Proceeding of the 2004 American Control Conference
Boston, Massachusetts June 30 - July 2, 2004

Closing the Loop in Sensor Fusion Systems: Stochastic Dynamic Programming Approaches

Michael K. Schneider, *Member, IEEE*, Gregory L. Mealy, *Member, IEEE*, and Felipe M. Pait, *Senior Member, IEEE*

***Abstract*—This paper provides an overview of the problem of managing sensor resources in a closed-loop sensor fusion system. We formulate the problem in a stochastic dynamic programming framework. In so doing, we expose structure in the problem resulting from target dynamics being independent and discuss how this can be exploited in solution strategies. We illustrate situations in which we believe such sensor management techniques are especially beneficial with two examples. One example is the management of a single sensor, and the other is the management of multiple sensors. The focus of both examples is on air-to-ground tracking.**

I. Introduction

In this paper, we address control aspects of sensor fusion. For the sensor fusion problem of interest here, one would like to infer the state of multiple targets from measurements made by one or more sensors over time. Targets are typically located on the ground and can include vehicles, buildings, and other man-made objects. States of interest could include position, velocity, mode (e.g. on- or off-road), vehicle type, etc. Estimates of the states are inferred by fusing information from multiple sensors over time. The fusion engine responsible for piecing together information from different types of sensors will typically create hypotheses by associating new observations with previously detected targets. Alternative hypotheses are formulated to deal with ambiguities caused by incomplete or even contradictory information. New hypotheses are created and abandoned as data is accumulated that indicates the current target states have changed or resolves ambiguities in the past states of targets. The data can be generated by many different types of sensors, including airborne surveillance radars, video sensors, etc. The sensors are managed to collect the appropriate measurements. We view sensor resource management (SRM) as the control problem of allocating available sensor resources to obtain the best awareness of the situation.

This material is based upon work supported in part by the U.S. Air Force under Contract Nos. F33615-02-C-1197, F33615-03-M-1515, and F3365-02-C-1129.

The authors are with ALPHATECH, Inc., Burlington, MA 01803 USA. (phone: 781-273-3388; fax: 781-273-9345; e-mail: {michaels, gmealy, fpait}@ALPHATECH.com).

Efficient sensor management requires consideration of the value of particular pieces of information to the fusion engine at each moment, so the plant to be controlled comprises not only the sensors and communication systems, but also the fusion engine that processes the information collected by them, as illustrated in Fig. 1. The plant's inputs are precisely the requests that the sensor management system is allowed to make, and its outputs include all the information obtained from the sensors. The state of the plant is then the total information available to the fusion engine, and in principle also to the SRM controller, at a given time. The dimension of the state is not fixed: it increases as information is collected, and new tracks are initiated. It also decreases when new information results in hypotheses being resolved, and when the hypothesis tree is pruned of alternatives that are considered less likely.

From this point of view the process model is completely deterministic, and full information about the process is available. Uncertainty enters the picture in the form of the actual measurements obtained by the sensors, which can be treated as external disturbances about which we, as designers of a sensor management and fusion system, have no control or previous knowledge. Additional disturbances include sensor actions over which the system has no control – for example, sensor systems which are allocated at a higher command level. Indeed, the current state of the fusion system represents the best possible guess about the actual ground truth – taking into account the information available and our capacity to process it. Since the estimate does not depend on probabilities of obtaining specific data

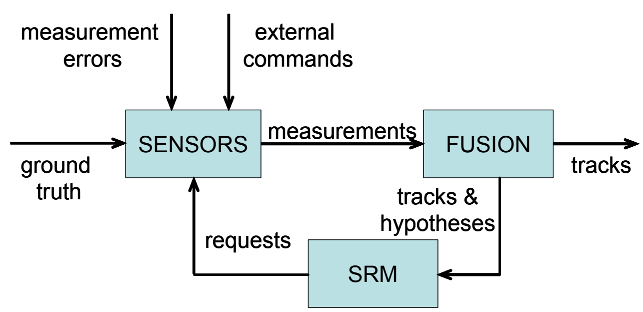

Fig. 1 Sensor Resource Management (SRM) closes the sensor/fusion control loop.

0-7803-8335-4/04/$17.00 ©2004 AACC

in the future, the system is essentially causal, a fact that simplifies conceptually the design of a sensor management algorithm. Of course the variable dimensionality of the state space precludes the use of textbook control design techniques, which are not likely to be applicable in any event.

A number of different approaches to the design of sensor managers have been proposed in the literature. They cover the different aspects of the sensor management problem including how to manage sensors to support detecting and localizing [3], [7], [8], [9]; tracking [2], [8], [10], [11], [12]; and classifying [4], [5], [6] targets. The proposed solutions include policies based on information-theoretic optimization criteria [8], [11] as well as policies for optimizing more traditional criteria (e.g., track error) generated using stochastic optimization techniques such as index rules [2], [5], [12]; Lagrangian relaxation [6]; et al. [3], [4], [7], [9], [10]. In this paper, we overview some of the technical issues in sensor management including structure in the problem that we believe can be exploited when designing solution techniques. This is discussed in a stochastic dynamic programming framework in Section II. In Section III, we illustrate situations in which we believe sophisticated sensor management strategies are especially beneficial with two examples. One example is the management of a single sensor, and the other is the management of multiple sensors. The focus of both examples is on air-to-ground tracking.

II. Approximate Stochastic Dynamic Programming Approach

We have conceived designs to the sensor management control problem in the framework of stochastic dynamic programming. A typical formulation starts with the system state at time t, $x(t)$. The state includes all target true positions and types. A control at time t, $u(t)$, specifies a measurement of the system to be taken. The measurement may be corrupted by a stochastic disturbance $v(t)$ and may be delayed so that it is not realized until a later time. The measurement process is given by the function h, so that

$$y(t_y) = h(x(t), u(t), v(t)) \tag{1}$$

is the measurement realized at time $t_y > t$. The information about the system at time t is summarized in the information state $I(t)$, consisting of all past measurements and controls

$$I(t) = \{y(t_y) : t_y \le t\} \cup \{u(t_u) : t_u < t\}. \tag{2}$$

The delay in realizing the measurement, Δ_y, taken at time t, is a function of the information state, control, and stochastic disturbance at time t so that

$$t_y = t + \Delta_y(I(t), u(t), v(t)). \tag{3}$$

Control decisions occur at discrete instants in time, $t_{u,0}$, $t_{u,1}$, $t_{u,2}$, ... Following time $t_{u,i}$, the next control is executed after the delay of Δ_u, which is a function of the information state, control, and stochastic disturbance at time $t_{u,i}$. Thus,

$$t_{u,i+1} = t_{u,i} + \Delta_u(I(t_{u,i}), u(t_{u,i}), v(t_{u,i})). \tag{4}$$

The control is chosen from a constraint set $U(I(t))$ according to a control law, μ, which is a function of the information state and time. Thus,

$$u(t_{u,i}) = \mu(I(t_{u,i}), t_{u,i}). \tag{5}$$

The sensor management policy is the collection of these control laws

$$\pi = \{\mu(I(t), t)\}. \tag{6}$$

Rewards are achieved upon executing the policy by attaining particular information states. The reward for attaining information state $I(t)$ is given by $R(I(t))$. These rewards are discounted by the factor $e^{-\gamma t}$ and integrated across time to yield an expected reward for executing policy π from the information state $I(0)$ of

$$J_\pi(I(0)) = E\int_0^\infty e^{-\gamma\tau} R(I(\tau))d\tau. \tag{7}$$

The optimal sensor management policy π^* is the one that maximizes (7) over all policies π. The optimal policy can be characterized in terms of Bellman's equation [1]. In this context, the equation states that the expected reward for the optimal policy satisfies

$$J^*(I(t)) = \max_{u(t)\in U(I(t))} \mathrm{E}\left[\begin{array}{l} \int_t^{t+\Delta_u(I(t),u(t),v(t))} e^{-\gamma\tau} R(I(\tau))d\tau + \\ J^*\left(I\left(t+\Delta_u(I(t),u(t),v(t))\right)\right) \end{array}\right]. \tag{8}$$

The first term on the right-hand side is the reward accrued until the next decision time after t. The second term is the expected reward after that time accrued from the resulting information state. The policy

$$\pi^* = \left\{\mu^*(I(t), t)\right\} \tag{9}$$

is optimal provided that the argument of the maximum in (8) is given by $\mu^*(I(t),t)$ for all $I(t)$ and t (the assumption here is that the set of candidate controls is compact, if not finite, so that the maximum is well-defined). Several computational techniques, including both policy and value iteration, exploit the characterization in (8) to compute policies. The difficulties in exploiting this characterization are tied to the size of the state space, the set of candidate controls, and the set of stochastic disturbances. In particular, Bellman's equation characterizes J^* for all possible information states $I(t)$ by evaluating the right-hand side of (8) for all possible controls $u(t)$, taking an expectation over all disturbances. This can be difficult to apply when the size of the sets involved is large.

However, there is special structure that can be exploited. Consider the following special case in which the system state is the aggregate state of n targets

$$x(t) = \{x_1(t), \ldots, x_n(t)\} \quad (10)$$

whose individual states $x_i(t)$ are independent and evolving in time as Markov processes. This would be the case, for example, when tracking independent, isolated targets. Moreover, suppose the measurements of the system state are conditionally independent given target state and sensor controls so that one can write

$$y(t) = \{y_i(t) : i = 1 \ldots n\} \quad (11)$$

where an individual measurement can be written

$$y_i(t) = h_i(x_i(t), u(t), v_i(t)) \quad (12)$$

for independent stochastic disturbances $v_i(t)$. Independence introduces considerable structure; however, the problem is still complex since the information states of the system do not have similar independence properties. For example, one can consider partitioning the information state as

$$I(t) = I_1(t) \cup I_2(t) \cup \ldots \cup I_n(t) \cup I_u(t) \quad (13)$$

where

$$I_j(t) = \{y_j(t_y) : t_y \le t\} \quad (14)$$

and

$$I_u(t) = \{u(t_u) : t_u < t\}. \quad (15)$$

However, the future information states $I_j(\tau)$ for $\tau > t$ are neither independent nor conditionally independent given the current control $u(t)$ and system state $x(t)$. The reason is that the information states of targets are coupled through the control decisions. Thus, one cannot rely on methods for computing sensor management policies that require the independence of the targets' information states.

One approach we have used to develop sensor management policies that exploit the special structure is the application of index rules [1], [13]. Index rules are optimal for the following type of sensor management problem. There are n targets, whose states are independent. A measurement can be made of only one target at a time, and the measurement is of fixed duration, i.e. Δ_y and Δ_u are constants and $\Delta_y < \Delta_u$. The state of the target can only change at instants when a measurement is made of it (e.g. the target state may not be changing, but the information state of the target may be as more measurements are acquired). In addition, the mission must be formulated such that the reward $R(I(t))$ accrued in a particular information state at time t depends only on the information state $I_j(t)$ of the target j being measured at that time. In this case, the optimal policy determining the next target at which to look from information state $I(t)$ is given by an index rule, which has the form

$$\mu(I(t)) = \arg\max_{j \in \{1, \ldots, n\}} m_j(I_j(t)) \quad (16)$$

where $m_j(I_j(t))$ is the index of the target. The index for target j can be represented in terms of a single target problem. We have been able to develop solutions to these single target problems and apply the resulting index rule policy. Although the assumptions required for the index rule to be optimal are often violated in sensor management problems (e.g. one may be able to measure the state of more than one target at a time), we have found that index rules may still be optimal or, at least, applicable as part of heuristics [5], [14].

Another approach we have used to develop sensor management policies is to use limited lookahead algorithms [1]. A limited lookahead policy is one for which the control action is chosen as the solution to

$$\max_{u(t) \in U(I(t))} \mathrm{E}\left[\begin{array}{l} \int_t^{t+\Delta_u(I(t),u(t),v(t))} e^{-\gamma\tau} R(I(\tau)) d\tau + \\ \tilde{J}_1\left(I\left(t + \Delta_u(I(t), u(t), v(t)) \right) \right) \end{array} \right]. \quad (17)$$

where

$$\tilde{J}_k(I(t)) = \max_{u(t) \in U(I(t))} \mathrm{E}\left[\begin{array}{l} \int_t^{t+\Delta_u(I(t),u(t),v(t))} e^{-\gamma\tau} R(I(\tau)) d\tau + \\ \tilde{J}_{k+1}\left(I\left(t + \Delta_u(I(t), u(t), v(t)) \right) \right) \end{array} \right]. \quad (18)$$

for $k=1,\ldots,N-1$ where N is the number of steps of lookahead and the terminal reward $\tilde{J}_N$ is chosen to approximate the expected reward. The algorithm for computing the limited lookahead policy is effectively enumerating possible controls and outcomes over N steps, calculating a reward for the resultant state based on an approximation, and selecting the control that yields the best outcome. Structure in the problem can be exploited in the construction of $\tilde{J}_N$. As noted previously, the problem has special structure in that individual target state evolutions are often independent. One approach to exploiting this is to use an approximate terminal reward that is separable so that

$$\tilde{J}_N = \sum_{j=1}^{n} \tilde{J}_{N,j}(I_j(t) \cup I_u(t)) \quad (19)$$

for per-target rewards $\tilde{J}_{N,j}$. These can be constructed a number of different ways. One technique we have used is to calculate the expected rewards associated with a single-target form of the problem, motivated by the index definition in [13]. Essentially, we use a function of the index m_j as the approximation $\tilde{J}_{N,j}$. We have also explored other methods for constructing $\tilde{J}_N$ including rollout and heuristic methods. In each case, we have tried to exploit structure in the problem such as the existence of independent target evolutions.

III. Application Examples

What follows are two examples of how we have been applying these techniques to sensor management problems.

The examples outline how we have applied the stochastic control techniques described above to the development of sensor managers and illustrate areas where we have found distinct advantages to using these techniques. In order to illustrate the breadth of applicability, the examples are drawn from two different types of problems. The first is the control of a single sensor; the second is the control of multiple, distributed sensors.

A. Control of a Single Sensor

In this first example, consider managing a single sensor air-to-ground radar tracking system. The sensor, tracker, and sensor manager are all colocated on the sensing platform. As a result, the latencies in transmitting information between components are minimal; so, the sensor manager is generating sensor controls on a fast time scale. In this context, two scenarios in which a stochastic control approach to sensor management has advantages are when there are differentiated targets and when the sensor mode must be matched to target state.

An example of the second scenario occurs when using an airborne radar to track ground targets. In the radar's standard ground moving target indicator (GMTI) mode, only targets moving against the background can be observed. However, the radar may have another mode such as a fixed-target indicator (FTI) mode, with which only stopped targets may be observed. In order to track the targets, the radar must be managed to periodically revisit targets in the appropriate mode to update the estimate of their position. Too long a period without observing the target will lead to the tracking system dropping the track. Longer track lifetimes are desirable. The sensor management problem is thus one of selecting the sequence of targets at which to look with the radar as well as the mode to use. One source of complexity in the problem is that targets may not be detected even if the appropriate mode is used. Thus, the sensor management policy must appropriately hedge to select the best mode based on past detections. Another potential source of complexity in the problem is that the measurements are taken over different durations Δ_y in the different modes. Thus, the policy must appropriately hedge in time so that longer duration modes are not chosen at poor instants in time. To address these two issues, we have developed a limited lookahead policy, of the form described by (17) and (18). The policy allows one to account for past detections as well as for predictions of future rewards that depend on the different measurement durations in each sensor mode. Initial results of performance are illustrated in Fig. 2. Here, a simple simulation is used to compute the average track lifetime for two different sensor policies. One is the limited lookahead policy; the other is a policy that only uses the GMTI mode. The simulation includes synthetic target motion, a simple tracker, and a simple sensor model. For this sensor model,

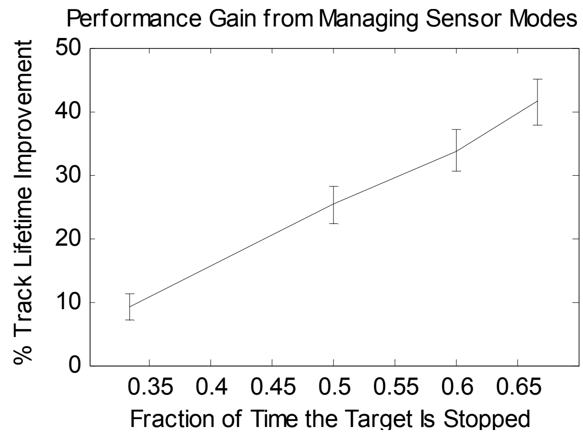

Fig. 2. The curve plots the fractional increase in track lifetime for using a limited lookahead sensor policy that controls sensor mode over a simple single mode policy. In this example, a different sensor mode must be used to observe the target when it is stopped than when it is moving. However, the sensor will only detect a target in the proper mode with some probability less than 1.

the measurement durations are the same for the two different modes. The results indicate that constructing a sensor policy that takes advantage of the FTI sensor mode has the potential to provide significant improvements in track lifetime. More realistic simulations would be required to determine the precise benefit.

The other example of a scenario for which we have noted benefits of sensor management is one in which there are differentiated targets. Specifically, a subset of the tracked targets is designated by a user to be higher priority than the others. The high-priority targets could have different tracking requirements than the low-priority targets. For example, they may have more stringent track accuracy requirements. The specific context considered here is air-to-ground tracking with a GMTI radar. Thus, there is no mode selection problem for the sensor manager, as in the previously discussed scenario. However, the problem of selecting the sequence of targets at which to look is more complex. The sensor management policy must account for the different numbers of high-and low-priority targets, the different tracking requirements, and the current state of tracks to generate a control sequence that generates measurements of targets to meet the tracking requirements. Some initial, simple simulations indicate that significant benefits can be realized from a good sensor management policy. In particular, we simulated a scenario with a high-priority target and several low-priority targets. Two limited lookahead sensor management policies were evaluated. One used one step of lookahead (*N=1*), and the other used two steps (*N=2*). Both policies performed equally well on the high-priority target. However, the two step lookahead policy achieved track accuracy requirements on the low-priority targets 86% more of the time than the one step lookahead policy. This suggests that significant benefits can be realized by appropriately managing the sensor to track differentiated targets. We are currently planning to evaluate the benefits in this type of scenario with a more realistic simulation.

B. Control of Multiple, Distributed Sensors

The second example differs from the previous one in two key respects. The first is a decomposition of the sensor resource management function into two parts: an information valuation step followed by a sensor allocation calculation (i.e., constructing a sensor scheduling plan that maximizes the value of collected information subject to constraints on sensor availability and routing). The second is the introduction of multiple sensors into the problem. In this example, we focus on the information valuation aspect for multiple sensors of differing capability.

As described above, the fusion state is determined by both the stochastic evolution of the real system and the stochastic results of sensor measurements of that system. Different sensor tasking choices will thus result in different evolutions of the system's state. The decision becomes one of determining the optimal valuation of sensor resources with respect to their impact on the fusion process. While there are multiple reasons for requesting particular sensor tasks, the approach described herein addresses an important subset – requesting sensor tasks that will either improve target track estimates or remove association ambiguity in the current or near-future fusion state. To emphasize this aspect of the approach, the algorithm has been termed FIND (Fusion Information Needs Determination).

The goal of the FIND algorithm is to maximize the time discounted reward J in (7) for the special case where the time between control actions Δ_u is constant so that one can rewrite it for a constant α as

$$J = \sum_{t=0}^{\infty} \alpha^t R(I(t)) \,. \tag{20}$$

The reward function R has the form

$$R(I(t)) = \sum_j R_j(I_j(t) \cup I_u(t)) \tag{21}$$

where the index j in this case ranges over the hypothesis space of the fusion system. The track hypothesis space contains information about the relative certainty of different data associations that are not reflected in the single global set of track estimates normally output. The individual rewards R_j are a function of a set of goals and priorities, specifically:

1. The required kinematic accuracy (expressed as tracking uncertainty, Σ_{Goal}) for tracking confirmed targets
2. The required classification accuracy (probability of correct classification, P_{Goal}) for declaring high confidence identification of a target
3. The relative priorities for meeting the kinematic and classification accuracy goals, both singly and in combination, for each of the expected target types
4. Indications of time-criticality of the information need.

Given this information, we can specify the reward for a given hypothesis H_j. The reward takes on differing values depending upon which combination of the goals is satisfied. For hypothesis H_j, with associated kinematic uncertainty $\sigma_j^2(t)$ (the maximum eigenvalue of the position error covariance) and classification probabilities $p_j(t)$ (defined as the vector of probabilities that the target is of a given type), the individual reward at time t is given by:

$$R_j(I_j(t) \cup I_u(t)) = \begin{cases} 0 & \sigma_j^2(t) > \Sigma_{Goal} \text{ and } \max(p_j(t)) < P_{Goal} \\ R_\Sigma & \sigma_j^2(t) \le \Sigma_{Goal} \text{ and } \max(p_j(t)) < P_{Goal} \\ R_Y & \sigma_j^2(t) > \Sigma_{Goal} \text{ and } \max(p_j(t)) \ge P_{Goal} \\ R_{Y\Sigma} & \sigma_j^2(t) \le \Sigma_{Goal} \text{ and } \max(p_j(t)) \ge P_{Goal} \end{cases}$$

Different candidate sensor tasks are valued using a 1-step limited lookahead approach given by (17) and (18). A heuristic approximation of the terminal award is given by the separable function

$$\tilde{J}_1 = \sum_j (p_j^T \rho\, p_j + R_\Sigma)\left(\frac{1}{\pi}\arctan\frac{(\Sigma_{Goal} - \sigma_j^2)}{\Gamma} + 0.5\right) \tag{22}$$

where the summation is over the different hypotheses within the track hypothesis space. The FIND values are computed as the increment in the expected reward of the one-step lookahead for a set of candidate sensor tasks

$$\underbrace{V(I(t), u(t), \Delta_u)}_{\text{FIND value}} = \underbrace{\mathrm{E}\left[\tilde{J}(I(t) \cup y(t+\Delta_u) \cup u(t))\right] - \tilde{J}(I(t))}_{\text{Incremental reward}} \tag{23}$$

Since FIND does not have information as to which specific sensors are available, the FIND value is computed for a set of candidate sensor controls u parameterized by hypothesis as well as a range of kinematic measurement accuracies and classification abilities.

The FIND valuation is used to determine the benefit derived from tasking a sensor to provide information on a specified hypothesis. In practice, these valuations are rank-ordered and filtered such that only a subset of the possible hypotheses is considered in the sensor allocation calculation. This portion of the solution balances the set of valuations (which vary with sensor performance) against competing requirements (e.g., requests produced at a higher command level) to produce a multiple sensor tasking plan.

To illustrate the performance of the FIND algorithm and demonstrate its utility for identifying (and quantifying) the benefits of candidate sensor taskings, consider the simple scenario. It begins with a single, stationary, high priority target. Initial information about the target consists of good classification, but poor kinematic information. A short time later, two distinct tracks are reported by an MTI system.

While these reports provide good kinematic information, target classification knowledge is poor. The problem becomes one of identifying which, if either, of the moving targets is the original high priority one.

Three approaches for generating sensor task valuations were examined:

1. **Raster** which simply tasks the sensor(s) to address each hypothesis in turn
2. **Myopic** which implements a 1-step lookahead, greedy approach. Defaults to Raster if no sensor task is expected to achieve a goal
3. **FIND** which implements a 1-step lookahead and uses the heuristic terminal reward in (22) to approximate the long-time reward.

Two sensors are available for tasking. Nominally, the first provides accurate kinematic information but no classification data, while the second provides classification information but has a poorer (i.e., larger) kinematic uncertainty. Each is assumed to report the results of the tasked observation. The FIND problem is to produce a sensor tasking (or set of sensor tasks) that resolves the inherent ambiguities in the hypothesis space while minimizing the number of such tasks. This is equivalent to producing a set of recommended sensor taskings that results in the best (minimum number of observations) solution to achieving the target tracking and classification goals.

Fig. 3 illustrates how the above algorithms perform for one set of evaluation conditions. The curves are the probability that the tracking and classification goals are exceeded as a function of the number of recommended sensor taskings. The results shown in the figure are representative; the FIND algorithm is clearly superior to the other approaches.

The valuations provided by the FIND algorithm can be viewed as providing different types of requests to improve fusion performance. The highest value requests are those which remove ambiguity in report associations to high priority targets. Requests that confirm ID and track likely high priority targets typically have medium values, while those with the lowest value are requests to ID unknown targets and are usually ignored unless no higher value tasks are requested for a given sensor resource.

IV. Conclusion

The examples in the previous section highlight issues in sensor management and indicate how one could exploit structure resulting from independent target motions to develop a sensor management policy. The results indicate that such policies will appropriately allocate sensor resources to improve the resolution of hypotheses in multi-target tracking systems and, specifically, to improve the surveillance of high-priority targets. Further experimentation is required to determine the precise degree to which benefits can be realized in practice. Planned development of high fidelity simulations will allow us to perform the necessary experiments. We expect results will indeed confirm that significant benefits can be realized.

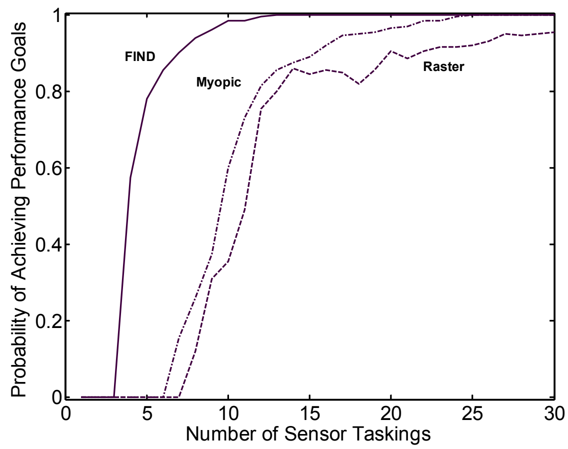

Fig. 3. FIND significantly reduces the number of sensor taskings required to achieve performance goals.

References

[1] D. P. Bertsekas. *Dynamic Programming and Optimal Control*. Athena Scientific, Belmont, MA, 2001.

[2] V. Krishnamurthy and R. J. Evans. Hidden Markov model multiarm bandits: A methodology for beam scheduling in multitarget tracking. *IEEE Transactions on Signal Processing*, 49(12):2893 - 2908, 2001.

[3] S. Singh and V. Krishnamurthy. The optimal search for a Markovian target when the search path is constrained: The infinite-horizon case. *IEEE Transactions on Automatic Control*, 48:493-497, 2003.

[4] V. Krishnamurthy. Algorithms for optimal scheduling and management of hidden Markov model sensors. *IEEE Transactions on Signal Processing*, pages 1382 - 1397, 2002.

[5] D. A. Castanon. Optimal search strategies in dynamic hypothesis testing. *IEEE Transactions on Systems, Man, and Cybernetics*, 25(7):1130-1138, 1995.

[6] D.A. Castanon. Approximate dynamic programming for sensor management. In *Proceedings of the 36th IEEE Conference on Decision and Control*, 1997.

[7] B. LaScala, B. Moran, and R. Evans. Optimal adaptive waveform selection for target detection. In *Proceedings of the International Radar Conference*, 2003.

[8] E. Ertin, J. W. Fisher, and L. C. Potter. Maximum mutual information principal for dynamic sensor query problems. In *Proceedings Information Processing in Sensor Networks*, pages 405-416, 2003.

[9] D. Sinno, D. Cochran, and D. Morrell. Multi-mode detection with markov target motion. In *Proceedings of the 3rd International Conference on Information Fusion*, pages 25-31, 2000.

[10] D. Sinno and D. Cochran. Dynamic estimation with selectable linear measurements. In *Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 2193-2196, 1998.

[11] F. Zhao, J. Shin, and J. Reich. Information-driven dynamic sensor collaboration. *IEEE Signal Processing Magazine*, 19:61-72, 2002.

[12] R.B. Washburn, M.K. Schneider, and J.J. Fox. Stochastic dynamic programming based approaches to sensor resource management. In *Proceedings of 5th International Conference on Information Fusion*, pages 608- 615, 2002.

[13] J. C. Gittins. Bandit processes and dynamic allocation indices. *Journal of the Royal Statistical Society: Series B (Methodological)*, 41(2):148-177, 1979.

[14] P. Whittle. Restless bandits: Activity allocation in a changing world. *Journal of Applied Probability*, 25A:287--298, 1988.