Distributed Tracking with a PHD Filter using Efficient Measurement Encoding

BIRUK K. HABTEMARIAM
AMPIKAITHASAN ARAVINTHAN
RATNASINGHAM THARMARASA
KUMARADEVAN PUNITHAKUMAR
THOMAS LANG
THIA KIRUBARAJAN

Probability Hypothesis Density (PHD) filter is a framework for multitarget tracking, which provides estimates for the number of targets as well as the individual target states. Sequential Monte Carlo (SMC) implementation of a PHD filter can be used for nonlinear non-Gaussian problems. However, the application of PHD-based state estimators for a distributed sensor network, where each tracking node runs its own PHD-based state estimator, is more challenging compared with single sensor tracking due to communication limitations. A distributed state estimator should use the available communication resources efficiently in order to avoid the degradation of filter performance. In this paper, a method that efficiently communicates encoded measurements between nodes while maintaining the filter accuracy is proposed. This coding is complicated in the presence of high clutter and instantaneous target births. This problem is mitigated using adaptive quantization and encoding techniques. The performance of the algorithm is quantified using a Posterior Cramér-Rao Lower Bound (PCRLB) that incorporates quantization errors. Simulation studies are performed to demonstrate the effectiveness of the proposed algorithm.

Manuscript received June 21, 2010; released for publication March 21, 2011.

Refereeing of this contribution was handled by Chee-Yee Chong.

Authors' addresses: B. K. Habtemariam, A. Aravinthan, R. Tharmarasa, and T. Kirubarajan, McMaster University, Hamilton, ON, Canada, E-mails: (habtembk@mcmaster.ca, aravinthan@ieee.org, tharman@mail.ece.mcmaster.ca, kiruba@mcmaster.ca); K. Punithakumar, GE Healthcare, London, ON, Canada, E-mail: (Kumaradevan.Punithakumar@ge.com); T. Lang, General Dynamics Canada, Ottawa, ON, Canada, E-mail: (tom.lang@gdcanada.com).

1. INTRODUCTION

The use of a large number of networked sensors, which can be deployed all over the surveillance region, has become feasible in tracking applications because of the availability of cheap sensors. The multisensor data need to be fused in order to fully utilize the information obtained in the network. A common practice in sensor network applications has been to process the collected data in a central processor. This architecture is known as the centralized sensor network [10, 26]. Centralized architectures are generally simpler to execute since the processing of data at one location can reduce the computational requirements of an algorithm. It is theoretically optimal if the network has enough communication bandwidth to send all the sensor data to the fusion node at every sampling time [3].

However, there are several drawbacks associated with the centralized architecture. First, the network relying on one processor to perform the task of every node in the network may result in a single-point failure. Second, in real-time applications, the central node may reside many hops away and sending data from one node to a central node may take too long. This may introduce latency, synchronization problems and imbalanced workload in the network. Further, the centralized architecture may utilize significant resources in communicating the data across the network. Distributed processing over the sensor network can be used to alleviate the problems inherent to the centralized architecture. Further, the distributed architecture requires lighter computational power at each fusion node due to the distribution of processing over multiple nodes.

Distributed algorithms based on particle filters have gained much attention. In [5], methods based on likelihood factorization of particles and adaptive data-encoding scheme are proposed for nonlinear/non-Gaussian systems with distributed architecture. An improvement to the approach proposed in [5] has been presented in [13] using a better encoding scheme and measurement vectorization. More particle-based implementations are given in [18], [22]. The adaptive data-encoding scheme uses the histogram of expected measurements to encode the target-generated measurements effectively. However, the false measurements might end up transmitting a larger number of bits than transmitting measurements without encoding. Hence, the effectiveness of the encoding scheme might degrade dramatically if no method is in place to identify and remove false measurements before transmitting over the network. Also, target birth must be taken care of while removing the false alarms in order to handle the time-varying number of targets.

The primary focus of this paper is on creating distributed algorithms that minimize network communication relating to sensor data fusion when multiple time-varying number of targets are present in the monitored

area. In this paper, a decentralized version of the Probability Hypothesis Density (PHD) filter is used to track multiple targets. The PHD filter eliminates the hard measurement-to-track association problem, unlike the Multiple Hypothesis Tracker (MHT) [4]. Furthermore, the PHD filter has been shown an effective way of tracking time-varying multiple number of targets that avoids model-data association problems [15]. Gaussian mixture implementation of PHD filter (GM-PHD) is presented in [25]. Sequential Monte Carlo (SMC) implementation of the PHD filter is used to handle the nonlinear measurements [24]. There are two options available to perform distributed tracking with a SMC-PHD filter in a sensor network. The first option is to send all the particles that represent the posterior density of targets. However, this option requires high bandwidth communications, which can not be handled by practical wireless sensor networks. The second option is to send the most relevant measurements after eliminating the false alarms to update the global estimates of the targets.

In this paper, measurements are communicated among nodes to update the filters. In this case, data transmission requires higher bandwidth channels unless the quantization of those data are done intelligently [16, 19]. To be effective, non-uniform quantization schemes can be made to match the distribution of the quantity to be discretized. Companding is a widely used method for implementing non-uniform quantizers [17]. It has been observed in non-uniform quantization that the communication can be considerably reduced with the right selection of the compander [16]. Quantized measurements need to be encoded before transmission. It is assumed that an optimal noiseless source code will be employed to minimize transmission needs between nodes. In this paper, Huffman coding is used to encode the quantized measurements. Handling multiple target-originated measurements at the quantization stage and producing identical symbols for encoding and decoding at each node are challenging. This paper proposes "cascaded companders" to nonlinearly quantize multiple target measurements. Predicted probability density is used in generating identical set of symbols and to place the companders at right positions. The measurement quantization and encoding techniques proposed this paper can be applied to distributed tracking with GM-PHD and other PHD filter realization algorithms as well.

Among the various methods to quantify the performance, verifying the closeness of the estimates mean square error matrix to the lower bound is a commonly known method in target tracking applications. The Posterior Cramer-Rao Lower Bound (PCRLB) is defined to be inverse of the Fisher Information Matrix (FIM) for random vector and provides lower bound on the performance of unbiased estimators of the unknown target state [23]. The PCRLB for state estimation with quantized measurements is complicated due to nonlinearity of the quantizer. Previously, in [28] the PCRLB for dynamic target tracking with measurement origin uncertainty and in [8] the PCRLB for state estimation with quantized measurement were developed. In this paper, the PCRLB calculation with quantized measurement is extended to incorporate measurement origin uncertainty for bearing only tracking.

This paper is structured as follows. Section 2 explains the proposed distributed implementation of SMC-PHD filter. Quantization and encoding methods are explaiined in Section 3. Section 4 provides the derivation of the PCRLB with quantized measurements and measurement origin uncertainty. Simulations that demonstrate the effectiveness of the proposed quantization strategy are presented in Section 5. Conclusions are given in Section 6.

2. DISTRIBUTED TRACKING USING SMC-PHD FILTER

2.1. State and Measurement Models

In this paper, the problem of tracking a time-varying number of multiple targets is considered. The general parameterized target dynamics is given by

$$x_{k+1} = F_k x_k + \nu_k \tag{1}$$

where x_k denotes the target state, F_k is a known matrix and ν_k is the process noise at time k.

The measurements originate from either targets or clutter. The target-originated measurement is given by

$$z_k = h_k(x_k) + \omega_k \tag{2}$$

where h_k is a nonlinear function and ω_k is the measurement noise at time k. For simplicity it is assumed that ν_k and ω_k are Gaussian with zero means and covariances Γ_k and Σ_k, respectively.

It is assumed that the number of false alarms is Poisson-distributed with the average rate of λ_k and that the probability density of the spatial distribution of false alarms is $c_k(\mathbf{z}_k)$.

2.2. PHD Filter

In tracking multiple targets, if the number of targets is unknown and varying with time, it is not possible to compare states with different dimensions using ordinary Bayesian statistics of fixed dimensional spaces. However, the problem can be addressed by using Finite Set Statistics (FISST) [15] to incorporate comparisons of state spaces of different dimensions. FISST facilitates the construction of multitarget densities from multiple-target transition functions by computing set derivatives of belief-mass functions [15], which makes it possible to combine states of different dimensions. The main practical difficulty with this approach is that the dimension of the full state space becomes large when many targets are present, which increases the computational load exponentially in the number of targets. Since the PHD is defined over the state space of one target in contrast

to the full posterior distribution, which is defined over the state space of all the targets, the computational cost of propagating the PHD over time is much lower than propagating the full posterior density.

In general, a PHD-based multitarget tracker will experience more difficulty in resolving closely-spaced targets than a tracker based on the full target posterior. However, if the probability density functions of individual targets is highly concentrated around their means compared to the target separation, such that the individual target pdfs do not overlap significantly, it will become possible to resolve the targets using the PHD filter as well. A theoretical explanation on the capability of the PHD filter to resolve closely-spaced targets in Gaussian context is given in [15]. By definition, the PHD $D_{k|k}(x_k \mid Z_{1:k})$, with single target state vector x_k, and given all the measurements up to and time step k, is the density whose integral on any region S of the state space is the expected number of targets $N_{k|k}$ contains in S. That is,

$$N_{k|k} = \int_X D_{k|}(x_k \mid Z_{1:k}) dx_k. \tag{3}$$

This property uniquely characterizes the PHD and the first-order statistical moment of the full target posterior distribution possesses this property. Hence, the first-order statistical moment of the full target posterior, or the PHD, given all the measurement $Z_{1:k}$ up to time step k, is given by the set integral [14]

$$D_{k|k}(x_k \mid Z_{1:k}) = \int f_{k|k}(\{x_k\} \cup Y \mid Z_{1:k}) \delta(Y). \tag{4}$$

More detailed mathematical explanations and derivation of the PHD filter can be found in [14]. The approximate expected target states are given by the local maxima of the PHD. The prediction and update steps of one cycle of PHD filter are given in the following section.

2.2.1. Prediction

In a general scenario of interest, there are target disappearances, target spawning and entry of new targets. The probability that a target with state x_{k-1} at time step $(k-1)$ will survive at time step k is denoted by $e_{k|k-1}(x_{k-1})$, the PHD of spawned targets at time step k from a target with state x_{k-1} by $b_{k|k-1}(x_k \mid x_{k-1})$, and the PHD of newborn spontaneous targets at time step k by $\gamma_k(x_k)$. Then, the predicted PHD, $D_{k|k-1}(x_k \mid Z_{1:k-1})$, at time k given all measurements up to time $k-1$ is given by

$$\begin{aligned} &D_{k|k-1}(x_k \mid Z_{1:k-1}) \\ &\quad = \gamma_k(x_k) + \int [e_{k|k-1}(x_{k-1}) f_{k|k-1}(x_k \mid x_{k-1}) + b_{k|k-1}(x_k \mid x_{k-1})] \\ &\qquad \times D_{k-1|k-1}(x_{k-1} \mid Z_{1:k-1}) dx_{k-1} \end{aligned} \tag{5}$$

where $f_{k|k-1}(x_k \mid x_{k-1})$ denotes the single-target Markov transition density. The prediction equation (5) is lossless since there are no approximations.

2.2.2. Update

The predicted PHD can be corrected with the availability of measurements Z_k at time step k to get the updated PHD. It is assumed that the number of false alarms is Poisson-distributed with the average rate of λ_k and that the probability density of the spatial distribution of false alarms is $c_k(z_k)$. Let the detection probability of a target with state x_k at time step k be $p_D(x_k)$. Then, the updated PHD at time step k is given by

$$\begin{aligned} &D_{k|k}(x_k \mid Z_{1:k}) \\ &\quad \cong \left[\sum_{z_k \in Z_k} \frac{p_D(x_k) f_{k|k}(z_k \mid x_k)}{\lambda_k c_k(z_k) + \psi_k(z_k \mid Z_{1:k-1})} + (1 - p_D(x_k)) \right] \\ &\qquad \times D_{k|k-1}(x_k \mid Z_{1:k-1}) \end{aligned} \tag{6}$$

where the likelihood function $\psi(\cdot)$ is given by

$$\begin{aligned} &\psi_k(z_k \mid Z_{1:k-1}) \\ &\quad = \int p_D(x_k) f_{k|k}(z_k \mid x_k) D_{k|k-1}(x_k \mid Z_{1:k-1}) dx_k \end{aligned} \tag{7}$$

and $f_{k|k}(z_k \mid x_k)$ denotes the single-sensor/single-target likelihood. The update equation (6) is not lossless since approximations are made on predicted multitarget posterior to obtain a closed-form solution. The reader is referred to [14] for further explanations.

2.3. Sequential Monte Carlo PHD Filter

This section describes the SMC approach to the PHD filter [24]. This approach provides a mechanism to represent the posterior probability hypothesis density by a set of random samples or particles, which consist of state information with associated weights, to approximate the PHD. The advantage of this method is that the number of particles can be adaptively allocated such that a constant ratio between the number of particles and the expected number of targets is maintained. This has a significant effect on the computational complexity of the algorithm. The complexity does not increase exponentially, but only linearly with the increasing number of targets. The SMC implementation considered here is structurally similar to the Sampling Importance Resampling (SIR) type of particle filter [2]. Let the posterior PHD $D_{k-1|k-1}(x_{k-1} \mid Z_{1:k-1})$ be represented by a set of particles $\{w_{k-1}^{(p)}, x_{k-1}^{(p)}\}_{p=1}^{L_{k-1}}$. That is,

$$D_{k-1|k-1}(\mathbf{x}_{k-1} \mid Z_{1:k-1}) = \sum_{p=1}^{L_{k-1}} w_{k-1}^{(p)} \delta(\mathbf{x}_{k-1} - \mathbf{x}_{k-1}^{(p)}) \tag{8}$$

where $\delta(\cdot)$ is the Dirac Delta function. In contrast to particle filters, the total weight of the particles $\sum_{p=1}^{L_{k-1}} w_{k-1}^{(p)}$ is not equal to one; instead, total weight gives the expected

number of targets n_{k-1}^X at time step $(k-1)$, which follows from the property that the integral of the PHD over the state space gives the expected number of targets.

2.3.1. Prediction

Importance sampling is applied to generate state samples that approximate the predicted PHD $D_{k|k-1}(x_k \mid Z_{1:k-1})$. State samples $\{x_{k|k-1}^{(s)}\}_{p=1}^{L_{k-1}}$ are generated from the proposal density $q_k(\cdot \mid x_{k-1}, Z_k)$ and i.i.d. state samples $\{x_{k|k-1}^{(p)}\}_{p=L_{k-1}+1}^{L_{k-1}+J_k}$ corresponding to new spontaneously born targets from another proposal density $p_k(\cdot \mid Z_k)$. That is,

$$x_{k|k-1}^{(s)} \sim \begin{cases} q_k(\cdot \mid x_{k-1}, Z_k) & p = 1,\ldots,L_{k-1} \\ p_k(\cdot \mid Z_k) & p = L_{k-1}+1,\ldots,L_{k-1}+J_k \end{cases}. \tag{9}$$

Then, the weighted approximation of the predicted PHD is given by

$$D_{k|k-1}(x_k \mid Z_{1:k-1}) = \sum_{p=1}^{L_{k-1}+J_k} w_{k|k-1}^{(p)} \delta(x_k - x_{k|k-1}^{(p)}) \tag{10}$$

where

$$w_{k|k-1}^{(p)} = \begin{cases} \dfrac{e_{k|k-1}(\mathbf{x}_{k|k-1}^{(p)}) f_{k|k-1}(\mathbf{x}_{k|k-1}^{(p)} \mid \mathbf{x}_{k-1}^{(p)}) + b_{k|k-1}(\mathbf{x}_{k|k-1}^{(p)} \mid \mathbf{x}_{k-1}^{(p)})}{q_k(\mathbf{x}_{k|k-1}^{(p)} \mid \mathbf{x}_{k-1}^{(p)}, Z_k)} w_{k-1}^{(s)} & p = 1,\ldots,L_{k-1} \\ \dfrac{\gamma_k(\mathbf{x}_{k|k-1}^{(p)})}{p_k(\mathbf{x}_{k|k-1}^{(p)} \mid Z_k)} \dfrac{1}{J_k} & p = L_{k-1}+1,\ldots,L_{k-1}+J_k \end{cases}. \tag{11}$$

The functions that characterize the Markov target transition density $f_{k|k-1}(.)$, target spawning $b_{k|k-1}$ and entry of new targets $\gamma_k(\cdot)$ in (11) are conditioned on the target motion model.

2.3.2. Update

With the available set of measurements Z_k at time step k, the updated particle weights can be calculated by

$$w_k^{*(p)} = \left[(1 - p_D(\mathbf{x}_{k|k-1}^{(p)})) + \sum_{i=1}^{N_k^Z} \frac{p_D(\mathbf{x}_{k|k-1}^{(p)}) f_{k|k}(\mathbf{z}_k^i \mid \mathbf{x}_{k|k-1}^{(p)})}{\lambda_k c_k(\mathbf{z}_k^i) + \Psi_k(\mathbf{z}_k^i)} \right] w_{k|k-1}^{(p)} \tag{12}$$

where

$$\Psi_k(z_k^i) = \sum_{p=1}^{L_{k-1}+J_k} p_D(\mathbf{x}_{k|k-1}^{(p)}) f_{k|k}(\mathbf{z}_k^i \mid \mathbf{x}_{k|k-1}^{(s)}), w_{k|k-1}^{(p)} \tag{13}$$

and $f_{k|k}(\cdot)$ is the single-target/single-sensor measurement likelihood function.

2.3.3. Resample

To perform resampling, since the weights are not normalized to unity in PHD filters, the expected number of targets is calculated by summing up the total weights, i.e.,

$$\hat{n}_k^X = \sum_{p=1}^{L_{k-1}+J_k} w_k^{*(p)}. \tag{14}$$

Then the updated particle set $\{w_k^{*(p)}/n_k^X, \mathbf{x}_{k|k-1}^{(p)}\}_{p=1}^{L_{k-1}+J_k}$ is resampled to get $\{w_k^{(p)}/n_k^X, \mathbf{x}_k^{(p)}\}_{p=1}^{L_k}$ such that the total weight after resampling remains n_k^X. Now, the discrete approximation of the updated posterior PHD at time step k is given by

$$D_{k|k}(\mathbf{x}_k \mid Z_{1:k}) = \sum_{p=1}^{L_k} w_k^{(p)} \delta(\mathbf{x}_k - \mathbf{x}_k^{(p)}). \tag{15}$$

2.4. Distributed Architecture

Distributed processing over the sensor network can be used to alleviate the problem inherent to centralized architectures. A sample distributed architecture is shown in Fig. 1, where S indicates the sensor. The underlying sensor network architecture consists of two different types of devices: sensors and nodes. Sensors collect measurements from the targets and report them to computational nodes. Nodes are responsible for running filters to track targets. Information gathered at one node are shared among various nodes. The efficient utilization of communication resources without compromising accuracy is essential.

2.5. Distributed Tracking Algorithm

The objective in this paper is to develop a distributed algorithm based on the SMC-PHD filter while minimizing the communication requirements of the distributed network in the presence of multiple time-varying number of targets and false alarms. It is assumed that the optimization of sensor resources to collect data and communication issues such as network protocols are already efficient enough. The proposed algorithm maintains SMC-PHD filters at all the computational nodes.

There are a number of different options to perform distributed tracking with an SMC-PHD filter in a sensor network. One option is to send all the particles that represent the posterior density of target states. Another is to send Gaussian mixture representation of the posterior density. These two options require high bandwidth communications, which cannot be handled by practical wireless sensor networks. The third option is to send only most relevant measurements after eliminating the

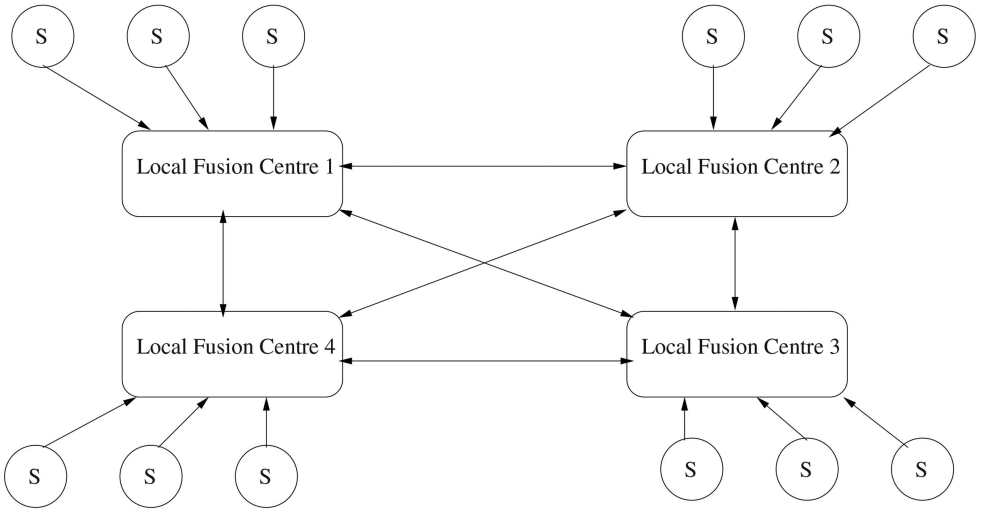

Fig. 1. A sample distributed architecture.

false alarms to update the global estimates of the targets. In this paper, the option of communicating the relevant measurements among nodes to update the filters is used. In a sensor network, it is possible that each node has enough active sensors to track an object by itself with reasonable tracking accuracy. Therefore, a PHD filter can be used to obtain the estimates based on the measurements collected from sensors local to that node. Since these nodes maintain PHD filters based on local measurements, they can also be used in the encoding strategy. The proposed framework will be performed in two layers. The first layer collects measurement data that are local to each node and maintains a local PHD filter using its associated sensors. In the second layer, all measurements are exchanged to all other nodes in the network and the global PHD filters are maintained.

In the proposed algorithm, identical copies of the SMC-PHD filter are maintained at each node. Initially, this is achieved by initializing filters using the same random seed. In order to encode the measurement data, an intelligent quantization and encoding strategy is used. From time step $k-1$ to k, particles are propagated while taking into account the measurement prediction covariance. The range of expected measurements is divided into bins depending on the required accuracy level. The contribution of each propagated particle's distribution is integrated over the bins to form the probability density. The measurements are quantized with a non-uniform quantizer where companders are used to perform non-uniform quantization. The probability density in the measurement space is then transformed to the companded measurement space. Then, the quantized measurements are encoded using Huffman encoding algorithm with the transformed bin probabilities. The encoded measurements are transmitted to all other nodes where each node decodes and decompands the data to obtain the quantized measurements. The details of quantization and encoding strategy used in this algorithm is presented in Section 3.

Each node performs filtering using quantized measurements to obtain the target state estimates. All nodes use the same set of measurement data to update the filter, thereby maintaining the identical copy of filter.

The steps of the distributed SMC-PHD filter are given below.

1) Initialization at $k = 0$:
   - Initialize SMC PHD filter on each node $n = 1,\ldots,N$ using the same random seed to generate identical particle distribution on all the nodes.
   - For each node $n = 1,\ldots,N$
     — Generate samples $\{\mathbf{x}_0^{(p)}\}_{p=0}^{L_0}$
2) Quantization and encoding (For implementation details of this step the reader is referred to Section 3):
   - Local Estimation
     — Perform filtering using the SMC PHD filter acting only on the measurements local to the node.
   - Quantization
     — For each node $n = 1,\ldots,N$
       * For $s = 1,\ldots,L_{k-1}$, predict $\mathbf{x}_{k|k-1}^{(p)}$
       * Calculate the bin probabilities, $p(z_k \mid b_j, \mathbf{z}_{1:k-1}^{(p)})$, in the measurement space using predicted measurements and construct the probability density where b_j is the jth measurement bin.
       * Identify the regions where the companders need to be placed and the number of companders needed. One compander per target is used and the width of the companding region is limited to $3\sigma_p^c$, where the σ_p^c is the standard deviation of the cth cluster. The compander is placed on the mean value, μ_p^c, of the cluster. In other regions linear quantizer is used.
       * Quantize the measurements, $\tilde{\mathbf{z}}_k$

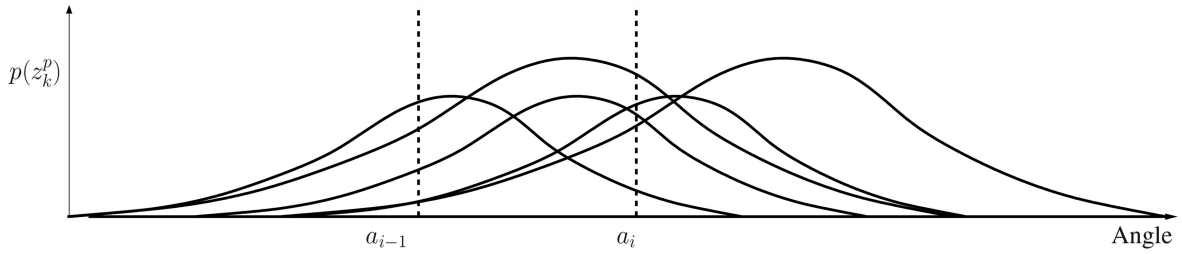

Fig. 2. Calculation of bin probabilities.

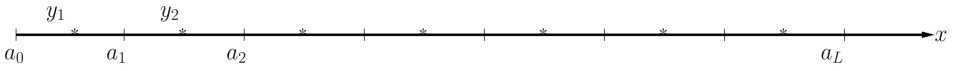

Fig. 3. Quantization.

- Encoding
  - — For each node $n = 1,\ldots,N$
    - * Calculate the bin probabilities, $\tilde{p}(z_k \mid b_j, \mathbf{z}_{1:k-1}^{(p)})$, in the transformed measurement space.
    - * Use the bin probabilities to form Huffman tree H_f^{k-1} and encode quantized measurements.

3) Reducing the false measurements transmitted over the network:
   - Remove the measurements from the queue if the number of bits in each encoded measurement exceeds a predefined threshold, l. This process is done using the local estimates of the target.
4) Global estimate:
   - For each node $n = 1,\ldots,N$, create the Huffman tree H_f^{k-1} and the quantizer to reconstruct the quantized data, $\tilde{\mathbf{z}}_k'$.
   - Using the obtained set of measurements, perform filtering to obtain the global state estimates.

3. QUANTIZATION AND ENCODING

Measurements reported by sensors in a sensor network need to be transmitted in order to perform tracking at high computational nodes called fusion centers. Quantization and encoding play a crucial role whereby measurements are quantized and encoded before being transmitted. Intelligent quantization and encoding schemes are necessary to effectively use the communication resources. This section explains how quantization and encoding can be effectively implemented to perform distributed target tracking with SMC-PHD filters.

The proposed algorithm needs an efficient nonlinear non-uniform quantization for measurements. Therefore, the concept of "cascaded companders," which can quantize measurements from multiple targets, is proposed. This section briefly explains the process of developing the compander. The first step is to construct a probability density of expected measurements to identify the regions where the target originated measurements would lie. The details of this process are given in Section 3.01. Measurements that fall in this region are quantized with minimum quantization error via Gaussian companders. Section 3.13 explains the cascaded companders. Details of encoding and decoding process using Huffman coding are given in Section 3.23. Sections 3.3 and 3.14 provide details on the false alarm elimination process and the incorporation of quantization errors into tracking, respectively.

3.0.1. Construction of a Probability Density

The necessity to have identical and accurate probability densities of targets at each node, where global SMC-PHD filter is running, is clear from the fact that the measurements are quantized, encoded and communicated across these nodes based on the probability density. The construction of probability density begins with propagating the densities of particles from time step $k-1$ to k, taking into account the measurement prediction covariance. The range of expected measurements is divided into bins depending on the required accuracy level. The contribution of each propagated particle's distribution is integrated over the bins to form the probability density. Figure 2 shows the distribution of three sample particles and the quantizer decision boundaries a_{i-1} and a_i. The probability density of predicted particles $p(z_k^s)$ in the measurement space is given by

$$p(z_k^p) = \mathcal{N}(z_k^s; h_k(x_{k|k-1}^p), S_k) \tag{16}$$

where $h_k(.)$ is a nonlinear function and S_k is the measurement prediction covariance. Then the bin probability is given by

$$p(z_k \mid b_j, \mathbf{z}_{1:k-1}^{(p)}) = \sum_{s=1}^{L_{k-1}} \int_{a_{i-1}}^{a_i} p(z_k^s) dz. \tag{17}$$

3.1. Quantization

One dimensional quantizer Q with L levels may be defined by a set of $L+1$ decision levels $a_0, a_1, \ldots, a_L$ and a set of L output levels $y_1, y_2, \ldots, y_L$, as shown in Fig. 3. When a sample x, the quantity to be quantized, lies in the ith quantizer interval $s_i = a_{i-1} < x \leq a_i$ the quantizer produces the output value $Q(x) = y_i$ [9]. The value of y_i is usually chosen to lie within the interval

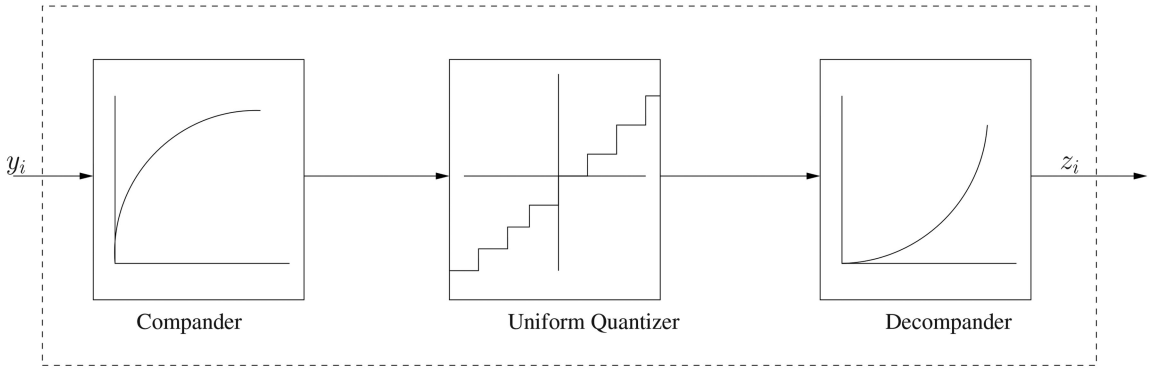

Fig. 4. Nonuniform quantization.

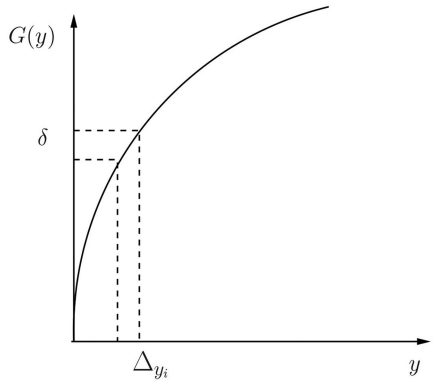

Fig. 5. A typical compander.

s_i. The end levels a_0 and a_L are generally chosen to be the smallest and largest values the input samples may obtain. The L output levels generally have a finite value and if $L = 2^n$, a unique n-bit binary word can identify a particular output level. The input-output characteristics of a one-dimensional quantizer resemble a staircase. The quantizer intervals, or steps, may vary in size.

Uniform and non-uniform quantizer strategies are investigated in this paper.

3.1.1. Uniform Quantization

Uniform quantizer is where the measurement space is divided into equal bins based on the number of bits used to encode. The output points are located at the midpoint of these intervals. If the step size is denoted by $\triangle$, then the maximum absolute error is given by $\triangle/2$. In general, uniform quantization is not the most effective way to obtain good quantizer performance [9].

3.1.2. Non-uniform Quantization

The non-uniform quantization essentially has a non-uniform spacing of decision levels based upon the input probability density [16]. The general model used to represent the non-uniform quantizer is shown in Fig. 4. The combined function of compression, quantization and expansion is termed companding [17]. The quantized samples are transmitted over the network while at the receiver end of the network the quantized samples are decompanded to its original values plus the quantization noise. The variance of the quantization noise associated with the received samples is related to the shape of the companding function $G(.)$ and the number of the bits, n, used for quantization. A typical companding function is shown in Fig. 5. With reference to the figure,

$$G(y + \Delta_y) - G(y) = \delta \tag{18}$$

in which the right hand side is the resolution of the uniform quantizer. Using standard companding techniques, Δ_y can be given as

$$\Delta_y \approx \frac{\delta}{\dot{G}(y)} \tag{19}$$

where $\dot{G}$ denotes differentiation of G.

3.1.3. Measurement Quantization with Cascaded Companders

The non-uniform quantization is performed based on probability density of the targets. Figures 6 and 7 show quantizers at two different time steps, when one and two targets are present in the environment, respectively. The companders are placed in the measurement space such that the target-originated measurements have less quantization errors than other measurements. In this paper, a Gaussian compander law, which is centered on the expected target position and whose curvature is dictated by the standard deviation of the expected position [16], is used. The compander and expander functions are as follows:

- Compander: $\text{erf}(\xi/\sigma\sqrt{6})$
- Expander: $\sigma\sqrt{6}\text{erf}(\xi)$

where $\text{erf}(\xi) = 2/\sqrt{pi}\int_0^{\xi}\exp(-t^2)dt$. One compander per target is used and the width of the companding region is limited to $3\sigma_p^c$, where σ_p^c is the standard deviation of the cth cluster. The compander is placed on the mean value, μ_p^c, of the cluster. A maximum quantization error is set in other regions of the measurement space, where the compander is not placed, by a liner quantizer. The companders are cascaded when multiple targets measurements are to be quantized.

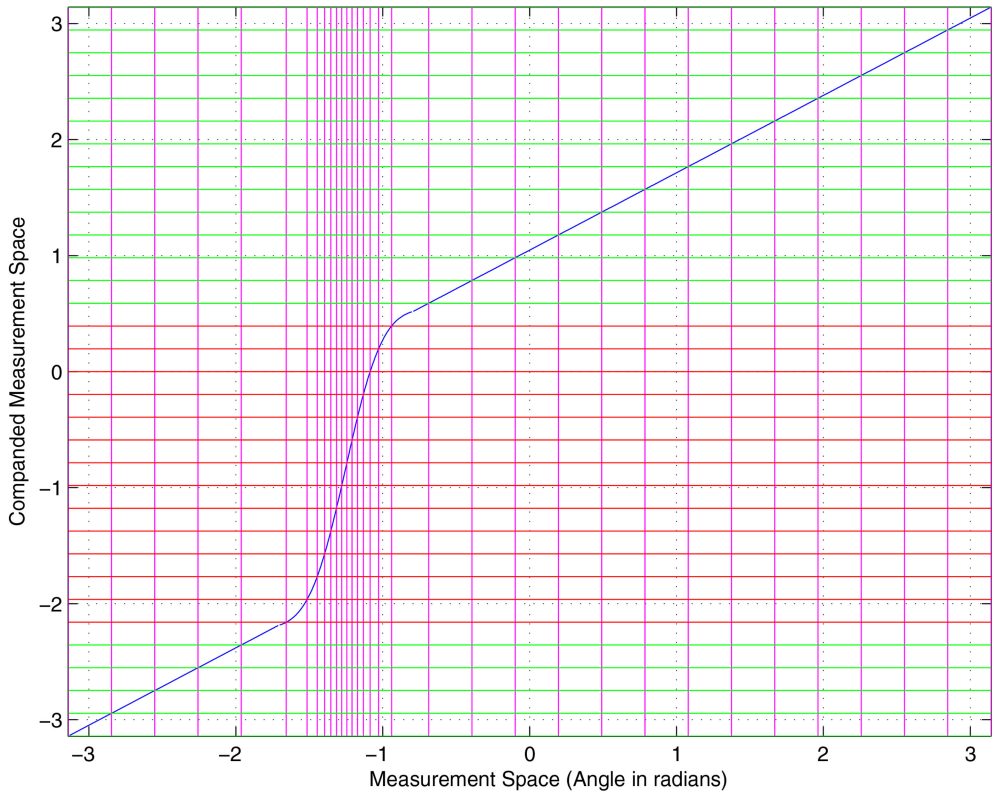

Fig. 6. A 32-bin compander with one target.

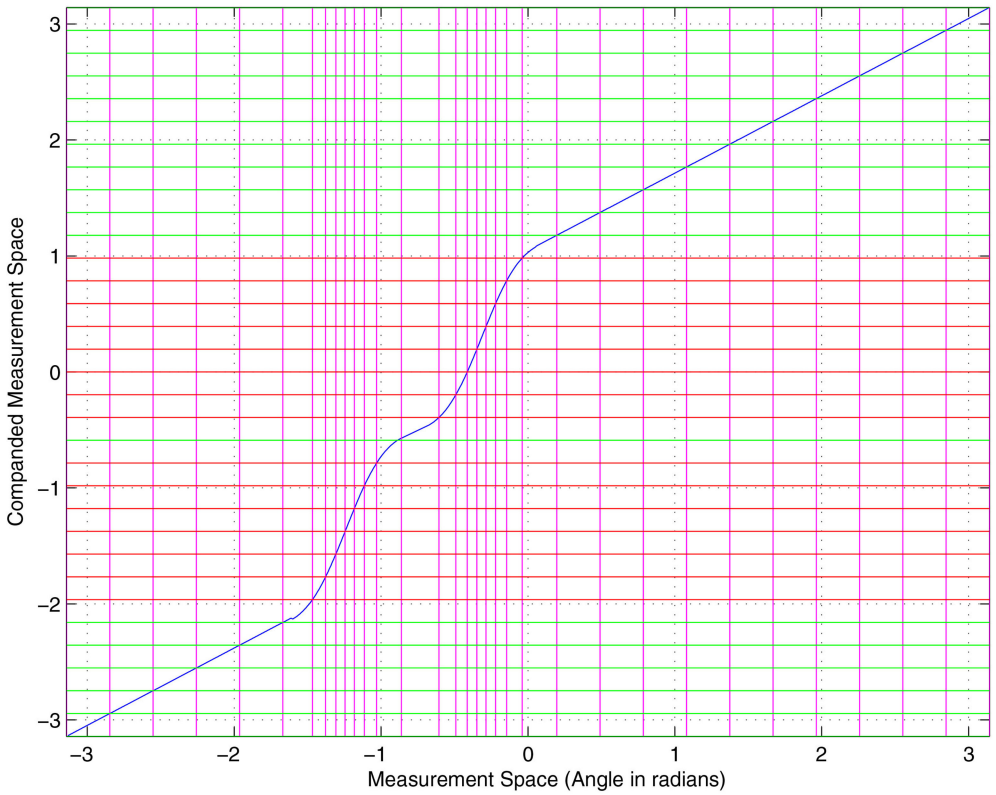

Fig. 7. A 32-bin compander with two targets.

3.1.4. Incorporating Quantization Errors

The insertion of quantized measurement to the SMC-PHD filter is done by updating the current particles by the quantized measurements while taking into account the extra error introduced by the quantization. The error arising from quantization has a uniform distribution.

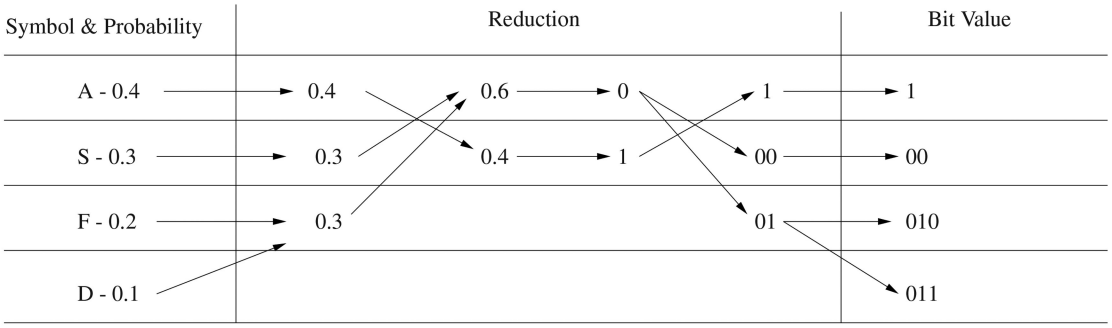

Fig. 8. Construction of Huffman Encoding Table.

The variances of errors introduced due to quantization is given by

- Uniform quantization

$$\mathrm{Var}(z_k^i \mid x_k) = \sigma_w^2 + \frac{\delta^2}{12}. \tag{20}$$

- Non-uniform quantization

$$\mathrm{Var}(z_k^i \mid x_k) = \sigma_w^2 + \frac{\delta^2}{12\dot{G}(y_k^i)^2}. \tag{21}$$

3.2. Encoding

In information theory, an entropy coding is a lossless data compression scheme that is independent of the specific characteristics of the medium. A common method of entropy coding defines a codebook by assigning a code to each symbol. By assigning smaller codes to the more frequent symbols, the average size of each coded symbol can be minimized. This leads to compression over sufficiently large number of encoded symbols. This technique is known as variable length coding. Generally, variable length coding shows a better performance than fixed-length codes where same size is assigned to all symbols [20].

Two widely used entropy coding techniques are Huffman coding [12, 6] and arithmetic coding [27]. Huffman coding is simple to implement and is efficient when the probabilities of symbols to be sent can be calculated in advance. Hence it is best suited for application in this paper.

Encoding will help reduce the communication load only for uniform quantization. In non-uniform quantization, the probability of getting measurement at each measurement bin is almost equal. As a result, it is not possible to achieve communication reduction by encoding for non-uniform quantization.

3.2.1. Huffman Coding

Huffman coding assigns a variable length code to each input symbol where the code and its size are based on the probability of occurrence of the associated symbol. It is necessary to calculate probability of symbols before the assignment and construction of a dictionary. By sorting and analyzing the probability of symbols, a conversion table is constructed so that the symbols with higher probability have the fewer number of bits and no symbol is a prefix to another symbol [20]. Greater compression can be achieved with the accurate estimation of probability distribution.

3.2.2. Building Huffman Codes

The construction of Huffman encoding table is a lengthy process. The probabilities must be sorted so that the two lowest probabilities can be found. These probabilities are added together to create a new probability table. This table is sorted, and the process is repeated until only two probabilities are left. These probabilities are assigned a value of zero and one. The process is now reversed. At each stage the two expanded probabilities are given a one or zero as they are expanded. The process continues until the table is expanded to its original state. For example, assume that the message "ASAFAFDAS" is being encoded. The first step is to find the probability for each symbol. "A" has a probability of 0.4, while S has 0.3, D has 0.1 and F has 0.2. These probabilities are sorted and added to create the table as in Fig. 8. Once the table is constructed, the data can be compressed. The compression process is accomplished by a direct conversion of symbols. The entire message is encoded as "10010101010011100," which requires 17 bits. The unencoded message would normally require 18 bits.

3.2.3. Measurement Encoding and Decoding

The original probability density constructed based on expected measurements is transformed to companded measurement space in order to create a global Huffman dictionary for encoding. The term global refers to the process or information that is related to global SMC-PHD filter running on every node. Companded measurements are encoded and transmitted over the network. In the receiver, measurements are decoded before expanding. The same steps are followed to construct a decoding dictionary.

3.3. False Alarm Elimination

Reducing the number of false measurements communicated over the network is important as they consume most of the communication resources. The number of bits in each encoded measurement, based on the

local Huffman dictionary, can effectively be used to reduce the number of false measurements transmitted over the sensor network. In this approach, it is assumed that since the local PHD filters have the most up-to-date information including the birth of a new target and the target generated measurements are most likely to be in a region in which the value of the probability is high. Thus the target-generated measurements are most likely to have a lesser number of bits in their encoded form compared to false measurements when encoded with local Huffman dictionary. It is reasonable to assume that the measurements that have a higher number of bits are not target generated and, by having a threshold value on the number of bits, they can be removed from the set of measurements that are transmitted over the network. Once the measurements are selected to be transmitted, those measurements are encoded with the global Huffman dictionary in order to transmit over sensor network. However, when measurements corresponding to new targets are encoded with global Huffman dictionary may produce higher number of bits. It could be noted that the new targets can be identified by the global PHD filter quickly. An indicator function, $\mathcal{I}_h^{(k,i)}$ is used to identify whether the measurement has been communicated or not.

$$\mathcal{I}_h^{(k,i)} = \begin{cases} 1 & H_f^{k-1}(\tilde{\mathbf{z}}_k^i) \le l \\ 0 & H_f^{k-1}(\tilde{\mathbf{z}}_k^i) > l \end{cases} \tag{22}$$

$H_f^{k-1}(\tilde{\mathbf{z}}_k^i)$ is a function that generates Huffman codes for each measurement. l is the cutoff number of bits per measurement. If a measurement in its encoded form is less than the cutoff number of bits, then the measurement is communicated and not otherwise.

4. POSTERIOR CRAMER-RAO LOWER BOUND

In this section, the recursive Riccati-like formula for the PCRLB is derived for state estimation using measurements with quantization and origin uncertainty. The Section 4.1 provides a brief review on PCRLB. Incorporating the measurement origin uncertainty in PCRLB is discussed in Section 4.2. In Section 4.3 the PCRLB with quantized measurements is derived.

4.1. Background

Consider the estimation of the state of a dynamical system given by (1) and (2). The quantized measurements at time k are denoted by $\tilde{\mathbf{z}}_k$. Let $\hat{x}_{k|k}$ denote the updated state estimate at time instant k, using measurement $\tilde{\mathbf{z}}_{1:k}$. The estimation error covariance matrix, $P_{k|k}$, for unbiased estimator is bounded as follows:

$$P_{k|k} = E[(x_k - \hat{x}_{k|k})(x_k - \hat{x}_{k|k})^T] \ge J_k^{-1} \tag{23}$$

where J_k is the Fisher information matrix, which is the inverse of PCRLB.

For linear Gaussian systems, Riccati-like recursion is given by [11]

$$J_{k+1} = (Q_k + F_k J_k^{-1} F_k^T)^{-1} + \underbrace{E[-\Delta_{x_{k+1}}^{x_{k+1}} \log p(\tilde{\mathbf{z}}_{k+1} \mid x_{k+1})]}_{J_{zk+1}} \tag{24}$$

with $J_0^{-1} = P_0$.

4.2. Effect of Measurement Origin Uncertainty

Consider n_s (≥ 1) sensors, and let $\tilde{\mathbf{z}}_k^s$ be the quantized measurement vector from sensor s. It is assumed that the measurement noises of sensors are independent. Also, due to false alarms, the total number of measurements can vary among sensors at each time step. Let m_k^s be the total number of measurements from sensor s at time k. Let the observation set at time k from sensor s be

$$\tilde{\mathbf{z}}_k^s = \{\tilde{\mathbf{z}}_k^s(i)\}_{i=1}^{m_k^s} \tag{25}$$

where m_k^s in general is random quantity.

Under the assumption that false alarms are uniformly distributed in the measurement space, and the number of false alarms is Poisson distributed, probability of getting m_k^s is given by [11]

$$p(m_k^s) = (1 - P_D^s)\frac{(\lambda V)^{m_k} \exp(-\lambda V)}{m_k!} + P_D^s \frac{(\lambda V)^{m_k - 1} \exp(-\lambda V)}{(m_k - 1)!} \tag{26}$$

where P_D^s is the probability of detecting the target by sensor s, V is the gated volume of the measurement space.

If false alarms are removed by setting a cut-off length for the number of bits to be sent after encoding, then $\bar{P}_D^s$ must be calculated by considering the possibility of removing a target originated measurement. In the PCRLB calculation, P_D^s must be replaced by $\bar{P}_D^s$. V is must also be calculated using the predicted target distribution and the false alarm removal cut-off limit. Even though the cut-off is set on the number of bits, it can be converted to the probability and can be used to decide the gate size.

Using measurement independent assumption, the measurement information, $J_k(\tilde{\mathbf{z}})$, is given by [11]

$$J_{zk}(\tilde{\mathbf{z}}) = \sum_{s=1}^{n_s} \sum_{m_k^s=0}^{\infty} p(m_k^s) J_{zk}^s(m_k^s) \tag{27}$$

where

$$J_{zk}^s(m_k^s) = E[-\Delta_{x_k}^{x_k} \log p(\tilde{\mathbf{z}}_k^s \mid x_k, m_k^s))] \tag{28}$$

$p(\tilde{\mathbf{z}}_k^s \mid x_k, m_k)$ is given by

$$p(\tilde{\mathbf{z}}_k^s(i)_{i=1}^{m_k^s} \mid x_k) = \left[\frac{(1 - \epsilon(m_k^s))}{V^{m_k^s}} + \frac{\epsilon(m_k^s)}{m_k^s V^{m_k^s - 1}} \sum_{i=1}^{m_k^s} p_1(\tilde{\mathbf{z}}_k^s(i)) \right] \tag{29}$$

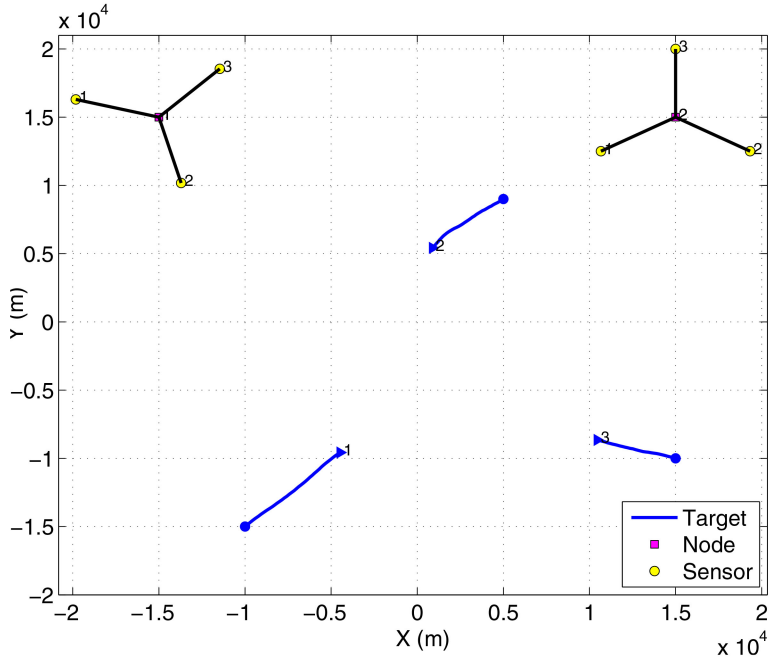

Fig. 9. The simulation environment.

where

$$\epsilon(m_k^s) = \frac{P_D^s}{p(m_k^s)} \frac{(\lambda V)^{m_k-1} \exp(-\lambda V)}{(m_k-1)!} \tag{30}$$

and $p_1(\tilde{\mathbf{z}}_k^s(i))$ is the pdf of the true observation, which is function of x_k. The details of obtaining $p_1(\tilde{\mathbf{z}}_k^s(i))$ with the quantized measurement are given in the following section.

4.3. Effect of Measurement Quantization

Due to the essence of quantization, it is known that $\tilde{\mathbf{z}}_k^s(i)$ has a discrete distribution and the only fact that can be inferred from $\tilde{\mathbf{z}}_k^s(i) = Q(z_k^s(i))$ is that $a_{(i,k)}^s \le \tilde{\mathbf{z}}_k^s(i) < a_{(i+1,k)}^s$ [8]. Under the assumption that the measurement error is a zero-mean Gaussian variable with standard deviation σ_ω^s, $p_1(\tilde{\mathbf{z}}_k^s(i))$ can be written as

$$\begin{aligned} p_1(\tilde{\mathbf{z}}_k^s(i)) &= P\{\tilde{\mathbf{z}}_k^s(i)) = Q(y_k^s(i)) \mid x_k\} \\ &= P\{a_{(i,k)}^s \le h_k(x_k) + v_k^s < a_{(i+1,k)}^s \mid x_k\} \\ &= \int_{a_{(i,k)}^s - h(x_k)}^{a_{(i+1,k)}^s - h(x_k)} \frac{1}{\sigma_w^s \sqrt{2\pi}} \exp\left\{-\frac{t^2}{2(\sigma_w^s)^2}\right\} dt. \end{aligned} \tag{31}$$

It can be shown that

$$\frac{\partial p_1(\tilde{\mathbf{z}}_k^s(i))}{\partial x_k^a} = -\frac{1}{\sigma_w^s \sqrt{2\pi}} \frac{\partial h(x_k)}{\partial x_k^a} \left(\exp\left[-\frac{(a_{(i+1,k)}^s - h(x_k))^2}{2(\sigma_w^s)^2}\right] - \exp\left[-\frac{(a_{(i,k)}^s - h(x_k))^2}{2(\sigma_w^s)^2}\right] \right). \tag{32}$$

From (28) and (32), it can be shown that

$$\frac{\partial \log p(\tilde{\mathbf{z}}_k^s \mid x_k, m_k^s)}{\partial x_k^a} = \frac{\epsilon(m_k^s)}{p(\tilde{\mathbf{z}}_k^s \mid x_k, m_k^s) m_k^s V^{m_k^s-1}} \sum_{i=1}^{m_k^s} \frac{\partial p_1(\tilde{\mathbf{z}}_k^s(i))}{\partial x_k^a}. \tag{33}$$

$J_{zk}^s(m_k^s)$ can be calculated using (33) and

$$\frac{-\partial^2 \log(p(\cdot))}{\partial x^a \partial x^b} = \frac{\partial \log(p(\cdot))}{\partial x^a} \frac{\partial \log(p(\cdot))}{\partial x^b}. \tag{34}$$

5. SIMULATION

In this section, results of the simulation studies for the proposed distributed algorithm with quantization and encoding strategies are presented.

5.1. Simulation Setup

In the simulations studies, a two dimensional tracking example is considered to show the effectiveness of the proposed algorithms. As shown in Fig. 9, it consists of two computational nodes placed at $(-15 \times 10^3, 15 \times 10^3)$ and $(15 \times 10^3, 15 \times 10^3)$. Each node has three sensors reporting bearing-only observations at a time interval of $T = 30$ s. The target motion model, which is nearly constant velocity, has the following linear-Gaussian target dynamics,

$$\mathbf{x}_{k+1} = \mathbf{F}\mathbf{x}_k + \mathbf{v}_k \tag{35}$$

where the target transition matrix F is given by

$$\mathbf{F} = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{36}$$

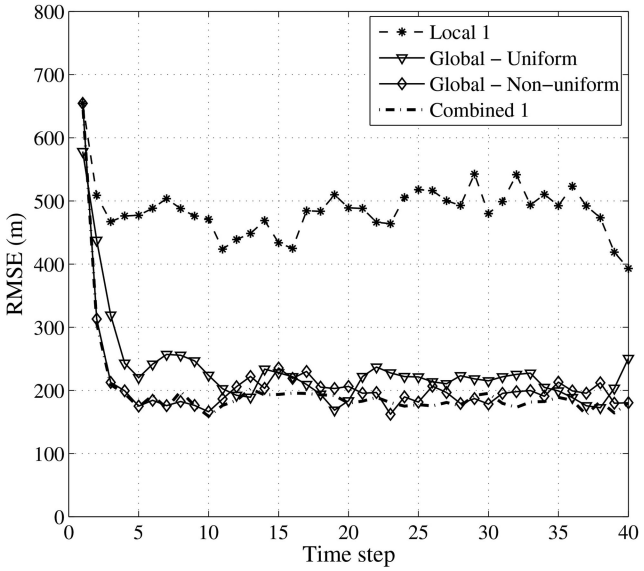

Fig. 10. Position RMSE comparison with 128-bit quantization for target 1.

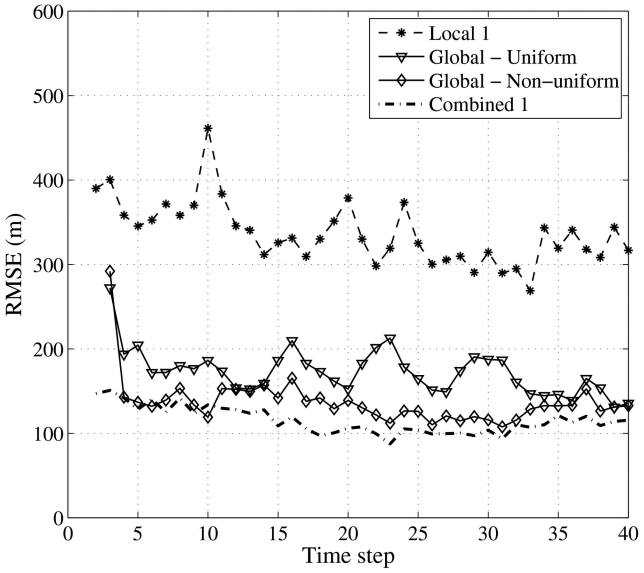

Fig. 11. Position RMSE comparison with 128-bit quantization for target 2.

and $\mathbf{v}_k$ is zero-mean white Gaussian noise with covariance $\mathbf{Q}$ given by

$$\mathbf{Q} = \begin{bmatrix} \frac{1}{3}T^3 & \frac{1}{2}T^2 & 0 & 0 \\ \frac{1}{2}T^2 & T & 0 & 0 \\ 0 & 0 & \frac{1}{3}T^3 & \frac{1}{2}T^2 \\ 0 & 0 & \frac{1}{2}T^2 & T \end{bmatrix} q \tag{37}$$

where $q = 0.001$ is the level of process noise in target motion.

Targets have different stating times and starting positions within the surveillance region. Target 1 and target 2 are present at $k = 0$, and their initial target positions are $(-10 \times 10^3, -15 \times 10^3)$ and $(-5 \times 10^3, 9 \times 10^3)$ m. Target 3 enters later at time $k = 10$ from the position

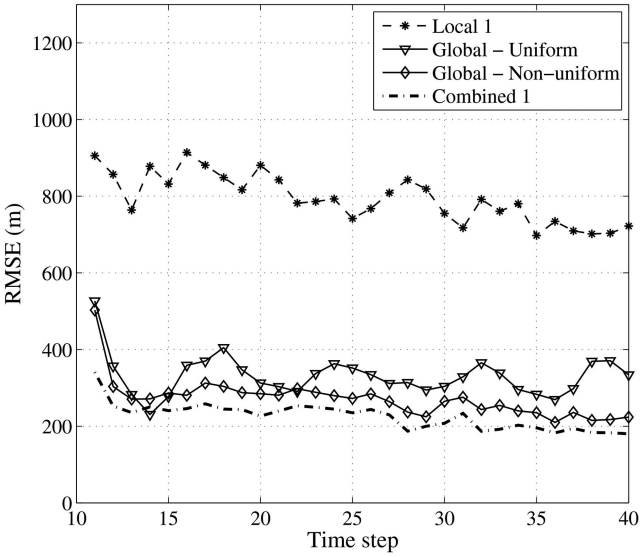

Fig. 12. Position RMSE comparison with 128-bit quantization for target 3.

$(15 \times 10^3, -10 \times 10^3)$ m. The targets' initial velocities are $(5,5)$, $(-4,3)$, $(-5,2)$ ms^{-1}. The target trajectories and sensor network arrangement are shown in Fig. 9.

The target generated measurements corresponding to target j on sensor i

$$z_k^{i,j} = \tan^{-1}\left(\frac{y_k^j - y_S^i}{x_k^j - x_S^i}\right) + v_k^i \tag{38}$$

where v_k^i is an i.i.d. sequence of zero-mean Gaussian variables with standard deviation 0.01 rad. The jth target location is denoted by (x_k^j, y_k^j) and that of ith sensor are denoted by (x_S^i, y_S^i). Additional parameters used in the simulations are: the probability of target survival = 0.99; the probability of target birth = 0.05; the probability of target spawning = 0; number of particles representing one target = 1000; the false alarm density $\lambda = 4 \times 10^{-3}$ rad^{-1}. The simulation results are based on 100 Monte Carlo runs.

5.2. Simulation Results

Figures 10, 11 and 12 show position Root Mean Square Errors (RMSEs) comparison for target 1, 2 and 3, respectively. RMSE values are computed from 100 Monte-Carlo runs. In those figures, 'Local 1' indicates the tracker at the fusion center 1 using only the measurement from local sensors; 'Global-Uniform' and 'Global-Non-uniform' indicate the trackers that use the uniformly and non-uniformly quantized measurement from all the fusion centers, respectively; 'Combined 1' indicates the tracker running at fusion center 1 that uses the quantized measurements from neighboring fusion center and the non-quantized local measurements. As expected, 'Combined 1' gives better performance than all the other trackers. Non-uniform quantization gives better performance than uniform quantization as well. Since the measurements from fusion center 2 are not

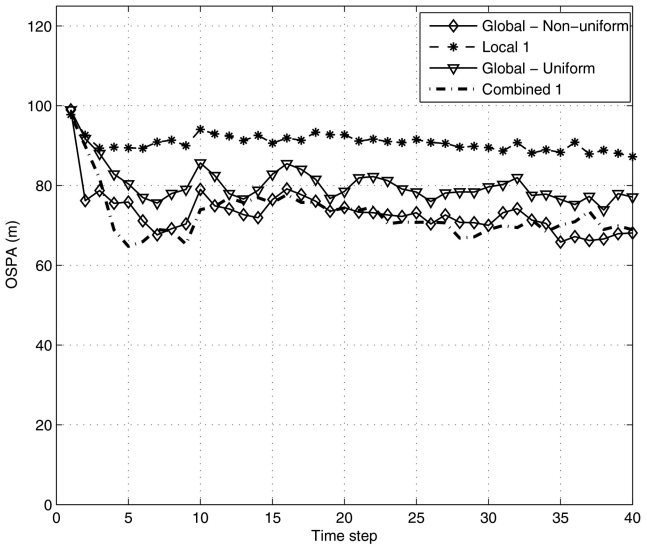

Fig. 13. OSPA comparison with 128-bin quantization.

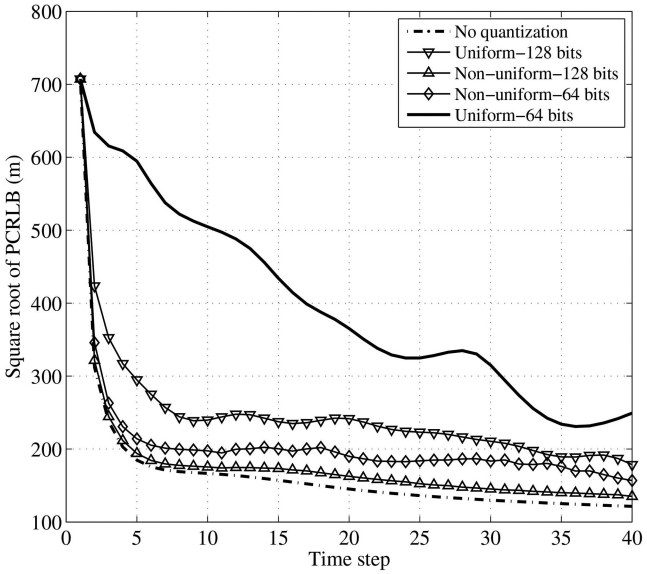

Fig. 14. Position PCRLB comparison for target 1.

used, the performance of the 'Local 1' tracker is the worst. Even though 'Combined 1' gives the best performance, its estimates cannot be used for quantization and encoding, since 'Combined 1's results, which are needed for decoding and decompanding, are not available at fusion center 2. However, 'Combined 1' can be used to eliminate the false alarms. Figure 13 shows Optimal Subpattern Assignment (OSPA) [21] distance comparison of the aforementioned approaches with OSPA parameters, $c = 25$ m, $p = 1$.

The PCRLB comparison for target 1 with various approaches is given in Fig. 14. From this figure, it can be noticed that non-uniform quantization performs close to the optimal performance, i.e., without quantization. Also, non-uniform quantization with 64 bits performs better than uniform quantization with 128 bits. Hence, non-uniform quantization can also be used to reduce the communication load in addition to improving the tracking performance.

The numbers of bits transmitted with and without Huffman coding are shown in Figs. 15, 16 and 17, where the overhead bits are not included. The effect of false alarms on communication load is shown in Fig. 15. In general, most of the false alarms are away from the target originated measurements. Hence, the number of bits allocated for the false alarms using Huffman coding, which used the probability density function of the target originated measurement, is very high. As a result Huffman coding will result in poor performance unless the false alarms are not eliminated.

After false alarms are eliminated as explained in Section 3.3, the number of bits transmitted is significantly reduced when Huffman coding is used with uniform quantization (see Fig. 16). When a new target enters and is detected by the local fusion center, the number of bit allocated for the new target originated measurement is high as the global estimate does not have information about the new target. Once the target is initialized the Huffman dictionary takes into account the new target so the encoded measurements have fewer bits. This can be observed at time step 11. Also, it is not possible to eliminate all the false alarms at all the times. Especially, it is hard to eliminate a false alarm if it falls close to any of the existing targets. This could be the reason for the slight increase in the number of bits at time step 24.

The number of bits transmitted using Huffman coding with non-uniform quantization is shown in Fig. 17. During the non-uniform encoding, the probability distribution is uniform over the measurement bins. As a result, there is no reduction in the number of bits transmitted. Hence, it is better to use no encoding with non-uniform quantization.

6. CONCLUSIONS

In this paper, a distributed implementation of SMC-PHD filter and an efficient quantization and encoding for communicating measurements were considered. Communication resources need to be handled efficiently in sensor networks while maximizing the tracking performance. False alarms take significant communication resources unless their communication is handled properly. A non-uniform quantization via companding was implemented to take advantages of the filter properties. It ensures that the target-originated measurements are quantized with less errors than others. An effective way of eliminating false alarms was also implemented. Posterior covariance was derived to access the algorithm using a recursive formula for the Fisher Information Matrix. Simulation studies confirm that the proposed quantization, encoding and false alarm elimination techniques are shown to be more efficient in terms of communication resource utilization and tracking performance than unencoded techniques. The pro-

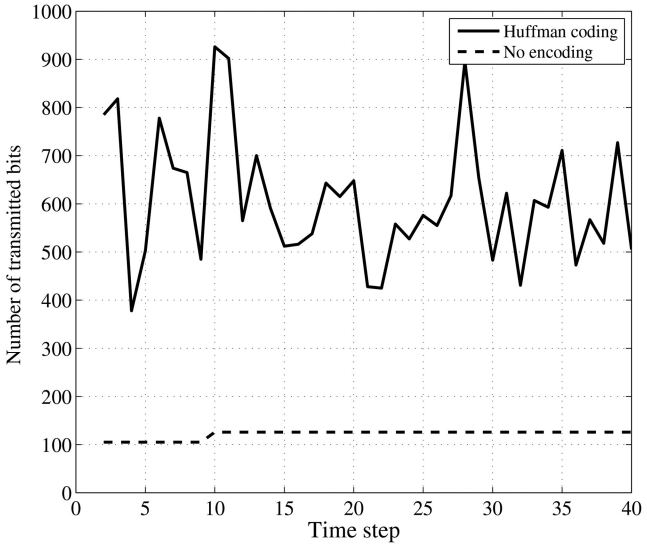

Fig. 15. Number of bits transmitted without false alarms elimination (with 3 false alarms at each time step).

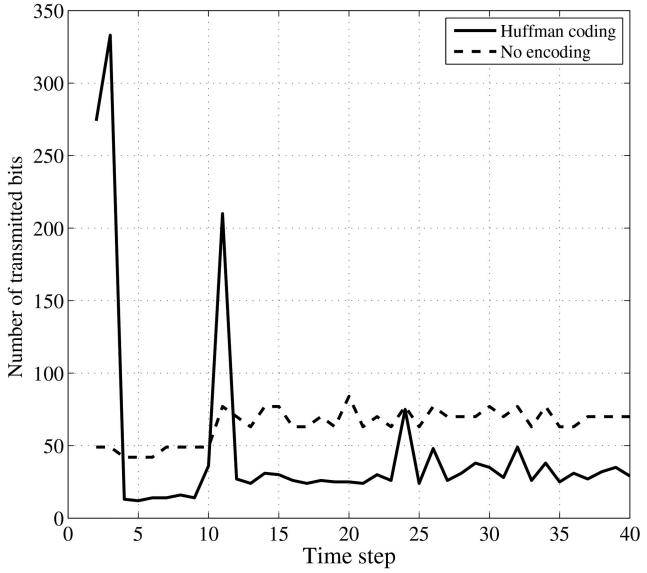

Fig. 16. Number of bits transmitted with uniform quantization and false alarm elimination.

posed distributed algorithm for SMC-PHD filter is also shown effective when the results were compared to its performance bound.

REFERENCES

[1] A. Aravinthan
Distributed tracking with probability hypothesis density filters using efficient measurement encoding.
Open Access Dissertations and Theses, Paper 4322, 2009.

[2] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp
A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking.
IEEE Transactions on Singal Processing, **50**, 2 (Feb. 2002), 173–188.

[3] Y. Bar-Shalom and X. R. Li
Multitarget-Multisensor Tracking: Principles and Techniques.
Storrs, CT: YBS Publishing, 1995.

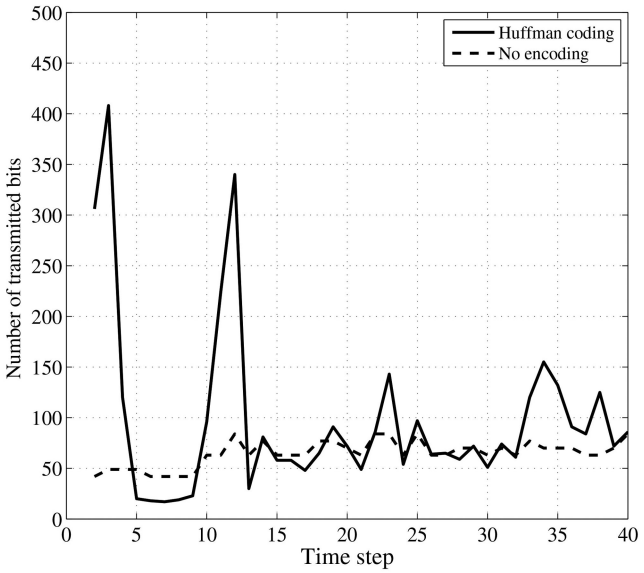

Fig. 17. Number of bits transmitted with non-uniform quantization and false alarm elimination.

[4] S. S. Blackman
Multiple hypothesis tracking for multiple target tracking.
IEEE Aerospace and Electronic Systems Megazine, **19**, 1 (Jan. 2004), 5–18.

[5] M. J. Coates
Distributed particle filters for sensor networks.
Proceedings of Third International Symposium on Information Processing in Sensor Networks, Apr. 2004, pp. 99–107.

[6] T. M. Cover and J. A. Thomas
Elements of Information Theory.
New York: Wiley, 1991.

[7] A. Doucet, N. De Freitas, and N. Gordon
An Introduction to Sequential Monte Carlo Methods.
New York: Springer-Verlag, 2001.

[8] Z. Duan, V. P. Jilkov, and X. R. Li
Posterior Cramer-Rao bounds for state estimation with quantized measurement.
Proceedings of 40th Southestern Symposium on System Theory, New Orleans, LA, Apr. 2008, pp. 376–380.

[9] R. M. Gray and D. L. Neuhoff
Quantization.
IEEE Transactions on Information Theory, **44**, 6 (Oct. 1998).

[10] D. L. Hall
Mathematical Techniques in Multisensor Data Fusion.
Norwood, MA: Artech House, 1992.

[11] M. L. Hernandez, T. Kirubarajan, and Y. Bar-Shalom
Multisensor resource deployment using posterior Cramer-Rao bounds.
IEEE Transaction on Aerospace and Electronic Systems, **40**, 2 (Apr. 2004), 399–416.

[12] D. A. Huffman
A method for the construction of minimum-redundancy codes.
Proceedings of the I.R.E., **40** (Sept. 1952), 1098–1101.

[13] G. Ing and M. J. Coates
Parallel particle filters for tracking in wireless sensor networks.
Proceedings of IEEE 6th Workshop on Signal Processing Advances in Wireless Communications, June 2005, pp. 935–939.

[14] R. P. S. Mahler
Multitarget Bayes filtering via first-order multitarget moments.
IEEE Transactions on Aerospace and Electronics Systems, **39**, 4 (Oct. 2003), 1152–1178.

[15] R. P. S. Mahler
Multitarget moments and their application to mulitarget tracking.
Proceedings of the Workshop on Estimation, Tracking and Fusion: A Tribute to Yaakov Bar-Shalom, Monterey, CA, 2001, pp. 134–166.

[16] F. Palmieri, S. Marano, and P. Willett
Measurement fusion for target tracking under bandwidth constraints.
Proceedings of the 2001 IEEE Aerospace Conference on Information Fusion, Big Sky, MT, Mar. 2001.

[17] P. Peebles
Digital Communication Systems.
Upper Saddle River, NJ: Prentice-Hall, 1987.

[18] M. Rosencrant, G. Gordon, and S. Thrun
Decentralized sensor fusion with distributed particle filters.
Proceedings of the Conference on Uncertainty in Artificial Intelligence, Acapulco, Mexico, Aug. 2003.

[19] Y. Ruan, P. Willett, A. Marrs, F. Palmieri, and S. Marano
Practical fusion of quantized masurements via particle filtering.
IEEE Transactions on Aerospace and Electronic Systems, **44**, 1 (Jan. 2008), 15–29.

[20] K. Sayood
Introduction to Data Compression.
Morgan Kaufman Publishers, 2nd ed., 2000.

[21] D. Schuhmacher, B-T. Vo, and B-N. Vo
A consistent metric for performance evaluation in multi-object filtering.
IEEE Transactions on Signal Processing, **56**, 8 (Aug. 2008), 3447–3457.

[22] X. Sheng, Y. H. Hu, and P. Ramanathan
Distributed particle filter with GMM approximation for multiple targets localization and tracking in wireless sensor network.
Proceedings of Fourth International Symposium on Information Processing in Sensor Networks, Apr. 2005, pp. 181–188.

[23] H. Van Trees
Detection, Estimation and Modulation Theory, vol. I.
New York: Wiley, 1968.

[24] B-N. Vo, S. Singh, and A. Doucet
Sequential Monte Carlo methods for multitarget filtering with random finite sets.
IEEE Transactions on Aerospace and Electronic Systems, **41**, 4 (Oct. 2005), 1224–1245.

[25] B-N. Vo and W-K. Ma
The Gaussian mixture probability hypothesis density filter.
IEEE Transactions on Signal Processing, **54**, 11 (Nov. 2006), 4091–4104.

[26] E. Waltz and J. Llinas
Multisensor Data Fusion.
Norwood, MA: Artech House, 1990.

[27] I. H. Witten, R. M. Neal, and J. G. Cleary
Arithmetic coding for data compression.
Communications of the ACM, **30**, 6 (June 1987), 520–540.

[28] X. Zhang and P. Willett
Cramer-Rao bounds for discrete-time linear filtering with measurement origin uncertainties.
Proceedings of the Workshop on Estimation, Tracking and Fusion: A Tribute to Yaakov Bar-Shalom, Monterey, CA, May 2001, pp. 546–560.

Biruk K. Habtemariam received the B.Sc. degree in electrical engineering from Mekelle University, Ethiopia, in 2007, and M.A.Sc. degree in electrical and computer engineering from McMaster University, Canada, in 2010.

Currently he is a research assistant/Ph.D. student in the Electrical and Computer Engineering Department at McMaster University. From 2007 to 2008 he was a graduate assistant in the Electrical Engineering Department at Mekelle University. His research interests include information fusion, detection/estimation theory, and target tracking. He is a recipent of International Excellence Award in 2011.

Ampikathasan Aravinthan received the B.Sc.Eng. and M.Sc.Eng. degrees in electronic and telecommunication engineering from University of Moratuwa, Sri Lanka, in 2002 and 2005, respecteively. He received his M.A.Sc. degree in electrical and computer engineering from McMaster University, Canada in 2009.

His research interests include estimation, target tracking and information theory.

Ratnasingham Tharmarasa received the B.Sc.Eng. degree in electronic and telecommunication engineering from University of Moratuwa, Sri Lanka in 2001, and the M.A.Sc. and Ph.D. degrees in electrical engineering from McMaster University, Canada in 2003 and 2007, respectively.

From 2001 to 2002 he was an instructor in electronic and telecommunication engineering at the University of Moratuwa, Sri Lanka. During 2002–2007 he was a graduate student/research assistant in ECE Department at McMaster University, Canada. Currently he is working as a research associate in the Electrical and Computer Engineering Department at McMaster University, Canada. His research interests include target tracking, information fusion and sensor resource management.

Kumaradevan Punithakumar received the B.Sc.Eng. (with First class Hons.) degree in electronic and telecommunication engineering from the University of Moratuwa, Moratuwa, Sri Lanka, in 2001, and the M.A.Sc. and Ph.D. degrees in electrical and computer engineering from McMaster University, Hamilton, ON, Canada, in 2003 and 2007, respectively.

From 2002 to 2007, he was a teaching assistant in the Electrical and Computer Engineering Department, McMaster University, where he became a postdoctoral research fellow in 2008. He is currently an imaging research scientist at GE Healthcare, London, ON, Canada. His research interests include medical image analysis, target tracking, sensor management, and computer vision.

Dr. Punithakumar was the recipient of the Industrial R&D Fellowship by the National Sciences and Engineering Research Council of Canada in 2008.

Tom Lang received Bachelor (1983) and Master (1985) of Engineering degrees in engineering physics from the Faculty of Engineering at McMaster University in Hamilton, Ontario, Canada.

Since 1985, he has been employed by General Dynamics Canada in Ottawa, Ontario, where he pursues research and development in sonar signal and data processing. In 2007, he was appointed an adjunct professor in the Department of Electrical and Computer Engineering at McMaster University. His primary research interests include sonar signal processing, multitarget tracking, and multisensor data fusion. He currently holds memberships in the IEEE Aerospace and Electronic Systems Society and the Acoustical Society of America.

Thiagalingam Kirubarajan received the B.A. and M.A. degrees in electrical and information engineering from Cambridge University, England, in 1991 and 1993, and the M.S. and Ph.D. degrees in electrical engineering from the University of Connecticut, Storrs, in 1995 and 1998, respectively.

Currently, he is a professor in the Electrical and Computer Engineering Department at McMaster University, Hamilton, Ontario. He is also serving as an Adjunct Assistant Professor and Associate Director of the Estimation and Signal Processing Research Laboratory at the University of Connecticut. His research interests are in estimation, target tracking, multisource information fusion, sensor resource management, signal detection and fault diagnosis. His research activities at McMaster University and at the University of Connecticut are supported by U.S. Missile Defense Agency, U.S. Office of Naval Research, NASA, Qualtech Systems, Inc., Raytheon Canada Ltd. and Defense Research Development Canada, Ottawa. In September 2001, Dr. Kirubarajan served in a DARPA expert panel on unattended surveillance, homeland defense and counterterrorism. He has also served as a consultant in these areas to a number of companies, including Motorola Corporation, Northrop-Grumman Corporation, Pacific-Sierra Research Corporation, Lockhead Martin Corporation, Qualtech Systems, Inc., Orincon Corporation and BAE systems. He has worked on the development of a number of engineering software programs, including BEARDAT for target localization from bearing and frequency measurements in clutter, FUSEDAT for fusion of multisensor data for tracking. He has also worked with Qualtech Systems, Inc., to develop an advanced fault diagnosis engine.

Dr. Kirubarajan has published about 100 articles in areas of his research interests, in addition to one book on estimation, tracking and navigation and two edited volumes. He is a recipient of Ontario Premier's Research Excellence Award (2002).