Dynamic Scheduling of Multiple Hidden Markov Model-Based Sensors

WOOSUN AN
SATNAM SINGH
KRISHNA R. PATTIPATI
DAVID L. KLEINMAN
SWAPNA S. GOKHALE

In this paper, a hidden Markov model (HMM)-based dynamic sensor scheduling problem is formulated, and solved using information gain and rollout concepts to overcome the computational intractability of the dynamic programming recursion. The problem involves dynamically sequencing a set of sensors to monitor multiples tasks, which are modeled as multiple HMMs with multiple emission matrices corresponding to each of the sensors. The dynamic sequencing problem is to minimize the sum of sensor usage costs and the task state estimation error costs. The rollout information gain algorithm proposed herein employs the information gain heuristic as the base algorithm to solve the dynamic sensor sequencing problem. The information gain heuristic selects the best sensor assignment at each time epoch that maximizes the sum of information gains per unit sensor usage cost, subject to the assignment constraints that at most one sensor can be assigned to a HMM and that at most one HMM can be assigned to a sensor. The rollout strategy involves combining the information gain heuristic with the Jonker-Volgenant-Castañòn (JVC) assignment algorithm and a modified Murty's algorithm to compute the κ-best assignments at each decision epoch of rollout. The capabilities of the rollout information gain algorithm are illustrated using a hypothetical scenario to monitor intelligence, surveillance, and reconnaissance (ISR) activities in multiple fishing villages and refugee camps for the presence of weapons and terrorists or refugees.

Manuscript received September 25, 2007; revised June 26, 2008; released for publication July 16, 2008.

Refereeing of this contribution was handled by Dr. Shozo Mori.

Authors' address: Electrical and Computer Engineering Department, University of Connecticut, Storrs, CT 06269-2157, E-mail: (woa05001 @engr.uconn.edu, satnam@engr.uconn.edu,krishna@engr.uconn.edu, dlkleinm@nps.edu, ssa@engr.uconn.edu).

A version of this paper was published in *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, Montreal, Quebec, Canada, 2007. The study reported in this paper was supported, in part, by the Office of Naval Research under contract no.N00014-06-1-0080. Any opinions expressed in this paper are solely those of the authors and do not represent those of the sponsor.

1557-6418/08/$17.00 © 2008 JAIF

1. INTRODUCTION

Complex applications involving threat detection, such as multi-target tracking and unmanned aerial vehicles for surveillance in remote or hostile environments, include heterogeneous sensors, which trade off performance (e.g., detection, identification, and tracking accuracies) versus the sensor usage cost (e.g., power and bandwidth consumption, distance traveled, risk of exposure, deployment requirements). The objective of dynamic sensor scheduling is to judiciously allocate sensing resources to exploit the individual sensors' capabilities, while minimizing their usage cost. As an example, consider a target identification scenario where an incoming aircraft needs to be identified as an enemy or a friendly target using active or passive sensors available at a surveillance station [16]. This scenario requires sensor scheduling because active sensors (e.g., radar) tend to reveal clues about the location of the surveillance station to a potential enemy aircraft, whereas the more stealthy passive sensors tend to be inaccurate [16]. Thus, in this case, the sensor scheduling algorithm needs to trade-off accuracy versus risk of exposure. As another example, unmanned aerial vehicles (UAVs) are preferred assets for monitoring nearly all the intelligence, surveillance, and reconnaissance (ISR) activities; however, they cannot be deployed in large numbers due to their limited availability. Thus, astute allocation of scarce resources is a major issue in sensor scheduling.

In this paper, we consider the sensor scheduling problem faced by an ISR officer of an expeditionary strike group (ESG) in coordinating the use of surveillance assets (sensors) to improve situational awareness [14]. An ESG provides a flexible Navy-Marine force, capable of tailoring itself to a wide variety of missions. An important ESG mission involves dealing with asymmetric threats, such as terrorist groups who carry out attacks while trying to avoid direct confrontation. Terrorist groups are elusive, secretive, amorphously structured and decentralized entities that often appear unconnected. This stealthy behavior makes it very difficult to predict when and where they will strike. Moreover, the increased geographical range and unpredictable nature of this behavior require effective allocation and appropriate scheduling of sensors to achieve mission objectives. Effectively performing the ISR activities is a key step to gain situational awareness, which, in turn, enables the allocation of resources for the interdiction of potential threats.

We model the asymmetric threats using hidden Markov models, because these activities are concealed and their true states can only be inferred through the observations obtained using various ISR sensors. A pattern of these observations and its dynamic evolution over time provides the information base for inferring a potential realization of a threat [25]. Performing the ISR activities requires multiple sensors to provide observations needed for accurately estimating the status of suspicious activities. The available sensors are limited in number, and possess different attributes requiring judicious sensor allocation over time. Therefore, an effective scheduling of ISR sensors over time is essential for accurate situation assessment and to the success of the overall mission.

1.1. Previous Work

The dynamic sensor scheduling problem, which has been widely studied in the area of target tracking (e.g., [8], [28]), is to solve a sequential stochastic optimization problem that seeks to minimize the expected scheduling cost under a given set of resource constraints over time [8]. For linear Gaussian state space systems, one can obtain an analytic solution for the posterior distribution of system state given sensor measurements, and a scheduling sequence via a Kalman filter [18]. Shakeri *et al.* [24] formulated the sensor scheduling problem subject to a fixed total budget and the cost of individual sensor varying inversely with its measurement variance. They obtained the optimal measurement variance distribution that minimizes the trace of a weighted sum of the estimation error covariance matrices of a discrete-time vector stochastic process, when the auto-correlation matrix of the process is given. The study showed that the problem can be transformed into an optimization problem with linear equality and inequality constraints. In the special case of a linear finite-dimensional stochastic system, they showed that the problem can be formulated as an optimal control problem, where the gradient and Hessian of the objective function with respect to the sensor accuracy parameters can be derived via a two-point boundary value problem. The resulting optimization problem was solved via a projected Newton Method [4], [24].

In [26], Singh *et al*. provided a summary of previous research on sensor scheduling for tracking targets, whose dynamics are modeled by linear Gauss-Markov processes. They formulated the sensor scheduling problem as one of minimizing the variance of the estimation error of hidden states of a continuous-time HMM with respect to a given action sequence [26]. The authors proposed a stochastic gradient algorithm to determine the optimal schedule for the HMM. Another effort, related to our work, using a discrete HMM framework was considered by Krishnamurthy in [16]. Here, the author proposed a stochastic dynamic programming (DP) framework to solve the sensor scheduling problem for a single HMM, which is intractable for all but simple HMMs with a few states (e.g., at most 15 states).

Sub-optimal approaches, based on information-theoretic criteria, have been developed to overcome the computational intractability of determining the optimal sensor schedule. For a linear Gauss-Markov system, Logothetis *et al.* [17] formulated the sensor scheduling problem as one of determining a sequence of active sensors to maximize the mutual information between the states of the unobserved dynamic process and the observation process generated by the sensors. In the context of sensor networks, Zhao *et al.* [29] and Chu *et al.* [9] formulated the target tracking problem as a sequential Bayesian estimation problem, where the participants for sensor collaboration are determined by minimizing an objective function comprised of information utility, measured in terms of entropy, Mahalanobis distance and the sensor usage cost.

Rollout algorithms were first proposed for the approximate solution of dynamic programming recursions by Bertsekas *et al.* in [5], [6]. They are a class of suboptimal solution methods inspired by the policy iteration of dynamic programming and the approximate policy iteration of neuro-dynamic programming. The rollout algorithm, combined with the information gain heuristic (IG), was first proposed in our previous research on sequential fault diagnosis [27], where the system state is fixed (i.e., static), but unknown. In [27], we showed that rollout strategy, which can be combined with the one-step or multi-step look-ahead heuristic algorithms as base algorithms, can solve test sequencing problems in real-world systems with a higher computational efficiency than the optimal strategies, while being superior to those using the base algorithms only. In order to coordinate multiple sensor resources to track and discriminate targets modeled as continuous-state HMMs, Schneider *et al.* [23] presented a rollout approach to approximate the dynamic programming recursion using a cost-to-go function based on feasible candidate scenarios. In contrast, our approach employs discrete-state HMMs to model tasks and an information gain heuristic to estimate the cost-to-go function.

In this paper, two-dimensional assignment algorithms, exemplified by the Jonker-Volgenant-Castañon (JVC) [15] and the auction [2], [3], are used to obtain an assignment for maximizing the sum of information gains per unit sensor usage cost accrued by assigning multiple sensors to multiple HMMs. The JVC and the auction are the most efficient algorithms for solving the two-dimensional (2-D) assignment problems. The JVC algorithm is a primal-dual optimization method that includes an effective initialization of dual variables, and an augmentation phase based on the Dijkstra's shortest path algorithm [11]. The auction algorithm, proposed by Bertsekas *et al.* [2], [3], consists of a bidding phase and an assignment phase, where an optimal assignment is found by employing a coordinate descent method on the dual function. However, scaling of the information gain matrix is critical to the success of the auction algorithm. The κ-best assignment algorithm, first proposed

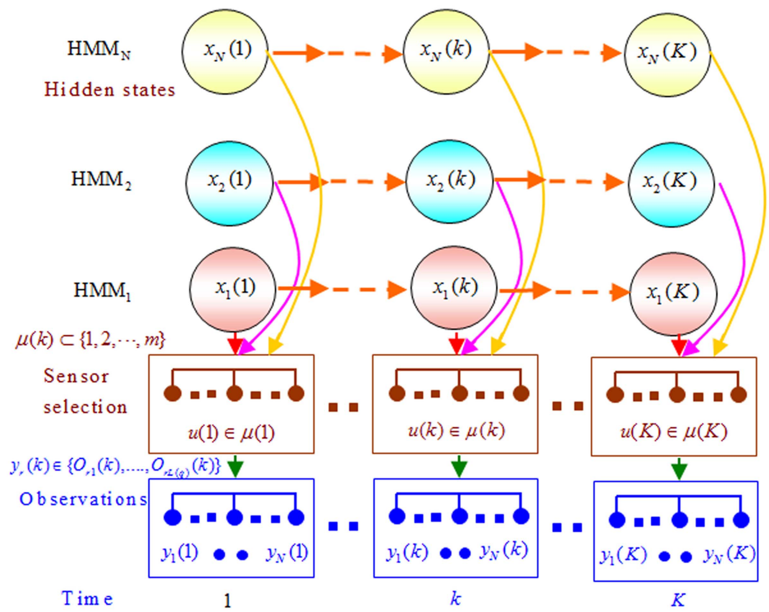

Fig. 1. Sensor scheduling problem for multiple HMMs.

by Murty [20], is independent of the algorithm chosen for solving the assignment problem. This algorithm ranks all the assignments in the order of decreasing objective function value by a clever partitioning of the search space of feasible assignments. The computational efficiency of Murty's algorithm has been enhanced by Cox *et al.* [10], Miller *et al.* [19], and Popp *et al.* [21], where the κ-best assignment algorithm is used to rank order assignment solutions for data association.

1.2. Scope and Organization of the Paper

This paper makes three novel contributions. First, motivated by the intractability of DP recursion even for a single HMM-based sensor scheduling problem [16] and its success in sequential probing for fault diagnosis [27], we propose a greedy heuristic algorithm based on information gain per unit sensor usage cost. We derive the information gain of a sensor for HMM models in the predictor-corrector form of state estimation equations, which are ideally suited for on-line implementation. Second, we improve the information gain heuristic algorithm by embedding it in a rollout algorithm to improve its scheduling performance. This is accomplished via the solution of a κ-best assignment algorithm. The multiple HMM scheduling problems using the combined rollout and assignment approach proposed herein have not been considered in the literature. Finally, the algorithms are applied to realistic ISR mission scenarios arising in ESG missions.

The paper is organized as follows. In Section 2, the multiple sensor scheduling problem is formulated. In Section 3, the DP recursion is developed. In Section 4, we present the rollout information gain heuristic algorithm based on JVC and κ-best assignment algorithm. We apply our solution approach to the ISR mission scenario, and present its results in Section 5. Finally, Section 6 concludes with a summary.

2. MULTIPLE HMM SENSOR SCHEDULING PROBLEM

2.1. The Factorial Hidden Markov Model (FHMM) for Dynamic Sensor Scheduling

Consider a scenario with N marginally independent discrete HMMs evolving independently and coupled via the observation process, as shown in Fig. 1. This model is also known as FHMM in the machine learning literature [13]. However, our framework is valid for coupled HMMs [7] and hierarchical HMMs [12] as well. Suppose there are m sensors, and $\mu(k) \subseteq \{1,2,\ldots,m\}$ are the set of available sensors at decision epoch $k \in \{1,2,\ldots,K\}$. We assume that at most a single sensor out of available sensors, $\mu(k)$, is assigned for observing the hidden state of a HMM at time epoch k. The FHMM is parameterized by the set of transition probability matrices $\mathbf{A}(k)$, the set of emission matrices $\mathbf{B}(k)$, and the set of initial probability vectors φ. We assume that the FHMM parameter sets $\Lambda(k) = (\mathbf{A}(k), \mathbf{B}(k), \varphi)$ $(k = 1,2,\ldots,K)$, are known *a priori*; however, they could also be estimated based on historical data using the Baum-Welch algorithm [1].

The set of transition probability matrices of the underlying Markov chains associated with the N HMMs is given by $\mathbf{A}(k) = \{A_1(k),\ldots,A_r(k),\ldots,A_N(k)\}$ at time epoch k, where $A_r(k)$ denotes the transition probability matrix of the rth HMM:

$$A_r(k) = [a_{rij}(k)] = [P(x_r(k) = s_{rj} \mid x_r(k-1) = s_{ri})], \quad i,j = 1,2,\ldots,n_r \qquad (1)$$

where $x_r(k)$ is the hidden state of rth HMM at time epoch k. The hidden state $x_r(k) \in \{s_{ri} : i = 1,2,\ldots,n_r\}$, where n_r is the number of states used for modeling the rth HMM. We denote the subset of emission matrices, corresponding to each of the m sensors associated with the rth HMM, as $B_r(k) = \{B_{r1}(k),\ldots,B_{rq}(k),\ldots,B_{rm}(k)\}$ at time epoch k. The set of emission matrices for the N HMMs is denoted by $\mathbf{B}(k) = \{B_1(k),\ldots,B_r(k),\ldots, B_N(k)\}$. The observation, measured by sensor $u_r(k) = q$ assigned to the rth HMM at time epoch k, is denoted by $y_r(k) \in \{O_{r1}(k),\ldots,O_{rL(q)}(k)\}$, i.e., it belongs to one of $L(u_r(k) = q)$ symbols. Evidently, the number of observation symbols $L(q)$ can be a function of the sensors. This models a realistic scenario in which different sensors have different capabilities in generating different observation symbols. If none of the sensors is assigned to a HMM at a given epoch, we assume that the observed symbol is null (ϕ). The probability of observing the symbol $O_{rl}(k)$ ($l = 1,2,\ldots,L(q)$) with the sensor $u_r(k) = q$ assigned to the rth HMM, given the state $x_r(k) = s_{ri}$, denoted by $b_{rliq}(k)$, is an element of the emission matrix, $B_{rq}(k)$. That is,

$$B_{rq}(k) = [b_{rliq}(k)] = [P(y_r(k) = O_{rl}(k) \mid x_r(k) = s_{ri}, u_r(k) = q],$$
$$i = 1,2,\ldots,n_r; \quad l = 1,2,\ldots,L(q);$$
$$q = 1,2,\ldots,m; \quad r = 1,2,\ldots,N. \tag{2}$$

The key point here is that the observation $y_r(k)$ depends upon the current state $x_r(k)$ and the selected sensor $u_r(k)$ from among the available sensors at time k. At time epoch k, we have, for each HMM ($r = 1,2,\ldots,N$), the information sets $\{Y_r^{k-1}, U_r^{k-1}\}$, where $Y_r^{k-1} = \{y_r(1),\ldots,y_r(k-1)\}$ and $U_r^{k-1} = \{u_r(1),\ldots, u_r(k-1)\}$, the previously observed symbols and the sensor sequence used from time epoch $t = 1$ to time epoch $t = k-1$. Evidently, $Y_r^0 = U_r^0 = \phi$. The initial probability of the underlying Markov states of the rth HMM at time $t = 0$ is denoted by

$$\underline{\varphi}_r = [\varphi_{ri} = p(x_r(0) = s_{ri})],$$
$$i = 1,2,\ldots,n_r; \quad r = 1,2,\ldots,N. \tag{3}$$

We denote the set of initial probability vectors of the N HMMs as $\boldsymbol{\varphi} = \{\underline{\varphi}_1,\ldots,\underline{\varphi}_r,\ldots,\underline{\varphi}_N\}$.

2.2. Dynamic Sensor Scheduling Cost

The sensor scheduling problem is the following: How to find the policy to optimally allocate the m sensors to the N HMMs from time epoch 1 to time epoch K, based on $\{\mathbf{Y}^{k-1}, \mathbf{U}^{k-1}\}_{k=1}^K$, where $(\mathbf{Y}^{k-1}, \mathbf{U}^{k-1}) = \{Y_r^{k-1}, U_r^{k-1}\}_{r=1}^N$, the information available to optimize the sensor schedule at time epoch k. The sensor scheduling cost function is a sum of sensor usage costs and the state estimation errors over the planning horizon. The information states $\boldsymbol{\Pi}(k \mid k-1) = \{\underline{\pi}_1(k \mid k-1),\ldots, \underline{\pi}_r(k \mid k-1),\ldots,\underline{\pi}_N(k \mid k-1)\}^T$ are sufficient statistics to describe the current state of the N HMMs, where $\underline{\pi}_r(k \mid k-1) = \{\pi_{r1}(k \mid k-1),\ldots,\pi_{rn_r}(k \mid k-1)\}^T$. Indeed, the information state is the *predicted* probability of the hidden state $\mathbf{X}(k) = \{\underline{x}_1(k),\ldots,\underline{x}_r(k),\ldots,\underline{x}_N(k)\}^T$ given the available information, $\{\mathbf{Y}^{k-1}, \mathbf{U}^{k-1}\}$, where $\underline{x}_r(k) = \{s_{r1}(k),\ldots,s_{rn_r}(k)\}^T$, i.e.,

$$\boldsymbol{\Pi}(k \mid k-1) = \mathbf{P}(\mathbf{X}(k) \mid \mathbf{Y}^{k-1}, \mathbf{U}^{k-1}) \tag{4}$$

where $\mathbf{P}(\mathbf{X}(k) \mid \mathbf{Y}^{k-1}, \mathbf{U}^{k-1}) = \{P(\underline{x}_1(k) \mid Y_1^{k-1}, U_1^{k-1}),\ldots, P(\underline{x}_r(k) \mid Y_r^{k-1}, U_r^{k-1}),\ldots,P(\underline{x}_N(k) \mid Y_N^{k-1}, U_N^{k-1})\}^T$. Let us denote the sensor scheduling policy from time epoch 1 to time epoch K by $\xi = \{\xi(k)\}_{k=1}^K$. For a given policy, the cumulative expected schedule cost from time epoch 1 to time epoch K, denoted by J_ξ, is assumed to be of the form:

$$J_\xi = E\left[\sum_{r=1}^N \left[\beta f_{rK}(\underline{\pi}_r(K \mid K)) + \sum_{k=0}^{K-1} \beta f_{rk}(\underline{\pi}_r(k \mid k)) + \sum_{k=1}^K g_{rk}(u_r(k), \underline{\pi}_r(k \mid k-1))\right]\right] \tag{5}$$

where $\underline{\pi}_r(k \mid k)$ is the updated (corrected) information state, $f_{rk}(\underline{\pi}_r(k \mid k))$ is the state estimation error, $g_{rk}(u_r(k), \underline{\pi}_r(k \mid k-1))$ is the sensor cost of the rth HMM, and β is a positive scalar weight. Here, the expectation is over the stochastic realizations of measurement sequences. Typical cost functions for the state estimation error are as follows:

$$f_{rk}(\underline{\pi}_r(k \mid k)) = 1 - \underline{\pi}_r^T(k \mid k)\underline{\pi}_r(k \mid k), \tag{6}$$

$$f_{rk}(\underline{\pi}_r(k \mid k)) = 1 - \max_{i \in \{1,\ldots,n_r\}} \pi_{ri}(k \mid k), \tag{7}$$

$$f_{rk}(\underline{\pi}_r(k \mid k)) = \min_{1 \le i \le n_r} \sum_{j=1}^{n_r} \pi_{rj}(k \mid k)\lambda_{ij}. \tag{8}$$

The first criterion as given in (6) can be interpreted as the L_2-norm of the updated state estimation error; the second criterion in (7) as the error probability of a maximum *a posteriori* probability (*MAP*)-based decision rule; while the third criterion in (8) as the expected cost of errors in estimating the information state. In (8), λ_{ij} represents the cost of erroneously estimating the hidden state as s_{ri} when the true state is s_{rj}. The sensor cost $g_{rk}(u_r(k), \underline{\pi}_r(k \mid k-1))$ is the sum of sensor usage cost $h_{rk}(u_r(k), \underline{\pi}_r(k \mid k-1))$ and sensor travel (movement) cost $c_m(u_r(k))$, i.e.,

$$g_{rk}(u_r(k), \underline{\pi}_r(k \mid k-1)) = h_{rk}(u_r(k), \underline{\pi}_r(k \mid k-1)) + c_m(u_r(k)) \tag{9}$$

where the sensor usage cost is given by

$$h_{rk}(u_r(k), \underline{\pi}_r(k \mid k-1)) = \sum_{i=1}^{n_r} c_{rk}(s_{ri}, u_r(k))\pi_{ri}(k \mid k-1) = \underline{c}_k^T(u_r(k))\underline{\pi}_r(k \mid k-1) \tag{10}$$

and $\underline{c}_k^T(u_r(k)) = \{c_{rk}(s_{r1}, u_r(k)), c_{rk}(s_{r2}, u_r(k)), \ldots, c_{rk}(s_{rn_r}, u_r(k))\}$ is the usage cost of sensor $u_r(k)$ corresponding to each of the states $\{s_{ri}\}_{i=1}^{n_r}$. We also considered the cost of moving a sensor, denoted by $c_m(u_r(k))$, from its current location to the location of the task. This cost is computed via a simplified travel cost model as

$$c_m(u_r(k)) = \frac{\|(a_r, b_r) - (a_{u_r(k)}, b_{u_r(k)})\|_2}{v(u_r(k))} w(u_r(k)) \tag{11}$$

where (a_r, b_r) and $(a_{u_r(k)}, b_{u_r(k)})$ denote the cartesian coordinates of the task location (indexed by the HMM) and the selected sensor $u_r(k)$ for monitoring the rth HMM, respectively. Here, $w(u_r(k))$ is a priority parameter that accounts for the scarcity of the sensor, $v(u_r(k))$ denotes the velocity of the sensor (or the mobile platform on which it is resident), and $\|\cdot\|_2$ denotes the Euclidean (2-) norm.

3. DYNAMIC PROGRAMMING ALGORITHMS FOR OPTIMAL SOLUTION

The optimal solution to the sensor scheduling problem is to find the sensor assignment policy ψ^* which minimizes the sensor scheduling cost defined in (5). Let us define a optimal cost-to-go function $J^*(\Pi(k \mid k))$as follows:

$$\begin{aligned} J^*(\mathbf{\Pi}(k \mid k)) &= E\left[\sum_{r=1}^{N}\left[\beta f_{rK}(\underline{\pi}_r(K \mid K)) + \sum_{l=k}^{K-1} \beta f_{rl}(\underline{\pi}_r(l \mid l)) \right.\right. \\ &\quad \left.\left. + \sum_{l=k}^{K} g_{rl}(\psi_r^*(\mathbf{\Pi}(l \mid l-1)), \underline{\pi}_r(l \mid l-1))\right]\right] \end{aligned} \tag{12}$$

where $\psi_r^*(\Pi(l \mid l-1)) = u_r^*(l)$ is the optimal sensor allocated to rth HMM in the optimal policy. The optimal cost-to-go function $J^*(\mathbf{\Pi}(k \mid k))$ satisfies the dynamic programming (DP) recursion:

$$\begin{aligned} J^*(\mathbf{\Pi}(k \mid k)) &= E\left[\sum_{r=1}^{N}[\beta f_{rk}(\underline{\pi}_r(k \mid k)) + g_{rk}(\psi_r^*(\mathbf{\Pi}(k \mid k-1)), \underline{\pi}_r(k \mid k-1))]\right. \\ &\quad \left. + J^*(\mathbf{\Pi}(k+1 \mid k+1))\right] \end{aligned} \tag{13}$$

with the terminal condition $J^*(\mathbf{\Pi}(K \mid K)) = \sum_{r=1}^{N} \beta f_{rK} \cdot (\underline{\pi}_r(K \mid K)) + g_{rK}(\psi_r^*(\mathbf{\Pi}(K \mid K-1)), \quad \underline{\pi}_r(K \mid K-1))$. Hence, the optimal solution of (5) can be obtained using the dynamic programming (DP) technique; however, the computational complexity is $\mathbf{O}(\prod_{r=1}^{N} D^{(n_r-1)} nLmK)$. Here, D is the number of quantization levels used to discretize the continuous-valued information probability state, m is the number of sensors, K is the number of time epochs, N is the number of HMMs, n_r is the number of states of rth HMM, $n = \max_{r\in\{1,2,\ldots,N\}} n_r$, $L = \max_{r\in\{1,2,\ldots,N\}, q\in\{1,2,\ldots,m\}}(L(u_r(.) = q))$, and $L(u_r(.) = q)$ is the number of observation symbols when sensor q is allocated to the rth HMM at any epoch. The computational complexity is intractable in both n and N. This motivates us to investigate suboptimal algorithms to solve the dynamic sensor scheduling problem. We propose the rollout information gain (RIG) algorithm with computational complexity of $\mathbf{O}(NnLm^2K)$ per rollout, which is significantly lower than that for the DP technique.

4. ROLLOUT STRATEGIES TO SOLVE SENSOR SCHEDULING PROBLEM WITH MULTIPLE HMMS

4.1. Information Gain Heuristic as a Base Policy

Multiple HMM sensor scheduling involves two-dimensional (2-D) assignment or a weighted bipartite matching problem, where one set of nodes corresponds to sensors and the other set to HMMs. When allocating m sensors among N HMMs at each time epoch using the information gain heuristic algorithm, one needs to consider the $m \times N$ matrix of information gains for each sensor-HMM pair, where the elements of qth row correspond to information gains obtained by assigning sensor q to each of the N HMMs, as shown in Fig. 2. The information gain heuristic algorithm selects the best sensor assignment at each time epoch k, $\delta^*(k)$, that maximizes the sum of information gains per unit sensor usage cost, subject to the assignment constraints that at most one sensor can be assigned to a HMM and that at most one HMM can be assigned to a sensor. The assignment problem at time epoch k is (assuming without loss of generality that $m < N$)[1]

$$\begin{aligned} \delta^*(k) &= \underset{\delta(k)\in\xi(k)}{\arg\max} \sum_{q=1}^{m}\sum_{r=1}^{N} \frac{I_{qr}(\underline{\pi}_r(k \mid k-1), u_r(k) = q)}{g_{rk}(u_r(k), \underline{\pi}_r(k \mid k-1))} \delta_{qr}(k) \\ \text{subject to} \quad & \sum_{q=1}^{m} \delta_{qr}(k) \le 1, \qquad r = 1,2,\ldots,N; \\ & \sum_{r=1}^{N} \delta_{qr}(k) = 1, \qquad q = 1,2,\ldots,m \end{aligned} \tag{14}$$

where $g_{rk}(u_r(k), \underline{\pi}_r(k \mid k-1))$ is the sensor usage cost when it is assigned to the rth HMM as defined in (9), and $I_{qr}(\underline{\pi}_r(k \mid k-1), u_r(k) = q)$ is the information gain

[1]This formulation can be extended to the case where multiple sensors may be needed to estimate a HMM state or a single sensor can estimate states of multiple HMMs. See [4].

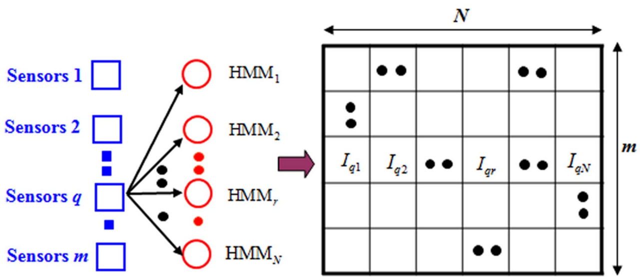

Fig. 2. Information gain matrix for multiple HMM-multiple sensor case.

given by:

$$
\begin{aligned}
I_{qr}(\underline{\pi}_r(k \mid k-1), u_r(k) = q) &= \sum_{i=1}^{n_r} \pi_{ri}(k \mid k-1) \sum_{l=1}^{L(q)} b_{rliq}(k) \log_2 b_{rliq}(k) \\
&\quad - \sum_{l=1}^{L(q)} \left(\sum_{i=1}^{n_r} b_{rliq}(k) \pi_{ri}(k \mid k-1) \right) \\
&\quad \cdot \log_2 \left(\sum_{i=1}^{n_r} b_{rliq}(k) \pi_{ri}(k \mid k-1) \right). \qquad (15)
\end{aligned}
$$

The derivation of information gain equation is provided in the Appendix. The formulation in (13) is an asymmetric assignment problem, because none of the sensors may be assigned to some HMMs, leading to *null* observations at that time epoch for the corresponding unassigned HMMs.

The Jonker-Volgenant-Castañòn (JVC) and the auction are the most efficient algorithms for solving the (2-D) assignment problems. The JVC algorithm [15] is a primal-dual method that includes an effective initialization of dual variables, and an augmentation phase based on the Dijkstra's shortest path algorithm [11]. The auction algorithm, proposed by Bertsekas *et al.* [2] [3], consists of a bidding phase and an assignment phase, where an optimal assignment is found by employing a coordinate descent method on the dual function. However, scaling of the weight (in our case the information gain per unit cost) matrix is critical to the success of the auction algorithm.

The JVC algorithm is used here for finding the best assignment of sensors among multiple HMMs at each time epoch. Thus, in the multiple HMM case, the information gain heuristic algorithm can be implemented using the following five steps (see Fig. 3 for IG heuristic processing steps of a single (rth) HMM):[2]

Step 1 (State Prediction): Predict the information state vector set $\mathbf{\Pi}(k \mid k-1) = \{\underline{\pi}_1(k \mid k-1), \ldots, \underline{\pi}_r(k \mid k-1), \ldots, \underline{\pi}_N(k \mid k-1)\}^T$. Here, $\underline{\pi}_r(k \mid k-1)$ at time epoch k is predicted using the current updated information state vector at time epoch $(k-1)$, $\underline{\pi}_r(k-1 \mid k-1)$, and the transition matrix $A_r(k)$:

$$\underline{\pi}_r(k \mid k-1) = A_r^T(k)\underline{\pi}_r(k-1 \mid k-1), \quad 1 \le r \le N. \qquad (16)$$

Here N is the number of HMMs being tracked and $\underline{\pi}_r(k \mid k-1) = \{\pi_{r1}(k \mid k-1), \ldots, \pi_{rn_r}(k \mid k-1)\}^T$. Evidently, the updated information state $\mathbf{\Pi}(k-1 \mid k-1)$ uses all the information available up to time epoch $k-1$, i.e., $\{\mathbf{Y}^{k-1}, \mathbf{U}^{k-1}\}$.

Step 2 (Generation of Information Gain Matrix): We construct the matrix of information gains per unit sensor cost for all sensor-HMM pairs,

$$\frac{I_{qr}(\underline{\pi}_r(k \mid k-1), u_r(k) = q)}{g_{rk}(u_r(k), \underline{\pi}_r(k \mid k-1))}, \quad r = 1,2,\ldots,N; \quad q = 1,2,\ldots,m.$$

Step 3 (Sensor Assignment): Select the best sensor assignment $\delta^*(k)$ that maximizes the sum of information gains in (14) using the JVC assignment algorithm.

Step 4 (Observation): The set of observations $\{y_1(k), y_2(k), \ldots, y_N(k)\}$ at time epoch k are obtained using the sensor set $u_r(k)$ $(r = 1,2,\ldots,N)$ based on the emission probability matrices given in (2).

Step 5 (State Update): Obtain the updated information state, $\pi_{ri}(k \mid k)$, by using the forward algorithm [22] as follows:

$$\pi_{ri}(k \mid k) = \frac{b_{rliq*}(k)\pi_{ri}(k \mid k-1)}{\sum_{j=1}^{n_r} b_{rljq*}(k)\pi_{rj}(k \mid k-1)} \qquad (17)$$

where $\pi_{ri}(k \mid k)$ is the ith element of $\underline{\pi}_r(k \mid k-1) = \{\pi_{r1}(k \mid k-1), \ldots, \pi_{ri}(k \mid k-1), \ldots, \pi_{rn_r}(k \mid k-1)\}^T$ and the b_{rliq*} is the (l,i) element of emission matrix $B_{rq*}(k)$. It is the probability of the symbol $O_{rl}(k)$ $(l = 1,2,\ldots, L(q))$ when the sensor $u_r(k) = q*$ is assigned to the rth HMM, given the state $x_r(k) = s_{ri}$.

[2] The information gain heuristic is derived in terms of predictor-corrector form of discrete HMM equations. These are similar to the dynamic Bayesian state estimation equations, when the HMM states and observations are continuous. In the latter case, $\pi_r(k \mid k-1) \triangleq p(\underline{x}_r(k) \mid Y_r^{k-1}, U_r^{k-1})$ and $\pi_r(k \mid k) \triangleq p(\underline{x}_r(k) \mid Y_r^k, U_r^k)$ should be interpreted as probability density functions of system state. The predicted information state for the next time epoch k can be obtained as:

$$\pi_r(k \mid k-1) = \int_{\underline{x}_r(k-1)} p(\underline{x}_r(k) \mid \underline{x}_r(k-1))\pi_r(k-1 \mid k-1) d\underline{x}_r(k-1); \quad 1 \le r \le N$$

Once the observation for selected sensor $u_r(k)$ is obtained, the updated information state is:

$$\pi_r(k \mid k) = \frac{p(\underline{y}_r(k) \mid \underline{x}_r(k), u_r(k) = q)}{\int_{\underline{x}_r(k)} p(\underline{y}_r(k) \mid \underline{x}_r(k), u_r(k) = q)\pi_r(k \mid k-1) d\underline{x}_r(k)} \pi_r(k \mid k-1); \quad 1 \le r \le N$$

Typically, these integrals are intractable. However, if we can obtain analytical approximations to the above equations (e.g., Gaussian sum approximation), information gain heuristic would still be a tractable approach.

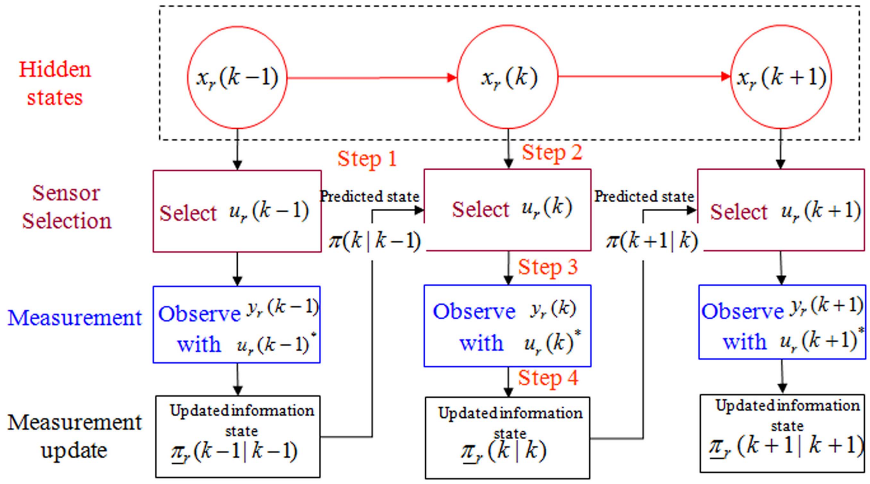

Fig. 3. Information gain heuristic (IG) processing steps for rth HMM.

4.2. Rollout Algorithms

Rollout algorithms are a class of suboptimal methods which are capable of improving the effectiveness of any given heuristic through sequential application. This is due to the policy improvement mechanism of the underlying policy iteration process [27]. They can be viewed as a single step of the classical policy iteration method, wherein we start from a given easily implementable and computationally tractable policy, and then try to improve on that policy using online learning and simulation. The attractive aspects of rollout algorithms are simplicity, broad applicability, and suitability for online implementation. The details of the rollout algorithms are provided in [5], [6], [27]. In our rollout framework, the information gain heuristic is used as a base policy where optimal cost-to-go function $J^*(\mathbf{\Pi}(k+1\mid k+1))$ is approximated by the cost-to-go function $J(\mathbf{\Pi}(k+1\mid k+1))$ of the information gain heuristic. The rollout policy for approximating (13) can be written in terms of Q-factor as follows:

$$
\begin{aligned}
\delta^*(k) &= \arg\min_{\delta^i(k)\in\delta(k)} Q(\mathbf{\Pi}(k\mid k-1),\delta^i(k)), \qquad i=1,\ldots,\kappa\\
&= \arg\min_{\delta^i(k)\in\delta(k)} E\left[\sum_{r=1}^{N}[f_{rk}(\underline{\pi}_r(k\mid k))\right.\\
&\qquad +g_{rk}(\psi_r^i(\mathbf{\Pi}(k\mid k-1)),\underline{\pi}_r(k\mid k-1))]\\
&\qquad \left. +J(\mathbf{\Pi}(k+1\mid k+1))\right] \qquad (18)
\end{aligned}
$$

where $\delta(k)=\{\delta^1(k),\ldots,\delta^\kappa(k)\}$ are the κ-best assignments used to reduce the search space, and $\psi_r^i(\mathbf{\Pi}(k\mid k-1))=u_r^i(k)$ is the sensor assigned to monitor the rth HMM in the ith-best assignment. The problem of computing the κ-best assignments is solved by combining the JVC algorithm with a modified Murty's algorithm [20], [10], [21]. However, the Q-factor driven by ith-best assignment $\delta^i(k)$ at time epoch k can not be computed in closed-form. A straightforward approach for computing the Q-factors is to use Monte Carlo simulations for $J(\mathbf{\Pi}(k+1\mid k+1))$. Unfortunately, the method suffers from increase in computational complexity. In our paper, given information state vector $\mathbf{\Pi}(k\mid k)$ at time epoch k, we approximated $J(\mathbf{\Pi}(k+1\mid k+1))$ by generating a single schedule trajectory computed from the information gain heuristic starting from $k+1$ to K. The rollout assignment is obtained by minimizing the approximated Q-factor in (18) from the κ-best assignments at time epoch k.

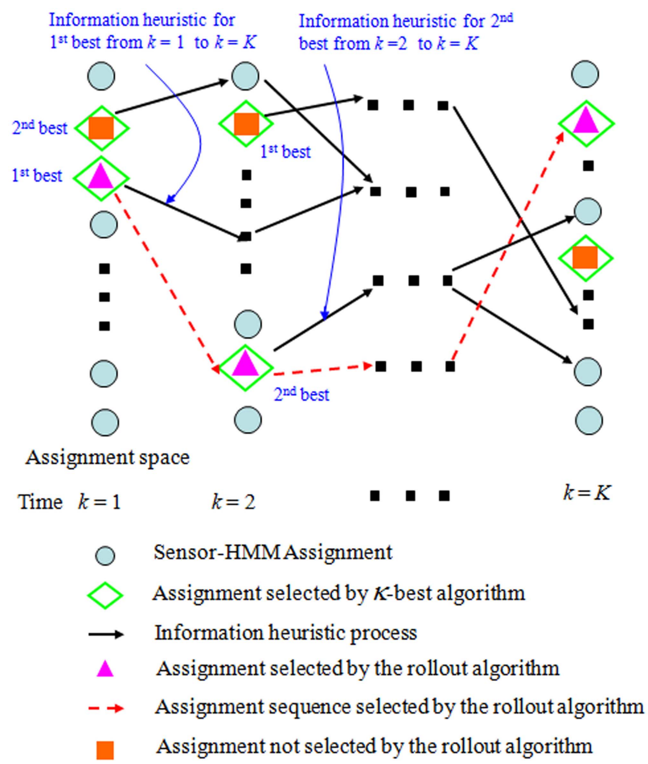

Fig. 4. Rollout information gain (RIG) algorithm coupled with κ-best assignment algorithm.

Fig. 4 graphically illustrates the RIG algorithm with two rollouts at each time epoch. The pseudo code of the

Rollout Information Gain Algorithm for Multiple HMMs:

Step 1: Predict the information state of each HMM using (16) as described in Section *3.1*.

Step 2: Generate the 2-D information gain matrix by computing the information gain per unit sensor cost for all sensor-HMM pairs at time k,

$$\frac{I_{qr}(\underline{\pi}_r(k|k-1), u_r(k)=q)}{g_{rk}(u_r(k), \underline{\pi}_r(k|k-1))}, \quad r=1,2,...,N; q=1,2,...,m.$$

Step 3: Find the κ-best sensor assignments, using the modified Murty's algorithm and the JVC algorithm. Usually, κ is in the range 2$\sim$5 to avoid excessive rollouts.

Step 4: Compute the approximated Q-factor in (18) for the κ-best sensor assignments. Select the sensor assignment which minimizes the approximated Q-factor in (18).

Step 5: If the cost has not improved over Z time epochs (Z in the range 3$\sim$5), stop. Else, update time epoch $k \to k+1$ and go back to *Step 1* until $k = K$.

Fig. 5. Pseudo code for the rollout information gain (RIG) algorithm.

Fig. 6. Notional area for scenario development.

RIG algorithm using the κ-best assignment algorithm is shown in Fig. 5.

5. COMPUTATIONAL RESULTS

5.1. A Hypothetical Mission Scenario

This scenario, motivated by ESG missions, involves simultaneous monitoring of multiple geographically dispersed threat activities. Here, an ISR officer needs to dynamically allocate sensors to monitor threats in a notional area (e.g., fishing villages, refugee camps) that involves primarily two fictitious countries, Asiland and Bartola [14]. Asiland is an unstable state, where maritime smugglers and anti-western terrorist groups have supported the insurgent factions hostile to the government of Bartola. Local terrorists and sea rovers use Asiland as a base. The scenario considers that nearly a month ago, the northern shore of Asiland was struck by a tsunami that destroyed several fishing villages and caused enormous casualties. Large numbers of Asiland citizens sought refuge in the south for help and assistance. However, this exodus quickly drained the resources of Asiland. Consequently, many Asiland refugees began to move to fishing villages and refugee camps in Bartola. Within a few days, insurgents and terrorist factions in and around Asiland began to exploit the situation, infiltrating their operations into Bartola by disguising as refugees and smuggling weapons onboard fishing boats and merchant ships. Bartola's military was overwhelmed by the massive influx of refugee boats, as well as tracking the terrorist/insurgent's activities using these boats and ships for illegal transfers. The government of Bartola sought help from the United States to provide Humanitarian Assistance/Disaster Relief (HA/DR) to Bartola and the organizations operating relief activities within it. The ESG sensor assets are deployed and begin to monitor strategically significant areas (e.g., major sea and air lanes as well as several major ports, villages, refugee camps, roads, and cities/sites) as shown in Fig. 6.

5.2. Single HMM Scenario: Monitoring a Fishing Village

We consider a scenario where an ISR officer needs to dynamically allocate sensors to monitor asymmet-

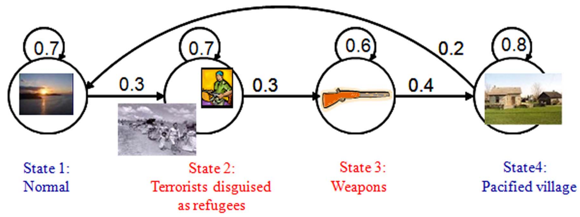

Fig. 7. A fishing village scenario for sensor scheduling problem.

ric threat activities in the notional area. The problem of monitoring the presence of terrorists and weapons in a fishing village is modeled using a four-state HMM. Activities such as the presence of terrorists and weapons, and ascertaining the crowd demeanor (normal or protesters or terrorists) are continually monitored using six sensors; labeled 1 through 6. As shown in Fig. 7, *State* 1 represents the normal state of the crowd in the fishing village. In *State* 2, refugees move into the village. Terrorists disguised as refugees arrive in the village and they also smuggle weapons into the village; this is modeled as *State* 3. In *State* 4, the weapons and terrorists are prosecuted/pacified and the village resumes normalcy, which is modeled by a transition to *State* 1. We specify the transition probability matrix, $A(k)$, based on state transitions and time spent in each state. In a discrete HMM model, the probability of staying in state j for a duration d is given by $p(d) = (a_{jj})^{d-1}(1-a_{jj})$, where the expected duration is obtained from the following equations:

$$E[d] = \sum_{d=1}^{\infty} d(a_{jj})^{d-1}(1-a_{jj}) = \frac{1}{(1-a_{jj})}. \quad (19)$$

The self transition probability of state j is set by substituting $E[d]$ in (19) with the duration provided by the scenario. The state transition probabilities depend on how many links exist from state j to other states. Suppose that state j has n state transitions and state i is linked to state j, then the probability a_{ji} is assumed to be given by $a_{ji} = (1-a_{jj})/n$. For simulations, we set the weighted priority vector $\underline{w}(u(k))$ in (11) to be a vector of constants and β is set to 10 in (5). The velocities of the six sensors are set as $\underline{v}^T(u(k)) = [300, 200, 300, 100, 450, 80]$. The sensor usage costs in (10) are selected as $\underline{c}_k(u(k)) = \{11, 4, 8, 5, 6, 2\}$, where for simplicity, each sensor usage cost vector is assumed to be independent of state. We used $\kappa = 6$ for the RIG algorithm. We assume that the initial probability distribution is known. The emission matrices are set by considering the sensing capability of each sensor, which are modeled by the probability of detection of the true states of each HMM. In this simulation, we assume that the observation capabilitities of a sensor decreases as the sensor label increases and each sensor provides one of four observation symbols at each time epoch k. The planning horizon $K = 15$ is set by considering the sum of expected durations as given in (19). Fig. 8 shows the total scheduling cost averaged over 1000 Monte Carlo runs. To assess the robustness of the algorithm, the cost was obtained by varying the observation probability using the variable p. Here, *Sensor* 1 or *Sensor* 5 curves indicate that a static schedule that employs *Sensor* 1 or *Sensor* 5 for all time epochs is substantially worse than a dynamic schedule. The rollout information gain algorithm (RIG)-based sensor schedule has approximately 5–18% lower cost as compared to one using only the information gain heuristic (IG); it also has $\approx 49\%$ lower cost as compared to a static schedule that employs *Sensor* 5 throughout. The dual advantages of using the RIG algorithm are that it significantly reduces the complexity of dynamic programming, while improving the accuracy over the base heuristic, viz., the information gain heuristic (IG) algorithm.

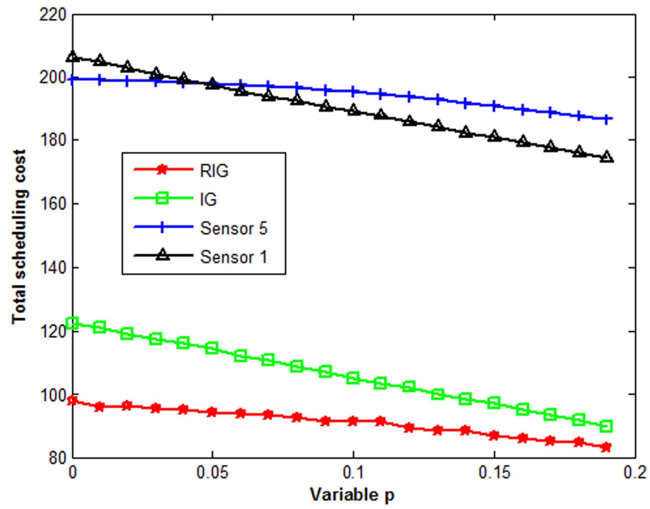

Fig. 8. Variation of the total scheduling cost with sensor accuracy variable p.

5.3 Multiple HMM Scenario: Monitoring Multiple Fishing Villages and Refugee Camps

In this scenario, we solve the problem of monitoring multiple fishing villages (FVs) and refugee camps (RCs) using multiple sensors. The problem of monitoring for the presence of refugees, weapons, and learning the crowd demeanor (normal or protesters or agitators or terrorists) in FVs and RCs is modeled using 17 HMMs, where their states are represented in a vector form (e.g., (refugees, weapons, crowd)), as shown in Fig. 9. For example, $(1, 1, 3)$ corresponds to the 16th state that indicates that refugees, weapons, and terrorists are present. Threat activities are continually monitored using a total of 17 sensors, which are comprised of 9 different types, as described in Table I. The states of 10 FVs and 7 RCs change dynamically by the departure and entry of refugees, and terrorists/insurgents (disguised as refugees). Fig. 10 shows the state transitions considered in this scenario.

The schedule cost considered for this scenario is given in (5), where sensor usage costs, $\underline{c}_k(u_r(k))$, are

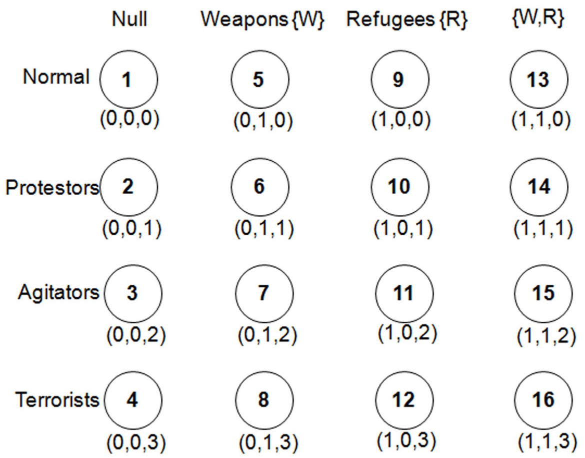

Fig. 9. States used for modeling multiple HMM scenario.

set by evaluating a weighted sum of unit price of the sensor and the crew required to operate the sensor. We employed the state estimation error criterion, given in (7). The cost of moving the sensors, $c_m(u_r(k))$, is set based on the actual mobility of the sensors and the distance from the task, as listed in Table I. The emission matrices are set by considering the sensing capability of each sensor as well as allocation preferences and geometrical constraints of sensing operations. If a sensor, $u_r(k) = q$, is irrelevant to monitoring a HMM (say, rth HMM), the entries of the emission matrix, given in (2), are set to uniform values, i.e., the emission matrix is a doubly stochastic matrix. We assume that each sensor provides one of the 16 observation symbols from the scenario. We specify the transition probability matrices, using the same process as that used in the single HMM case. However, since the scenario does not provide information on all the self-transition probabilities, the undefined self-transition probabilities are set to reasonable values and the remaining transition probabilities in the same row are uniformly distributed. For example, the transition matrix of Glorisabay, A_7, is set as shown in Table II, based on the state transition sequence of Glorisabay as shown in Fig. 10.

Fig. 11 shows the assignment distributions over 1500 mission scenarios, each averaged over 50 Monte Carlo runs. The value of β is set as 40 in (5) and the values of priority vector are set to a constant vector as in the single HMM case. The assignment distributions of RIG algorithm are obtained using $\kappa = 2$-best assignments. We assume that the initial probability distribution is known. The planning horizon $K = 30$ is set by considering the sum of expected durations as given in (19). Table III shows the state transitions and RIG-based sensor assignments in Glorisabay FV. Note assignments of *Sensor* 8 for RCs and *Sensor* 1 for FVs. The assignments are reasonable because *Sensor* 8 has difficulty in sensing the operations in RCs (e.g., patrol ships) and an ISR officer always assigns *Sensor* 1 to RCs as a first priority. To model the sensing constraints of *Sensor* 1 and *Sensor* 8 in the extreme case, the *Sensor* 1 emission matrix probabilities for FVs and the *Sensor* 8 emission matrix entries for RCs are distributed uniformly. The realization of assignment constraints by the setup of emission matrix is shown in Fig. 11, where columns correspond to sensors and rows correspond to HMMs. The brightness represents the number of assignments of HMM-sensor pair. Fig. 12 shows the total scheduling cost. The cost of assignment by distance was obtained using sensor assignments to minimize the sum

TABLE I
Setup of Usage Cost and Sensing Capability

		Sensor 1	Sensor 2	Sensor 3	Sensor 4	Sensor 5	Sensor 6	Sensor 7	Sensor 8	Sensor 9
	Crew	5	0	20	2	10	1	20	5	15
	Unit Price	51	13	0	12	36	22	40	26	40
Usage Cost	0.2 X Cr +0.16XUP	11	2	10	3	11	4	17	7	14
Mobility	Speed (kts)	333	49	30	180	330	547	30	35	30
Sensing Capability		Long range radar	Radar Camera	Human	Short Range Radar	Radar Sonar Buoy	Mid-range radar	Long Range Radar	Short range Radar	Mid-range radar
	Range	9	3	1	3	6	6	9	3	6
	Resolution	9	9	6	5	4	3	2	2	2
Sensing Capability	0.1 X Ra +0.9 X Re	9	8	6	5	4	3	3	2	2
Quantity		3	1	3	2	2	3	2	1	1

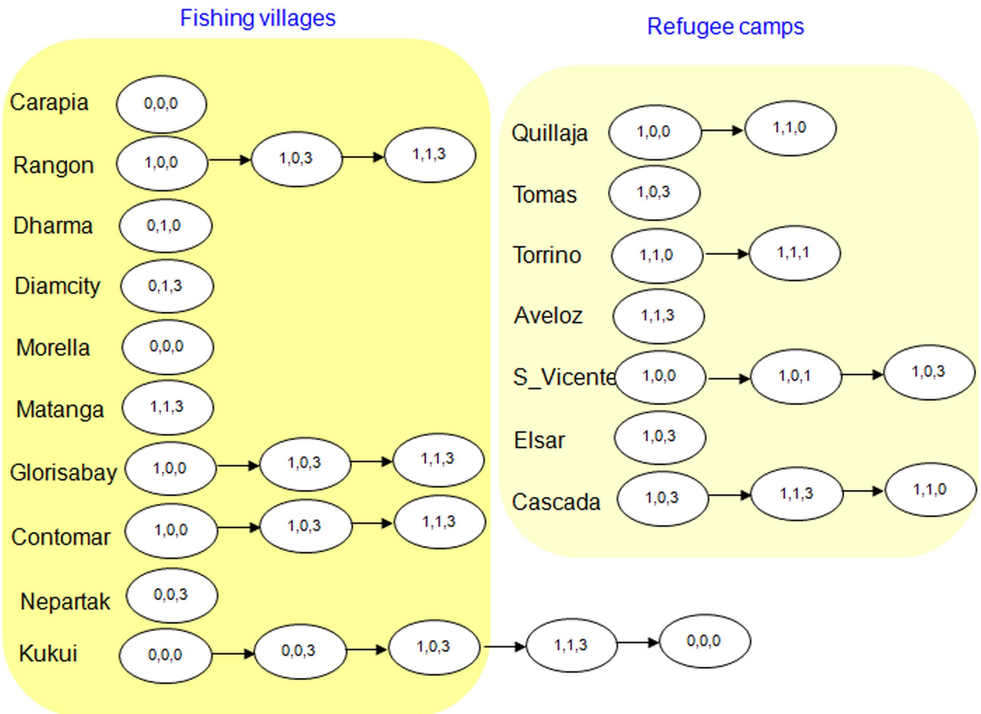

Fig. 10. States transitions of multiple fishing villages and refugee camps.

TABLE II
Transition Matrix A_7 Set Up of a Fishing Village Glorisabay

0.8	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013
.013	0.8	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013
.013	.013	0.8	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013
.013	.013	.013	0.8	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013
.013	.013	.013	.013	0.8	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013
.013	.013	.013	.013	.013	0.8	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013
.013	.013	.013	.013	.013	.013	0.8	.013	.013	.013	.013	.013	.013	.013	.013	.013
.013	.013	.013	.013	.013	.013	.013	0.8	.013	.013	.013	.013	.013	.013	.013	.013
0	0	0	0	0	0	0	0	.89	0	0	.11	0	0	0	0
.013	.013	.013	.013	.013	.013	.013	.013	.013	0.8	.013	.013	.013	.013	.013	.013
.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	0.8	.013	.013	.013	.013	.013
0	0	0	0	0	0	0	0	0	0	0	.89	0	0	0	.11
.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	0.8	.013	.013	.013
.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	0.8	.013	.013
.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	0.8	.013
.000 6	.000 6	.000 6	.000 6	.000 6	.000 6	.000 6	.000 6	.000 6	.000 6	.000 6	.000 6	.000 6	.000 6	.000 6	.99

of travelled distances. Sensor scheduling via the RIG algorithm ($\kappa = 7$) has approximately $\approx 2.1\%$ lower cost as compared to one using only the information gain heuristic (IG) and $\approx 4\%$ lower cost as compared to scheduling by distance. The results also suggest that, while the RIG algorithm in multiple HMM sensor scheduling problem improves the performance of information gain heuristic, it is less effectiveness when compared to a single

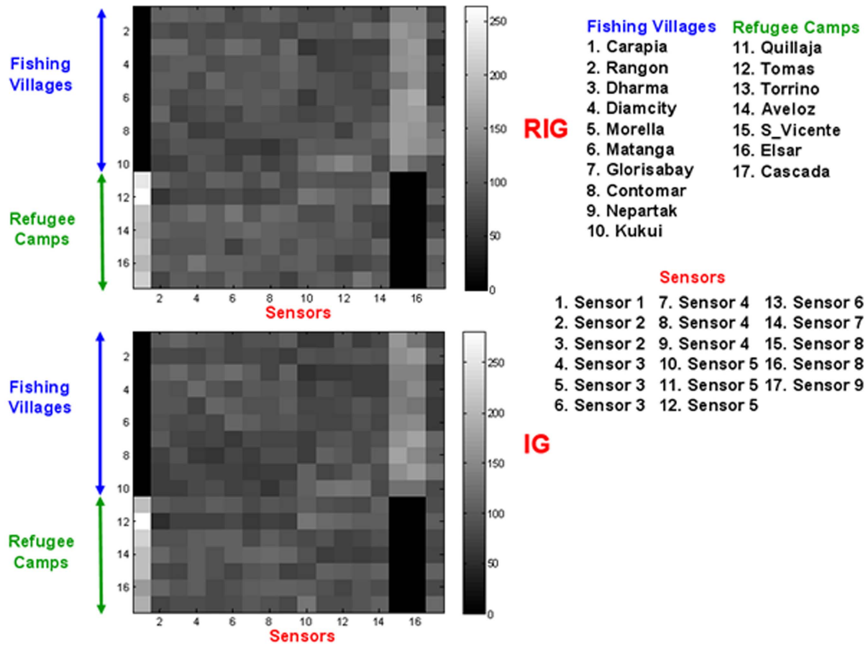

Fig. 11. Assignment distribution of IG ($\kappa = 1$) and RIG ($\kappa = 2$).

TABLE III
State Transition and RIG Sensor Assignment in Fishing Village Glorisabay

k	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
$x_7(k)$	9	9	9	9	9	9	9	9	9	9	9	9	9	12	12
$u_7(k)$	9	17	13	7	14	6	2	17	12	4	4	11	14	8	2
k	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
$x_7(k)$	12	12	12	12	12	12	12	16	16	16	16	16	16	16	16
$u_7(k)$	8	11	5	10	16	9	1	5	10	12	7	16	3	12	17

HMM sensor scheduling problem. This is due to the assignment of all available sensors to multiple HMMs. In addition, the differences in information gains, obtained from the κ-best assignments, are much less than those obtained from the κ-best sensors in the single HMM case. However, the fact that RIG and IG have nearly identical performance gives us confidence that the IG-based sensor schedules are near-optimal.

6. CONCLUSION

This paper formulated the sensor scheduling problem using HMM formalisms. The optimal solution of the sensor scheduling problem via dynamic programming (DP) is intractable for both single and multiple HMM scheduling problems due to computational explosion caused by the curse of dimensionality. To overcome this, we proposed a RIG algorithm by combining rollout concepts with the JVC and the κ-best assignment algo-

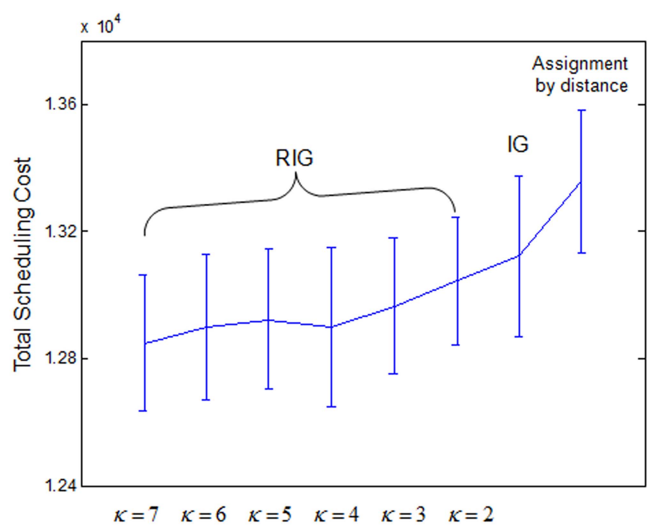

Fig. 12. Total scheduling cost of RIG, IG, and assignment by distance.

rithms. We illustrated its application on realistic mission scenarios involving the monitoring of threat activities in a fishing village modeled using a single HMM, and multiple fishing villages and refugee camps modeled using multiple HMMs.

Our work on HMM-based dynamic sensor scheduling model assumed that the tasks are independent. In practice, however, tasks may exhibit dependencies or may have a hierarchical structure. We plan to develop extensions to HMM-based sensor scheduling model to handle dependencies and hierarchical structure among tasks using coupled HMMs [7] and hierarchically structured HMMs [12]. In addition, we assumed that at most one sensor is assigned to a HMM at each time epoch. However, due to the various assignment constraints, imposed by organizational structure, this assumption may need to be relaxed. Finally, sensor scheduling is a cooperative process among multiple decision makers. Auction-based algorithms may provide a mechanism for implementing distributed and coordinated sensor scheduling algorithms. We plan to pursue these extensions in the future.

APPENDIX

Let $H(x) = -\sum_i p_i(x)\log_2 p_i(x)$ denote the entropy of the state with a probability mass function $\{p_i(x)\}$. Let us derive the information gain defined in (14) obtained by assigning a sensor $u_r(k) = q$ to rth HMM, where the information state is defined in (15). The entropy of the information state is:

$$H(\underline{\pi}_r(k \mid k-1)) = -\sum_{i=1}^{n_r} \pi_{ri}(k \mid k-1)\log_2 \pi_{ri}(k \mid k-1). \tag{20}$$

Recall that the joint entropy is given by:

$$H(x,u) = H(x) + H(u \mid x) = H(u) + H(x \mid u) \tag{21}$$

and mutual information (or information gain) is:

$$I(x,u) = H(x) - H(x \mid u). \tag{22}$$

$$= H(u) - H(u \mid x). \tag{23}$$

Using (21) and (22),

$$I_{qr}(\underline{\pi}_r(k \mid k-1), u_r(k) = q) = H(\underline{\pi}_r(k \mid k-1)) - H(\underline{\pi}_r(k \mid k-1) \mid u_r(k) = q), \tag{24}$$

$$I_{qr}(\underline{\pi}_r(k \mid k-1), u_r(k) = q) = H(u_r(k) = q) - H(u_r(k) = q \mid \underline{\pi}_r(k \mid k-1)) \tag{25}$$

where

$$H(u_r(k) = q) = -\sum_{l=1}^{L(q)} P(y_r(k) = O_{rl}(k) \mid Y_r^{k-1}, U_r^{k-1}, u_r(k) = q) \cdot \log_2 P(y_r(k) = O_{rl}(k) \mid Y_r^{k-1}, U_r^{k-1}, u_r(k) = q) \tag{26}$$

and

$$\begin{aligned} P\left(y_r(k) = O_{rl}(k) \mid Y_r^{k-1}, U_r^{k-1}, u_r(k) = q\right) &= \sum_{i=1}^{n_r} P(y_r(k) = O_{rl}(k), x_r(k) = s_{ri} \mid Y_r^{k-1}, U_r^{k-1}, u_r(k) = q) \\ &= \sum_{i=1}^{n_r} P(y_r(k) = O_{rl}(k) \mid x_r(k) = s_{ri}, u_r(k) = q)\pi_{ri}(k \mid k-1) \\ &= \sum_{i=1}^{n_r} b_{rliq}(k)\pi_{ri}(k \mid k-1). \end{aligned} \tag{27}$$

The conditional entropy of a random variable X, conditioned on a random variable Y

$$H(X \mid Y) = \sum_y p_Y(y) H(X \mid Y = y). \tag{28}$$

Using (26):

$$H(u_r(k) = q \mid \underline{\pi}_r(k \mid k-1)) = -\sum_{i=1}^{n_r} \pi_{ri}(k \mid k-1)\left[\sum_{l=1}^{L(q)} b_{rliq}(k)\log_2 b_{rliq}(k)\right]. \tag{29}$$

Using (24) and (27), the information gain is:

$$\begin{aligned} I_{qr}(\underline{\pi}_r(k \mid k-1), u_r(k) = q) &= \sum_{i=1}^{n_r} \pi_{ri}(k \mid k-1)\sum_{l=1}^{L(q)} b_{rliq}(k)\log_2 b_{rliq}(k) \\ &\quad - \sum_{l=1}^{L(q)}\left(\sum_{i=1}^{n_r} b_{rliq}(k)\pi_{ri}(k \mid k-1)\right) \cdot \log_2\left(\sum_{i=1}^{n_r} b_{rliq}(k)\pi_{ri}(k \mid k-1)\right). \end{aligned} \tag{30}$$

We can also derive the information gain using (23). In this case,

$$H(\underline{\pi}_r(k \mid k-1)) = -\sum_{i=1}^{n_r} \pi_{ri}(k \mid k-1)\log_2 \pi_{ri}(k \mid k-1), \tag{31}$$

$$H(\underline{\pi}_r(k \mid k-1) \mid u_r(k) = q)$$
$$= -\sum_{l=1}^{L(q)} P(y_r(k) = O_{rl}(k) \mid Y_r^{k-1}, u_r(k) = q)$$
$$\cdot \sum_{i=1}^{n} P(x_r(k) = s_{ri} \mid Y_r^{k-1}, u_r(k) = q, y_r(k) = O_{rl}(k))$$
$$\cdot \log_2 P(x_r(k) = s_{ri} \mid Y_r^{k-1}, u_r(k) = q, y_r(k) = O_{rl}(k)). \tag{32}$$

Using the forward algorithm [22],

$$P(x_r(k) = s_{ri} \mid Y_r^{k-1}, u_r(k) = q, y_r(k) = O_{rl}(k)) = \frac{b_{rliq}(k)\pi_{ri}(k \mid k-1)}{\sum_{j=1}^{n_r} b_{rljq}(k)\pi_{rj}(k \mid k-1)} \tag{33}$$

and

$$P(y_r(k) = O_{rl}(k) \mid Y_r^{k-1}, u_r(k) = q) = \sum_{j=1}^{n_r} b_{rljq}(k)\pi_{rj}(k \mid k-1). \tag{34}$$

Inserting (31) and (32) in (30), we get:

$$H(\underline{\pi}_r(k \mid k-1) \mid u_r(k) = q)$$
$$= -\sum_{l=1}^{L(q)} \sum_{i=1}^{n_r} b_{rliq}(k)\pi_{ri}(k \mid k-1) \cdot \log_2 \frac{b_{rliq}(k)\pi_{ri}(k \mid k-1)}{\sum_{j=1}^{n_r} b_{rljq}(k)\pi_{rj}(k \mid k-1)}$$
$$= -\sum_{i=1}^{n_r} \pi_{ri}(k \mid k-1) \sum_{l=1}^{L(q)} b_{rliq}(k) \log_2 b_{rliq}(k)$$
$$- \sum_{i=1}^{n_r} \pi_{ri}(k \mid k-1) \log_2 \pi_{ri}(k \mid k-1)$$
$$+ \sum_{l=1}^{L(q)} \left(\sum_{i=1}^{n_r} b_{rliq}(k)\pi_{ri}(k \mid k-1) \right) \cdot \log_2 \left(\sum_{i=1}^{n_r} b_{rliq}(k)\pi_{ri}(k \mid k-1) \right). \tag{35}$$

Since $\sum_{l=1}^{L(q)} b_{rliq}(k) = 1$ for $i = 1, 2, \ldots, n_r$, combining (35) and (31) indeed gives (30).

REFERENCES

[1] L. E. Baum, T. Petrie, G. Soules and N. Weiss
A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains.
Ann. of Math. Stat, **41** (1970), 164–171.

[2] D. P. Bertsekas
An auction algorithm for shortest paths.
SIAM Journal on Optimization, **1** (1991), 425–447.

[3] D. P. Bertsekas
Auction algorithms for network flow problems: A tutorial introduction.
Computational Optimization and Application, **1** (1992), 7–66.

[4] D. P. Bertsekas
Network optimization: Continuous and discrete models.
Athena Scientific, Belmont MA, 1998.

[5] D. P. Bertsekas and D. A. Castanon
Rollout algorithms for stochastic scheduling problems.
Journal of Heuristics, **5** (1999), 89–108.

[6] D. P. Bertsekas, J. N. Tsitsiklis and C. Wu
Rollout algorithms for combinatorial optimization.
Journal of Heuristics, **3** (1997), 245–262.

[7] M. Brand, N. Oliver and A. Pentland
Coupled hidden Markov models for complex action recognition.
In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, San Juan, Puerto Rico, 1997, 994–999.

[8] S. Chhetri
Sensor scheduling and efficient algorithm implementation for target tracking.
Ph. D. Dissertation, Arizona State University, 2006.

[9] M. Chu, H. Haussecker and F. Zhao
Scalable information-driven sensor querying and routing for Ad Hoc heterogeneous sensor networks.
International Journal of High Performance Computing Applications, **16** (2002), 293–313.

[10] I. Cox and M. Miller
On finding ranked assignments with application to multitarget tracking and motion correspondence.
IEEE Transactions on Aerospace and Electronic Systems, **31** (1995), 486–489.

[11] E. W. Dijkstra
A note on two problems in connexion with graphs.
Numerische Mathematik, **1** (1959), 296–271.

[12] S. Fine, Y. Singer and N. Tishby
The hierarchical hidden Markov model: Analysis and applications.
Machine Learning, **32** (1997), 41–62.

[13] Z. Ghahramani and M. I. Jordan
Factorial hidden Markov models.
Machine Learning, **29** (1997), 245–273.

[14] S. G. Hutchins, S. A. Weil, D. L. Kleinman, S. P. Hocevar, W. G. Kemple, K. Pfeiffer, D. Kennedy, H. Oonk, G. Averett and E. E. Entin
An investigation of ISR coordination and information presentation strategies to support expeditionary strike groups.
12th ICCRTS, Newport, RI, 2007.

[15] R. Jonker and A. Volgenant
A shortest augmenting path algorithm for dense and sparse linear assignment problems.
Computing, **38** (1987), 325–340.

[16] V. Krishnamurthy
Algorithms for optimal scheduling and management of hidden Markov model sensors.
IEEE Transactions on Signal Processing, **50** (2002), 1382–1397.

[17] A. Logothetis and A. Isaksson
On sensor scheduling via information theoretic criteria.
IEEE American Control Conference, San Diego, CA, vol. 4, 1999, 2402–2406.

[18] L. Meier, J. Perschon and R. M. Dressier
Optimal control of measurement subsystems.
IEEE Transactions on Automatic Control, **12** (1967), 528–536.

[19] M. L. Miller, H. S. Stone and I. J. Cox
Optimizing Murty's ranked assignment method.
IEEE Transactions on Aerospace and Electronic Systems, **33** (1997), 851–862.

[20] K. G. Murty
Algorithm for ranking all the assignments in order of increasing cost.
Operations Research, **16** (1968), 682–687.

[21] R. L. Popp, K. R. Pattipati and Y. Bar-Shalom
Dynamically adaptable M-Best 2D assignment and multi-level parallelization.
IEEE Transactions on Aerospace and Electronic Systems, **35** (1999), 1145–1160.

[22] L. R. Rabiner
A tutorial on hidden Markov models and selected applications in speech recognition.
Proceedings of the IEEE, **77** (1989), 257–286.

[23] M. Schneider and C. Chong
A rollout algorithm to coordinate multiple sensor resources to track and discriminate targets.
In *Proceeding of SPIE*, vol. 6235, Orlando, FL, 2006, 62350E.

[24] M. Shakeri, K. R. Pattipati and D. L. Kleinman
Optimal measurement scheduling for state estimation.
IEEE Transactions on Aerospace and Electronic System, **31** (1995), 716–729.

[25] S. Singh, W. Donat, H. Tu, K. Pattipati and P. Willett
Anomaly detection via feature-aided tracking and hidden Markov models.
IEEE Aerospace Conference, Big Sky, MT, 2007, 1–18.

[26] S. S. Singh, N. Kantas, B. Vo, A. Doucet and R. J. Evans
Simulation based optimal sensor scheduling with application to observer trajectory planning.
Technical Report, Cambridge University, 2005.

[27] F. Tu and K. R. Pattipati
Rollout strategies for sequential fault diagnosis.
IEEE Transactions of System, Man and Cybernetics, Part A, **33** (2003), 86–99.

[28] He. Ying and K. P. Chong
Sensor scheduling for target tracking in sensor networks.
IEEE Conference on Decision and Control, **1** (2004), 743–748.

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

Woosun An is currently a Ph.D. student in electrical and computer engineering at the University of Connecticut (UConn). He received his B.S. degree in electrical engineering from the University of Chung Ang, South Korea in 2001. He received his M.S. degree in electrical engineering from the University of Florida in 2005.

He received UConn engineering summer fellowship award in 2007 and 2008. His research interests include dynamic sensor scheduling, graphical models, and optimization theory.

Satnam Singh is currently a Ph.D. student in electrical and computer engineering at the University of Connecticut (UConn). He received his B.S. degree in chemical engineering from the Indian Institute of Technology (IIT) Rookree, India in 1996. He received his M.S. degree in electrical engineering from the University of Wyoming in 2003.

Prior to his graduate studies, he worked as a design engineer in Engineers India Limited, New Delhi, India for 4 years. He is a reviewer for IEEE Potentials magazine for the past three years and he has also served as the President of IEEE Student Branch at UConn during 2003–04. He received UConn engineering summer fellowship award in 2006. His research interests include data mining, graphical models, pattern recognition, fault detection and diagnosis, and optimization theory.

Krishna R. Pattipati (S'77—M'80—SM'91—F'95) received the B.Tech. degree in electrical engineering with highest honors from the Indian Institute of Technology, Kharagpur, in 1975, and the M.S. and Ph.D. degrees in systems engineering from the University of Connecticut in 1977 and 1980, respectively.

From 1980–1986 he was employed by ALPHATECH, Inc., Burlington, MA. Since 1986, he has been with the University of Connecticut, where he is a Professor of Electrical and Computer Engineering. His current research interests are in the areas of adaptive organizations for dynamic and uncertain environments, multi-user detection in wireless communications, signal processing and diagnosis techniques for complex system monitoring, and multi-object tracking. Dr. Pattipati has published over 370 articles, primarily in the application of systems theory and optimization (continuous and discrete) techniques to large-scale systems. He has served as a consultant to Alphatech, Inc., Aptima, Inc., and IBM Research and Development. He is a cofounder of Qualtech Systems, Inc., a small business specializing in intelligent diagnostic software tools.

Dr. Pattipati was selected by the IEEE Systems, Man, and Cybernetics Society as the Outstanding Young Engineer of 1984, and received the Centennial Key to the Future award. He was elected a Fellow of the IEEE in 1995 for his contributions to discrete-optimization algorithms for large-scale systems and team decision making. Dr. Pattipati has served as the Editor-in-Chief of the IEEE Transactions on SMC: Part B—Cybernetics during 1998–2001, Vice-President for Technical Activities of the IEEE SMC Society (1998–1999), and as Vice-President for Conferences and Meetings of the IEEE SMC Society (2000–2001). He was corecipient of the Andrew P. Sage award for the Best SMC Transactions Paper for 1999, Barry Carlton award for the Best AES Transactions Paper for 2000, the 2002 and 2008 NASA Space Act Awards for "A Comprehensive Toolset for Model-based Health Monitoring and Diagnosis," the 2003 AAUP Research Excellence Award and the 2005 School of Engineering Teaching Excellence Award at the University of Connecticut. He also won the best technical paper awards at the 1985, 1990, 1994, 2002, 2004 and 2005 IEEE AUTOTEST Conferences, and at the 1997 and 2004 Command and Control Conferences.

David L. Kleinman (S'62—M'68—SM'93—F'94) received his B.E.E. degree from The Cooper Union, New York, in 1962 followed by M.S. and Ph.D. degrees in electrical engineering (control systems) from MIT in 1963 and 1967, respectively.

He works part-time as a research professor in the Information Sciences Department at the Naval Postgraduate School in Monterey, CA, and is also professor emeritus in the Electrical and Computer Engineering Department at the University of Connecticut. His current research is in the areas of team decision-making/coordination in dynamic task environments, and structural adaptation processes in human organizations. From 1967 to 1971, he worked at Bolt Beranek and Newman, Cambridge, MA where he pioneered in the application of modern control and estimation theory to develop and validate an analytical model for describing human control and information processing performance in manned vehicle systems. From 1971–73 he directed the East Coast office of Systems Control, Inc., where he led applied research projects in both manual and automatic control. From 1973–1994, while a Professor at UConn, he directed the CYBERLAB, a laboratory for empirical research in cybernetic systems. In 1993 Dr. Kleinman was named an IEEE Fellow for his work in human modeling. He has authored or coauthored over 200 journal and conference papers, many of which are in the area of military Command and Control.

Dr. Kleinman has been a Vice-President of Finance and Long-range Planning, for the IEEE-SMC Society. He was the Program Chairman for the 1989 IEEE International Conference on Systems, Man, and Cybernetics, and for the 1990 JDL Symposium on Command and Control Research.

Dr. Kleinman was a cofounder of Alphatech, Inc., Qualtech Systems, Inc., and Aptima, Inc. He continues to serve on the Board of Directors of the latter two companies.

Swapna S. Gokhale received her B.E. (Hons.) in electrical and electronics engineering, and computer science from the Birla Institute of Technology and Science, Pilani, India in 1994, and her M.S. and Ph.D. in electrical and computer engineering from Duke University in 1996 and 1998 respectively.

She is currently an assistant professor in the Department of Computer Science and Engineering at the University of Connecticut. Prior to joining UConn, she was a research scientist at Telcordia Technologies (formerly Bell Communications Research). Her dissertation research was in the area of software reliability and performance. Prior to joining Telcordia, she was a Post Graduate Researcher at the University of California, Riverside. Her research interests include software testing and reliability, Quality of Service issues in wireless and wireline networks and Voice-over-IP. She received the Best Paper awards at Intl. Symposium on Software Reliability Engineering (ISSRE) in 1998 and at Intl. Conference on Reliability and Safety Engineering (INCRESE) in 2005. She received a CAREER award from the National Science Foundation (NSF) to conduct research in the area of architecture-based software reliability assessment in January 2007.