MONTE CARLO METHODS FOR SENSOR MANAGEMENT IN TARGET TRACKING

Christopher M. Kreucher

General Dynamics
Michigan R&D Center
Ypsilanti, MI

Alfred O. Hero III

The University of Michigan
EECS Department
Ann Arbor, MI

ABSTRACT

Surveillance for multi-target detection, identification and tracking is one of the natural problem domains in which particle filtering approaches have been gainfully applied. Sequential importance sampling is used to generate and update estimates of the joint multi-target probability density for the number of targets, their dynamical model, and their state vector. In many cases there are a large number of degrees of freedom in sensor deployment, e.g., choice of waveform or modality. This gives rise to a resource allocation problem that can be formulated as determining an optimal policy for a partially observable Markov decision process (POMDP). In this paper we summarize approaches to solving this problem which involve using particle filtering to estimate both posterior state probabilities and the expected reward for both myopic and multistage policies.

1. INTRODUCTION

This paper describes an application of particle filtering methods to the sensor management problem. Sensor management, as defined here, refers to the process of automatically choosing the best action (e.g., pointing direction, waveform, mode) for a collection of agile sensors at each time epoch for the purposes of providing surveillance. In our setting, a surveillance region consists of an unknown number of targets with unknown positions, velocities, movement modes, and identification. The sensor management problem studied here is therefore one of directing sensor actions over time so as to generate the best estimate of target number and state.

Our method of sensor management directs sensing assets to take actions that are expected to maximize information flow. Maximizing information flow is a flexible method that simultaneously captures many relevant surveillance goals, including detection, tracking, mode estimation, and identification. In fact, it can be shown that a closely related method bounds any task driven metric [1].

This work was supported under the United States Air Force contract F33615-02-C-1199, AFRL 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.

The key enabling element is an on-line recursive estimation of the joint multitarget probability density (JMPD), which is done using particle filtering methods. In order to choose actions that result in maximal information flow, one needs to construct a probability density that captures the knowledge of the surveillance region provided by the measurements made. This probability density then allows prediction of what actions will be valuable as measured by information flow. The correct probability density for this setting is the JMPD, which is a very high dimensional non-parametric object. It is this density which is estimated using particle filtering methods. Due to the high dimensionality of the JMPD, efficient importance density design is of paramount importance.

This paper proceeds as follows. In Section 2, we define the JMPD and describe the temporal evolution and measurement update. In Section 3, we describe the particle filter implementation which uses an adaptive importance density tailored to our particular problem. We highlight the novel characteristics of this problem which allow for an efficient implementation, which relies on adaptively factoring the density when permissible and using biassed sampling techniques that preferentially place particles in important regions of state space. Next, in Section 4, we describe how methods from information theory are combined with the JMPD to choose action sets that maximize information flow. We first describe single sensor myopic methods, and then show the extensions to multisensor and non-myopic methods. Section 5 presents a simulation showing the efficacy of the method. Finally, Section 6 provides some concluding remarks.

2. THE JMPD

Estimation of the joint multitarget probability density (JMPD) is a Bayesian method of fusing models of target behavior, sensor capability, and actual measurements into a single picture. In this sense, the JMPD captures all of the uncertainty about the surveillance region. Others have studied related methods based on Bayesian reasoning, e.g., the work in [2, 3]. This section and the following are abbreviated versions of [4].

The joint multitarget probability density (JMPD)

$$p(\mathbf{x}_1^k, \mathbf{x}_2^k, ...\mathbf{x}_{T-1}^k, \mathbf{x}_T^k, T^k | \mathbf{Z}^k) \ , \qquad (1)$$

is the probability density for exactly T targets with states $\mathbf{x}_1, \mathbf{x}_2, ...\mathbf{x}_{T-1}, \mathbf{x}_T$ at time k based on a set of past observations $\mathbf{Z}^k$. The JMPD is a continuous discrete hybrid as it is a product of the probability mass function $p(T^k|\mathbf{Z}^k)$ and the probability density function $p(\mathbf{x}_1^k, \mathbf{x}_2^k, ...\mathbf{x}_{T-1}^k, \mathbf{x}_T^k|T^k, \mathbf{Z}^k)$.

The number of targets at time k, T^k, is a variable estimated simultaneously with the states of the T^k targets. The JMPD is defined for all T^k, $T^k = 0 \cdots \infty$. The observation set $\mathbf{Z}^k$ refers to the collection of all measurements, 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 from time i.

Each $\mathbf{x}_t$ in $p(\mathbf{x}_1^k, \mathbf{x}_2^k, ...\mathbf{x}_{T-1}^k, \mathbf{x}_T^k, T^k|\mathbf{Z}^k)$ is a vector quantity and may (for example) be of the form $[x, \dot{x}, y, \dot{y}]$. For convenience, the JMPD will be written more compactly in the traditional manner as $p(\mathbf{X}^k, T^k|\mathbf{Z}^k)$, which implies that the state-vector $\mathbf{X}^k$ represents a variable number of targets each possessing their own state vector. We will omit the time subscript k when convenient and no confusion will arise, and e.g., write simply $p(\mathbf{X}, T|\mathbf{Z})$.

The likelihood $p(\mathbf{z}|\mathbf{X}, T)$ and the JMPD $p(\mathbf{X}, T|\mathbf{Z})$ are conventional Bayesian objects manipulated by the usual rules of probability and statistics. Thus, a multitarget system has state $\mathbf{X} = (\mathbf{x}_1, \cdots, \mathbf{x}_T)$ with probability distribution given by $p(\mathbf{X}, T|\mathbf{Z})$. This can be viewed as a hybrid stochastic system where the discrete random variable T governs the dimensionality of $\mathbf{X}$. The normalization is therefore

$$\sum_{T=0}^{\infty} \int d\mathbf{X} p(\mathbf{X}, T|\mathbf{Z}) = 1 \ , \tag{2}$$

where the single integral denotes the T integrations required.

The temporal update of the posterior proceeds according to the usual rules of Bayesian filtering. The model of JMPD time evolution is given by $p(\mathbf{X}^k, T^k|\mathbf{X}^{k-1}, T^{k-1})$ and will be referred to as the kinematic prior (KP). The kinematic prior includes models of target motion, target birth and death, and any additional prior information that may exist such as terrain and roadway maps. In the case where target identification is part of the state being estimated, different kinematic models may be used for different target types. Multiple models are also possible [5]. The time-updated (prediction) density is computed via the *model update* equation as

$$p(\mathbf{X}^k, T^k|\mathbf{Z}^{k-1}) = \tag{3}$$
$$\sum_{T^{k-1}=0}^{\infty} \int_{\mathbf{X}} d\mathbf{X}^{k-1} p(\mathbf{X}^k, T^k|\mathbf{X}^{k-1}, T^{k-1}) p(\mathbf{X}^{k-1}, T^{k-1}|\mathbf{Z}^{k-1})$$

The *measurement update* uses Bayes' rule to update the posterior density with a new measurement $\mathbf{z}^k$ as

$$p(\mathbf{X}^k, T^k|\mathbf{Z}^k) = \frac{p(\mathbf{z}^k|\mathbf{X}^k, T^k) p(\mathbf{X}^k, T^k|\mathbf{Z}^{k-1})}{p(\mathbf{z}^k|\mathbf{Z}^{k-1})} \ . \tag{4}$$

3. PARTICLE FILTER ESTIMATION OF THE JMPD

The sample space of the JMPD is very large since it contains all possible configurations of state vectors $\mathbf{x}_t$ for all possible values of T. Thus, to estimate the JMPD in a computationally tractable manner, a sophisticated approximation method is required. This section describes our particle filter implementation with special attention given to the adaptive importance density that allows tracability.

3.1. Notation

In particle filtering, the probability density of interest (here the JMPD) is represented by a set of N_{part} weighted samples. Here, a particle is more than just the estimate of the state of a target; it incorporates both an estimate of the states of all of the targets as well as an estimate of the number of targets.

The multitarget state vector for T targets is

$$\mathbf{X} = [\mathbf{x}_1, \ \mathbf{x}_2, \ ..., \ \mathbf{x}_{T-1}, \ \mathbf{x}_T] \ , \tag{5}$$

and particle p will be written as

$$\mathbf{X}_p = [\mathbf{x}_{p,1}, \ \mathbf{x}_{p,2}, \ \ldots \ \mathbf{x}_{p,T_p}] \ , \tag{6}$$

which says particle p estimates there are T_p targets, where T_p can be any non-negative integer. With δ_D denoting the Dirac delta, we define

$$\delta(\mathbf{X} - \mathbf{X}_p) = \begin{cases} 0 & T \neq T_p \\ \delta_D(\mathbf{X} - \mathbf{X}_p) & \text{otherwise} \end{cases} \tag{7}$$

Then the particle filter approximation to the JMPD is given by a set of particles $\mathbf{X}_p$ and corresponding weights w_p as

$$p(\mathbf{X}, T|\mathbf{Z}) \approx \sum_{p=1}^{N_{part}} w_p \delta(\mathbf{X} - \mathbf{X}_p) \tag{8}$$

where $\sum_{p=1}^{N_{part}} w_p = 1$.

The JMPD is defined for all possible numbers of targets, $T = 0, 1, 2, \cdots$. As each of the particles is a sample drawn from the JMPD, a particle may estimate $0, 1, 2, \cdots$ targets. Here, different particles in the approximation may correspond to different estimates of the number of targets.

3.2. Multitarget SIR

With these definitions, the SIR particle filter extends directly to filtering with the JMPD. The method is to simply proposes new particles at time k from the particles at time $k-1$ by projecting through the kinematic prior. This model includes both the dynamics of persistent targets and the model of how targets enter and exit the surveillance region. Specifically, targets entering or leaving the surveillance region are accounted for as the proposed particle $\mathbf{X}_p^k$ may have either fewer targets or more targets than $\mathbf{X}_p^{k-1}$ (i.e. $T_p^k = T_p^{k-1} - 1$ or $T_p^k = T_p^{k-1} + 1$). The weight update is simply

$$w_p^k = w_p^{k-1} p(\mathbf{z}^k|\mathbf{X}_p^k) \ . \tag{9}$$

3.3. The Inefficiency of the SIR Method

The SIR particle filter has the benefit of being simple to describe and easy to implement. However, the SIR is so numerically inefficient that multitarget problems are intractable.

Assume for discussion that the sensor is pixelated, returning energy in one of C sensor cells. Target birth may occur in any unoccupied cell at any time step. Target death may occur in any occupied cell at any time step. One method of handling this would be to have a very large number of particles, capable of encoding all possibilities of the next state, i.e. no new target, one new target (in each of the possible unoccupied cells), two new targets (in each possible pair of unoccupied cells), etc. and likewise with target removal. This must still retain the particle diversity required for efficient filtering. This method requires an enormous number of particles even with a first-order approximation that at most one target enters or leaves the region at each time step.

Furthermore, even with no birth and death, target proposals using the kinematics are too inefficient to be useful on multitarget problems. Consider the case where there are T targets in the surveillance region. In order for a particle to be a good estimate of the multitarget state, all T targets must be proposed to good locations. Without knowledge of the measurements, the probability an individual target is proposed to a good location is much less than 1. Therefore, as the number of targets grows, the number of particles required to perform good tracking with high probability grows exponentially.

Both of these problems are remedied via an importance density that more closely approximates the optimal importance density (i.e., uses current measurements to direct proposals to higher likelihood multitarget states). In the following subsections, we describe the importance density.

3.4. Importance Density Design for Target Birth/Death

In order to reach the efficiency required for tractable detection of multiple targets, we advocate a measurement directed sampling scheme for target birth and death. Specifically, we keep an existence grid (separate from the particles and tied to the sensor grid) which contains the probability that a single target is in cell i at time k given the measurements. Therefore, the existence grid is merely a single vector of floating point numbers, one for each sensor cell.

The existence grid cells are initialized with a prior probability which may be spatially varying. The probability of target existence in each cell is propagated forward via an addition/removal model analogous to the target motion model, and updated with new measurements according to Bayes' rule.

To handle target birth, new targets are preferentially added in locations according to the rate dictated by the existence grid. This bias is removed during the weight update process so that the Bayesian recursions are still exactly implemented. This implementational technique allows particles to be used more efficiently as new targets are only added in highly probable areas. Target death is handled analogously by removing according to the existence grid.

3.5. Importance Density Design for Persistent Targets

The kinematic prior does not take advantage of the fact that the JMPD state vector is made up of individual target state vectors. In particular, targets that are far apart in measurement space behave independently and should be treated as such. Furthermore, similar to that of the uniformed birth/death proposal, the current measurements are not used when proposing new particles. These two considerations taken together result in an inefficient use of particles and therefore require large numbers of particles to successfully track.

To overcome these deficiencies, we use a technique which biases the proposal process towards the measurements and allows for factorization of the multi-target state when permissible. These strategies propose each target in a particle separately, and form new particles as the combination of the proposed targets. We describe the use of two methods here, the independent partitions (IP) method of [6] and the coupled partitions (CP) method. The basic idea of both CP and IP is to construct particle proposals at the target (or group-of-targets) level, incorporating the measurements so as to bias the proposal towards the optimal importance density. This biased sampling is removed in the weight update stage. Therefore the Bayes recursions are still exactly implemented. We advocate an adaptive partition (AP) method which automatically switches between the two as appropriate. All of the methods are performed only on the persistent targets, and the algorithm is done in conjunction with the addition and removal of targets as described in the preceding section.

3.5.1. Independent-Partition (IP) Method

The independent partition (IP) method given by Orton [6] is a convenient way to propose particles when part or all of the joint multitarget density factors. As employed here, the IP method proposes a new target as follows. For a target t, each particle at time $k-1$ has it's t^{th} partition proposed via the kinematic prior and weighted by the measurements. From this set of N_{part} weighted estimates of the state of the t^{th} target, we select N_{part} samples with replacement to form the t^{th} partition of the particles at time k.

With well separated targets, this method allows many targets to be tracked with the same number of particles needed to track a single target. Indeed, in the case of well separated targets, the multitarget tracking problem breaks down into many single-target problems. The IP method is useful for just this case, as it allows the targets to be treated independently when their relative spacing deems that appropriate. Note, however, that applying this method on a target by target basis is not appropriate when there is any measurement-to-target associa-

tion ambiguity. Therefore, when targets are close together in sensor space, an alternative approach must be used.

3.5.2. *Coupled Partition (CP) Proposal Method*

When the posterior distributions on target position begin to overlap, we say that the corresponding partitions are coupled. In these instances, another method of particle proposal such as Coupled Partitions (CP) must be used. An alternative method would be to simply use the IP strategy on groups of partitions as alluded to in [6]. However, we find that the CP method described here provides a benefit by giving extra computation at those points where it is most necessary.

We apply the CP method as follows. To propose partitions $t_1 \cdots t_M$ of particle p, CP proposes R possible realizations of the future state using the kinematic prior. The R proposed futures are then given weights according to the current measurements and a single representative is selected. This process is repeated for each particle until the t^{th} partition for all particles has been formed. As in the IP method, the final particle weights are adjusted for this biased sampling.

3.5.3. *Adaptive Particle Proposal Method*

We use a hybrid if the IP and CP method, called the Adaptive-Partition (AP) method. The adaptive-partition method again considers each target separately. Those targets sufficiently well separated from all other targets are treated as independent and proposed using the IP method. When targets are not sufficiently distant, the CP method is used. To determine when targets are sufficiently separated, we use filter estimate of targets states and then threshold based on distance in sensor space between the estimated states.

3.5.4. *An Improvement*

In certain circumstances, the optimal importance density can be more efficiently approximated that the sample based approach discussed here. In particular, if target dynamics are linear/Gaussian and measurements are made on a grid, the optimal proposal involves sampling from truncated normals. In this case, a similar AP approach is used wherein partitions are first separated into groups that are uncoupled and then each group is treated by sampling from truncated normals. In the more generic case, one does not have a convenient (semi-) closed form and instead relies on the purely sample driven methods of IP and CP as described above.

4. INFORMATION-BASED SENSOR MANAGEMENT

In this section, we give an overview of information theoretic sensor scheduling. This is based on [7].

In our method of sensor management, actions are ranked based on the amount of information expected to be gained from their execution. In principle, we compute the expected gain in information between the current JMPD and the JMPD that would result after taking action $\mathbf{r}$ and making a measurement, for all $\mathbf{r}$. Then the sensor management decision is to select the best $\mathbf{r}$ using on expected information gain. In practice, one may have a continuous action space and need more sophisticated methods (e.g., vector force approaches) as the possible sensor actions can not be enumerated. In our application, we envision the sensor management problem being one of choosing where to move a sensor (although the method is general), so $\mathbf{r}$ will be treated as a position in the following.

4.1. The Rényi Divergence Between JMPDs

The calculation of information gain between two densities p_1 and p_0 is done using the Rényi information divergence, [8] also known as the α-divergence:

$$D_\alpha(p_1||p_0) = \frac{1}{\alpha - 1} \ln \int p_1^\alpha(x) p_0^{1-\alpha}(x) dx \tag{10}$$

In our application, we compute the divergence between the predicted density $p(\mathbf{X}^k, T^k|\mathbf{Z}^{k-1})$ and the updated density after a measurement $\mathbf{z}$ is made at new location $\mathbf{r}$, denoted $p(\mathbf{X}^k, T^k|\mathbf{Z}^{k-1}, \mathbf{z}, \mathbf{r})$. Therefore, the relevant divergence is

$$D_\alpha\Big(p(\cdot|\mathbf{Z}^{k-1}, \mathbf{z}, \mathbf{r})||p(\cdot|\mathbf{Z}^{k-1})\Big) = \frac{1}{\alpha - 1} \times \tag{11}$$
$$\ln \sum_{T^k} \int p^\alpha(\mathbf{X}^k, T^k|\mathbf{Z}^{k-1}, \mathbf{z}, \mathbf{r}) p^{1-\alpha}(\mathbf{X}^k, T^k|\mathbf{Z}^{k-1}) d\mathbf{X}^k \ ,$$

where the integral is interpreted as in (2).

4.2. The Expected Rényi Divergence for a Sensing Action

The value of an action must be predicted *before receiving* the measurement $\mathbf{z}$. Therefore, we calculate the expected value of the divergence for each possible action and use this to select the next action. The expectation may be written as an integral over all possible outcomes $\mathbf{z}$ when taking action $\mathbf{r}$ as

$$\mathbb{E}\Big(D_\alpha\Big(p(\cdot|\mathbf{Z}^{k-1}, \mathbf{z}, \mathbf{r})||p(\cdot|\mathbf{Z}^{k-1})\Big)\Big) \equiv \tag{12}$$
$$\int d\mathbf{z} p(\mathbf{z}|\mathbf{Z}^{k-1}, \mathbf{r}) D_\alpha\Big(p(\cdot|\mathbf{Z}^{k-1}, \mathbf{z}, \mathbf{r})||p(\cdot|\mathbf{Z}^{k-1})\Big) \ .$$

And then the method of scheduling is to choose

$$\hat{\mathbf{r}} = \arg\max_{\mathbf{r} \in \mathbb{C}} \mathbb{E}\Big(D_\alpha\Big(p(\cdot|\mathbf{Z}^{k-1}, \mathbf{z}, \mathbf{r})||p(\cdot|\mathbf{Z}^{k-1})\Big)\Big) \ , \tag{13}$$

where $\mathbb{C}$ is a set of physically feasible actions.

4.3. Multi-platform Sensor Management

Information theoretic scheduling for a collection of N sensors requires choosing the set of future locations of the N sensors

$\vec{\mathbf{r}} \equiv (\mathbf{r}_1, \cdots, \mathbf{r}_N)$ which satisfies

$$\hat{\vec{\mathbf{r}}} = \arg\max_{\vec{\mathbf{r}} \in \mathbb{C}'} \mathbb{E}\Big(D_\alpha\Big(p(\cdot|\mathbf{Z}^{k-1}, \vec{\mathbf{z}}, \vec{\mathbf{r}}) || p(\cdot|\mathbf{Z}^{k-1}) \Big) \Big) \quad . \quad (14)$$

where $\mathbb{C}'$ represents the multiplatform constraint set which includes physical constraints but also prevents collision.

The joint optimization can be rewritten as a sum of single sensor optimizations plus a correction factor as

$$\sum_{i=1}^{N} \mathbb{E}\Big(D_\alpha\big(p(\cdot|\mathbf{Z}^{k-1}) || p(\cdot|\mathbf{Z}^{k-1}, \mathbf{z}_i, \mathbf{r}_i) \big) \Big) + \mathbb{E}\big[h(\vec{\mathbf{z}}, \vec{\mathbf{r}}, \mathbf{Z}^{k-1}) \big]$$

The function h in this expression is an "information coupling" term which accounts for the fact (among other things) that the gain in information for two sensors taking the same action is not double the information gain for a single sensor taking the action.

The constraint that sensors cannot collide deals with action sets and not simply with individual actions, and so cannot be handled by simply censoring actions that violate the constraint. Therefore, define the Lagrangian

$$\begin{aligned} L(\vec{\mathbf{r}}) =& \mathbb{E}\Big(D_\alpha\Big(p(\cdot|\mathbf{Z}^{k-1}, \vec{\mathbf{z}}, \vec{\mathbf{r}}) || p(\cdot|\mathbf{Z}^{k-1}) \Big) \Big) + \lambda f(\vec{\mathbf{r}}) \\ =& \sum_{i=1}^{N} \mathbb{E}\Big(D_\alpha\big(p(\cdot|\mathbf{Z}^{k-1}) || p(\cdot|\mathbf{Z}^{k-1}, \mathbf{z}_i, \mathbf{r}_i) \big) \Big) \\ & + \mathbb{E}\big[h(\vec{\mathbf{z}}, \vec{\mathbf{r}}, \mathbf{Z}^{k-1}) \big] + \lambda f(\vec{\mathbf{r}}) \end{aligned} \quad (15)$$

where the function f is a term that penalizes action sets that move the sensors too close together. The joint constrained optimization then becomes an unconstrained optimization

$$\hat{\vec{\mathbf{r}}} = \arg\max_{\vec{\mathbf{r}}} L(\vec{\mathbf{r}}) \quad . \quad (16)$$

This optimization in principle requires a selection between N^M possible action sets, where M is the number of possible actions for each sensor. In the continuous action setting, the optimization requires choosing the best vector from $\mathbb{R}^N$. In some settings, convexity or other physical conditions exist which ameliorate the tractability situation. No such convenience exists in our setting.

To avoid this intractable optimization, we make an approximation and then exactly optimize. Note that both the Lagrangian penalty term f and the information correction term h reflect that actions sets which involve moving sensors close together are poor choices. Therefore, we simultaneously approximate both terms with a function that reduces the value of action sets involving sensors moving close together. We have chosen to use a physicomimetic force to provide this approximation, although other similar approximations are valid. Evaluating this force has a very small computational burden, and furthermore allows a decentralized optimization.

We use a generalization of the Lennard-Jones (LJ) potential that serves as a zeroth order model of the intermolecular forces of liquids. The LJ force for a pair of sensor nodes i, j separated by $r_{i,j}$ is radial with magnitude

$$F_{LJ}(r_{i,j}) = -\epsilon \left[m \frac{\gamma^m}{r_{i,j}^{m+1}} - n \frac{\gamma^n}{r_{i,j}^{n+1}} \right] \quad . \quad (17)$$

This is strongly repulsive as the radius between sensors $r_{i,j}$ gets small. The terms γ and ϵ are chosen based on sensor kinematic properties. The total force node i feels is the vector sum of the forces from all other nodes. To compute the total force, a node need only know the positions of the other nodes; in fact, since the force falls off so rapidly sensors much more distant that γ have negligible effect. Therefore a node only needs to know the positions of nearby neighbors.

Denote by $\mathbf{F}_{LJ}^{i,j}(\mathbf{r}_i)$ the vector force node i feels from node j when positioned at $\mathbf{r}_i$. Then the total force node i feels from all other nodes when positioned at $\mathbf{r}_i$ is simply $\mathbf{F}_{LJ}^{i}(\mathbf{r}_i) = \sum_{j \neq i} \mathbf{F}_{LJ}^{i,j}(\mathbf{r}_i)$. This specification of relaxation term results in the final approximate optimization

$$\begin{aligned} \hat{\vec{\mathbf{r}}} = & \arg\max_{\vec{\mathbf{r}}} \\ & \sum_{i=1}^{N} \mathbb{E}\Big(D_\alpha\big(p(\cdot|\mathbf{Z}^{k-1}) || p(\cdot|\mathbf{Z}^{k-1}, \mathbf{z}_i, \mathbf{r}_i) \big) \Big) + \lambda \mathbf{F}_{LJ}^{i}(\mathbf{r}_i) \quad . \end{aligned} \quad (18)$$

4.4. Non-myopic Sensor Management

The sensor management approach presented so far is myopic in that it only considers immediate benefit when making decisions. In certain applications, this approach is inferior to one that accounts for the long term consequences. As with multiple sensors, computational burden grows exponentially. Therefore, in this section, which is a condensed version of [9], we present a principled approximation that provides scheduling benefit at a modest computational cost.

Let the value of state s at time k be denoted $V_k(s)$. We will use $c(s, \mathbf{r})$ as shorthand for the myopic expected gain associated with an action $\mathbf{r}$ in state s, i.e.,

$$c(s, \mathbf{r}) \doteq \mathbb{E}\Big(D_\alpha\big(p(\cdot|\mathbf{Z}^k, \mathbf{r}, \mathbf{z}) || p(\cdot|\mathbf{Z}^{k-1}) \big) \Big) \quad (19)$$

where s is used as a surrogate for $p(\mathbf{X}^k|\mathbf{Z}^{k-1})$.

Bellman's equation describes the value of being in state s at time k as a sum of the immediate reward and the long term reward (neglected heretofore) as

$$V_k(s) = \max_{\mathbf{r}} \{ c(s, \mathbf{r}) + \gamma E_{s'}[V_{k+1}(s')] \} \quad (20)$$

where $E_{s'}[V_{k+1}(s')] = \int_{j \in S} p(j|s, \mathbf{r}) V_{k+1}(j)$.

The optimal non-myopic action $\hat{\mathbf{r}}$ is then given by

$$\hat{\mathbf{r}} = arg\max_{\mathbf{r}} \{ c(s, \mathbf{r}) + \gamma E_{s'}[V_{k+1}(s')] \} \quad (21)$$

As alluded to earlier, this non-myopic optimization is in general intractable. One method of approximation that has been successful in the past is to simply replace the value-to-go-term with a easily computed function, e.g., choose

$$\hat{\mathbf{r}} = arg \max_{\mathbf{r}} \{c(s, \mathbf{r}) + \gamma N(s, \mathbf{r})\} \tag{22}$$

In our application, we have used as $N(s, \mathbf{r})$ the "gain in information for waiting", which is an information based quantity that rewards actions that have less value in the future and penalizes actions more valuable in the future. In this manner, those actions that are important to take now are elevated in terms of value and more likely to be taken immediately.

This technique applies to a variety of scenarios. For example, consider the scenario where a sensor has time-varying visibility of the surveillance region due to topological features. In this scenario it is important to predict that certain areas of the region will not be visible in the future and therefore they should be interrogated preferentially at present.

5. A SIMULATION

The following simulation compares myopic and approximate non-myopic scheduling algorithms with a random scheduling algorithm on a model problem. The model problem (detailed further in [9]) has a single moving target in a visibility obscured area that is to be detected and tracked by choosing where to point a single agile sensor.

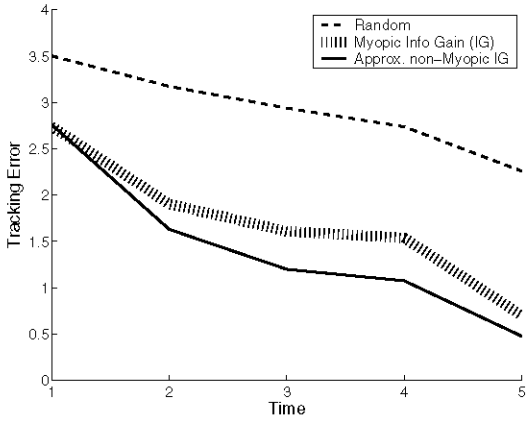

Fig. 1. A comparison of the information maximization scheulding methods and a random scheduling policy.

6. CONCLUSION

This paper has described a method of sensor management based on maximizing information flow. The key enabling element to this method is an on-line recursive estimate of the joint multitarget probability density (JMPD), which is accomplished using particle filtering methods. Particular attention has been given to designing an efficient sampling strategy to allow tractable estimation of the JMPD. The importance density so designed simultaneously exploits the fact that the JMPD often factors into densities over smaller numbers of targets, and also uses biassed sampling strategies to ensure particles are used in an efficient manner.

7. ACKNOWLEDGEMENTS

The authors gratefully acknowledge the contributions to this work of Keith Kastella and Mark Morelande.

8. REFERENCES

[1] C. Kreucher, A. Hero, and K. Kastella, "A comparison of task driven and information driven sensor management for target tracking," in *The 44th IEEE Conference on Decision and Control*, December 12-15 2005, pp. 4004–4009.

[2] L. D. Stone, T. L. Corwin, and C. A. Barlow, *Bayesian Multiple Target Tracking*, Artech House, 1999.

[3] I. R. Goodman, R. P. S. Mahler, and H. T. Nguyen, *Mathematics of Data Fusion*, Kluwer Academic Publishers, 1997.

[4] C. Kreucher, K. Kastella, and A. Hero, "Multitarget tracking using the joint multitarget probability density," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 41, no. 4, pp. 1396–1414, October 2005.

[5] C. Kreucher, A. Hero, and K. Kastella, "Multiple model particle filtering for multitarget tracking," in *The Proceedings of The Twelfth Annual Conference on Adaptive Sensor Array Processing (ASAP)*, March 16 - 18 2004.

[6] M. Orton and W. Fitzgerald, "A Bayesian approach to tracking multiple targets using sensor arrays and particle filters," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 216–223, 2002.

[7] C. Kreucher, K. Kastella, and A. Hero, "Sensor management using an active sensing approach," *Signal Processing*, vol. 85, no. 3, pp. 607–624, March 2005.

[8] A. O. Hero III, B. Ma, O. Michel, and J. D. Gorman, "Alpha divergence for classification, indexing and retrieval," *Technical Report 328, Comm. and Sig. Proc. Lab., Dept. EECS, Univ. Michigan*, 2001.

[9] C. Kreucher, A. Hero, K. Kastella, and D. Chang, "Efficient methods of non-myopic sensor management for multitarget tracking," in *The Proceedings of the 43rd IEEE Conference on Decision and Control (CDC)*, December 14 - 17 2004, vol. 1, pp. 722–727.