An Information Based Sensor Management Method for Multitarget Tracking

Chris Kreucher[a], Keith Kastella[a], and Alfred O. Hero III[b]

[a]Veridian's Ann Arbor Research and Development Center, Ann Arbor, MI, USA
[b]The University of Michigan Department of EECS, Ann Arbor, MI, USA

ABSTRACT

This paper presents an information based method for sensor management based on tasking a sensor to make the measurement that maximizes the *expected* gain in information. The method is applied to the problem of tracking multiple targets. The underlying tracking methodology is a multiple target tracking scheme based on recursive estimation of a Joint Multitarget Probability Density (JMPD), which is implemented using particle filtering methods. The sensor management scheme is predicated on choosing to make the measurement that maximizes the expected Rényi Information Divergence (a generalization of the Kullback-Leibler divergence) between the current JMPD and the JMPD after a measurement has been made.

Keywords: Sensor Management, Particle Filtering, Multi-target Tracking

1. INTRODUCTION

The problem of sensor management is to determine the best way to task a sensor where the sensor may have many modes and pointing options. A typical application, and one that we focus on in our model problems, is to direct an electronically scanned aperture (ESA) radar.[1] An ESA provides great flexibility in pointing and mode selection. For example, the beam can be redirected in microseconds, enabling targets to be illuminated at will.

We propose here a sensor tasking algorithm that is motivated by information theory. In this work, we utilize an information measure called the Rényi Information Divergence, which reduces to the Kullback-Leibler divergence under a certain limit. The Rényi divergence has additional flexibility in that in allows for emphasis to be placed on specific portions of the information.

We apply our sensor management scheme to the problem of tracking a collection of moving targets. First, we utilize a target tracking algorithm to recursively estimate the joint multitarget probability density for the set of targets. We then strive to task the sensor in such a way that the sensing action it makes results in the maximum amount of information gain. The decision as to how to use a sensor then becomes one of determining which sensing action will maximize the expected information gain between the current joint multitarget probability density and the joint multitarget probability density after a measurement has been made.

2. THE JOINT MULTITARGET PROBABILITY DENSITY

The joint multitarget probability density (JMPD) provides a means for tracking an unknown number of targets in a Bayesian setting*. In short, the JMPD $p(\mathbf{x}_1^k, \mathbf{x}_2^k, ...\mathbf{x}_{T-1}^k, \mathbf{x}_T^k|\mathbf{Z}^k)$ is the probability that there are exactly T targets with states $\mathbf{x}_1^k, \mathbf{x}_2^k, ...\mathbf{x}_{T-1}^k, \mathbf{x}_T^k$ in the surveillance region at time k based on the set of observations $\mathbf{Z}^k$. The number of targets T is a variable to be estimated simultaneously with the states of the T targets. The observation set $\mathbf{Z}^k$ refers to the collection of measurements up to and including time k, i.e. $\mathbf{Z}^k = \{\mathbf{z}^1, \mathbf{z}^2, ...\mathbf{z}^k\}$, where each of the $\mathbf{z}^i$ may be a single measurement or a vector of measurements made at time i.

Each state vector $\mathbf{x}_i$ in the density $p(\mathbf{x}_1^k, \mathbf{x}_2^k, ...\mathbf{x}_{T-1}^k, \mathbf{x}_T^k|\mathbf{Z}^k)$ is a vector quantity and may (for example) be of the form $[x, \dot{x}, y, \dot{y}]'$. For convenience, the density will be written compactly in the traditional manner as $p(\mathbf{X}^k|\mathbf{Z}^k)$, with the understanding that the state-vector $\mathbf{X}$ represents a variable number of targets each possessing their own state vector. We refer to each of the T target state vectors $\mathbf{x}_1^k, \mathbf{x}_2^k, ...\mathbf{x}_{T-1}^k, \mathbf{x}_T^k$ as a partition of $\mathbf{X}$.

Please Direct correspondence to Christopher.Kreucher@veridian.com

*More detail on JMPD and the subsequent particle filter implementation may be found in Kreucher[4,5]

The temporal update of the posterior likelihood proceeds according to the usual rules of Bayesian filtering. Given a model of state dynamics $p(\mathbf{X}^k|\mathbf{X}^{k-1})$, we may compute the time-updated or prediction density via

$$p(\mathbf{X}^k|\mathbf{Z}^{k-1}) = \int d\mathbf{X}^{k-1} p(\mathbf{X}^k|\mathbf{X}^{k-1}) p(\mathbf{X}^{k-1}|\mathbf{Z}^{k-1}) \tag{1}$$

Bayes rule enables us to update the posterior density as new measurements $\mathbf{z}^k$ arrive as

$$p(\mathbf{X}^k|\mathbf{Z}^k) = \frac{p(\mathbf{z}^k|\mathbf{X}^k)p(\mathbf{X}^k|\mathbf{Z}^{k-1})}{p(\mathbf{z}^k|\mathbf{Z}^{k-1})} \tag{2}$$

In practice, the sample space of $\mathbf{X}^k$ is very large. It contains all possible configurations of state vectors $\mathbf{x}_i$ for all possible values of T. The original formulation of JMPD given by Kastella[3] approximates the density by discretizing on a grid. It was found that the computational burden in this scenario makes realistic problems intractable, even with the simple model of targets moving between discrete locations in one-dimension. The Monte Carlo methods collectively known as particle filtering break this logjam.

3. THE PARTICLE FILTER IMPLEMENTATION OF JMPD

To implement JMPD via a particle filter (PF), we first approximate the joint multitarget probability density $p(\mathbf{X}|\mathbf{Z})$ by a set of N_{part} weighted samples, $\mathbf{X}_p, (p = 1...N_{part})$:

$$p(\mathbf{X}|\mathbf{Z}) \approx \sum_{p=1}^{N_{part}} w_p \delta(\mathbf{X} - \mathbf{X}_p) \tag{3}$$

Recall from Section 2 that our multitarget state vector $\mathbf{X}$ has T partitions, each corresponding to a target:

$$\mathbf{X} = [\mathbf{x}_1,\ \mathbf{x}_2,\ ...,\ \mathbf{x}_{T-1},\ \mathbf{x}_T] \tag{4}$$

Furthermore, the joint multitarget probability $p(\mathbf{X}|\mathbf{Z})$ is defined for $T = 0...\infty$. Each of the particles $\mathbf{X}_p$, $p = 1...N_{part}$ is a sample drawn from $p(\mathbf{X}|\mathbf{Z})$. Therefore, a particle $\mathbf{X}_p$ may have any number of partitions from 0 to ∞, each partition corresponding to a different target. In practice, of course, the maximum number of targets under surveillance is truncated at some finite number T. We will denote the number of partitions in particle $\mathbf{X}_p$ by n_p, where n_p may be different for different $\mathbf{X}_p$. Since a partition corresponds to a target, the number of partitions that a particle has is that particle's estimate of the number of targets in the surveillance area.

4. RÉNYI INFORMATION DIVERGENCE FOR SENSOR MANAGEMENT

Our information based method for tasking the sensor is to choose the sensing action that maximizes the expected information gain. To that end, our algorithm proceeds by first enumerating all possible sensing actions. A sensing action may consist of choosing a particular mode (i.e. SAR mode versus GMTI mode), a particular dwell point, or a combination of the two. We next calculate the *expected* information gain in making each of the possible sensing actions, and select to take the action that yields the maximum expected information gain.

The calculation of information gain between two densities f_1 and f_0 is done using the Rényi information divergence (5), also known as the α-divergence:

$$D_\alpha(f_1||f_0) = \frac{1}{\alpha - 1} ln \int f_1^\alpha(x) f_0^{1-\alpha}(x) dx \tag{5}$$

Since we are interested in computing the divergence between the predicted density $p(\mathbf{X}|\mathbf{Z}^{k-1})$ and the updated density after a measurement is made, $p(\mathbf{X}|\mathbf{Z}^k)$ we write

$$D_\alpha\left(p(\mathbf{X}|\mathbf{Z}^k)||p(\mathbf{X}|\mathbf{Z}^{k-1})\right) = \frac{1}{\alpha-1} ln \sum_{\mathbf{X}} p(\mathbf{X}|\mathbf{Z}^k)^\alpha p(\mathbf{X}|\mathbf{Z}^{k-1})^{1-\alpha} \tag{6}$$

The integral in equation (5) reduces to a summation since any discrete approximation of $p(\mathbf{X}|\mathbf{Z}^{k-1})$ only has nonzero probability at a finite number of target states. After some algebra, this quantity simplifies to

$$D_\alpha\left(p(\mathbf{X}|\mathbf{Z}^k)||p(\mathbf{X}|\mathbf{Z}^{k-1})\right) = \frac{1}{\alpha-1} ln \frac{1}{p(\mathbf{z}|\mathbf{Z}^{k-1})^\alpha} \sum_{\mathbf{X}} p(\mathbf{X}|\mathbf{Z}^{k-1}) p(\mathbf{z}|\mathbf{X})^\alpha \tag{7}$$

Our particle filter approximation of the density reduces equation (7) to

$$D_\alpha\left(p(\mathbf{X}|\mathbf{Z}^k)||p(\mathbf{X}|\mathbf{Z}^{k-1})\right) = \frac{1}{\alpha-1} ln \frac{1}{p(\mathbf{z})^\alpha} \sum_{p=1}^{N_{part}} w_p p(\mathbf{z}|\mathbf{X_p})^\alpha \tag{8}$$

We would like to choose to perform the measurement that makes the divergence between the current density and the density after a new measurement has been made as large as possible. This indicates that the sensing action has maximally increased the information content of the measurement updated density with respect to the density before a measurement was made. We propose, then, as a method of sensor management calculating the expected value of equation (8) for each possible sensing action and choosing the action that maximizes the expectation. A sensing action refers to any activity under consideration, including but not limited to mode selection and beam positioning.

The expected value of equation (8) may be written as an integral over all possible outcomes z_m when performing sensing action m. In the special case where measurements are thresholded and are therefore either detections or no-detections (i.e. $z = 0$ or $z = 1$), this integral reduces to a summation over the possible measurements, which using equation (8) becomes simply

$$< D_\alpha >_m = \frac{1}{\alpha-1} \sum_{z=0}^{1} p(z) ln \frac{1}{p(z)^\alpha} \sum_{p=1}^{N_{part}} w_p p(z|\mathbf{X}_p)^\alpha \tag{9}$$

5. SIMULATION RESULTS

We test the performance of the sensor management (SM) scheme by considering the following model problem. Three targets move on a $12x12$ sensor grid. Each target is modeled using the four-dimensional state vector $[x, \dot{x}, y, \dot{y}]'$. Target motion is simulated using a constant-velocity (CV) model with a diffusive component. The trajectories have been shifted and time delayed so that there are two times during the simulation where targets cross paths (i.e. come within sensor resolution of each other), to make the problem challenging.

The target kinematics assumed by the filter are CV as in the simulation. At each time step, a set of L (not necessarily distinct) cells are measured. The sensor is at a fixed location above the targets and all cells are always visible to the sensor. When measuring a cell, the imager returns either a 0 (no detection) or a 1 (detection) governed by P_d, P_f, and SNR. This model is known by the filter and used to evaluate (2). In this illustration, we take $P_d = 0.5$, and $P_f = P_d^{(1+SNR)}$, which is a standard model for thresholded detection of Rayleigh returns.

We contrast the performance of the tracker when the sensor uses a non-managed (periodic) scheme versus the performance using the managed scheme. The periodic scheme measures cells in sequence. At time 1, cells $1...L$ are measured. At time 2, cells $L+1...2L$ are measured. This sequence continues until all cells have been measured, at which time the scheme resets. The managed scheme uses the expected information divergence to calculate the best L cells to measure at each time.

Fig. 1 presents a single-time snapshot from the tracker illustrating the difference between the two schemes. The managed scheme is shown on the left and the periodic sheme on the right. In both panes, the three targets are marked with an asterisk, the covariance ellipses of the estimated target position are shown, and we use gray scale to indicate the number of times each cell has been measured at this time step. Qualitatively, in the managed scenario the measurements are focused in or near the cells that the targets are in. Furthermore, the covariance ellipses are much tighter. In fact, the non-managed scenario has confusion about which tracks correspond to which target as the covariance ellipses overlap.

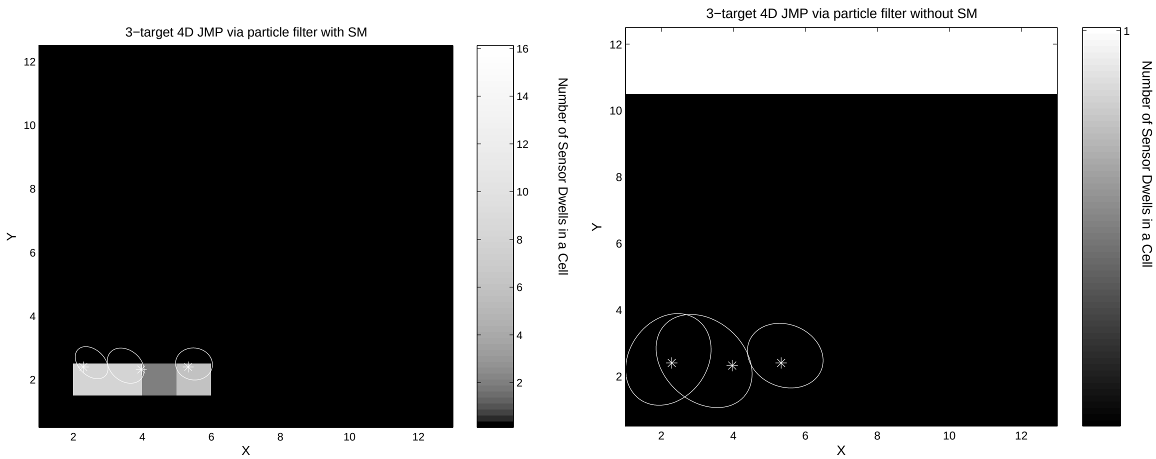

Figure 1. Comparison of Non-Managed and Managed Tracking. (L) Using SM, and (R) A Periodic Scheme.

A more detailed examination is provided in the Monte Carlo simulation results of Figure 2. The SM algorithm was run with $L = 24$ (i.e. was able to scan 24 cells at each time step) and is compared to the non-managed scheme with 24 to 312 looks. The unmanaged scenario needs approximately 312 looks to equal the performance of the managed algorithm in terms of RMSE error. We say that the sensor manager is approximately 13 times as efficient as allocating the sensors without management.

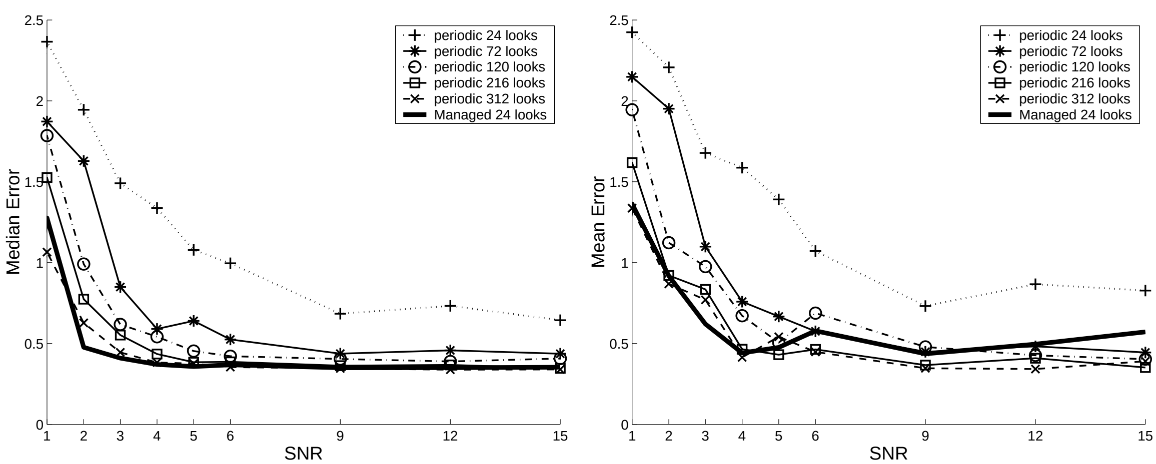

Figure 2. Median and Mean Error vs. Signal To Noise Ratio (SNR). Managed Performance With 24 Looks is Similar to Unmanaged Performance With 312 Looks.

As a second simulation, we test the SM algorithm in a situation intended to demonstrate the technique in a scenario of increased realism. Here we have ten targets moving in a $5000m$ x $5000m$ surveillance area. Each target is modeled using the four-dimensional state vector $[x, \dot{x}, y, \dot{y}]'$. Target trajectories come directly from a set of recorded data based on GPS measurements of vehicle positions over time collected as part of a battle training exercise at NTC. Targets routinely come with in sensor cell resolution (i.e. cross). Target positions are recorded at 1 second intervals, and the simulation duration is 1000 time steps.

We compare the performance of the managed and unmanaged scenarios in Figure 3. Our method of comparison here is to determine empirically the number of Looks needed in the unmanaged scenario to achieve the same performance as the managed algorithm with $L = 50$ looks. We see that the unmanaged scenario needs approximately 600 to 700 looks to equal the performance of the managed algorithm in terms of RMSE error. Therefore, the sensor manager is approximately 13 times as efficient as allocating the sensors without management.

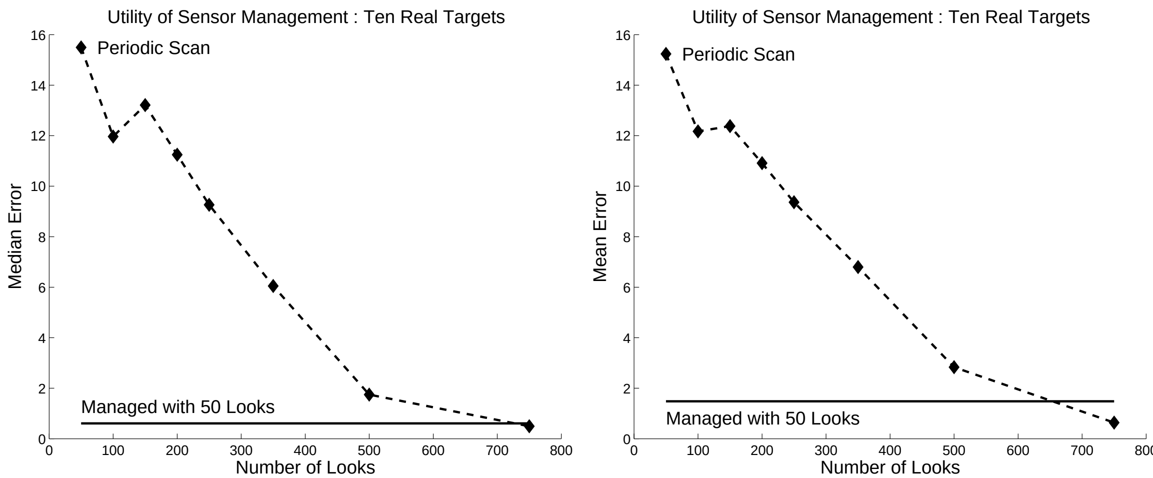

Figure 3. Error vs. Number of Looks. Managed Performance With 50 Looks Similar to Unmanaged with 600 Looks.

6. DISCUSSION

The information-based sensor management scheme presented here is based on computing the expected information gain for each sensor tasking under consideration. The sensor management algorithm is integrated with the target tracking algorithm in that it uses the posterior density $p(\mathbf{X}|\mathbf{Z})$ approximated by the multitarget tracker. The posterior is used in conjunction with target kinematic models and sensor models to predict which measurements will provide the most information gain. In simulated scenarios, we find that the tracker with sensor management gives similar performance to the tracker without sensor management with more than a ten-fold improvement in sensor efficiency.

ACKNOWLEDGMENTS

This work was supported under the United States Air Force contract F33615-02-C-119, Air Force Research Laboratory contract SPO900-96-D-0080 and by ARO-DARPA MURI Grant DAAD19-02-1-0262. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the United States Air Force.

REFERENCES

1. S.S. Blackman, Mulitple-Target Tracking with Radar Applications. Norwood, MA. Artech House, 1986.
2. A. O. Hero, B. Ma, O. Michel and J. Gorman, "Applications of entropic spanning graphs", IEEE Signal Processing Magazine (Special Issue on Mathematics in Imaging), Vol 19, No. 5, pp 85-95, Sept. 2002.
3. K. Kastella, "Joint multitarget probabilities for detection and tracking", SPIE Proceedings, Acquisition, Tracking and Pointing XI, 21-25 April, 1997, Orlando, FL.
4. Kreucher, C., Kastella, K., and Hero III, Alfred O., "Tracking Multiple Targets Using a Particle Filter Representation of the Joint Multitarget Probability Density", SPIE Signal and Data Processing of Small Targets Conference, San Diego, CA, August 2003.
5. Kreucher, C., Kastella, K., and Hero III, Alfred O., "Information Based Sensor Management for Multitarget Tracking", SPIE Signal and Data Processing of Small Targets Conference, San Diego, CA, August 2003.