

Approximate Nonmyopic Sensor Selection Via Submodularity and Partitioning

Wenhui Liao, wenhui.liao@thomsonreuters.com

R&D, Thomson Reuters, Saint Paul, MN 55123, USA

Qiang Ji, jiq@rpi.edu

Department of Electrical, Computer, and Systems Engineering

William A. Wallace, wallaw@rpi.edu, IEEE Fellow

Department of Decision Sciences and Engineering Systems

Rensselaer Polytechnic Institute, Troy, NY 12180, USA

Abstract

As sensors become more complex and prevalent, they present their own issues of cost-effectiveness and timeliness. It becomes increasingly important to select sensor sets that provide the most information at the least cost and in the most timely and efficient manner. Two typical sensor selection problems appear in a wide range of applications. The first type involves selecting a sensor set that provides the maximum information gain within a budget limit. The other type involves selecting a sensor set that optimizes the tradeoff between information gain and cost. Unfortunately, both require extensive computations due to the exponential search space of sensor subsets. This paper proposes efficient sensor selection algorithms for solving both of these sensor selection problems. The relationships between the sensors and the hypotheses that the sensors aim to assess are modeled with Bayesian networks (BNs), and the information gain (benefit) of the sensors with respect to the hypotheses is evaluated by mutual information. We first prove that mutual information is a submodular function in a relaxed condition, which provides theoretical support for the proposed algorithms. For the budget-limit case, we introduce a greedy algorithm that has a constant factor of $(1 - \frac{1}{e})$ guarantee to the optimal performance. A partitioning procedure is proposed to improve the computational efficiency of the algorithms by efficiently computing mutual information as well as reducing the search space. For the optimal-tradeoff case, a submodular-supermodular procedure is exploited in the proposed algorithm to choose the sensor set that achieves the optimal tradeoff between the benefit and cost in a polynomial-time complexity.

I. INTRODUCTION

Many real-world applications use sensors to obtain information that will help them improve their activities. Here, sensor is a general term; it could refer to a test, a feature, an observation, an evidence, etc. Rarely is only one sensor needed, however. Most applications use several sensors of different kinds, which we refer to as sensor sets. As sensors become more prevalent and ubiquitous, they present their own issues of cost-effectiveness and timeliness. It becomes critically important to select sensor sets that provide the most information at the least cost in a timely and efficient manner. For example, in fault-diagnosis problems (medical diagnosis, computer system troubleshooting, etc.) [7], [46], a set of informative tests needs to be selected to provide an optimal tradeoff between the cost of performing the tests and the accuracy of diagnoses; in sensor networks [10], [24], [25], a subset of sensors needs to be selected to achieve a suboptimal tradeoff between the energy consumption of operating the sensors and the information gain; in pattern recognition [19], [30], [39], good features need to be selected to guarantee the performance of the classifiers.

Purposely choosing an optimal subset from multiple sensing sources can save computational time and physical costs, avoid unnecessary or unproductive sensor actions, reduce redundancy, and increase the chance of making correct and timely decisions. Because of these benefits, sensor selection plays an especially important role for time-critical and resource-limited applications, including computer vision, control systems, sensor networks, diagnosis systems, and many military applications.

Basically, sensor selection problems can be divided into two types. The first type, called a budget-limit case, involves choosing a sensor set with maximum information gain given a budget limit. The other type, called an optimal tradeoff case, involves deciding a sensor set that achieves an optimal tradeoff between the information gain and the cost. Unfortunately, both of them are NP-hard, since the number of sensor subsets grows exponentially with the total number of sensors.

Most work formulates sensor selection as an optimization problem based on information-theory or decision-theoretic criteria. However, solving the optimization problem efficiently is difficult since the search space usually is large. To be practical, some methods [42], [10], [24] select the best sensor myopically, or select the first m sensors greedily. But the selected sensors could

lead to poor performance, since the selection methods cannot provide performance guarantees. Recently, some sensor selection algorithms have been proposed to achieve a balance between performance and efficiency. Isler and Bajcsy [16] present an approximation algorithm for sensor selection that aims to minimize the error in estimating the position of a target within a sensor network. They prove that at least one of the sensor subsets whose sizes are less or equal to six can guarantee that the resulting estimation error is within a factor 2 of the least possible error under certain assumptions. Thus the algorithm is to enumerate all l -subsets ($l \leq 6$) of sensors and choose the best one. Zheng et. al. [46] use a greedy test-selection strategy to find an optimal subset of tests in a fault-diagnosis system. Similarly, the paper provides a theoretical justification for the greedy test selection approach, along with some performance guarantees. (The expected number of tests produced by the greedy strategy is within an $O(\log N)$ factor from the optimal solution.) Krause and Guestrin [25] present an efficient greedy approach to select the most informative subset of variables in a graphical model for sensor networks. The algorithm provides a constant factor $(1 - \frac{1}{e} - \epsilon)$ approximation guarantee for any ϵ with high confidence. However, both Zheng and Krause's algorithms assume that the sensors are conditionally independent given the hypotheses (i.e., the event/situation that sensor fusion aims to assess).

We seek to achieve a balance between the performance and the efficiency of the proposed sensor selection algorithms by fully utilizing the properties of submodular functions and the probabilistic relationships among sensors. The probabilistic relationships between the sensors and the hypotheses are modeled by a Bayesian network (BN). In addition, the informativeness of the sensors with respect to the hypotheses is measured by mutual information. We then prove that mutual information is a submodular function under several conditions. Based on the theory of submodular functions, in the budget-limit case, the proposed sensor selection algorithms provide a constant factor of $(1 - 1/e)$ guarantee to the optimal performance. Furthermore, we propose a partitioning procedure by exploiting sensor dependence embedded in the BN model to compute mutual information efficiently as well as to reduce the search space, so that the efficiency of the algorithms is further improved. In the optimal tradeoff case, a submodular-supermodular procedure is embedded within the proposed sensor selection algorithm to decide the optimal or near-optimal sensor set that maximizes the difference between benefit and cost with a polynomial-time complexity.

The following sections are organized as follows. Section II gives a brief overview of related

work. Section III presents the Bayesian network model for sensor selection and fusion. Section IV introduces the sensor selection criteria. The sensor selection algorithms for the budget-limit case and the optimal-tradeoff case are described in Sections V and VI respectively. Section VII discusses the experimental results based on synthetic data, and Section VIII provides an illustrative application. This paper ends in Section IX with a summary and some suggestions for future work.

II. RELATED WORK

Sensor selection usually consists of two problems: selection criterion definition and sensor set selection based on the defined sensor selection criterion. In general, sensor selection can be treated as an optimization problem: finding an optimal sensor set that maximizes the objective function defined by the selection criterion. Most existing work can be divided into two groups based on the sensor selection criterion they used: information-theoretic methods and decision-theoretic methods.

A. *Sensor selection criteria*

Information-theoretic methods apply information theory to define the objective function for sensor selection. The common functions include Shannon's entropy, mutual information, entropy difference, and Kullback-Leibler's (KL) cross-entropy. Hintz [15] uses the expected change in Shannon entropy when tracking a single target moving in one-dimension with Kalman Filters. Zhang et al. [45] apply mutual information to select an optimal sensor set based on a dynamic Bayesian network. Denzler et al. [8] use mutual information to determine the optimal action (set of camera parameters, including focal length, pan and tilt angles) that will maximally decrease the uncertainty of the object-state estimation process. Wang et al. [42] propose an entropy-based sensor selection heuristic for target localization. Instead of using mutual information, they use maximal entropy difference as the criterion to choose one sensor at each step until the required uncertainty level of the target state is achieved. The method is computationally simpler than the mutual-information-based approaches and their experiments demonstrate that the method can sort candidate sensors into exactly the same order as the mutual-information method does in most cases.

An active sensing approach, based on an entropy measure called the Rényi divergence, is proposed by Kreucher et al. [26] to schedule sensors for multiple-target tracking applications. At each time step, only one sensor action is chosen to provide measurement and thus update the probability density of the target states. Kullback-Leibler's (KL) cross-entropy is used for optimal multisensor allocation [32] and sensor management [22]. Ertin et al [10] employ expected posterior entropy to choose the next measurement node (sensor) in a distributed Bayesian sequential estimation framework. They show that minimizing the expected posterior uncertainty is equivalent to maximizing the mutual information between the sensor output and the target state.

Although information-based criteria (entropy, mutual information, etc.) are the most commonly used functions for ranking information sources in terms of uncertainty reduction, these criteria require intensive computations, especially when computing for multiple sensors. Therefore, the research reported in the aforementioned papers often selects only one sensor at one time with the information-theoretic criteria. We propose an efficient procedure in this paper to efficiently compute mutual information by exploiting the statistical independences among sensors.

For decision-theoretic approaches, the objective function is defined based on decision theory, where the goal is to choose the optimal sensory action by maximizing an overall utility function (e.g. trade-off between cost and benefit). Wu and Cameron [44] describe a mathematical framework using Bayesian decision theory to select optimal sensing actions for achieving a given goal, for example, for object recognition in robot vision applications where one optimal action is decided at each time step. Rimey and Brown [37], [38] build a task-oriented computer vision system that uses Bayesian network and a maximum-expected-utility decision rule to choose a sequence of task-dependent and opportunistic visual operations on the basis of their utilities.

Gaag and Wessels [41] build a decision tree for selective gathering of evidence for diagnostic belief networks. Lindner et al. [29] estimate the expected utility of each sensor to predict the minimum cost subset of sensors for a mobile robot application. Kristensen [27] develops a Bayesian decision tree to solve the problem of choosing proper sensing actions from a family of candidates. Each sensing action is evaluated by its EISI (expected interest from sample information). The Bayesian decision rule simply selects the one with the maximum EISI value as the sensing action to be performed one at a time.

The most common decision-theoretic criterion is the value of information (VOI). The VOI of

a sensor set is defined as the difference in the maximum expected utilities with and without the information collected from the sensor set. It evaluates the utility of the sensor set by considering both the benefit and cost of using the sensors. The cost of operating the sensor for evidence collection includes the computational cost, physical cost, etc., while the benefits include financial benefits, performance improvements, reduced loss, etc. Oliver and Horvitz [34], [35] apply an EVI (expected-value-of-information) based policy to selective perception in SEER, a multimodal system of recognizing office activity that relies on a layered Hidden Markov Model (LHMMs) representation. Although the system uses a dynamic programming strategy to compute the EVI of each feature combination based on the LHMMs, it needs to enumerate all the feature combinations and then selects the best one. The process is, therefore, time-consuming.

A special group in the decision-theoretic methods is based on Markov decision process (MDP). Bayer [1] formalizes the diagnosis process as a MDP to find an optimal diagnostic policy that achieves optimal tradeoffs between the costs of tests and the costs of misdiagnoses. Cassandra et al. [2] model the problem of deciding optimal actions for mobile-robot navigation as a POMDP. Castanon [3] formulates the problem of dynamic scheduling of multi-mode sensor resources for classifying a large number of stationary objects as a POMDP.

The decision-theoretic methods provide a precise formulation for sensor selection problems. These measures are usually different from the information-theoretic measures such as mutual information since they directly relate sensor/resource allocation to decision-making, while information theoretic criterion relates sensor management to situation/event assessment. On the other hand, like the information-theoretic criteria, decision-theoretic criteria also require significant computation for multiple sensors. As a result, like the information-theoretic criteria, most of the above methods adopt the myopic approach, i.e., choose only one sensing action at each step. For example, MDP-based approaches suffer from combinatorial explosion when solving practical problems of even moderate size. Another problem of the decision-theoretic methods is the subjective definition of utility functions. The utilities are usually problem-dependent and may vary as their environment changes. In addition, for some applications, it may not be possible to derive appropriate utilities.

B. Optimization methods for sensor selection

For both the information-theoretic and decision-theoretic criteria, optimization methods are needed to identify the optimal sensor set. Methods for solving the optimization include search methods and mathematical programming (e.g. approximate dynamic programming, integer programming, and greedy selection strategies). One option is to use heuristic searches. Heuristic searches use some function that estimates the cost from the current state to the goal, presuming that such a function is efficient. Heuristic search techniques incorporate domain knowledge to improve efficiency over blind search. Some heuristic searches can guarantee an optimal solution, but they could be very slow. Thus, another search strategy is to give up completeness and risk losing optimal subsets in exchange for efficiency, such as *sequential forward selection* [43], *floating search selection* [36], and simulated annealing.

However, due to the high computational cost of search algorithms, in most applications involved with sensor selection, the greedy strategy is used. This strategy can be regarded as the simplest form of sequential search, where at each iteration the best sensor is incorporated into the candidate sensor set until there is no improvement in the value of the objective function. A more complex sequential search approach, called EA2 (Entropy Adaptive Aggregation) algorithm, is proposed in [11]. It includes an aggregative phase to heuristically choose the initial subset and an adaptive phase to iteratively aggregate and disaggregate the current subset until it converges. Kalandros and Pao [21] explore the use of randomization and super-heuristics search [28] for sensor selections in target-tracking applications. The search begins with a base sensor set and then generates more alternative solutions via a probabilistic assignment rule. The best solution is decided only from the generated solutions, and so it may not be optimal.

Our work differs from the foregoing work in that it explicitly exploits the theory of submodular functions with respect to the sensor selection criterion, and the probabilistic relationships among sensors to achieve both efficient and accurate sensor selections in two typical scenarios: the budget-limit case and the optimal-tradeoff case. Hence, the proposed algorithms are not only efficient, but also provide performance guarantees.

III. SENSOR SELECTION AND FUSION MODEL

We use dynamic Bayesian networks (DBNs) as shown in Figure 1 to model the relationships between sensors and the hypotheses the sensors aim to assess. Given the DBN, sensor fusion is

performed through probabilistic inference. A BN is a directed acyclic graph (DAG) that represents a joint probability distribution among a set of variables [4], [20]. In a BN, nodes denote variables and the links between nodes denote the conditional dependencies among the variables. The dependency for each node is characterized by a conditional probability table (CPT). A dynamic Bayesian network (DBN) additionally models the temporal relationships of the variables. Such a model is capable of representing the relationships among different sensors in a coherent and unified hierarchical structure, accounting for sensor uncertainties and dependencies, modeling the dynamics in situation development with temporal links, and providing principled inference and learning techniques to systematically combine the domain information and statistical information extracted from the sensors. These capabilities make DBN a good choice to model sensor selection and fusion.

As shown in Figure 1(a), the root node Θ of such a network, named as the hypothesis node, represents a mutually exclusive and exhaustive set of possible hypotheses about the events/situations we want to assess. For example, Θ could be the system states in a fault-diagnosis system, class labels for a classification problem, enemy intents in a battlefield situation-assessment scenario, etc.. Sensors occupy the lowest level nodes without any children. Evidence is gathered through sensors. To model the integrity/reliability with sensory readings, an information node I_i is introduced for each sensor S_i . I_i contains the information that sensor S_i measures. The conditional probabilities between the information node I_i and the corresponding sensor node S_i quantify the reliability of sensor measurements, and the sensor reliability may change over time. The intermediate nodes X model the probabilistic relations between the hypothesis and information nodes at different abstraction levels. The DBN in Figure 1(a) represents a typical structure for active sensor selection and fusion using BN. However, the BN structure could be more flexible. For example, Θ is not necessarily a root node, and there are may be multiple Θ s. In addition, S_i is not necessarily a leaf node either, and the intermediate nodes X are not needed for some applications. Figure 1 (b), (c), and (d) give different BN configurations for sensor fusion.

IV. INFORMATION GAIN AND SENSOR COST

The goal of sensor selection is to select a group of sensors that achieves a good balance between the information gain (benefit) and the cost of the sensors. In recent years, a commonly

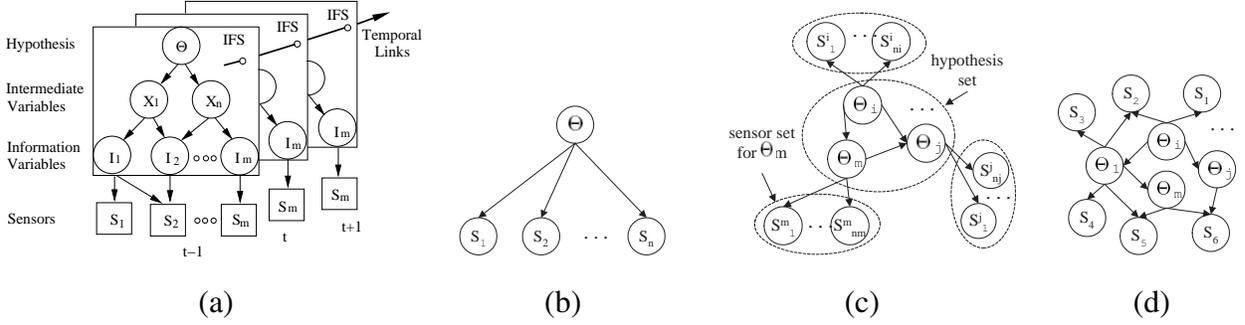


Fig. 1. (a) A dynamic Bayesian network for active sensor selection and fusion, where Θ , X , I and S denote hypothesis variables, intermediate variables, information variables, and sensors respectively; (b) All the hypotheses share the same sensors (the naive BN); (c) Each hypothesis has its own individual sensor set; (d) Each hypothesis has individual sensors and/or share sensors with other hypotheses. The BN in (a) is more appropriate for high-level sensor fusion, while (b), (c), and (d) can be regarded as special cases of the BN in (a) and are more appropriate for low-level sensor fusion.

used method to measure information gain is mutual information. with respect to Figure 1 (a), let S be a sensor set, $S = \{S_1, S_2, \dots, S_n\}$, the mutual information of S with respect to Θ , $I(\Theta; S)$, is defined as: $I(\Theta; S) = H(\Theta) - H(\Theta|S)$, where H indicates the entropy function. $I(\Theta; S)$ measures the uncertainty reduction of Θ given the sensors in S .

Mutual information may have a very interesting property, called *submodularity*. Before we prove it, we first give background information about submodular functions.

Let V be a finite set, and f be a set function: $2^V \mapsto \mathfrak{R}$. A function f is *submodular* if the inequality

$$f(A) + f(B) \geq f(A \cap B) + f(A \cup B)$$

holds for every pair of sets $A, B \subseteq V$ [31]. Equivalently,

$$f(A \cup X) - f(A) \geq f(B \cup X) - f(B)$$

for $\forall A \subset B \subset V, X \in V$. It means that the marginal value of X with respect to A is larger than the marginal value of X with respect to a larger set such as B . In other words, for submodular functions, adding X into a smaller set helps more than adding it into a larger set. For example, entropy functions, cut capacity functions, and matroid rank functions are submodular.

The negative of a submodular function is called a *supermodular* function. In other words, a function f is supermodular if the inequality

$$f(A) + f(B) \leq f(A \cap B) + f(A \cup B)$$

holds for every pair of sets $A, B \subseteq V$. Also, a function that is both submodular and supermodular is called a *modular* function.

In general, mutual information is not a submodular function [25]. However, we show that it is submodular under several conditions.

Proposition 1: *Let A be any subset on S and $f(A) = I(\Theta; A)$. If any sensor in S is conditionally independent of each other given Θ , then f is submodular.*

Proof. See Appendix.

Although people usually make the assumption of conditional independence in their frameworks such as [25], [46], it is still a strong assumption for most applications. Proposition 2 shows that f is still a submodular function under a relaxed assumption.

Proposition 2: *Let A, B be any subset on S and $f(A) = I(\Theta; A)$. If $I(\Theta; B \setminus A | A \cap B) \geq I(\Theta; B \setminus A | A)$, or $I(\Theta; A \setminus B | A \cap B) \geq I(\Theta; A \setminus B | B)$, then f is a submodular function.*

Proof. See Appendix.

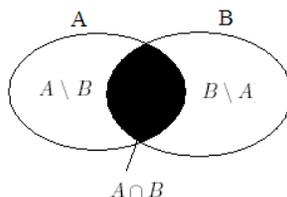


Fig. 2. A graph illustration of Proposition 2; $I(\Theta; B \setminus A | A \cap B) \geq I(\Theta; B \setminus A | A) \Leftrightarrow I(\Theta; A \setminus B | A \cap B) \geq I(\Theta; A \setminus B | B)$. The “\” sign means set subtraction.

$I(\Theta; B \setminus A | A \cap B) \geq I(\Theta; B \setminus A | A)$ shows that the mutual information between Θ and the sensors in $B \setminus A$ is related to how much information is already known. It is reasonable to assume that observing the sensors in $B \setminus A$ reduces the uncertainty of Θ less if more sensors have already been observed. Thus, the mutual information between Θ and the sensors in $B \setminus A$ given A , is less than that when a smaller set $A \cap B$ is observed. Figure 2 shows such a concept.

Also, the following proposition shows that f is a non-decreasing function.

Proposition 3: *Let A be any subset on S . $f(A) = I(\Theta; A)$, then f is non-decreasing, and $f(\emptyset) = 0$.*

Proof. See Appendix. We will show that the submodularity and nondecreasing of the information gain function help to provide a performance guarantee to our sensor selection algorithms.

The cost function for each sensor or sensor set can be simple or complex. The sensor cost could be computational cost, operational cost, cost of energy consumption, and others. The cost could be constant and the same for different sensors, or the cost could vary with the sensors. How to define the cost is application-dependent. In order to simplify the sensor selection problems, most applications assume that each sensor S_i has a constant cost $c(S_i)$, and the cost of a sensor set A_i is $\sum_{S_i \in A_i} c(S_i)$. However, it is often the case that the cost of operating a collection of sensors (jointly) is less than the combined cost of operating each sensor individually. In that case, it is reasonable to model c as a submodular function.

No matter how the benefit and cost are defined, the sensor selection problems can be divided into two categories. One involves finding a best sensor set with maximum information gain when the cost is within a budget limit. Another involves finding a best sensor set that achieves the best tradeoff between the information gain and the cost. Since both of these categories play an important role in the vast majority of applications, we propose efficient sensor selection algorithms for both of them.

V. SENSOR SELECTION WITH A BUDGET LIMIT

In this section, we present the sensor selection algorithms for the budget-limit case. And in the next section, we will present the selection algorithms for the optimal-tradeoff case. Let $S = \{S_1, S_2, \dots, S_n\}$ indicate the available sensors, $A = \{A_1, \dots, A_m\}$ be a collection of sets over S . The cost of each A_i is defined as $c(A_i) = \sum_{S_i \in A_i} c(S_i)$, and the benefit of each A_i is $f(A_i) = I(\Theta; A_i)$. Given a budget-bound L , the optimal sensor set A^* is:

$$A^* = \arg \max_{A_i \in A} \{f(A_i) : c(A_i) \leq L\}. \quad (1)$$

A. An Initial Greedy Algorithm

Based on the above definition, the sensor selection problem can be regarded as a budgeted maximum coverage problem [40]. Although in general such a problem is NP-hard, an approximate solution is available, which is near optimal and computable in polynomial time. We modified the algorithm from [25] as shown below. Similar algorithms can also be found at [23], [40].

Table I illustrates the pseudo code of Algorithm 1. In the first phase, Algorithm 1 arrives at a solution A_1^* by enumerating all possible l -element ($l < k$) subsets that satisfy the budget

TABLE I
THE PSEUDO CODE OF ALGORITHM 1: GREEDY-BRUTE FORCE

<p>Algorithm 1: $A^* = \text{SensorSelect1}(BN, S, L, k)$</p> <p>Let $G = \{A_i : A_i = k, c(A_i) \leq L, A_i \in A\}$; $A_1^* \leftarrow \arg \max_{A_i} \{f(A_i) : A_i < k, A_i \in A, c(A_i) \leq L\}$; $A_2^* \leftarrow \emptyset$; for each $A_i \in G$ $G' \leftarrow S \setminus A_i$; while $G' \neq \emptyset$ $S^* \leftarrow \arg \max_{S_i} \left\{ \frac{f(A_i \cup S_i) - f(A_i)}{c(S_i)} : S_i \in G' \right\}$; if $c(A_i) + c(S^*) \leq L$ then $A_i \leftarrow A_i \cup \{S^*\}$; $G' \leftarrow G' \setminus S^*$; if $f(A_i) > f(A_2^*)$ then $A_2^* = A_i$; Return $A^* = \arg \max(f(A_1^*), f(A_2^*))$;</p>

constraint. In the second phase, Algorithm 1 starts from all possible k -element subsets for some constant k and then uses a greedy approach to supplement these sets so that to produce a solution A_2^* . Finally, the algorithm outputs A_1^* if $f(A_1^*) > f(A_2^*)$ and A_2^* otherwise. The time complexity of the algorithm is $O(n^{k+1}\gamma \log n)$, where n is the size of the whole sensor set, and γ is the time required to compute the function value of f .

Theorem 1: *The worst-case performance guarantee of Algorithm 1 for solving the problem (1) is equal to $(1 - 1/e)$, if $k \geq 3$ and f is a submodular and non-decreasing function.*

Theorem 1 shows that the information gain of any sensor set selected by Algorithm 1 will be not less than $(1-1/e)$ of the information gain of the optimal sensor set. The theorem has been proven respectively by [12], [40], and [25].

Since f is a submodular and non-decreasing function as we proved in the previous section, Algorithm 1 can be used to find a near-optimal solution for Problem (1) in polynomial time with a performance guarantee.

B. Algorithm Speed-up with Partitioning

In Algorithm 1, for each updated A_i , we need to compute $f(A_i \cup S_i)$ for each possible candidate sensor S_i at each loop. As the size of A_i increases, the computation could be time consuming.

Therefore, it is important to speed up Algorithm 1. We propose a partitioning procedure to compute f efficiently by exploiting the probabilistic dependence among the sensors.

Assume the target is to compute $f(S) = I(\Theta; S_1, \dots, S_n)$. The partitioning procedure is to divide S into several groups, where the sensors in one group are conditionally independent of the nodes in other groups given Θ . The partitioning procedure consists of three steps:

- 1) Decide whether two sensors, S_i, S_j , are conditionally independent given Θ by exploring the Bayesian network structure based on four rules: i) if there is a path between S_i and S_j without passing Θ , S_i and S_j are dependent; ii) if both S_i and S_j are the ancestors of Θ , S_i and S_j are dependent given Θ ; iii) after removing the links to and from Θ from the original Bayesian network, if S_i and S_j have common ancestors, or S_i is S_j 's ancestor, or vice versa, then S_i and S_j are dependent; and iv) in all the other cases, S_i and S_j are conditionally independent given Θ . (Time complexity for step 1: $O(h)$, where h is the longest path in a Bayesian network.)
- 2) Build an undirected graph to model the relationships among the sensors. In such a graph, each vertex represents a sensor S_i , and each edge between two vertices indicates that the two corresponding sensors are dependent according to the rules in Step 1.
- 3) Partition the graph into disjoint connected subgraphs. A depth first search (DFS) algorithm [5] is used to partition the graph into several *connected components* (disjoint connected subgraphs) so that each component is disconnected from other components. The sensors in each connected component are conditionally independent of the sensors in any other connected components. Therefore, each connected component corresponds to one group. (Time complexity for step 3: $O(|V| + |E|)$, where V is the set of vertices, and E is the set of edges in the graph.)

Based on the time complexity, we can see that the partitioning procedure is quite efficient. Also, we only need to perform the partitioning procedure once for the whole sensor set. To divide the sensors in any subset into several groups, we need only look at the subgraph related with the targeted sensors. Thus, the whole procedure is efficient. The partitioning procedure returns several independent groups $A_c^1, A_c^2, \dots, A_c^m$. Then, the following lemmas stand.

Lemma 1: $H(\Theta, S_1, \dots, S_n) = \sum_{i=1}^m H(A_c^i | \Theta) + H(\Theta)$.

Proof. See Appendix

Lemma 2: The mutual information $I(\Theta; S_1, S_2, \dots, S_n) = H(A_c^1, \dots, A_c^m) - \sum_{i=1}^m H(A_c^i | \Theta)$.

Proof. See Appendix

Based on Lemma 2, computing $I(\Theta; S_1, \dots, S_n)$ is equal to computing $H(A_c^1, \dots, A_c^m)$ and each $H(A_c^i|\Theta)$, $i=1, \dots, m$. To compute $H(A_c^1, \dots, A_c^m)$ efficiently, one key issue is to compute $p(A_c^1, \dots, A_c^m)$. We notice that:

$$\begin{aligned} & p(S_1, \dots, S_m) \\ &= \sum_{\Theta} \{p(A_c^1, \dots, A_c^{m-1}, A_c^m|\Theta)p(\Theta)\} \\ &= \sum_{\Theta} \left\{ \prod_{i=1}^{m-1} p(A_c^i|\Theta) \cdot p(A_c^m|\Theta) \cdot p(\Theta) \right\} \end{aligned}$$

Since Algorithm 1 always starts from a smaller sensor set to a larger set, each $p(A_c^i|\Theta)$, $i = 1, \dots, m - 1$, is usually already computed before computing $p(A_c^1, \dots, A_c^{m-1}, A_c^m)$. Then $p(A_c^m|\Theta)$ is the only new factor that must be computed. This factor can be easily computed with a BN inference algorithm. Thus, $p(A_c^1, \dots, A_c^{m-1})$ can be computed much more easily. Similarly, for the second term in Lemma 2, $\sum_{i=1}^m H(A_c^i|\Theta)$, each $H(A_c^i|\Theta)$, $i = 1, \dots, m - 1$, is usually already computed, so $H(A_c^m|\Theta)$ is the only term that needs to be computed. Overall, the partitioning procedure allows computation sharing between different sensor sets. Especially, the f function of each large sensor set becomes easy to compute because of sharing computations with its subsets. Therefore, the partitioning procedure is able to speed up Algorithm 1 significantly as shown in the experiments. The pseudo code to compute mutual information $I(\Theta; S_1, \dots, S_n)$ is shown in Table II.

C. A New Algorithm with Partitioning

The partitioning procedure helps only with computing mutual information and doesn't affect the selection procedure in Algorithm 1. However, in order to further speed up Algorithm 1, we can also apply the partitioning procedure to sensor selection and thus get a new selection algorithm. This algorithm exploits both the submodularity property and the sensor dependence embedded in the BN model through the partitioning procedure. The pseudo code is shown in Table III.

Algorithm 2 consists of three phases. In the first phase, it uses the partitioning procedure to divide all the sensors into several groups, $A_c^1, A_c^2, \dots, A_c^m$. Then, in the second phase, for each

TABLE II
THE PSEUDO CODE TO COMPUTE $I(\Theta; S_1, \dots, S_n)$

<p>Partitioning S into conditionally independent groups $A_c^1, A_c^2, \dots, A_c^m$ by the DFS (depth first search) algorithm;</p> <p>for each $A_c^i, i = 1, \dots, m$</p> <p> if $H(A_c^i)$ is computed before</p> <p> Reuse the values of $p(A_c^i)$ and $H(A_c^i)$</p> <p> else</p> <p> compute $H(A_c^i)$</p> <p>compute $\sum_{\Theta} \{\prod_i^{m-1} p(A_c^i \Theta) \cdot p(A_c^m \Theta) \cdot p(\Theta)\}$</p> <p>compute $H(A_c^1, \dots, A_c^m)$</p> <p>compute $\sum_{i=1}^m H(A_c^i \Theta)$</p> <p>compute $I(\Theta; S_1, \dots, S_n)$ based on lemma 2</p>

TABLE III
THE PSEUDO CODE OF ALGORITHM 2: OPTIMAL SELECTION WITH PARTITIONING

<p>Algorithm 2: $A^* = \text{SensorSelect2}(BN, S, L)$</p> <p>Partitioning S into conditionally independent groups $A_c^1, A_c^2, \dots, A_c^m$ by the DFS (depth first search) algorithm;</p> <p>for each $A_c^i, i = 1, \dots, m$</p> <p> $L_i' = L(c_1 \frac{n \sum_{S_i \in A_c^i} f(S_i)/c(S_i)}{ A_c^i \sum_{S_i \in S} f(S_i)/c(S_i)} + c_2 \frac{ A_c^i }{n})$</p> <p> if $A_c^i \leq l$</p> <p> use brute-force method to select the best sensor set $A_i^{*'}$ from A_c^i under the cost constraint of L_i';</p> <p> else</p> <p> $A_i^{*'} = \text{SensorSelect1}(BN, A_c^i, L_i', 3)$</p> <p>$A^{*'} = \{A_1^{*'}, \dots, A_m^{*'}\}$;</p> <p> if $A^{*'} \leq l$</p> <p> A^* is decided from $A^{*'}$ with brute-force method;</p> <p> else</p> <p> $A^* = \text{SensorSelect1}(BN, A^{*'}, L, 3)$;</p> <p>Return A^*;</p>
--

group A_c^i , a subfunction, Algorithm 1 or a brute-force algorithm, is called to select a local optimal sensor set subject to the local budget L_i' . If the size of the group is small ($\leq l$), the brute-force algorithm is used to find the local optimal set. Otherwise, Algorithm 1 is applied to decide the local optimal set. In the third phase, a new sensor set $A^{*'}$ is constructed by combining all the local optimal sets. For this set, Algorithm 1 or the brute-force method is applied again to decide the global optimal sensor set.

There are two dynamic parameters, l and L_i' , in the algorithm. l decides which algorithm is applied to each group to obtain the local optimal set: the brute-force method or Algorithm 1. If l is too large, the brute-force algorithm will slow down the speed. Empirically, l is set as 6. Another parameter L_i' decides the budget for each group. It is dynamically decided by two factors: 1) $\frac{1}{|A_c^i|} \sum_{S_i \in A_c^i} f(S_i)/c(S_i)$, the average ratio of $f(S_i)/c(S_i)$ in each group A_c^i ; and 2) $|A_c^i|$, the size of A_c^i .

Obviously, the time complexity of Algorithm 2 depends on the two subfunctions: Algorithm 1 and the brute-force method. For the brute-force method, since the size of the input group is always less than l ($l \leq 6$), it is time efficient. Since the input is a subset when calling Algorithm 1, it costs much less time than when the input is the whole sensor set. Therefore, Algorithm 2 is more efficient than Algorithm 1 in general. If all the sensors are conditionally independent of one another, Algorithm 2 degenerates to Algorithm 1.

In addition, the performance of Algorithm 2 can be justified by two factors: 1) Algorithm 2 performs sensor selection on each sensor subset, instead of starting from the whole sensor set, thus it is more efficient; 2) The sensor subsets are conditionally independent from each other through partitioning, thus the sensors selected from each subset are more likely to be included in the optimal sensor set, compared to dependent sensor subsets. Therefore, Algorithm 2 significantly improves the speed of Algorithm 1, yet with comparable accuracy.

VI. SENSOR SELECTION WITH OPTIMAL TRADEOFF

The algorithms in section V maximize the benefit of a set of sensors as long as the cost is within a budget. A further goal is to find a sensor set with the best tradeoff between the cost of the sensors and the potential gain (benefit) obtained by using the sensors. In certain applications,

it is more reasonable to achieve the optimal sensor set A^* such that:

$$A^* = \arg \max_{A_i} \{f(A_i) - c(A_i)\} \quad (2)$$

Take sensor networks for example, c could be sensor power usage. It is desirable to have sensors that maximize the gain while consuming as little power as possible in sensor networks. This is an NP-hard problem too. Based on the previous analysis, f is a non-decreasing submodular function. If c is a modular function as assumed by most researchers, $f(A_i) - c(A_i)$ would still be a submodular function, but not necessarily a non-decreasing one. Therefore, Problem (2) becomes a submodular function maximization problem, which can be solved with global solutions by some well-known polynomial algorithms such as the combinatorial strongly polynomial algorithm IFF (Iwata, Fleischer, and Fujishige) developed by Iwata et. al. [18], a faster scaling algorithm in [17], etc. However, in some cases, it is not reasonable to define c as a modular function. For example, usually, the cost of operating a collection of sensors (jointly) is less than the combined cost of operating each sensor individually. Thus assuming the cost of a sensor set is the sum of the cost of each individual sensor may not be appropriate. In that case, it is more practical to model c as a submodular function.

Unfortunately, if c is also a submodular function, the difference between two submodular functions like f and c is not necessarily a submodular function. However, if we could transfer $f(A_i) - c(A_i)$ into a submodular function, the optimal sensor set A^* can be obtained efficiently with one of these well-known algorithms. One solution is to seek a modular function h that closely approximates the cost function c . Since h is a modular function, $f(A_i) - h(A_i)$ will still be a submodular function. Proposition 4 shows how to find such a function h , which is referred from [9], [33].

Proposition 4: *Let π be a random permutation of the sensor set $S = \{S_1, \dots, S_n\}$. Let $W_i = \{\pi(1), \pi(2), \dots, \pi(i)\}$. A function $h : S \rightarrow \mathfrak{R}$ is defined as follows:*

$$h(\pi(i)) = \begin{cases} c(W_1) & \text{if } i = 1 \\ c(W_i) - c(W_{i-1}) & \text{otherwise} \end{cases}$$

In addition,

$$h(A_i) = \sum_{S_j \in A_i} h(S_j), \forall A_i \subseteq S$$

Then,

- 1) $h(A_i) \leq c(A_i)$ for $\forall A_i \subseteq S$
- 2) $h(W_m) = c(W_m)$, for $1 \leq m \leq n$.

Proof see [14].

With this procedure, the generated function h is modular and bounded above by c . Furthermore, this modular approximation is the tightest possible approximation to c in the following sense [33]:

Proposition 5 *Every h obtained in Proposition 4 is a vertex in the extended base polymatroid of c , and every vertex in the extended base polymatroid of c can be obtained by picking an appropriate permutation.*

Proof see [14].

Now, we can use Algorithm 3 as depicted in Table IV to decide the optimal or near optimal solution for the problem (2) by using Proposition 4 and the submodular function maximization algorithm.

TABLE IV
THE PSEUDO CODE OF ALGORITHM 3: SELECTION WITH MODULAR APPROXIMATION

Algorithm3: SensorSelect3(BN, S, δ)
<pre> max ← -∞; mark ← true; A* ← ∅; while mark = true do if A* is empty π ← a random permutation of S; else π ← random permutation starting with A*; define h with the method in Proposition 4 given π; A* ← arg max_{A_i ⊆ S} (f - h)(A_i) if (f - h)(A*) - max > δ mark = true; max = (f - h_n)(A*); else mark = false; Return A*; </pre>

As shown in the pseudo code, it repeats generating new permutation of the sensor set S based

on the selected sensor set A^* at the previous step until no obvious improvement is achieved, which is controlled by the parameter δ . Corresponding to each new permutation, a modular function h is generated to approximate c and $A^* \leftarrow \arg \max_{A_i \subseteq S} (f - h)(A_i)$ can be decided with the IFF algorithm by Iwata [18]. The time complexity of IFF algorithm is $O(n^\tau \gamma \log(n))$, where γ is the time required for computing the function value of $(f - h)$.

If the cost function c is a modular function, no modular approximation is necessary, and Algorithm 3 actually returns the global optimal solution. When the cost function c is a submodular function, because of introducing h to approximate c , local optimum can be returned. However, since h is very close to c (as shown in Proposition 5), and we also use δ to control the precision effects, the returned solution is usually optimal or near-optimal (as will be shown in the experiments).

VII. EXPERIMENTS

In the experiments, we analyze the performance of the proposed sensor selection algorithms in terms of both accuracy and speed. To demonstrate the robustness of the algorithms to different BN structures and parameters, we generate BNs with random structures and parameters, whose maximal number of nodes is 50. In each BN, 12 nodes are randomly selected as the sensor nodes and 1 node is selected as the hypothesis node. For the budget-limit case, the cost of each sensor is randomly set at each testing case and the budget limit L is set as a constant for all the cases. For the optimal-tradeoff case, the sensor cost is defined as a submodular function. The ground-truth of the optimal sensor subset is obtained by a brute-force approach.

A. Sensor Selection with a Budget Limit

To demonstrate the performance of the proposed algorithms, we compare Algorithm 1, Algorithm 2, a greedy approach, and the brute-force method. For Algorithm 1, k is set as 3. The greedy approach is similar to Algorithm 1 with $k=1$.

In order to demonstrate whether the performances of the algorithms are affected by the relationships among the sensors, we designed two sets of testing cases. The first set consists of 500 BNs where all the sensors are conditionally independent of one another; and the second set consists of 500 BNs where the sensors can be dependent of one another. Figure 3 gives two example BNs.

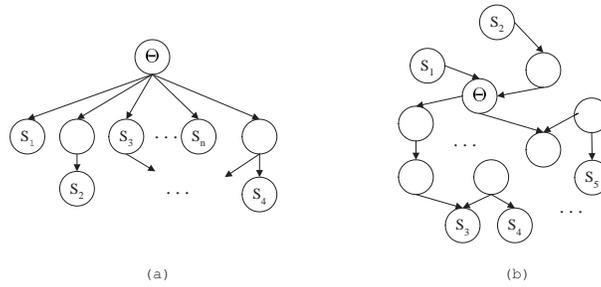


Fig. 3. (a) An example BN where all the sensors are conditionally independent of one another given Θ ; (b) An example BN where the sensors can be dependent of one another given Θ .

TABLE V

PERFORMANCE OF THE SENSOR SELECTION ALGORITHMS WHEN SENSORS ARE CONDITIONALLY INDEPENDENT IN 500 TESTING CASES.

	Alg. 1a	Alg. 1b	Greedy
Error Ratio	0.05	0.05	0.21
Mutual Information Ratio	0.98	0.98	0.91
Running Time Ratio	0.35	0.06	0.03

Table V and VI demonstrate the experimental results, where Alg.1a represents Algorithm 1 without using the partitioning procedure to compute mutual information while Alg.1b represents Algorithm 1 with the partitioning procedure. Table V doesn't list the performance of Alg.2 because it has the same performance as Alg.1 when all the sensors are conditionally independent. The *error ratio* is the ratio between the number of mis-selection cases and the overall number of cases, where a mis-selection case is defined as the case that the selected sensor set has more than one sensor different from the optimal selection. The *mutual information ratio* is the mutual

TABLE VI

PERFORMANCE OF THE SENSOR SELECTION ALGORITHMS WHEN SENSORS ARE DEPENDENT IN 500 TESTING CASES.

	Alg. 1a	Alg. 1b	Alg. 2	Greedy
Error Ratio	0.06	0.06	0.10	0.22
Mutual Information Ratio	0.99	0.99	0.97	0.91
Running Time Ratio	0.35	0.10	0.06	0.03

information between the selected sensors and Θ , divided by the mutual information between the optimal sensor set and Θ . The *running time ratio* is the ratio between the computational time of the algorithm in each column and that of the brute-force method. Both the mutual information ratio and the running time ratio are averaged over the 500 testing cases.

As shown in the Tables, the performance of each algorithm is rarely affected by the relationships among the sensors. Compared to the greedy approach, Alg.1a, Alg.1b, and Alg. 2 have much lower error ratio, which means they are able to return the near-optimal solutions for most testing cases. In addition, Alg.2 is faster than Alg.1a and Alg.1b because the partitioning procedure is applied to both mutual information computation and selection procedure. Alg.1b is faster than Alg.1a because of the savings in mutual information computation. Greedy approach, on the other hand, is faster than both algorithms 1 and 2 for both types of networks, but with much worse error ratios.

B. Sensor Selection with Optimal Tradeoff

To test Algorithm 3, 500 BNs are randomly generated. But the cost of sensors is defined as a submodular function. We compare Algorithm 3 to a greedy method, which is very similar to Algorithm 1 except that the budget limit is removed. The overall performance is shown in Table VII. The *DIC rate* is the ratio between the DIC (difference of mutual information and cost) of the selected sensor set and that of the optimal sensor set. Both the DIC and error ratio are averaged over the 500 testing cases.

TABLE VII
PERFORMANCE OF ALGORITHM 3 VS GREEDY APPROACHES IN 500 TESTING CASES

	Error Ratio	DIC rate
Algorithm 3	0	0.99
Greedy approach (k=3)	0.25	0.94
Greedy approach (k=1)	0.42	0.82

Table VII shows that the error ratio of Algorithm 3 is 0. It demonstrates that Algorithm 3 always returns an optimal or near-optimal solution. For the greedy approach, although it has good performance for the budget-limit case since the objective function is submodular and non decreasing, it performs worse in the optimal tradeoff case. Since $(f - c)$ is neither a submodular

function nor non decreasing, there is no performance guarantee for the greedy approach even with $k \geq 3$ in the optimal tradeoff case. Figure 4 demonstrates the performance of Algorithm 3 vs greedy approaches in 50 cases. In the majority of the cases, the DIC is 1, which means the proposed algorithm returns the optimal sensor sets, while the greedy approaches rarely return the optimal sensor sets.

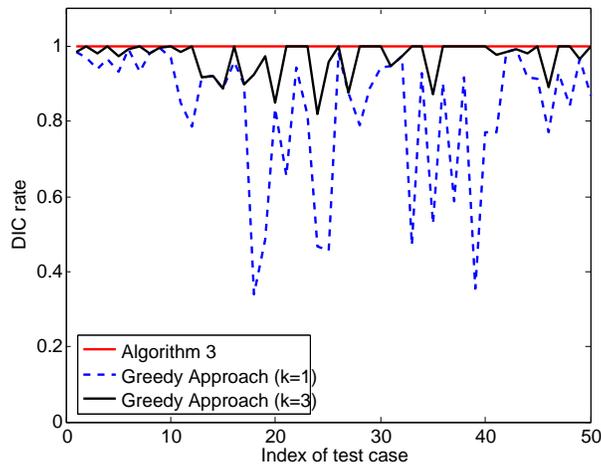


Fig. 4. Performance of Algorithm 3 vs greedy approaches. The x-coordinate is the test case index, and the y-coordinate is the DIC rate.

When n is small, Algorithm 3 is not much faster than the brute-force method although the time complexity of the former is polynomial and the latter is exponential. But when n is large (e.g., $n \geq 25$), it is much faster than the brute-force method, in addition to its capability of providing optimal or near-optimal solutions.

VIII. AN ILLUSTRATIVE APPLICATION

We apply the proposed sensor selection algorithms to an application of multistage battlefield situation assessment. The scenario here is inspired from [6]. In this application, dynamic Bayesian network is used to model the sensors and hypothesis. Although we focus on static BNs in the previous sections, our algorithms can also apply to dynamic BNs one frame at a time, as shown in this application.

The scenario develops during a period of growing hostility between the Blue force and the Red force who poses a threat. The goal of this scenario is for the blue force to selectively use its

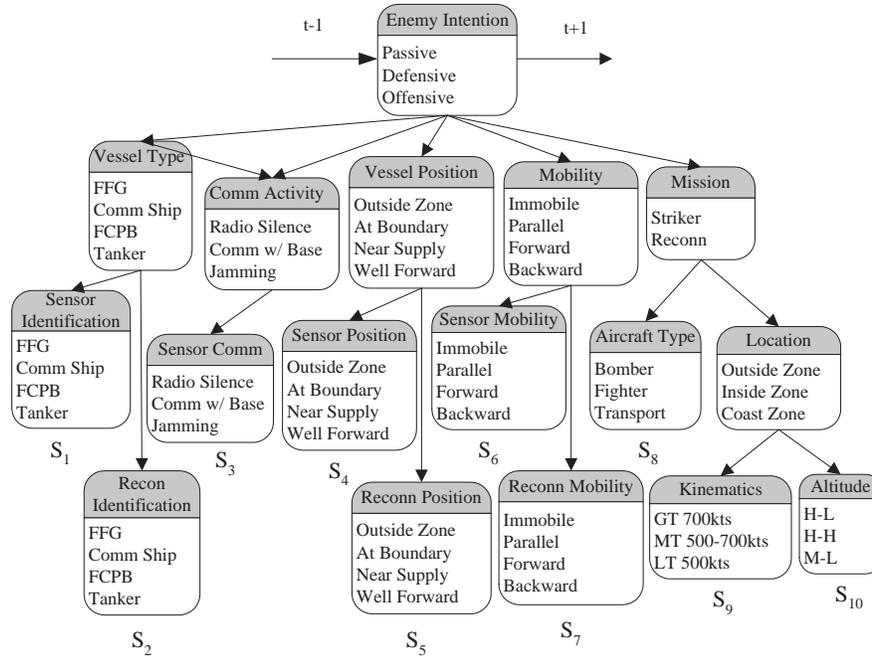


Fig. 5. A Bayesian network model for the battlefield scenario. $S_1 - S_{10}$ are information sources.

surveillance assets to quickly and efficiently infer the intent of the Red force. More detail is given in [6]. The Blue force surveillance facilities include a number of offshore sensors, unmanned aerial vehicles (UAVs), surveillance helicopters (RAH66 Comanche), etc. The Blue forces on duty in the restricted zone consist of 1) a Fremantle Class Patrol Boat (FCPB); 2) a Maritime Patrol Aircraft (MPA); 3) a Night Hawk Helicopter; 4) one F111 (Maritime Strike Aircraft). The Red forces include: 1) a major fleet unit; a Guided Missile Frigate (FFG); 2) one FCPB; 3) a communication ship. In addition, the Red force has two surface units armed with an M386 Rocket Launcher and a 110 SF Rocket Launcher that are ready to move to the locations where the Blue force is within their fire range.

A dynamic Bayesian network as shown in Figure 5 is constructed to assess the situations for the scenario above. A set of hypotheses representing possible Red force intents include: 1) Passive: monitor the Blue forces in the restricted zone and assume that the Blue forces will not interfere with the fuel supply; 2) Defensive: conduct active reconnaissance and maintain a defensive presence to guard the supply routes against from the Blue force interference; 3) Offensive: mount a naval attack or infantry artillery engagement (surface-to-air or surface-to-

surface attack) on the Blue forces with the intent of destroying the Blue forces as well as their offshore surveillance facilities.

We assume that there are external modules that receive sensor data and make the data available as input evidence to the network. The model in Figure 5 shows that there are 10 sensors available to supply information. The conditional probabilities and sensor costs are given subjectively. A Blue force commander needs to select appropriate sensors over time in order to assess the hypothesis of the Red force intent (Passive, Defensive, or Offensive in a timely and efficient manner).

In order to further demonstrate how the sensor selection algorithms help predict enemy intents, a simulation system is developed to generate synthetic data. The simulator consists of two independent but related models: a *source model* simulating the intents of the Red force to produce evidence reflecting them, and a *working model* simulating the Blue force that estimates enemy (Red force) intents and determines appropriate sensory actions by selectively collecting the evidence. We assume the enemy intents is passive in the beginning, gradually changes to offensive, and finally becomes defensive.

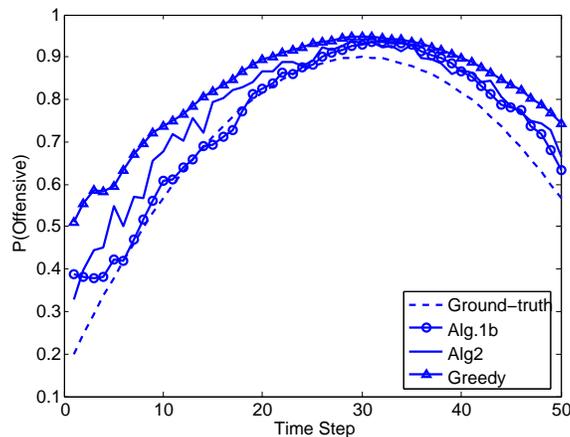


Fig. 6. Sensor selection results of the military example

Figure 6 illustrates the estimation results when different sensor selection algorithms are used. The dotted line represents the ground-truth enemy intent $P(\text{Offensive})$, and the other three curves represent the inferred enemy intent by collecting evidence from the selected sensors with the corresponding three sensor selection algorithms: Alg.1b, Alg.2, and the greedy approach. As

shown in the figure, the enemy intent estimated by Alg.1b and Alg.2 is quite close to the true enemy intent after 5 time steps, while the enemy intent estimated by the greedy algorithm is not close to the truth until after more than 20 time steps. After about 30 time steps, as $P(\text{offensive})$ decreases, Alg.1b and Alg2 are able to follow enemy intent more closely than the greedy algorithm as well. In other words, Alg.1b and Alg.2 are able to select optimal sensors in time and thus track the enemy intent better. In addition Alg.2 is 40% faster than Alg.1b. Table VIII shows the specific sensors used in certain time slices.

TABLE VIII
A CASE STUDY OF SENSOR SELECTION WITH ALG. 1B

Assessment Stage	Probability of Hypothesis	Sensors Selected
1	$P(\text{Pas})=0.33, P(\text{Def})=0.33, P(\text{Off})=0.33$	S_9, S_{11}
8	$P(\text{Pas})=0.17, P(\text{Def})=0.32, P(\text{Off})=0.51$	S_9, S_{11}
19	$P(\text{Pas})=0.01, P(\text{Def})=0.18, P(\text{Off})=0.81$	S_9, S_{14}, S_{16}
30	$P(\text{Pas})=0.03, P(\text{Def})=0.05, P(\text{Off})=0.92$	S_9, S_{13}, S_{17}
45	$P(\text{Pas})=0.01, P(\text{Def})=0.24, P(\text{Off})=0.75$	S_9, S_{14}, S_{16}

Note: Pas, Def, and Off represent Passive, Defensive, and Offensive, respectively.

IX. CONCLUSIONS

In this paper, we propose several algorithms to perform efficient and accurate sensor selection in two typical scenarios: the budget-limit case in which the best sensor set is the one with maximum information gain under a budget limit; and the optimal-tradeoff case in which the best sensor set is the one that achieves the optimal tradeoff between the information gain and the cost. Although finding an optimal solution is NP-hard for both of them, we introduce efficient and near-optimal solutions by fully utilizing the properties of the sensor selection criterion and the probabilistic dependences among sensors.

Specifically, for the budget-limit case, to ensure performance of the proposed algorithms, we first prove that mutual information is a submodular function under a relaxed condition. Based on this property, we introduce an efficient greedy approach with a constant factor of $(1 - 1/e)$ performance guarantee to the optimal performance. Furthermore, to improve the efficiency of the

algorithms, we propose a partitioning procedure for both efficient sensor selection and efficient mutual information computation. For the optimal tradeoff case, if the cost function is a modular function, the proposed algorithm can provide the global optimal solution in polynomial time; if the cost function is a submodular function, a submodular-supermodular procedure is embedded with the proposed sensor selection algorithm to choose the optimal or near-optimal sensor set in polynomial time. The experimental results with both synthetic and real data demonstrate the performance and efficiency of our algorithms.

This paper focuses only on sensor selection. Our future goal is to model sensor selection, sensor fusion, and decision-making in a unified framework. More issues will be addressed, e.g., how to fuse the information collected from the sensors efficiently, how to decide the optimal action based on the fused results, and how to learn the parameters of the BN framework. In addition, we will apply the framework as well as the algorithms to more real-world applications.

APPENDIX

Proof of Proposition 1:

$$\begin{aligned}
 f(A) &= H(\Theta) - H(\Theta|A) \\
 &= H(\Theta) - [H(A, \Theta) - H(A)] \\
 &= H(\Theta) - (H(\Theta) + H(A|\Theta)) + H(A) \\
 &= H(A) - H(A|\Theta)
 \end{aligned}$$

Thus, $\forall A, B \subseteq S$:

$$\begin{aligned}
 &f(A) + f(B) \\
 &= H(A) + H(B) - H(A|\Theta) - H(B|\Theta) \\
 &f(A \cap B) + f(A \cup B) \\
 &= H(A \cap B) + H(A \cup B) - H(A \cap B|\Theta) - H(A \cup B|\Theta)
 \end{aligned}$$

From the fact that Entropy is a submodular function [13], thus $H(A) + H(B) \geq H(A \cap B) + H(A \cup B)$. Also, since the sensors are conditionally independent given Θ ,

$$\begin{aligned}
& H(A|\Theta) + H(B|\Theta) \\
&= \sum_{S_i \in A} H(S_i|\Theta) + \sum_{S_i \in B} H(S_i|\Theta) \\
&= \sum_{S_i \in A \cap B} H(S_i|\Theta) + \sum_{S_i \in A \cup B} H(S_i|\Theta) \\
&= H(A \cap B|\Theta) + H(A \cup B|\Theta)
\end{aligned}$$

Therefore: $f(A) + f(B) \geq f(A \cap B) + f(A \cup B)$, in other words, f is a submodular function.

Proof of Proposition 2:

Based on the chain rule of mutual information, we find the following:

$$\begin{aligned}
& I(\Theta; B \setminus A|A \cap B) \\
&= H(\Theta|A \cap B) - H(\Theta|(B \setminus A) \cup (A \cap B)) \\
&= H(\Theta|A \cap B) - H(\Theta|B) \\
& I(\Theta; B \setminus A|A) \\
&= H(\Theta|A) - H(\Theta|(B \setminus A) \cup (A)) \\
&= H(\Theta|A) - H(\Theta|(A \cup B))
\end{aligned}$$

Thus,

$$\begin{aligned}
& I(\Theta; B \setminus A|A \cap B) \geq I(\Theta; B \setminus A|A) \\
&\Rightarrow H(\Theta|A \cap B) - H(\Theta|B) \geq H(\Theta|A) - H(\Theta|A \cup B) \\
&\Rightarrow H(\Theta) - H(\Theta|A) + H(\Theta) - H(\Theta|B) \\
&\quad \geq H(\Theta) - H(\Theta|A \cap B) + H(\Theta) - H(\Theta|A \cup B) \\
&\Rightarrow f(A) + f(B) \geq f(A \cap B) + f(A \cup B)
\end{aligned}$$

If $I(\Theta; A \setminus B|A \cap B) \geq I(\Theta; A \setminus B|B)$ is true, the proof is similar, we thus skip the details. Actually, $I(\Theta; B \setminus A|A \cap B) \geq I(\Theta; B \setminus A|A)$ is equivalent with $I(\Theta; A \setminus B|A \cap B) \geq I(\Theta; A \setminus B|B)$. We skip the proof since it is very straightforward.

Proof of Proposition 3:

Let X be any sensor belonging to S , but not in A .

$$\begin{aligned}
& f(A \cup X) - f(A) \\
&= H(\Theta) - H(\Theta|A \cup X) - (H(\Theta) - H(\Theta|A)) \\
&= H(\Theta, A) - H(A) + H(A, X) - H(\Theta, A, X)
\end{aligned}$$

Since entropy is a submodular function[13]

$\Rightarrow H(\Theta, A) + H(A, X) \geq H(A) + H(\Theta, A, X)$ (because $(\Theta, A) \cap (A, X) = A$ and $(\Theta, A) \cup (A, X) = (\Theta, A, X)$).

Thus, $f(A \cup X) - F(A) \geq 0$; in other words, f is non decreasing. It is also obvious that $f(\emptyset) = 0$.

Proof of Lemma 1:

$$\begin{aligned}
& H(\Theta, S_1, S_2, \dots, S_n) \\
&= H(\Theta, A_c^1, A_c^2, \dots, A_c^m) \\
&= - \sum_{\Theta, A_c^1, \dots, A_c^m} \{p(\Theta, A_c^1, \dots, A_c^m) \log p(\Theta, A_c^1, \dots, A_c^m)\} \\
&= - \sum_{\Theta, A_c^1, \dots, A_c^m} \{p(\Theta, A_c^1, \dots, A_c^m) \log [p(\Theta) \prod_{i=1}^m p(A_c^i|\Theta)]\} \\
&= - \sum_{\Theta, A_c^1, \dots, A_c^m} \{p(\Theta, A_c^1, \dots, A_c^m) [\sum_{i=1}^m \log p(A_c^i|\Theta) + \log p(\Theta)]\} \\
&= - \sum_{i=1}^m \sum_{\Theta, A_c^1, \dots, A_c^m} \{p(\Theta, A_c^1, \dots, A_c^m) \cdot \log p(A_c^i|\Theta)\} \\
&\quad - \sum_{\Theta} p(\Theta) \log p(\Theta) \\
&= \sum_{i=1}^m H(A_c^i|\Theta) + H(\Theta)
\end{aligned}$$

Proof of Lemma 2:

$$\begin{aligned}
& I(\Theta; S_1, S_2, \dots, S_m) \\
&= H(\Theta) - H(\Theta | A_c^1, \dots, A_c^m) \\
&= H(\Theta) - [H(\Theta, A_c^1, \dots, A_c^m) - H(A_c^1, \dots, A_c^m)] \\
&= H(A_c^1, \dots, A_c^m) - \sum_{i=1}^m H(A_c^i | \Theta) \text{(from Lemma 1)}
\end{aligned}$$

REFERENCES

- [1] V. Bayer-Zubek, *Learning diagnostic policies from examples by systematic search*, UAI04 (2004), 27–34.
- [2] Anthony R. Cassandra, Leslie Pack Kaelbling, and James A. Kurien, *Acting under uncertainty: Discrete bayesian models for mobile robot navigation*, Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (1996).
- [3] D. A. Castanon, *Approximate dynamic programming for sensor management*, Proceedings of the 36th IEEE Conference on Decision and Control **2** (1997), 1202–1207.
- [4] E. Charniak, *Bayesian networks without tears*, AI Magazine **12(4)** (1991), 50–63.
- [5] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to algorithms*, MIT Press, 2002.
- [6] B. Das, *Representing uncertainties using bayesian networks*, Tech. Rep. DSTO-TR-0918 (1999), DSTO Electronics and Surveillance Research Laboratory, Salisbury South Australia 5108, Australia.
- [7] R. Debouk, S. Lafortune, and D. Teneketzis, *On an optimization problem in sensor selection for failure diagnosis*, Proceedings of the 38th IEEE Conference on Decision and Control **5** (1999), 49990–4995.
- [8] Joachim Denzler and Christopher M. Brown, *Information theoretic sensor data selection for active object recognition and state estimation*, IEEE Trans. Pattern Anal. Machine Intell. **24** (2002), no. 2, 145–157.
- [9] J. Edmonds, *Submodular functions, matroids and certain polyhedra*, Proc. Calgary Int. Conf. Combinatorial structures and their applications (1970), 69–87.
- [10] E. Ertin, J. Fisher, and L. Potter, *Maximum mutual information principle for dynamic sensor query problems*, IPSN'03: Proceedings of the third international symposium on information processing in sensor networks, 2004, pp. 405–416.
- [11] Bienvenu Fassinut-Mombot and Jean-Bernard Choquel, *A new probabilistic and entropy fusion approach for management of information sources*, Information Fusion **5** (2004), 35–47.
- [12] U. Feige, *A threshold of $\ln n$ for approximating set cover*, Journal of ACM **45** (1998), 634–652.
- [13] S. Fujishige, *Polymatroidal dependence structure of a set of random variables*, Information and Control **39** (1978).
- [14] M. Grötschel, L. Lovász, and A. Schrijver, *The ellipsoid method and its consequences in combinatorial optimization*, Combinatorica **1** (1981), 169–197.
- [15] K. J. Hintz, *A measure of the information gain attributable to cueing*, IEEE Trans. on Systems, Man, and Cybernetics **21** (1991), no. 2, 434–442.
- [16] Volkan Isler and Ruzena Bajcsy, *The sensor selection problem for bounded uncertainty sensing models*, Fourth International Symposium on Information Processing in Sensor Networks (IPSN 2005) (2005), 151–158.
- [17] Satoru Iwata, *A faster scaling algorithm for minimizing submodular functions*, SIAM Journal on Computing **32** (2003), no. 4, 833–840.

- [18] Satoru Iwata, Lisa Fleischer, and Satoru Fujishige, *A combinatorial, strongly polynomial-time algorithm for minimizing submodular functions*, Journal of the ACM **48** (2001), no. 4, 761–777.
- [19] A. Jain and D. Zongker, *Feature selection: Evaluation, application, and small sample performance*, IEEE Trans. Pattern Analysis and Machine Intelligence **19** (1997), no. 2, 153–158.
- [20] F.V Jensen, *Bayesian networks and decision graphs*, Springer Verlag, New York, 2001.
- [21] M. Kalandros, L.Y. Pao, and Yu-Chi Ho, *Randomization and super-heuristics in choosing sensor sets for target tracking applications*, Proceedings of the 38th IEEE Conference on Decision and Control **2** (1999), 1803–1808.
- [22] K. Kastella, *Discrimination gain to optimize classification*, IEEE Trans. Systems Man Cybernet. Systems and Humans **27** (1997), no. 1, 112–116.
- [23] S. Khuller, A. Moss, and J. Naor, *The budgeted maximum coverage problem*, Inf Proc Let **70** (1999), no. 1, 39–45.
- [24] F. Kobayashi, T. Fukui, F. Arai, T. Fukuda, F. Kojima, M. Onoda, and Y. Hotta, *Sensor selected fusion with sensor selection based gating neural network*, Proceedings of the 2002 IEEE International Conference on Fuzzy Systems, 2002, pp. 1482–1487.
- [25] Andreas Krause and Carlos Guestrin, *Near-optimal nonmyopic value of information in graphical models*, In the 21st Conference on Uncertainty in Artificial Intelligence (UAI 2005) (2005).
- [26] C. Kreucher, K. Kastella, and A.O. Hero III, *Sensor management using an active sensing approach*, Signal Processing **85** (2005), no. 3, 607–624.
- [27] S. Kristensen, *Sensor planning with bayesian decision theory*, Robotics and Autonomous Systems **19** (1997), no. 3, 273–286.
- [28] Tak Wing Edward Lau and Yu-Chi Ho, *Super-heuristics and their applications to combinatorial problems*, Asian Journal of Control **1** (1999), no. 1, 1–13.
- [29] J. Lindner, R.R. Murphy, and E. Nitz, *Learning the expected utility of sensors and algorithms*, IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (1994), 583–590.
- [30] H. Liu and L. Yu, *Toward integrating feature selection algorithms for classification and clustering*, IEEE Transactions on Knowledge and Data Engineering **17** (2005), no. 4, 491–502.
- [31] L. Lovasz, *Submodular functions and convexity*, Mathematical Programming – The State of the Art (A. Bachem, M. Grotchel, and B. Korte, eds.), Springer-Verlag, 1983, pp. 235–257.
- [32] R. Mahler, *A unified foundation for data fusion*, Selected Papers on Sensor and Data Fusion, SPIE **124** (1996), 325–345.
- [33] Mukund Narasimhan and Jeff Bilmes, *A supermodular-submodular procedure with applications to discriminative structure learning*, Proceedings of the 21th conference on uncertainty in artificial intelligence (UAI05) (2005).
- [34] Nuria Oliver and Eric Horvitz, *Selective perception policies for guiding sensing and computation in multimodal systems: A comparative analysis*, Proceedings of the Fifth International Conference on Multimodal Interaction (2003), 36–43.
- [35] ———, *S-seer: Selective perception in a multimodal office activity recognition system*, First International Workshop on Machine Learning for Multimodal Interaction (MLMI2004) (2004), 122–135.
- [36] P. Pudil, J. Novovi, and J. Kittler, *Floating search methods in feature selection*, Pattern Recognition Letter **15** (1994), no. 11, 1119–1125.
- [37] R.D. Rimey and C.M. Brown, *Task-specific utility in a general bayes net vision system*, IEEE Conf. on Computer Vision and Pattern Recognition (1992), 142–147.
- [38] R.Rimey and C.Brown, *Control of selective perception using bayes nets and decision theory*, Int. Journal of Computer Vision **12** (1994), 173–207.

- [39] P. Somol, P. Pudil, and J. Kittler, *Fast branch & bound algorithms for optimal feature selection*, PAMI **26** (2004), no. 7, 900–912.
- [40] M. Sviridenko, *A note on maximizing a submodular set function subject to knapsack constraint*, Operations Research Letters **32** (2004), 41–43.
- [41] L. van der Gaag and M. Wessels, *Selective evidence gathering for diagnostic belief networks*, AISB Quarterly **86** (1993), 23–34.
- [42] Hanbiao Wang, Kung Yao, Greg Pottie, and Deborah Estrin, *Entropy-based sensor selection heuristic for target localization*, IPSN'04: Proceedings of the third international symposium on Information processing in sensor networks, 2004, pp. 36–45.
- [43] A. Whitney, *A direct method of non parametric measurement selection*, IEEE Trans. Computing **C20** (1971), 11001103.
- [44] H. Wu and A. Cameron, *A bayesian decision theoretic approach for adaptive goal-directed sensing*, ICCV **90** (1990), 563–567.
- [45] Yongmian Zhang, Qiang Ji, and Carl Looney, *Active information fusion for decision making under uncertainty*, Proc. of the Int. Conf. on Information Fusion 2002 **1** (2002), 643–650.
- [46] A.X. Zheng, I. Rish, and A. Beygelzimer, *Efficient test selection in active diagnosis via entropy approximation*, UAI-05 (2005).