NON-MYOPIC APPROACHES TO SCHEDULING AGILE SENSORS FOR MULTISTAGE DETECTION, TRACKING AND IDENTIFICATION

Chris Kreucher and Alfred O. Hero III

The University of Michigan
Department of Electrical Engineering and Computer Science
Ann Arbor, MI 48109-4122

ABSTRACT

This paper addresses the problem of sensor scheduling for simultaneous target detection, tracking and identification. We consider sensors with agility in waveform and pointing direction. Scheduling decisions are made using an information based approach, where the merit of competing actions is judged by the information expected to be gained when taking the action. We focus on non-myopic scheduling, where the long-term ramifications of scheduling decisions are accounted for in decision making. Since an exact non-myopic solution is computationally prohibitive, we investigate two approximate approaches: Direct approximation of Bellman's equation and reinforcement learning. We show via simulation that both techniques provide substantial gains over myopic scheduling.

1. INTRODUCTION

The term sensor scheduling refers to the problem of determining the best way to task an agile sensor to detect, track and identify targets. Sensor tasking often includes choosing pointing angle, waveform, and how to direct a platform.

In this paper, we consider a situation where an agile sensor can choose from two waveforms and also decide in which direction to point. The first waveform simulates an X-band radar, which has good detection performance but is susceptible to line of sight obstructions. The second waveform is a high frequency (HF) band radar, which has poorer detection ability but is unaffected by line of sight obstructions. Furthermore, due to sensor and target motion, target visibility changes over time.

Some researchers have used information measures as a means of sensor scheduling [1][2][3]. In the context of Bayesian estimation, a good measure of the quality of an action is the reduction in entropy expected to be induced by the measurement. Using expected information gain for sensor scheduling has the desirable property that the different goals of identification, tracking, and detection can be simultaneously optimized through a single metric. Therefore, a sensor with multiple action types, some of which contribute to identification, others to detection and others to tracking, can be tasked by evaluating a single global metric.

Scheduling strategies may be myopic or non-myopic. In the myopic case, sensing actions are taken so as to maximize immediate reward. Myopic methods have the advantage that they are more computationally tractable than non-myopic methods. However, they do not account for the long term ramifications of current actions. Non-myopic methods, on the other hand, explicitly account for the long-term effects when making current scheduling decisions. However, an exact non-myopic solution is computationally intractable in all but the simplest of problems.

In this paper, we investigate two approximate methods for tractable non-myopic scheduling. Both methods rely on information as a measure of utility. The first method, presented in Section 3, is an approximation which replaces the value-to-go term in Bellman's equation with an easily computed function of current and future information gaining ability. The second method, presented in Section 4, is a reinforcement learning strategy where a non-myopic policy is learned from example episodes. We give a simulation result showing the merit of the two methods in Section 5 and discuss the performance in Section 6.

2. INFORMATION THEORY FOR SCHEDULING

In this section, we describe how information theory is used for myopic sensor scheduling. We extend the framework to non-myopic scheduling in Sections 3 and 4.

Information theory for sensor scheduling requires a probability density which captures uncertainty in the current state estimate. In our situation, uncertainty arises in the number of targets, their kinematic states and identification. The relevant density is known as the joint multitarget probability

This work was supported by the USAF Contract No. F33615-02-C-1199, AFRL contract SPO900-96-D-0080, and 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.

density (JMPD) [1] and is defined as

$$p(\mathbf{x}_t^1, \mathbf{x}_t^2, ...\mathbf{x}_t^{T-1}, \mathbf{x}_t^T, T_t|\mathbf{Z}_t) \ , \quad (1)$$

which is the probability for T targets with states $\mathbf{x}^1, \cdots, \mathbf{x}^T$ at time t based on the observations $\mathbf{Z}_t$. Each $\mathbf{x}^i$ in the JMPD is a vector quantity and may (for example) be of the form $[x, \dot{x}, y, \dot{y}, id]'$. For convenience, the density will be written compactly as $p(\mathbf{X}_t, T_t|\mathbf{Z}_t)$. The JMPD is estimated using a sophisticated multitarget particle filtering scheme [4].

Uncertainty in the JMPD drives sensor scheduling decisions. To schedule a sensor, we enumerate all possible sensing actions (e.g. sensor modes and sensor pointing directions) and calculate the *expected* gain in information associated with each possible action. The calculation of information gain between two densities f_1 and f_0 is done using the Rényi information divergence [5],

$$D_\alpha(f_1||f_0) = \frac{1}{\alpha - 1} ln \int f_1^\alpha(x) f_0^{1-\alpha}(x) dx \ . \quad (2)$$

In this work, we use $\alpha = 0.5$. We are interested in computing the divergence between the predicted density and the updated density [6]. A particle filter approximation of the densities with particles $\mathbf{X}_p$ and weights w_p, $p = 1 \cdots n$, simplifies (2) to

$$D_\alpha\left(p(\cdot|\mathbf{Z}_{t+1})||p(\cdot|\mathbf{Z}_t)\right) \propto ln \frac{-1}{p(\mathbf{z})^\alpha} \sum_{p=1}^{n} w_p p(\mathbf{z}|\mathbf{X_p})^\alpha \quad (3)$$

We'd like to choose the action that maximizes divergence between the current density and the density after a new measurement. However, we do not know the outcome until after the action is taken. Therefore, we instead use the conditional mean estimate of divergence, i.e. choose to take the action m that maximizes the expected gain in information:

$$\int_z dz p(z|\mathbf{Z}_t, m) D_\alpha\left(p(\cdot|\mathbf{Z}_{t+1})||p(\cdot|\mathbf{Z}_t)\right) \ . \quad (4)$$

3. APPROXIMATION TO BELLMAN'S EQUATION

The optimal non-myopic method for scheduling a sensor is given by Bellman's equation, which recursively gives the value of being in state s at time k (the value function),

$$V_k(s) = \max_m \{E[c(s,m)] + \gamma E_{s'}[V_{k+1}(s')|s,m]\} \ . \quad (5)$$

The state is denoted s and the reward for taking action m in s is $c(s,m)$. γ is a weight factor used to emphasize future rewards less than current rewards, and $E_{s'}$ indicates the expectation is taken with respect to the future state s'. The state is described by the JMPD, the models of target and sensor kinematics and any ancillary information (e.g. visibility maps or terrain elevation maps). The immediate reward is given by the gain in information as measured by the Rényi Divergence. This leads to a policy that chooses

$$\hat{m} = \arg\max_m \{E[c(s,m)] + \gamma E_{s'}[V_{k+1}(s')|s,m]\} \ . \quad (6)$$

Solving (6) is intractable for all but the simplest problems. We therefore advocate a method which approximates the value-to-go term, $E_{s'}[V_{k+1}(s')]$, by a function $N(s,m)$ which captures the long term value of an action and is easily computable. Specifically, we use a policy that selects

$$\hat{m} = \arg\max_m \{E[c(s,m)] + \gamma N(s,m)\} \ . \quad (7)$$

In this work, use as $N(s,m)$ the "gain in information for waiting". This approximation to the long term value function is based on the information theoretic framework of Section 2, and provides a method using long-term effects to influence the selection of current actions.

Specifically, let $\bar{g}_m^k$ denote the expected myopic gain when taking action m at time k. Furthermore, let $p_m^k(\cdot)$ denote the distribution of myopic gains when taking action m at time k. Then we approximate the long-term value of taking action m by the gain (loss) in information received by waiting until a future time step to take the action,

$$N(s,m) \approx \sum_{j=1}^{J} \gamma^j sgn\left(\bar{g}_m^k - \bar{g}_m^{k+j}\right) D_\alpha\left(p_m^k(\cdot)||p_m^{k+j}(\cdot)\right) \quad (8)$$

where J is the horizon length.

Each term in the summand has two components. First, $sgn\left(\bar{g}_m^k - \bar{g}_m^{k+m}\right)$ signifies if the expected information gain in the future is more or less than the present. A negative value implies that the future is better and that the action ought to be discouraged at present. A positive value implies that the future is worse and that the action ought to be encouraged. The second term, $D_\alpha\left(p_m^k(\cdot)||p_m^{k+j}(\cdot)\right)$, measures the Rényi divergence between the density on current myopic gains and future myopic gains. A small number implies the two are very similar and therefore the non-myopic term will have little impact on the decision making.

In the situation where sensor to target visibility changes with time, this approximation encourages the sensor to preferentially look at areas of the surveillance region that will become obscured in the near future.

To completely specify the technique advocated here, we introduce a weighting w which gives relative precedence to the non-myopic and myopic terms in the approximation to Bellman's equation, i.e. we schedule a sensor by choosing

$$\hat{m} = c(s,m) + w \sum_{j=1}^{J} \gamma^j sgn\left(\bar{g}_m^k - \bar{g}_m^{k+j}\right) D_\alpha\left(p_m^k(\cdot)||p_m^{k+j}(\cdot)\right) \quad (9)$$

As $w \rightarrow 0$ the technique schedules myopically, and as $w \rightarrow \infty$ the technique considers only the future. An appropriate choice for w balances the present and the future.

4. REINFORCEMENT LEARNING

An alternate approach to solving (5) is a technique known as reinforcement learning (RL). Rather than attempting to explicitly evaluate the value function (either by direct computation or approximation as in Section 3), the RL approach learns the value function through a large set training episodes. Specifically, we define the Q-function as

$$Q(s,m) = E[c(s,m)] + E[\gamma V(s')|s,m] \ . \qquad (10)$$

Given the Q-function, optimal actions can be computed according to $\hat{m} = \arg\max_m Q(s,m)$.

The Q-function is estimated from multiple example trajectories of the process. Assume first that both the number of states and actions are finite. Then there is a lookup table representation of $Q(s,m)$. In this case, given an arbitrary initial value of $Q(s,m)$, the one-step Q-learning algorithm is given by the repeated application of the update equation

$$\begin{aligned} Q(s,m) \leftarrow & (1-\beta)Q(s,m) \\ & + \beta\left(r + \gamma \max_{m'} Q(s',m')\right) \end{aligned} \qquad (11)$$

where the 4-tuples $\{s, m, s', r\}$ are incurred during training, and β is the learning rate. Under certain exploration conditions, the algorithm converges to the optimal Q function [7].

Unfortunately, in most realistic problems it is infeasible to represent the Q-function in a lookup table, because the number of states is too large or because the state space is continuous. Therefore, function approximation is required. The simplest class of Q-function approximators are linear combinations of basis functions (also called features), i.e.

$$Q(s,m) = \theta_m^T \phi(s) \ , \qquad (12)$$

where $\phi(s)$ is a feature vector associated with s and $\theta_m, m = 1 \cdots M$ is to be estimated, i.e., the training data is used to learn the best approximation to $Q(s,m)$ among all linear combinations of the features. Gradient descent is used with the training data to update the estimate of θ_m, i.e.

$$\theta_m \leftarrow \theta_m + \beta\left(r + \beta \max_{m'} Q(s',m') - Q(s,m)\right)\nabla Q(s,m)$$

where the gradient is given by $\nabla Q(s,m) = \phi(s)$. Once learning of θ is completed, optimal actions can be computed according to $\hat{m} = \arg\max_m \theta_m^T \phi(s)$.

The features that constitute ϕ are selected in accordance with our information theoretic paradigm. The JMPD is used along with kinematic and measurement models to compute the expected gain in information for each possible sensing action. This fixed dimension feature vector is then used to characterize the state in the Q-learning algorithm.

5. SIMULATION RESULT

We consider a problem where we wish to detect and track a ground target by choosing the best waveform and pointing direction. The sensor has two possible waveforms. The first corresponds to an X-band radar and provides good sensing capabilities but is affected by line of sight obscuration. The second is an HF radar and provides poorer capabilities but is unaffected by visibility constraints. At each time step, the sensor can measure exactly one cell with one waveform to determine the presence or absence of a target. Each waveform choice is characterized by a P_d and P_f which give the correct detection probability and the false alarm probability, respectively. The simulation is illustrated in Figure 1.

Fig. 1. An illustration of the model problem. A moving sensor is to best choose the waveform and pointing direction to detect and track a ground target. Visibility between the sensor and ground affects one waveform but not the other.

The sensor is moving, so line of sight between the sensor and detection cells is time varying. Coupled with terrain elevation, this makes the ability of the sensor to see a cell change with time. Therefore, scheduling will benefit from non-myopic decisions, specifically those that encourage the sensor to measure cells that are about to become obscured. A myopic strategy makes tasking decisions based only on immediate reward. In particular, at the first time step all visible cells are equally desirable. A non-myopic strategy favors interrogating cells that will soon become obscured to the sensor. In the present situation where sensor motion may make portions of the region invisible for extended portions of time, non-myopic scheduling becomes critical.

Figures 2 and 3 compare the performance of the value-to-go approximation and the RL method with myopic and random strategies. For each simulation, the target is placed randomly in the surveillance area and the filter is initialized with complete uncertainty in target position. The strategies

select waveform and pointing direction for a 25s vignette to best detect and track the target. Performance is measured in terms of mean tracking error over the episodes.

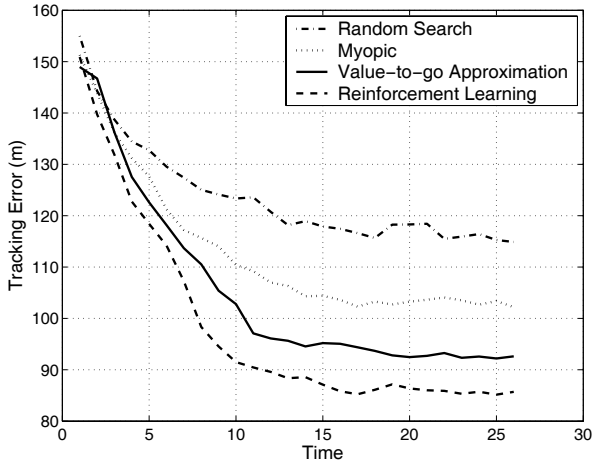

Fig. 2. Performance of the approximate policies in terms of tracking error. Shown for the purposes of comparison is the error for a purely random sensor allocation strategy and the information based myopic strategy.

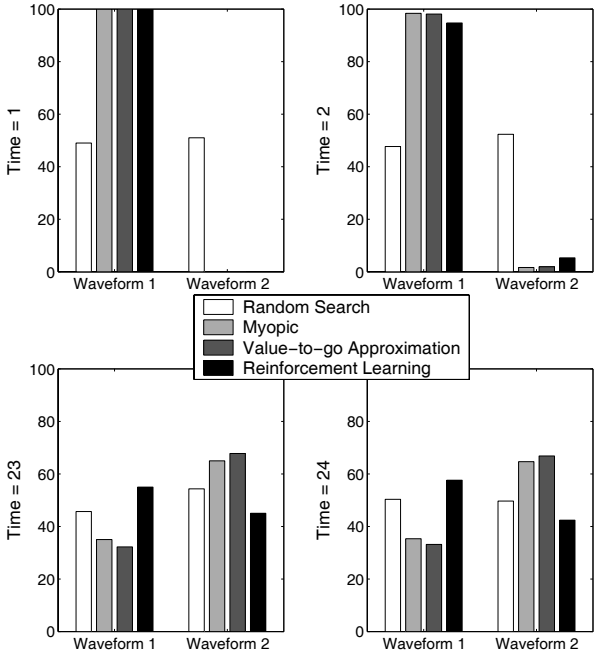

Fig. 3. Bar charts showing waveform choices made by the strategies at four different time steps. The random policy uses each waveform equally, while the other policies use the X-band waveform initially and use the HF waveform later when visibility becomes a factor.

6. DISCUSSION AND CONCLUSION

The simulation result shows the approximate non-myopic strategies outperform myopic and random policies. In particular, targets are localized significantly faster.

There are benefits and drawbacks to each strategy. RL requires feature extraction which may not be obvious in all settings. Afterwards, Q-learning is a turn-key solution which often provides good results. Once the Q-function is learned, scheduling is done nearly as fast as myopic scheduling. However, learning the Q-function is a time consuming process. Furthermore, policies learned off-line may not perform well in scenarios dissimilar to the training scenario.

The value-go-approximation requires design of approximating function. The function given here is not applicable to all scenarios but is valuable in a number of common situations. This method has complexity linear in horizon length, and so it is nearly as tractable as the myopic scheme.

7. REFERENCES

[1] K. Kastella, "Discrimination gain for sensor management in multitarget detection and tracking," *IEEE-SMC and IMACS Multiconference*, vol. 1, pp. 167–172, 1996.

[2] R. Mahler, "Global optimal sensor allocation," *Proceedings of the Ninth National Symposium on Sensor Fusion*, vol. 1, pp. 167–172, 1996.

[3] K. J. Hintz and E. S. McVey, "Multi-process constrained estimation," *IEEE Transactions on Man, Systems, and Cybernetics*, vol. 21, no. 1, pp. 434–442, January/February 1991.

[4] C. M. Kreucher, K. Kastella, and A. O. H. III, "Tracking multiple targets using a particle filter representation of the joint mulitarget proabbility density," *Proceedings of SPIE Conference on Signal and Data Processing of Small Targets*, 2003.

[5] 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, 2002.

[6] C. M. Kreucher, K. Kastella, and A. O. H. III, "Information based sensor management for multitarget tracking," *Proceedings of SPIE Conference on Signal and Data Processing of Small Targets*, 2003.

[7] D. P. Bertsekas and D. Castanon, "Rollout algorithms for stochastic scheduling problems," *Journal of Heuristics*, vol. 5, no. 1, pp. 89–108, 1999.