Proceedings of the
44th IEEE Conference on Decision and Control, and
the European Control Conference 2005
Seville, Spain, December 12-15, 2005

Stochastic Control Bounds on Sensor Network Performance

David A. Castañón

Abstract— **Consider a network of sensors, each of which has limited sensing resources, which is tasked with collecting noisy classification information on objects. The amount of resources required a given sensor to measure an object depends on the specific sensor-object geometry. Sensors exchange collected information to estimate object identities and coordinate which measurements to collect. This paper describes a computable lower bound on the classification error that can be achieved by a causal adaptive sensing schedule. This bound is based on solving a partially observed stochastic control problem. Expanding the admissible control space of this problem leads to a relaxed problem with simpler decision structure for which the bounds can be computed. The bound computations are illustrated for examples involving 100 unknown objects, and compared with the Monte Carlo performance of specific scheduling algorithms. These comparisons illustrate the tightness of the bounds.**

I. INTRODUCTION

There are many recent applications for networks of sensors, each of which has a given amount of resources, such as available power or duty cycle. Often, each sensor has multiple sensing modes that it can use to collect different types of information; the amount of resources required to collect a measurement by a sensor depends on the specific sensor-object geometry and the mode used. The network is tasked with using its available resources to obtain information on a given number of objects. In order to achieve the best information possible, it is important to coordinate the allocation and scheduling of the different sensors and sensor modes across objects. Sensors exchange collected information to determine the current state of information on objects. The adaptive sensing problem consists of selecting and scheduling the sensor modes which are applied to objects of interest based on the collected past information.

This paper develops a model for a class of adaptive sensing problems involving the objective of classifying a known number of objects of unknown types at known locations, given a fixed number of sensor with finite resources and finite modes. We assume that sensor performance parameters are time-invariant. This class of problems arises in several applications, from object classification using multiple airborne platforms, dynamic search, and fault inspection and isolation in manufacturing systems. In these applications, individual measurements provide noisy estimates of object type whose quality depends on the specific mode used by the sensor. This noisy information can be used to prioritize which objects to look at next, from which sensor, and to assign appropriate sensor modes to the objects.

This work was supported in part by grants NSF DMI-0330171 and DARPA F33615-02-C-1197

Dept. Electrical & Computer Eng., Boston University, Boston, MA
dac@bu.edu

Because of the uncertain nature of the underlying object types and the adaptive nature of the desired schedules, adaptive sensing problems can be formulated as partially observed Markov decision problems (POMDP) [1], [2], [9]. As such, this class of problems can be solved using stochastic dynamic programming [3]. For large numbers of objects, the required state space is very high-dimensional, consisting of the conditional probability distributions of all of the objects. This leads to intractable computational problems, even with the fastest POMDP algorithms.

Sensing problems have been formulated previously as dynamic optimization problems with partial information [6], [14]. These formulations restrict the sensors to a single sensor with a single mode. Because of the complexity of these problems, most practical algorithms are heuristic algorithms based on information-theoretic metrics [5]. To date, there has been no effective approach that can characterize the achievable adaptive sensing performance performance to determine whether such heuristic algorithms are performing well.

In this paper, we consider sensing problems involving multiple distributed sensors with multiple modes per sensor. This model is an extension of the model discussed in [7]. We show that the resulting POMDP models admit a lower bound on classification error performance based on modifying the constraint structure to expand the space of admissible strategies. The resulting problem becomes a dynamic optimization problem subject to expected value constraints, a class of problems recently studied by in [20]. We develop a hierarchical algorithm that solves this problem efficiently through decomposition into single object POMDPs. The hierarchical algorithm avoids the exponential growth of the dimensions of the resulting state space in the POMDP problem as a function of the number of objects.

The paper includes several examples where the lower bound performance is computed, and compared with the Monte Carlo performance achieved by suboptimal SM algorithms. In particular, we compute bounds for a special problem for which the optimal sensing strategy is known, and compare the bounds to the optimal performance to show how tight the bounds are.

II. PROBLEM FORMULATION

Assume that there are N objects of interest in the problem, with known locations. Each object can belong to one and only one of K different classes, and the object identity does not change over time. Let $x_i \in X \equiv \{1, \ldots, K\}$ denote the true class of object i. We define the complete (but unknown) system state as:

$$\underline{x}^T = \begin{pmatrix} x_1 & x_2 & \cdots & x_N \end{pmatrix} \tag{1}$$

Since the identities do not change over time, the complete system state is constant over time. We assume that x_i are independent random variables with values in the finite space X. Associated with each object i is a prior probability vector $\pi_i(0)$ which describes the probability distribution of the random variable x_i. That is,

$$\pi_{ij}(0) = \text{Prob}\{x_i = j\} \tag{2}$$

To obtain information about the state of each object, selected objects are examined with different modes from different sensors. In order to simplify the notation in the exposition, we consider the case of a single sensor with multiple modes $m \in \{1, \ldots, M\}$. Using a sensor mode m on object i produces an observable y_m in a finite set Y_m, with a conditional probability distribution that depends only on the object i, its type x_i and the mode m, denoted by $p(y_m|i, x_i, m)$. We assume that the observation outcomes of these sensing actions are conditionally independent of each other given the object types.

Obtaining a measurement of object i with mode m requires sensor resources $R_{im} > 0$ (e.g. power), which depend on the object location, sensor location and specific mode selected. The sensor has a finite amount of sensor resources R. The objective is to classify, with minimal error cost, the objects after the sensor resource R is exhausted.

Without loss of generality, we restrict our attention to sensing policies that execute only one action at a time. Such strategies are optimal in that they provide maximal information for adaptation, and will achieve minimal error cost under the assumption of time invariant performance. Let $u(k) = (i(k), m(k))$ denote the $k+1$-th action (starting at $k = 0$) taken by the sensor, consisting of measuring object $i(k)$ with mode $m(k)$. Let U denote the set of possible sensor actions, and let $y_{m(k)}(k)$ denote the measured value resulting from action $u(k) \in U$. The past information available to adaptively select $u(k)$ is $I(k) = \{u(0), y_{m(0)}(0), \ldots, u(k-1), y_{m(k-1)}(k-1)\}$. The sensing problem decisions are selected adaptively until a final random stopping instance T, selected based on the information $I(T)$. At the end of this stopping instance, the information $I(T)$ is used to select a final classification decision $v_i \in X$ for each object, based on $I(T)$, to minimize the expected classification cost.

An admissible adaptive sensing policy is a set of measurable feedback policies $\{\gamma(0), \ldots, \gamma(T)\}$ and stopping time T such that

$$\begin{aligned} &\gamma(k) : I(k) \to U, \ k < T \\ &T : I(T) \to \{\mathit{stop}, \mathit{continue}\} \\ &\gamma(T) : I(T) \to X^N \end{aligned} \tag{3}$$

Let Γ denote the set of all admissible sensing policies. Since the observation space is finite and the decision space is also finite, Γ is a countable space.

Denote by $c(v, x)$ the cost of selecting classification decision v when the true object type is x. The adaptive sensing problem is to minimize the expected total classification cost

$$J(\gamma) = E_\gamma\{\sum_{i=1}^{N} c(v_i, x_i)\} \tag{4}$$

over adaptive sensing policies $\gamma \in \Gamma$ satisfying the resource utilization constraint

$$\sum_{k=0}^{T-1} R(u(k)) \leq R \tag{5}$$

with the notation $R(u(k)) \equiv R_{i(k)m(k)}$.

The constraint in (5) is a sample path constraint; for every realization of the information sets $I(k)$, the adaptive policy γ must not exceed the total sensor resources available. Finite observation sets and decision spaces imply that there is only a finite number of possible admissible sensing policies that satisfy the constraint (5).

The above problem is a class of finite-state, finite-observation POMDP studied in [1], [2], [9], with the special structure that the underlying state dynamics are constant, and decisions are constrained by the sample path constraints of (5). Such problem scan be transformed into fully-observed MDPs in terms of a sufficient statistic: the conditional probability distribution of the state $\underline{x}$ given information $I(k)$, denoted as $P(\underline{x}|I(k)) \in S_N$. The recursive evolution of this information state in response to an action $u(k) = (i(k), m(k))$ can be described by Bayes' rule as

$$\begin{aligned} P(\underline{x}|I(k+1)) &= P(\underline{x}|I(k), u(k), y_{m(k)}(k)) \\ &= \frac{P(y_{m(k)}(k)|x_{i(k)}, m(k))P(\underline{x}|I(k))}{P(y_{m(k)}(k))|I(k), u(k))} \end{aligned} \tag{6}$$

with the initial condition

$$P(\underline{x}|I(0)) = \prod_{i=1}^{N} \pi_i(0) \tag{7}$$

Under the previous independence assumptions, the following lemma establishes a convenient representation:

Lemma 2.1: Under the adaptive sensing problem assumptions, the conditional probability $P(\underline{x}|I(k))$ can be factored as

$$P(\underline{x}|I(k)) = \prod_{i=1}^{N} P(x_i|I(k)) \tag{8}$$

where the evolution of $P(x_i|I(k))$ under sensing action $u(k) = (i(k), m(k))$ and observation $y_{m(k)}(k)$ is given by

$$P(x_i|I(k+1)) = \begin{cases} P(x_i|I(k)) & \text{if } i(k) \neq i \\ \frac{P(y_{m(k)}(k)|x_{i(k)}, m(k))P(x_i|I(k))}{\sum_{j=1}^{K} P(y_{m(k)}(k)|x_i=j, I(k))P(x_i=j|I(k))} & \text{otherwise} \end{cases} \tag{9}$$

The proof of this lemma is straightforward by induction.

Define $\pi_i(k) = P(x_i|I(k))$ to be the conditional probability distribution of x_i given information $I(k)$, with components $\pi_{ij}(k) = P(x_i = j|I(k))$. Lemma 2.1 establishes that the conditional probability distribution of the entire state, $P(\underline{x}|I(k))$, can be computed as the product of $\pi_i(k), i = 1, \ldots, N$. Define the information vector $\vec{\pi} = \begin{pmatrix} \pi_1^T & \ldots & \pi_N^T \end{pmatrix}^T$ For a given observation y_m using mode m on object index i, define the observation probability matrix as the $K \times K$ diagonal matrix

$$B_i(y_m) = \text{diag}\{P(y_m|x_i = 1, m), \ldots, P(y_m|x_i = K, m)\}$$

The information vector evolves in response to a measurement y_m obtained from a sensing action (i, m) according to an operator T, where

$$T(\vec{\pi}, u = (i', m), y_m) = \begin{pmatrix} T_1(\pi_1, u = (i', m), y_m) \\ \vdots \\ T_n(\pi_N, u = (i', m), y_m) \end{pmatrix}$$

$$T_i(\pi_i, u = (i', m), y_m) = \begin{cases} \pi_i & \text{if } i \neq i' \\ \frac{B_i(y_m)\pi_i}{\vec{e}^T B_i(y_m)\pi_i} & \text{if } i = i' \end{cases}$$

where $\vec{e}$ is a K-dimensional vector of all ones.

The adaptive sensing problem described above can be solved by stochastic dynamic programming [3]. The resource constraint in (5) can be incorporated into the dynamics to obtain a dynamic programming recursion, as in [20]. The value function $V(\vec{\pi}, C)$ is the optimal solution of (3)-(5) when the initial information is $\vec{\pi}$ and the available sensor resource level is $R = C$ satisfies the following Bellman's equation:

$$V(\vec{\pi}, R) = \min[\sum_{i'=1}^{N} \min_{v_{i'} \in X} \sum_{j=1,\ldots,K} c(v_{i'}, j)\pi_{i'j}, \\ \min_{(i,m)\in U(R)} E_{y_m}\{V(T(\vec{\pi}, (i, m), y_m), R - R_{im})\}] \quad (10)$$

where $U(R) \subset U$ is the set of feasible sensor actions (i, m) such that $R_{im} \leq R$, and

$$E_{y_m}\{V(T(\vec{\pi}, (i, m), y_m), R - R_{im})\} = \\ \sum_{y_m \in Y_m} P(y_m|I(k), (i, m))V(T(\vec{\pi}, u, y_m), R - R_{im}) \\ = \sum_{y_m \in Y_m} e^T B_i(y_m)\pi_i V(T(\vec{\pi}, (i, m), y_m), R - R_{im}) \quad (11)$$

This recursion starts from the following boundary conditions: Let $R_{min} = \min_{i,m} R_{im}$. Then, the set of admissible modes $U(R)$ is empty for $R < R_{min}$. Thus,

$$V(\vec{\pi}, R) = \sum_{i=1}^{N} \min_{v_i \in X} \sum_{j=1,\ldots,K} c(v_i, j)\pi_{ij} \quad \text{if } R < R_{min} \quad (12)$$

Eqs. (10)-(12) can be used recursively to compute the optimal value for all information states and nonnegative levels.

The initialization of the recursion decouples into N independent optimizations. However, the recursion (10) does not preserve this decomposability. The coupling arises primarily because of the resource use constraints in (5); the decision of which object to view and which mode to use depends on the information vector of all the objects and the available resources. Thus, the dynamic programming induction must be carried out for the entire state $\vec{\pi}(t)$, which becomes a formidable problem even for small numbers of objects.

III. RELAXED FORMULATION AND BOUNDS

We relax the sample path sensor resource use constraints (5) and use an averaged version of the same constraints, as

$$E\{\sum_{k=1}^{T} R(u(k))\} \leq R \quad (13)$$

This replaces a large set of constraints by a single aggregate constraint. Sensing policies that satisfy (5) will also satisfy (13). Let J^* and J_A^* denote the optimal classification cost in (3)-(4) with constraints (5) and (13), respectively.

Lemma 3.1: $J^* \geq J_A^*$

Let $\lambda \geq 0$ denote a Lagrange multiplier for (13). For any admissible policies in Γ, consider the objective

$$J(\lambda, \gamma) = E_\gamma\{\sum_{i=1}^{N} c(v_i, x_i)\} + \lambda[E_\gamma\{\sum_{k=0}^{T-1} R(u(k))\} - R] \quad (14)$$

Consider minimizing (14) for fixed $\lambda \geq 0$. Denote by $J^*(\lambda)$ the optimal value of (14) over all adaptive sensing policies $\gamma \in \Gamma$. Then,

Lemma 3.2: For all values of $\lambda \geq 0$, $J^* \geq J_A^* \geq J^*(\lambda)$. In particular, $J^* \geq \sup_{\lambda \geq 0} J^*(\lambda)$.

Lemma 3.2 is a consequence of weak duality in nonlinear programming [4]. Since the number of adaptive sensing policies that satisfy (14) is finite, computation of J_A^* is an integer programming problem, and computation of $\sup_{\lambda \geq 0} J^*(\lambda)$ is its dual problem.

The key issue is computing the lower bounds $J^*(\lambda)$ efficiently. Rewrite (14) for $\gamma \in \Gamma$ as

$$J(\lambda, \gamma) = E_\gamma\{\sum_{i=1}^{N}[c(v_i, x_i) + \lambda \sum_{k=0}^{T-1} R(u(k))\delta_{i(k)-i}]\} - \lambda R \quad (15)$$

where the indicator function $\delta_i = 1$ if $i = 0$, and 0 otherwise. This suggests that optimization of $J(\lambda, \gamma)$ may be separable across individual objects i.

Partition the information $I(k)$ into disjoint sets $I_i(k)$, where $I_i(k)$ are the sensing actions and measurement actions applied to object i:

$$I_i(k) = \{(u(j), y(j))|j < k, i(j) = i\} \quad (16)$$

Note that the conditional probability vector π_i only changes on measurements included in $I_i(k)$. We wish to restrict the set of adaptive sensing policies to a subset where the decision to apply a sensor action for object i depends only on the information previously collected for object i. We refer to this subset of policies as adaptive *local* sensing policies, defined as:

Definition 3.1: An adaptive *local* sensing policy is an adaptive sensing policy γ and stopping times $T_i, i = 1, \ldots, N$, with the properties that, for each sensing action instance k,

1) If $u(k) = (i(k), m(k))$, then $i(k) = k \mod N + 1$.
2) The selected sensor mode $m(k)$ depends only on the information $I_{i(k)}$.
3) For each object i, there is a stopping time T_i which depends only on $I_i(T_i)$ such that, for all $k \geq T_i$, if $i = k \mod N + 1$, no sensing action is taken. If $k < T_i$ and $i = k \mod N + 1$, then $u(k) = (i, m)$ for some mode m in $\{1, \ldots, \mathrm{M}\}$.
4) At time T_i, the local decision v_i for object i is selected as a function of $I_i(T_i)$.

Adaptive local sensing policies use a fixed round-robin schedule for selecting which objects to measure, and the choice of sensing mode, stopping time and final classification on each object depends only on the prior information collected on that object. The *effective* stopping time of an adaptive local sensing policy is the earliest time at which every object has a final classification decision. Adaptive local sensing policies are a subset of adaptive sensing policies.

Let Γ_L denote the set of adaptive local sensing policies. Given sensor resources R, there are a finite number of feasible adaptive local sensing policies. Thus, Γ_L is a countable discrete set. For the purposes of bound computation, we will include *mixed* policies:

Definition 3.2: A *mixed* local sensing policy is a probability distribution $q(\gamma)$ over Γ_L such that local SM policy γ is selected for use with probability $q(\gamma)$. The set of mixed local sensing policies is denoted by $Q(\Gamma_L)$.

Consider the problem of minimizing the relaxed cost (15) over local sensing policies Γ_L. Since $\Gamma_L \subset \Gamma$, we have

$$\min_{\gamma \in \Gamma} J(\lambda, \gamma) \leq \min_{\gamma \in \Gamma_L} J(\lambda, \gamma) \tag{17}$$

Furthermore, since (15) is an unconstrained objective, the minimum in mixed local sensing policies is achieved by a pure local sensing policy, so

$$\min_{\gamma \in \Gamma} J(\lambda, \gamma) \leq \min_{q \in Q(\Gamma_L)} \sum_{\gamma \in \Gamma_L} q(\gamma) J(\lambda, \gamma) \tag{18}$$

The importance of mixed local sensing policies is highlighted in the theorem below, proven in [21]:

Theorem 3.1: For any admissible adaptive sensing policy $\gamma \in \Gamma$, there exists a mixed local sensing policy $q \in Q(\Gamma_L)$ such that the expected classification costs in (4) and the expected total resource use in (13) are equal under both policies γ and q.

This result implies the following inequality:

$$\min_{\gamma \in \Gamma} J(\lambda, \gamma) \geq \min_{q \in Q(\Gamma_L)} \sum_{\gamma \in \Gamma_L} q(\gamma) J(\lambda, \gamma) \tag{19}$$

Combining (18) and (19) yields the following:

$$\min_{\gamma \in \Gamma} J(\lambda, \gamma) = \min_{q \in Q(\Gamma_L)} \sum_{\gamma \in \Gamma_L} q(\gamma) J(\lambda, \gamma) = \min_{\gamma \in \Gamma_L} J(\lambda, \gamma) \tag{20}$$

Eq. (20) implies that lower bounds for the achievable classification performance can be computed by optimizing over local sensing policies only. For each local policy $\gamma \in \Gamma_L$, let γ_i denote the policy that is used for instances k when actions are taken for object i, and let Γ_{L_i} be the set of such admissible local policies for object i. Thus, γ_i selects actions for object i based on past observations $I_i(k)$, and selects a stopping time T_i and a final classification v_i at that stopping time. The importance of local sensing policies is that the optimization in (20) decouples over objects as

$$\min_{\gamma \in \Gamma_L} J(\lambda, \gamma) = \sum_i \min_{\gamma_i \in \Gamma_{L_i}} J_i(\lambda, \gamma_i) - \lambda R \tag{21}$$

$$J_i(\lambda, \gamma_i) = E_{\gamma_i}\{c(v_i, x_i) + \lambda \sum_{k=0}^{T-1} R(u(k)) \delta_{i(k)-i}\} \tag{22}$$

This implies that computation of the bounds can be achieved with N independent optimization problems for each value of λ. Furthermore, Lemma 3.2 yields

$$J^* \geq \sup_{\lambda \geq 0} \min_{\gamma \in \Gamma_L} J(\lambda, \gamma) \tag{23}$$

Note that the right hand side of (23) is the dual of the following linear programming problem:

$$\min_{q \in Q(\Gamma_L)} \sum_{\gamma \in \Gamma_L} q(\gamma) E_\gamma J(\gamma) \tag{24}$$

subject to

$$\sum_{\gamma \in \Gamma_L} q(\gamma) E_\gamma [\sum_{k=0^{T-1}} R(u(k))] \leq R \tag{25}$$

$$q(\gamma) \geq 0 \ \forall \ \gamma \in \Gamma_L; \quad \sum_{\gamma \in \Gamma_L} q(\gamma) = 1 \tag{26}$$

which is a linear program over the choice of probability distributions $q \in Q(\Gamma_L)$, subject to two constraints, which implies that the optimal mixed local sensing policy q will have support only on two pure local sensing policies.

IV. BOUND ALGORITHMS

There are two potential approaches to compute a lower bound: a dual approach, based on Lagrangian relaxation [13], that optimizes (23) over the choice of dual variable λ, and a primal approach based on solving the linear program (24)-(26). The dual approach is straightforward, and uses techniques from nondifferentiable optimization [16] to search the space of possible λ. The primal approach optimizes over a very large space of of mixture probabilities q. However, this mixture has very sparse support, which makes it ideal for column generation algorithms [15].

A fundamental step in either approach is the computation of the optimal local sensing policies given λ for each object i:

$$\min_{\gamma_i} E_{\gamma_i}[c(v_i, x_i) + \lambda \sum_{k \geq 0 : i = k \bmod N+1}^{T_i - 1} R(u(k))] \tag{27}$$

This problem is a single object POMDP, with sufficient statistic given by the marginal probability distribution $\pi_i(k)$. One can reduce the action instants to a new counter k' indexing only the action opportunities for object i, to obtain

$$\min_{\gamma_i \in \Gamma_{Li}} E_{\gamma_i}[c(v_i, x_i) + \lambda \sum_{k'}^{T_i'-1} R_{im(k')}] \tag{28}$$

The resulting POMDP problems are small enough to solve using existing algorithms such as those overviewed in [2], [9], [11], [12].

Solution of the N decoupled problems (28) yields a local policy $\gamma \in \Gamma_L$, for which the expected classification cost $E_\gamma[\sum_{i=1}^{N} c(v_i, x_i)]$ and expected resource use $E_\gamma[\sum_{k=0}^{T-1} R(u(k))]$ are computed from the solution. This provides the starting point for the use of column generation [15] for solution of (24)-(26), similar to the approaches used for MDPs and POMDPs in [18], [19], [8]. Solving the linear program in (24)-(26) restricted to mixtures of the $d = 1, \ldots, D$ initial policies yields an upper bound J^{UB} to the optimal cost, and an optimal dual price λ_D for the resource constraint (26). The constraint generation algorithm uses this optimal dual price value in (28) to generate a new candidate local policy γ^{D+1},by solving N independent POMDP problems with this value of λ. The combined solution of the N subproblems also provides a lower bound J^{LB} on the optimal performance, as described in Lemma 3.2. The key result in the constraint generation algorithm is stated as follows [15]:

Lemma 4.1: Consider the pure local policy generated by the solution of (28). If $J^{LB} = J^{UB}$, the optimal solution over all mixtures of local policies is a mixture of the local policies indexed by $d = 1, \ldots, D$. Otherwise, the pure local policy γ^{D+1} can be used as part of a mixed policy which provides a cost lower than J^{UB}.

V. EXTENSION TO MULTIPLE SENSORS

The development of the previous sections carries through with little modification when multiple sensors are used. The key difference is that there is a separate resource constraint for each sensor. Thus, there will be a vector of sensor resources R_s, where s is a sensor index, thus resulting in a vector of averaged constraints (13). The Lagrange multipliers λ will thus be vectors instead of scalars. Nevertheless, all of the lemmas and theorems can be extended to the multisensor case with minor modifications.

The main assumption that was used in the single sensor formulation was that only one sensor action would be performed simultaneously. This assumption is still used for the multiple sensor problem to derive the lower bound, although the results in the previous section indicate that optimal local sensing strategies that achieve the lower bound may use simultaneous sensing by multiple sensors.

VI. EXAMPLES

For our first example, we consider a case where the optimal strategies are known [22]. There are 100 unknown objects with one of two types, with equal priors for each object and one sensor with a single mode. Measurement outcomes are binary-valued, identifying one of the two types, with a symmetric probability of error P_e. The objective is to minimize the expected number of classification errors after N measurements. The optimal strategy derived in [22] is to assign the next measurement to the object with conditional probability with greatest entropy.

Table I shows the results of 1000 Monte Carlo simulations of the optimal strategy, compared with the predicted performance of the lower bound, in terms of expected number of classification errors for 3 different conditions of symmetric single measurement P_e and four levels of number of measurements N. As the table indicates, the bound predictions are very tight for this case. The gap between bound and optimal strategy increases slightly with the number of measurements N because the likelihood of errors decreases, and the bound strategy allows the use of more resources than available in unlikely cases.

	$P_e = 0.25$		$P_e = 0.2$		$P_e = 0.15$	
N	Bound	Opt.	Bound	Opt.	Bound	Opt.
100	25	25.03	20	20.02	15	15.067
200	18.182	18.185	12.727	12.765	7.888	7.988
300	11.364	11.432	5.749	6.038	2.518	2.593
400	7.833	7.905	3.468	3.543	0.927	0.987

TABLE I

COMPARISON OF EXPECTED NUMBER OF ERRORS BY LOWER BOUND AND MONTE CARLO OF OPTIMAL STRATEGY

For the second set of experiments, we consider another 100 object scenario where objects can be of three different types ($K = 3$): cars, trucks and military vehicles (MV). There is a single sensor, with two modes: a low resolution mode 1 that takes 1 second per image ($R_{i1} = 1$), and a high resolution mode 2 that requires 5 seconds per image, ($R_{i2} = 5$). Low resolution imagery is useful in separating cars from trucks and MVs, but separating trucks from MVs requires high resolution imagery. Apriori, each object has a probability of 0.1 as a military vehicle, 0.2 truck and 0.7 car. Imagery generated by the sensor is processed into a binary decision as to whether the object is MV or not. Hence $y_{ij} \in \{0, 1\}$, where 1 indicates that the decision is MV.

	low-resolution		high-resolution	
Type	$y = 0$	$y = 1$	$y = 0$	$y = 1$
Car	0.9	0.1	0.95	0.05
Truck	0.1	0.9	0.85	0.15
MV	0.1	0.9	0.8	0.2

TABLE II

MEASUREMENT LIKELIHOODS FOR DIFFERENT MODES

The objective of the problem is to determine as accurately as possible which objects are military vehicles (type 1). Thus, the classification costs are given by $c(v_i, x_i)$ as a 3×3 matrix where v_i is the row index:

$$(c(v_i, x_i)) = \begin{pmatrix} 0 & MD & MD \\ FA & 0 & 0 \\ FA & 0 & 0 \end{pmatrix} \qquad (29)$$

where $FA = 1$ and MD will vary in the experiments. The conditional probability distributions $p(y|x, m)$ are given in Table II. R seconds of sensor time can be used before all objects need to be classified. This number will also be varied from 300 seconds to 700 seconds.

We compare the bound with the performance of two adaptive SM algorithms: a variation of Kastella's discrimination gain (DG) algorithm [5], which selects the best sensor mode and target on the basis of maximizing the expected entropy reduction in the distribution of object type per unit sensor resource applied, and a dynamic scheduling algorithm (ADP) based on Lagrangian relaxation and POMDP approximations described in [7]. Each algorithm was simulated for 100 independent Monte Carlo runs using the same measurement outcomes to evaluate its average performance for three different levels of sensor resources: 300, 500 and 700 seconds. Table III includes the results for 300 and 700 seconds for 5 levels of missed detection (MD) costs. The bound is close to the more complex ADP algorithm, and far from the DG algorithm as MD increases.

	700 Seconds			300 Seconds		
MD	Bound	ADP	DG	Bound	ADP	DG
1	1.6	1.58	1.91	4.61	9.7	9.17
5	4.5	4.46	6.75	15.66	17.03	18.62
10	6.5	6.49	9.87	19.56	21.18	20.71
20	8	8.25	14.87	21.67	22.38	22.11
40	10	10.01	23.05	24.18	24.53	24.91

TABLE III

PERFORMANCE OF SCHEDULING ALGORITHMS VS. BOUND

Figure 1 shows the results for the two algorithms and the lower bound for 500 seconds of sensing. The results show that the ADP policy approaches the bound in performance.

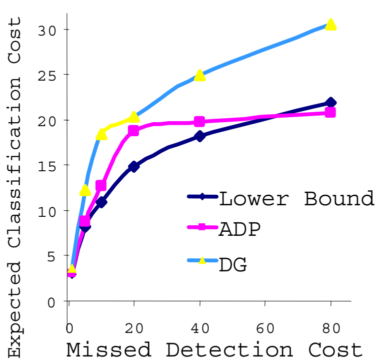

Fig. 1. Monte Carlo performance of algorithms and lower bound for 500 seconds of sensor resource.

VII. DISCUSSION

In this paper, we have presented a lower bound for the achievable classification performance for a network of sensors with finite sensing resources. The approach is based on a POMDP approximation of the formulation of the adaptive sensing problem that can be solved efficiently. We presented experimental results that compared the lower bound with the performance of two suboptimal adaptive sensing algorithms available in the literature. The experimental results established that the bound is tight in that the performance of suboptimal algorithms is close to the predicted performance of the bound for several conditions.

For sensor networks, the bound in this paper neglects the cost of communications as compared to the cost of active sensing. This is the case when sensors are in near vicinity of each other, and sensing requires active emissions by the sensors, so that the two-directional path loss is significant. In situations where communications also consume significant number of resources, the bound is optimistic, and would not be a good prediction for sensor network performance.

REFERENCES

[1] R. D. Smallwood and E. J. Sondik, "The optimal control of partially observable Markov processes over a finite horizon" *Op. Res.,* V. 21, p 1071-1088, 1973.

[2] G. E. Monahan, "A survey of partially observable Markov decision processes: Theory, models and algorithms," *Mgmt. Sci.,* V. 28, p1-16, Jan. 1982.

[3] D. P. Bertsekas, *Dynamic Programming and Optimal Control,* Vols. I-II, Athena Scientific, Belmont, MA 1995.

[4] D. P. Bertsekas, *Nonlinear Programming*, Athena Scientific, Belmont, MA, 1999.

[5] K. Kastella, "Discrimination Gain to Optimize Detection and Classification,"*IEEE Trans. on Systems, Man and Cybernetics, Part A,* V. 27, No. 1, Jan. 1977.

[6] D. A. Castañón, "Optimal search strategies for dynamic hypothesis testing," *IEEE Trans. Sys., Man & Cybernetics,* v. 25, 1995.

[7] D. A. Castañón, "Approximate Dynamic Programming for Sensor Management," *Proc. 36th IEEE Conference on Decision and Control,* San Diego, CA, December 1997.

[8] D. A. Castañón and J. M Wohletz, "Model Predictive Control for Unreliable Dynamic Task Assignment," *Proc. 2002 Conf. Decision and Control,* Las Vegas, NV, Dec. 2002.

[9] W. S. Lovejoy, "A survey of algorithmic methods for partially observable Markov decision processes," *Annals of Operations Research,* v. 28, 1991.

[10] M. L. Littman, A. R. Cassandra and L. Pack-Kaelbling, "Efficient dynamic programming updates in partially observable Markov decision processes," working paper, Brown University, Dec. 1995.

[11] A. R. Cassandra, *Exact and Approximate Algorithms for Markov Decision Processes,* Ph. D. Diss., Brown Univ., Providence, RI 1998.

[12] A. R. Cassandra, M. L. Littman and N. L. Zhang, "Incremental Pruning: A Simple, Fast Exact Method for Partially Observed Markov Decision Processes," *Proc. 13th Conf. Uncertainty in Artificial Intelligence,* Providence, RI 1997.

[13] A. M. Geoffrion, "Lagrangian relaxation for integer programming," *Math. Prog. Studies,* v. 2, 1974.

[14] V. Krishnamurthy and R. J. Evans, "Hidden Markov Model Multiarm Bandits: A Methodology for Beam Scheduling in Multitarget Tracking," *IEEE Trans. Signal Processing,* V. 49, N. 12, Dec. 2001.

[15] P. C. Gilmore and R. E. Gomory, "A Linear Programming Approach to the Cutting Stock Problem" *Operations Research,* V. 9, 1961.

[16] V. M. Demyanov and L. V. Vasilev, *Nondifferentiable Optimization,* Optim. Software, New York 1985.

[17] S. J. Benkoski, M. G. Monticino, and J. R. Weisinger, "A Survey of the Search Theory Literature," *Naval Research Logistics,* Vol. 38, No. 4, 1991, pp. 469-494.

[18] K. A. Yost, *Solution of Large-Scale Allocation Problems with Partially Observed Outcomes,* Ph. D. Thesis, Naval Postgraduate School, Monterey, CA, Sept. 1998.

[19] K. A. Yost and A. R. Washburn, "The LP/POMDP Marriage: Optimization with Imperfect Information," *Naval Research Logistics,* Vol 47, No. 8, 607-619, 2000.

[20] R. Chen and G. L. Blankenship, "Dynamic Programming Equations for Discounted Constrained Stochastic Control, " *IEEE Trans. Automatic Control,*" v.49, no. 5, May 2004.

[21] D. A. Castañón, "A Lower Bound on Adaptive Sensor Management Performance for Classification," CISE Report 2005-3, Boston University, January 2005.

[22] M. Schneider, "An Optimal Policy for Sensor Management with Symmetric Sensor Measurements," ALPHATECH paper, August 2004.