defense and Security Symposium*, edited by O. E. Drummond, International Society for Optics and Photonics, SPIE Paper 69691H, Bellingham, WA, April 2008.
- [8] Govaers, F., and Koch, W., "Out-of-Sequence Processing of Cluttered Sensor Data Using Multiple Evolution Models," *13th International Conference on Information Fusion*, IEEE Publ., Piscataway, NJ, July 2010, pp. 1–7.
- [9] Zhang, S., and Bar-Shalom, Y., "Out-of-Sequence Measurement Processing for Particle Filter: Exact Bayesian Solution," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 48, No. 4, 2012, pp. 2818–2831.
doi:10.1109/TAES.2012.6324663
- [10] Berntorp, K., and Robertsson, A., "Rao–Blackwellized Particle Filters with Out-of-Sequence Measurement Processing," *IEEE Transactions on Signal Processing*, Vol. 62, No. 24, 2014, pp. 6454–6467.
doi:10.1109/TSP.2014.2365763
- [11] Lerner, U., "Hybrid Bayesian Networks for Reasoning About Complex Systems," Ph.D. Thesis, Stanford Univ., Palo Alto, CA, 2002.
- [12] Mahler, R. P. S., *Statistical Multisource-Multitarget Information Fusion*, Artech House, Boston, MA, 2007, pp. 305–538.
- [13] Pannetier, B., Moras, J., Dezert, J., and Sella, G., "Study of Data Fusion Algorithms Applied to Unattended Ground Sensor Network," *Proceedings of SPIE: Signal Processing, Sensor/Information Fusion, and Target Recognition XXIII*, Vol. 9091, 2014, Paper 9091.
doi:10.1117/12.2049819
- [14] Chen, Y., Jilkov, V., and Li, X., "Multilane-Road Target Tracking Using Radar and Image Sensors," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 51, No. 1, 2015, pp. 65–80.
doi:10.1109/TAES.2014.120766
- [15] Ulmke, M., and Koch, W., "Road-Map Assisted Ground Moving Target Tracking," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 42, No. 4, 2006, pp. 1264–1274.
doi:10.1109/TAES.2006.314571
- [16] Agate, C., and Sullivan, K., "Road-Constrained Target Tracking and Identification a Particle Filter," *Optical Science and Technology, SPIE's 48th Annual Meeting*, No. 805, International Society for Optics and Photonics, Bellingham, WA, 2003, pp. 532–543.
- [17] Cheng, Y., and Singh, T., "Efficient Particle Filtering for Road-Constrained Target Tracking," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 43, No. 4, 2007, pp. 1454–1469.
doi:10.1109/TAES.2007.4441751
- [18] Bar-Shalom, Y., Daum, F., and Huang, J., "The Probabilistic Data Association Filter: Estimation in the Presence of Measurement Origin Uncertainty," *IEEE Control Systems Magazine*, Vol. 29, No. 6, Dec. 2009, pp. 82–100.
doi:10.1109/MCS.2009.934469
- [19] Blackman, S. S., "Multiple Hypothesis Tracking for Multiple Target Tracking," *IEEE Aerospace and Electronic Systems Magazine*, Vol. 19, No. 1, 2004, pp. 5–18.
doi:10.1109/MAES.2004.1263228
- [20] Bailey, T., Upcroft, B., and Durrant-Whyte, H., "Validation Gating for Non-Linear Non-Gaussian Target Tracking Computing," *9th International Conference on Information Fusion (FUSION 2006)*, IEEE Publ., Piscataway, NJ, 2006, pp. 1–6.
- [21] Schulz, D., Burgard, W., Fox, D., and Cremers, A. B., "People Tracking with Mobile Robots Using Sample-Based Joint Probabilistic Data Association Filters," *International Journal of Robotics Research*, Vol. 22, No. 2, Feb. 2003, pp. 99–116.
doi:10.1177/0278364903022002002
- [22] Wyffels, K., and Campbell, M., "Dynamic Extended Multi-Object Tracking," *IEEE Transactions on Robotics*, Vol. 31, No. 2, 2015, pp. 425–442.
doi:10.1109/TRO.2015.2409413
- [23] Koch, W., "On Exploiting "Negative" Sensor Evidence for Target Tracking and Sensor Data Fusion," *Information Fusion*, Vol. 8, No. 1, 2007, pp. 28–39.
doi:10.1016/j.inffus.2005.09.002
- [24] Rabiner, L., "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," *Proceedings of the IEEE*, Vol. 77, No. 2, 1989, pp. 257–286.
doi:10.1109/5.18626
- [25] Cao, Y., Casbeer, D., Ahmed, N., and Kingston, D., "Probabilistic Learning of Target Density Distribution," *AIAA InfoTechAerospace Forum at SciTech 2017*, AIAA Paper 2017-0790, 2017.
- [26] Grinstead, C. M., and Snell, J. L., *Introduction to Probability*, American Mathematical Soc., Providence, NY, 1998, pp. 405–470.
- [27] Bishop, C., *Pattern Recognition and Machine Learning*, Springer, New York, 2006, pp. 605–647.
- [28] Runnalls, A. R., "Kullback-Leibler Approach to Gaussian Mixture Reduction," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 43, No. 3, 2007, pp. 989–999.
doi:10.1109/TAES.2007.4383588

- [29] Särkkä, S., Vehtari, A., and Lampinen, J., “Rao–Blackwellized Particle Filter for Multiple Target Tracking,” *Information Fusion*, Vol. 8, No. 1, Jan. 2007, pp. 2–15.
doi:10.1016/j.inffus.2005.09.009
- [30] Liu, J. S., *Monte Carlo Strategies in Scientific Computing*, Springer, New York, 2001, pp. 39–49.
- [31] Rabiner, L., and Juang, B.-H., “An Introduction to Hidden Markov Models,” *IEEE ASSP Magazine*, Vol. 3, No. 1, 1986, pp. 4–16.
doi:10.1109/MASSP.1986.1165342
- [32] Arulampalam, M., Maskell, S., Gordon, N., and Clapp, T., “A Tutorial on Particle Filters for Online Nonlinear/Nongaussian Bayesian Tracking,” *IEEE Transactions on Signal Processing*, Vol. 50, No. 2, 2002, pp. 723–737,

J. P. How
Associate Editor