

Target Localization Based on Energy Considerations in Distributed Sensor Networks

Yi Zou and Krishnendu Chakrabarty
Department of Electrical and Computer Engineering
Duke University
Durham, NC 27708, USA
E-mail: {yz1, krish}@ee.duke.edu

Abstract—Wireless distributed sensor networks (DSNs) are important for a number of strategic applications such as coordinated target detection, surveillance, and localization. Energy is a critical resource in wireless sensor networks and system lifetime needs to be prolonged through the use of energy-conscious sensing strategies during system operation. We propose an energy-aware target detection and localization strategy for cluster-based wireless sensor networks. The proposed method is based on an *a posteriori* algorithm with a two-step communication protocol between the cluster head and the sensors within the cluster. Based on a limited amount of data received from the sensor nodes, the cluster head executes a localization procedure to determine the subset of sensors that must be queried for detailed target information. This approach reduces both energy consumption and communication bandwidth requirements, and prolongs the lifetime of the wireless sensor network. Simulation results show that a large amount of energy is saved during target localization using this strategy.

Index Terms—Cluster-based sensor networks, communication protocol, energy management, event reporting.

I. INTRODUCTION

Wireless sensor networks are important for a number of strategic applications such as coordinated target detection, surveillance, and localization [3]. Energy management in these networks is crucial since battery-driven sensor nodes are severely energy-constrained. Considerable research has been recently carried out in an effort to make sensor network energy-efficient. In [2], a mathematical model is presented to determine a bound on sensor network lifetime, with and without sensing activities. The hardware-based energy model for transmission and reception described in [1] is widely used as the basic energy consumption model for a wireless sensor network node. Heinzelman et al. [6] proposed a cluster-based routing algorithm called LEACH as an energy-efficient communication protocol for wireless sensor networks. The self-selected cluster heads collect raw data from the neighboring sensing nodes, aggregate them by data fusion methods, and transmit the aggregated data back to base stations for higher level processing. PEGASIS, an improvement over LEACH, is another example of an energy-aware protocol [10], which tends to increase the sensor network lifetime by decreasing the bandwidth via local collaboration among nodes. Another

example is the TEEN protocol proposed in [5]. Dynamic power management [4] has also been used for the design of energy-efficient wireless sensor networks. Other related work includes energy-saving strategies for the link layer [11], data aggregation [8], and system partitioning [9]. Sensor deployment for collaborative target detection is discussed in [13], where path exposure is used as a measure of the effectiveness of the sensor deployment. In [14], the authors propose a dual-space approach to event tracking and sensor resource management.

We focus here on reducing energy consumption in wireless sensor networks for target localization. In general, a sensor network has an almost constant rate of energy consumption if no target activities are detected in the sensor field [2]. The minimization of energy consumption for an active sensor network with target activities is more complicated since target detection involves collaborative sensing and communication involving different nodes. The transmission of detailed target information consumes a significant amount of energy due to the large volume of raw data. Contention for the limited bandwidth among the shared wireless communication channels causes additional delay in the relaying of detailed target information to the cluster head. In this work, we attempt to prolong the sensor network lifetime by adopting a new target localization procedure. We propose an *a posteriori* energy-aware target localization strategy, which is based on a two-step communication protocol between the cluster head and the sensors reporting the target detection events. In the first step, sensors detecting a target report the event to the cluster head using a very short binary yes/no message. The cluster head subsequently queries a subset of sensors that are in the vicinity of these likely target positions. This subset is determined from the localization procedure executed by the cluster head. Our simulation results show that a large amount of energy is saved by using the proposed target localization procedure. The simulation also illustrates the built-in advantages of the proposed target localization procedure in reducing the communication bandwidth and filtering out false alarms.

The organization of the rest of the paper is as follows. In Section II, we present the preliminaries and assumptions. In Section III, we present details of the energy-aware target localization approach. In Section IV, we present simulation

results, comparing the proposed approach to a base method in which every sensor always reports compete target data to the cluster head. Section V concludes the paper and outlines directions for future work.

II. PRELIMINARIES

A. Assumptions

We make the following assumptions in this paper,

- All sensor nodes are able to communicate with the cluster head after initial deployment. The cluster head also knows the location of the sensors through an initial setup process.
- The cluster head has more computation power than the sensor nodes. The cluster-based approach in this work assumes that the cluster head is responsible for computation, and the sensor nodes are functioning mainly as data collection devices.
- To simplify the energy analysis, the time for sending a certain amount of data is assumed to be the same as the time for receiving the same amount of data. The distance from the different nodes to the cluster head is ignored in the discussion on energy consumption. Also, all sensor nodes are assumed to be homogeneous, therefore the energy consumption for sensing is the same to each sensor node.
- To focus on energy consumption due to external activities in wireless sensor network, this work does not consider the energy consumed by sensor nodes when they are in the idle state. This does not however imply that the energy consumption of idle sensor nodes can always be ignored.

B. Sensor Detection Model

The sensor detection model converts the physical sensing signals to probability-based values in evaluating the confidence level about the data collected by the sensor. Consider an n by m sensor field grid and assume that there are k sensors deployed in the initial sensor deployment stage. Each sensor has a detection range r . Assume that sensor $s_i, 1 \leq i \leq k$ is deployed at point (x_i, y_i) . For any point P with coordinates (x, y) on the sensor field grid, we denote the Euclidean distance between s_i and P as $d(s_i, P)$, i.e. $d(s_i, P) = \sqrt{(x_i - x)^2 + (y_i - y)^2}$. Equation (1) shows the probability-based sensor detection model [12] that expresses the coverage $c_{xy}(s_i)$ of the grid point P at (x, y) by sensor s_i :

$$c_{xy}(s_i) = \begin{cases} 0, & \text{if } r + r_e \leq d(s_i, P) \\ e^{-\lambda a^\beta}, & \text{if } r_e > |r - d(s_i, P)| \\ 1, & \text{if } r - r_e \geq d(s_i, P) \end{cases} \quad (1)$$

where $r_e (r_e < r)$ is a measure of the uncertainty in sensor detection, $a = d(s_i, P) - (r - r_e)$, and λ and β are parameters that measure detection probability when a target is at distance greater than $r - r_e$ but within a distance $r + r_e$ from the sensor. The probabilistic sensor detection model is shown in Fig. 1. Note that distances are measured in units of grid points. Fig. 1 also illustrates the translation of a distance response from a sensor to the confidence level as a probability value about this sensor response. The discontinuities of $c_{xy}(s_i)$ at $r - r_e$

and $r + r_e$ are simplified translations for the extreme cases of too much and too little confidence in sensor responses. They reduce the complexity of the sensor model without significant degradation in the translation. Different values of the parameters α and β yield different translations reflected by different detection probabilities, which can be viewed as the characteristics of various types of physical sensors. This model reflects the behavior of range sensing devices such as infrared and ultrasound sensors. Since the proposed target localization approach is independent of the sensor model, alternative sensor models can also be used.

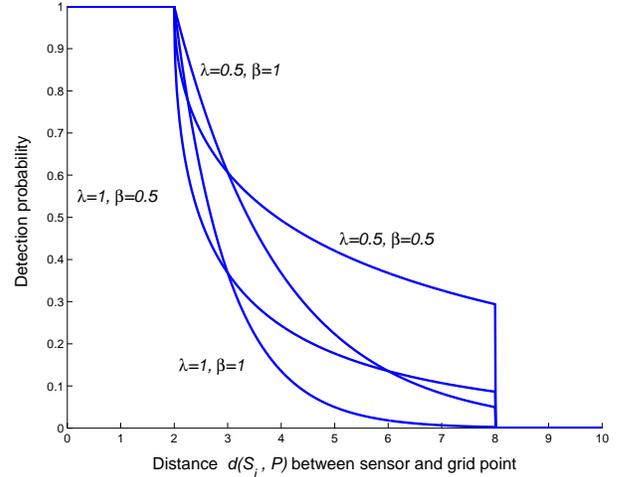


Fig. 1. Probabilistic sensor detection model.

C. Energy Consumption Model

Based on [1], [2], [7], a simplified sensor node energy consumption model is used here as a metric for evaluating energy consumption. Suppose a sensor node has three basic energy consumption types—sensing, transmitting and receiving, and these power values (joule per second) are ψ_s , ψ_t and ψ_r , respectively. Assume at time instant t , there are $k(t)$ sensors that have detected the target, where $k(t) \leq k$. Therefore, the energy for sensing activities in the wireless network, denoted as $E_s(k(t))$, is

$$E_s(k(t)) = k(t)\psi_s T_s \quad (2)$$

where T_s is the time duration that a sensor node is involved in sensing. For a fixed time interval, E_s is a constant if all sensor nodes are assumed to be homogenous. The energy used for communication between nodes and the cluster head can be categorized into two types, E_b and E_c . The parameter E_c is the energy consumed by a sensor node for communication with the cluster head. This includes the energy for transmitting data and the energy for receiving data. The parameter E_b is the energy needed for broadcasting data from the head to the nodes. Both E_b and E_c are functions of T and $k(t)$, where T is the time required for either retrieving data from a sensor node or the broadcasting of data from the cluster head, and $k(t)$ is the number of sensors involved in this communication

at time instant t . We define E_c and E_b as follows:

$$E_c(k(t), T) = (\psi_t T + \psi_r T)k(t) \quad (3)$$

$$E_b(k(t), T) = \psi_t T + \psi_r T k(t) \quad (4)$$

The parameter T is directly proportional to the volume of data involved in the communication. In this work, T can be one of three values, T_d for raw target data, T_e for target event reporting, and T_q for query request. They satisfy the relationship $T_e \leq T_q \ll T_d$ since raw data collected by a sensor node can be up to hundreds of bytes in size. We assume that target detection and localization are discrete processes, which are derived from a discrete sampling of target activities in the sensor network. Also, since the sensor network is designed to track target activities, T_s , T_e , T_q , and T_d are assumed to be less than the granularity of the time t . Thus for the case that a target is moving in the sensor field during the time interval $[t_{start}, t_{end}]$, the corresponding instantaneous energy consumption $E(t)$ and total energy consumption E in the wireless sensor network can be expressed as

$$E(t) = E_s(k(t)) + E_c(k(t), T_d) \quad (5)$$

$$E = \sum_{t=t_{start}}^{t_{end}} E(t). \quad (6)$$

III. TARGET LOCALIZATION PROCEDURE

In our two-step communication protocol, when a sensor detects a target, it sends an event notification to the cluster head. In order to conserve power and bandwidth, the message from the sensor to the cluster head is kept very small; in fact, the presence or absence of a target can be encoded in just one bit. Detailed information such as detection strength level, imagery and time series data are stored in the local memory and provided to the cluster head upon subsequent queries. Based on the information received from the sensors within the cluster, the cluster head executes a probabilistic localization algorithm to determine candidate target locations, and it then queries the sensor(s) in the vicinity of the target.

A. Detection Probability Table

The *a posteriori* side of this method is based on the detection probability table, which contains entries for all possible detection reports from those sensors that can detect a target at all grid points. The detection probability table is created by the cluster head for each grid point. Let us assume that a grid point $P(x, y)$ is covered by a set of k_{xy} sensors, denoted as S_{xy} , $|S_{xy}| = k_{xy}$, $0 \leq k_{xy} \leq k$, and $S_{xy} \subseteq \{s_1, s_2, \dots, s_k\}$. The probability table is built on the power set of S_{xy} including the event that none of the sensors detect anything (represented by the binary string as "00...0") as well as the event that all of the sensors (represented by the binary string as "11...1"). Thus the probability table for grid point (x, y) then contains $2^{k_{xy}}$ entries, defined as:

$$p_{table_{xy}}(i) = \prod_{s_j \in S_{xy}} p_{xy}(s_j, i) \quad (7)$$

where $0 \leq i \leq 2^{k_{xy}}$, and $p_{xy}(s_j, i) = c_{xy}(s_j)$ if s_j detects a target at $P(x, y)$; otherwise $p_{xy}(s_j, i) = 1 - c_{xy}(s_j)$.

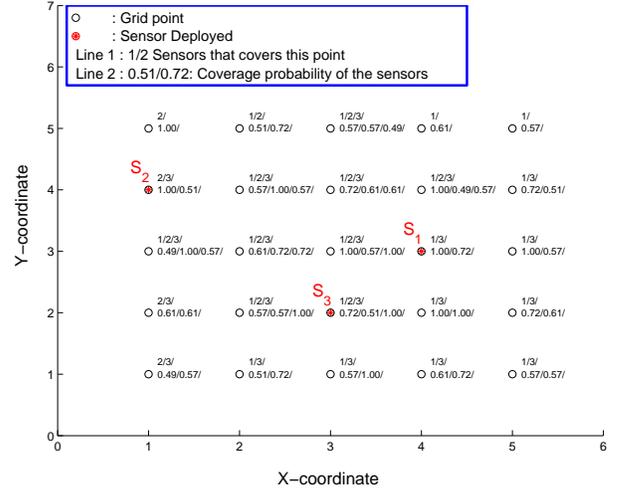


Fig. 2. The detection probability table.

Consider the grid point $(2, 4)$ in Fig. 2 which is covered by all three sensors s_1 , s_2 and s_3 with probabilities as 0.57, 1, and 0.57 respectively. For the three sensors s_1 , s_2 and s_3 , there are a total of 8 possibilities for their combined event detection at grid point $(2, 4)$. For example, the binary string 110 denotes the possibility that s_1 and s_2 report a target but s_3 does not report a target. For each such possibility $d_1 d_2 d_3$ ($d_1, d_2, d_3 \in \{0, 1\}$) for a grid point, we calculate the conditional probabilities that the cluster head receives $d_1 d_2 d_3$ given that a target is present at that grid point. For our example, these conditional probabilities are listed in Table I. Consider the binary string 110, the conditional probability associated with this possibility is given by $p_{table_{24}}(6) = p_{24}(s_1, 6)p_{24}(s_2, 6)p_{24}(s_3, 6) = 0.57 \times 1 \times (1 - 0.57) = 0.24$. Note that the probability table generation is only a one-time cost. Once the probability table is generated, there is no need to refresh it unless sensor locations are changed. Table I gives an example of the probability tables of Fig. 2.

TABLE I
EXAMPLE PROBABILITY TABLE.

| i | $d_1 d_2 d_3$ | $p_{table_{xy}}(i)$, $0 \leq i < 2^{k_{xy}}$, $k_{xy} = 3$ |
|-----|---------------|--|
| 0 | 000 | $(1 - 0.5736) \times (1 - 1) \times (1 - 0.5736) = 0.0$ |
| 1 | 001 | $(1 - 0.5736) \times (1 - 1) \times 0.5736 = 0.0$ |
| 2 | 010 | $(1 - 0.5736) \times 1 \times (1 - 0.5736) = 0.1819$ |
| 3 | 011 | $(1 - 0.5736) \times 1 \times 0.5736 = 0.2446$ |
| 4 | 100 | $0.5736 \times (1 - 1) \times (1 - 0.5736) = 0.0$ |
| 5 | 101 | $(1 - 0.5736) \times 1 \times (1 - 0.5736) = 0.0$ |
| 6 | 110 | $0.5736 \times 1 \times (1 - 0.5736) = 0.2446$ |
| 7 | 111 | $0.5736 \times 1 \times 0.5736 = 0.3290$ |

B. Score-based Ranking

Suppose $S_{rep}(t)$ is the set of sensors that have reported the detection of an object at time t , $S_{rep,xy}(t)$ is the set of sensors that can detect a target at point $P(x, y)$ and have also reported the detection of an object at time t , and the set of the sensors selected by the cluster head for querying at time t is denoted as $S_q(t)$. Their cardinalities are given as $k_{rep}(t)$, $k_{rep,xy}(t)$,

and $k_q(t)$, respectively. Obviously, $S_{rep,xy}(t) = S_{rep}(t) \cap S_{xy}$ and $S_q(t) \subseteq S_{rep}(t)$. We use an inference method based on the established probability table. To save both communication energy and bandwidth, at any time instant t during the target localization process, the cluster head uses the probability table to determine $k_q(t)$ most suitable sensors out of the reported $k_{rep}(t)$ sensors to be queried for more detailed information. The score of the grid point $P(x, y)$ at time instant t is calculated as follows:

$$SCORE_{xy}(t) = p_table_{xy}(i(t)) \times w_{xy}(t) \quad (8)$$

where $i(t)$ is the index of the p_table_{xy} at time t . The parameter $i(t)$ is calculated from S_{xy} and $S_{rep,xy}$. The parameter $p_table_{xy}(i(t))$ corresponds to the conditional probability that the cluster head receives this event information given that there was a target at $P(x, y)$. The weight $w_{xy}(t)$ reflects the confidence level in this reporting event for this particular grid point and it refines the grid point scores to narrow down grid points that are most probably close to the current target location. $w_{xy}(t)$ is defined as:

$$w_{xy}(t) = \begin{cases} 0 & \text{if } S_{rep,xy}(t) = \{\phi\} \\ 4^{-\Delta k_{rep,xy}(t)} & \text{otherwise} \end{cases} \quad (9)$$

where $\Delta k_{rep,xy}(t)$ measures the degree of difference in the set of sensors that reported and those sensors that can detect point $P(x, y)$ at time instant t . $\Delta k_{rep,xy}(t)$ is given as:

$$\Delta k_{rep,xy}(t) = |k_{rep}(t) - k_{rep,xy}(t)| + |k_{rep}(t) - k_{xy}| \quad (10)$$

In Equation (10), the first term $|k_{rep}(t) - k_{rep,xy}(t)|$ represents the absolute difference between the number of nodes that have reported, i.e. $k_{rep}(t)$, and the number of nodes that have reported and can also detect a target at point (x, y) , i.e. $k_{rep,xy}$. The second term $|k_{rep}(t) - k_{xy}|$ represents the absolute difference between the number of nodes that have reported, i.e. $k_{rep}(t)$, and the number of nodes that can detect the grid point (x, y) from the pre-calculated detection probability table, i.e. k_{xy} . The parameter w_{xy} is a decaying factor that is 1 only if $S_{rep}(t) = S_{xy}$. The number 4 in Equation (9) was chosen empirically through simulations. We are using $w_{xy}(t)$ to filter out grid points that are not likely to be close to the actual target location. The score is based on both the probability value from the probability table and the current relationship between $S_{rep}(t)$, $S_{rep,xy}(t)$ and S_{xy} . Consider the 10 by 10 grid shown in Fig. 3. There are five sensors deployed, $k = 5$, $r = 2$ and $r_e = 1$. The zigzag shaped line is the target movement trace. The target starts to move at $t = t_{start}$ from the grid point marked as ‘‘Start’’ and finishes at $t = t_{end}$ at the grid point marked as ‘‘End’’. Table II gives some score calculation examples for the grid points in Fig. 3 at the time instant t_{start} .

C. Selection of Sensors to Query

Assume that the maximum number of sensors that are allowed to report an event is k_{max} . To select the sensor to query based on the event reports and the localization procedure, we first note that for time instant t , if $k_{max} \geq$

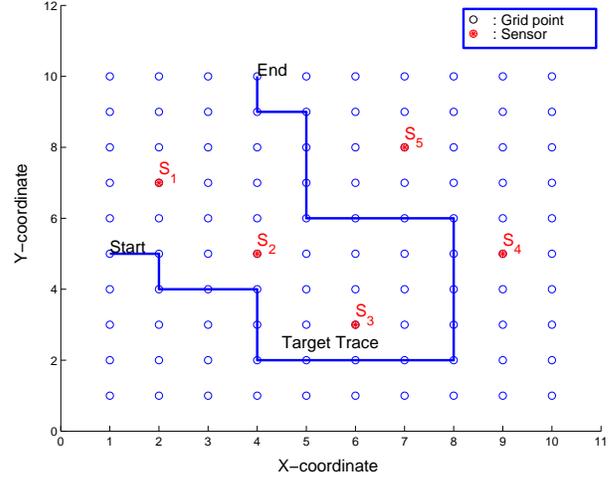


Fig. 3. Examp sensor field with a moving target.

TABLE II
SCORING CALCULATION EXAMPLE AT t_{start} .

| (x, y) | (1, 6) | (2, 6) | (3, 6) | (4, 6) |
|-------------------------------|--------|------------|------------|------------|
| S_{xy} | s_1 | s_1, s_2 | s_1, s_2 | s_1, s_2 |
| $S_{rep,xy}(t_{start})$ | s_1 | s_1, s_2 | s_1, s_2 | s_1, s_2 |
| $w_{xy}(t_{start})$ | 0.25 | 1.00 | 1.00 | 1.00 |
| $p_table_{xy}(i(t_{start}))$ | 0.7248 | 0.5736 | 0.5254 | 0.5736 |
| $SCORE_{xy}(t_{start})$ | 0.0453 | 0.5736 | 0.5254 | 0.5736 |

$k_{rep}(t)$, then all reported sensors can be queried. Otherwise, we select sensors based on a score-based ranking. The sensors selected correspond to the ones that have the shortest distance to those grid points with the highest scores. This selection rule is defined as:

$$S_q(t) : d(S_q(t), P_{MS}) = \min\{d(s_i, P_{MS})\} \quad (11)$$

where $S_i \in S_{rep}(t)$, and P_{MS} denotes the set of grid points with the highest scores. Note it is possible that there are multiple grid points that have the maximum score. When this happens, we calculate the score concentration by averaging the scores of the current grid point and its eight neighboring grid points. The grid point with the highest score (or the score concentration) is the most likely current target location. Therefore, selecting sensors that are closest to this point guarantee that the selected sensors can provide the most detailed and accurate data in response to the subsequent queries. Note target identification is not possible as at this stage since the cluster head has no additional information other than $S_{rep}(t)$. However, the selected sensors provide enough information in the subsequent stage to facilitate target identification. We evaluate the accuracy of this target localization procedure by calculating the distance between the grid point with the highest score and the actual target location. For the example of Fig. 3, Table III gives some results for the selected sensor when the target is moving from ‘‘Start’’ ($t = 1$) to ‘‘End’’ with $k_{max} = 1$. From Equation (2) to Equation (6), we evaluate the energy consumption using the above target localization method as following:

$$E^*(t) = E_s(k_{rep}(t)) + E_c(k_{rep}(t), T_e)$$

TABLE III
SELECTED SENSORS IN FIG. 3.

| t | $S_{rep}(t)$ | $S_q(t)$ | t | $S_{rep}(t)$ | $S_q(t)$ |
|-----|--------------|----------|-----|--------------|----------|
| 1 | s_1, s_2 | s_1 | 2 | s_1, s_2 | s_1 |
| 3 | s_1, s_2 | s_1 | 4 | s_2 | s_2 |
| 5 | s_2, s_3 | s_3 | 6 | s_2, s_3 | s_3 |
| 7 | s_2, s_3 | s_3 | 8 | s_3 | s_3 |
| ... | ... | ... | ... | ... | ... |
| 21 | s_5 | s_5 | 22 | s_5 | s_5 |
| 23 | s_1 | s_1 | 24 | NONE | NONE |

$$+ E_b(k_q(t), T_q) + E_c(k_q(t), T_d) \quad (12)$$

$$E^* = \sum_{t=t_{start}}^{t_{end}} E^*(t) \quad (13)$$

Therefore, let $k(t) = k_{rep}(t)$ in Equations (2) and (3), the difference in energy consumption, $\Delta E = E - E^*$ can be expressed as:

$$\begin{aligned} \Delta E(t) &= (k_{rep}(t) - k_q(t))(\psi_t + \psi_r)T_d \\ &\quad - (k_q(t)\psi_r + \psi_t)T_q \\ &\quad - k_{rep}(t)(\psi_t + \psi_r)T_e \end{aligned} \quad (14)$$

$$\Delta E = \sum_{t=t_{start}}^{t_{end}} \Delta E(t) \quad (15)$$

The last two terms in Equation (14) indicate the overhead for the proposed target localization procedure. Since $T_d \gg T_e$, and $T_d \gg T_q$, the overhead is small. As $k_q < k_{max}$, with k_{max} properly selected, from Equation (14) and Equation (15), energy consumption is greatly reduced with the passage of time.

D. Procedural Description

Fig. 4 shows the pseudocode of the procedure to generate the probability table for each grid point. Fig. 5 shows the pseudocode for the simulation of the probabilistic localization algorithm. For an n by m grid with k sensors, the computational complexity involved in generating the probability table is $O(nm2^k)$ since the maximum number of sensors that can detect a grid point is k for the worst case. The computational complexity of the localization procedure is $O(nmk_{max})$, $k_{max} \leq k$. Therefore, the computational complexity of the probabilistic localization algorithm is $\max\{O(nmk_{max}), O(nm2^k)\} = O(nm2^k)$. Even though the worst-case complexity of the localization procedure is exponential in k , in practice, the localization procedure can execute in less time since the number of sensors that can effectively detect a target at a given grid point is quite small.

IV. CASE STUDY AND DISCUSSIONS

We present results for a case study carried out using MatLab. The simulation is done on a 30 by 30 sensor field grid with 20 sensors randomly placed in the sensor field. The parameters of the sensor detection model are $r = 5$, $r_e = 4$, $\lambda = 0.5$, and $\beta = 0.5$. We choose energy the consumption model parameters as $\psi_r \approx 400$ nJ/sec, $\psi_t \approx 400$ nJ/sec, and

Procedure

Generate_Probability_Table($P(x, y), \{s_1, \dots, s_k\}$)

```

1 /* find  $S_{xy}$ , the set of sensors that can detect  $P(x, y)$  */
2 For  $i \in \{s_1, s_2, \dots, s_k\}$ 
3   If  $d(s_i, P(x, y)) \leq r + r_e$ 
4      $S_{xy} = S_{xy} \cup \{s_i\}$ ;
5 End
6 End
7 /* fill up the probability table */
8 For  $i, 0 \leq i \leq k_{xy}, k_{xy} = |S_{xy}|$ ;
9   If  $s_j$  detects  $P(x, y)$ 
10    Set  $p_{xy}(s_j, i) = c_{x,y}(s_j)$ ;
11 Else
12   Set  $p_{xy}(s_j, i) = 1 - c_{x,y}(s_j)$ ;
13 End
14 Set  $p\_table_{xy}(i) = \prod_{s_j \in S_{xy}} p_{xy}(s_j, i)$ ;
15 End

```

Fig. 4. Pseudocode for generating the detection probability table.

Procedure

Target_Localization(Grid, $\{s_1, \dots, s_k\}$, TargetTrace)

```

/*  $k_{max}$  is the maximum number of sensors that are allowed
for querying,  $p_{rep}$  is the threshold level for a sensor to report
to the cluster head of an event. TargetTrace starts at  $t_{start}$ 
and it ends at  $t_{end}$ . The simulation time unit is 1. */
1 Set  $t = t_{start}$ ;
2 While ( $t \leq t_{end}$ )
3   /* current target location */
4   Set Target = TargetTrace( $t$ );
5   /* calculate the scores */
6   Calculate  $S_{rep}(t)$  from  $\{s_1, s_2, \dots, s_k\}$ , Target( $t$ ),  $p_{rep}$ ;
7   Set  $k_{rep}(t) = |S_{rep}(t)|$ ;
8   For  $P(x, y)$  in Grid,  $x \in [1, width]$ ,  $y \in [1, height]$ 
9     Set  $k_{xy} = |S_{xy}|$ ;
10    Calculate  $S_{rep,xy}(t)$  from  $S_{rep}(t)$  and  $P(x, y)$ ;
11    Calculate the index  $i(t)$  of  $p\_table_{xy}$  from
 $S_{rep}(t)$ , and  $S_{rep,xy}(t)$ ;
12    Set  $k_{rep,xy}(t) = |S_{rep,xy}(t)|$ ;
13    If  $S_{rep,xy}(t) = \{\phi\}$ 
14       $w_{xy}(t) = 0$ ;
15    Else
16      Set  $\Delta k_{rep,xy}(t) = |k_{rep}(t) - k_{rep,xy}(t)|$ 
+  $|k_{rep}(t) - k_{xy}|$ ;
17       $w_{xy}(t) = 4^{-\Delta k_{rep,xy}(t)}$ ;
18    End
19    Set  $SCORE_{xy}(t) = p\_table_{xy}(i(t)) \times w_{xy}(t)$ ;
20  End
21  /* select sensors for querying */
22  Calculate  $S_q(t)$  from  $SCORE_{xy}(t)$  and  $k_{max}$ ,
 $x \in [1, width]$ ,  $y \in [1, height]$ ;
23  /* next time instant */
24  Set  $t = t + 1$ ;
25 End

```

Fig. 5. Pseudocode of the target localization procedure.

$\psi_s \approx 1000$ nJ/sec. These values are based on the typical values given in [1], [2], [7], assuming the sensing rate for a the sensor is 8 bits/sec. We have no physical data available for T_d and T_e ; however, their values do not affect the target localization

procedure, therefore we only need to set them manually to satisfy the relationship $T_d \gg T_e$ and $T_d \gg T_q$. In this case, $T_d = 100$ ms, $T_e = 2$ ms, and $T_q = 4$ ms.

A. Random Target Movement Trace

The layout of the sensor field is given in Fig. 6, with a target trace randomly generated in the sensor field. The target travels from the position marked as ‘‘Start’’ to the position marked as ‘‘End’’. We assume the target locations are updated at discrete time instants in unit of seconds, and the granularity of time is long enough for sampling by two neighboring locations in the target trace with negligible errors. We have evaluated the algorithm for $k_{max} = 1$, $k_{max} = 2$, and $k_{max} = 3$. Fig. 7 presents the instantaneous energy saving in percentage, and Fig. 8 presents the absolute value of the cumulative energy saving for the case study as the target moves along its trace in the sensor field. The energy savings are compiled relative to the base case when all sensors report complete target information in one step everywhere.

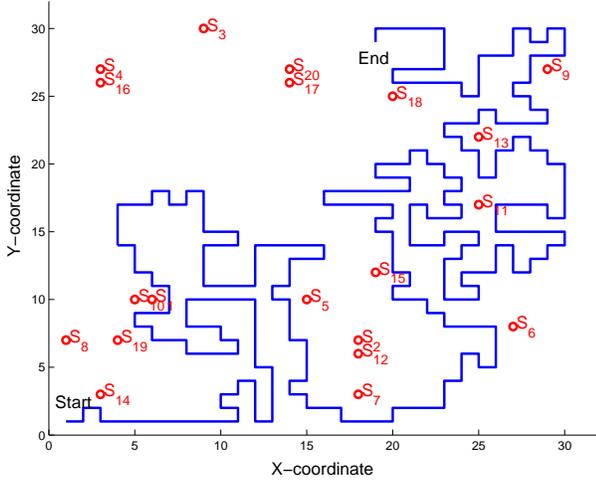


Fig. 6. Sensor field layout with target trace.

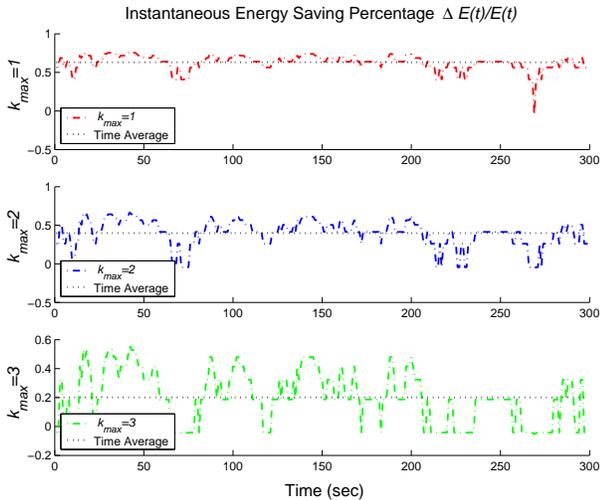


Fig. 7. Instantaneous energy saving percentage during target localization relative to the ‘‘always report’’ one-step base case.

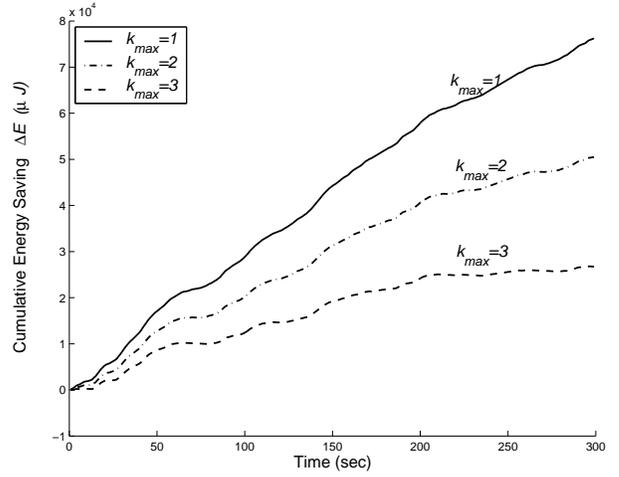


Fig. 8. Cumulative energy saved during target localization relative to the ‘‘always report’’ one-step base case.

From Fig. 7 and Fig. 8, we note that a large amount of energy is saved during target localization. Note that when k_{max} approaches $k_{rep}(t)$, the saving is less apparent due to the additional communication overhead of the two-stage query protocol. Nevertheless, there is still a considerable amount of energy saved in target localization, even for the case that $k_{max} = 3$. With an appropriate selection of k_{max} , the proposed algorithm performs exceptionally well.

Next, we consider the latency in the localization of a target by the cluster head. By latency, we refer to the time that it takes for the cluster head to collect the detailed target information from sensor nodes from the time sensor nodes detect an event, assuming that the wireless sensor network uses the time division multiple access (TDMA) protocol [3]. The results are shown in Fig. 9. The latency is reduced here compared to the base case using a ‘‘report once’’ strategy, since a large amount of communication for transmitting raw data has been reduced to a smaller amount of data sent by a selected set of sensors. This is an added advantage to the proposed energy-aware target localization procedure.

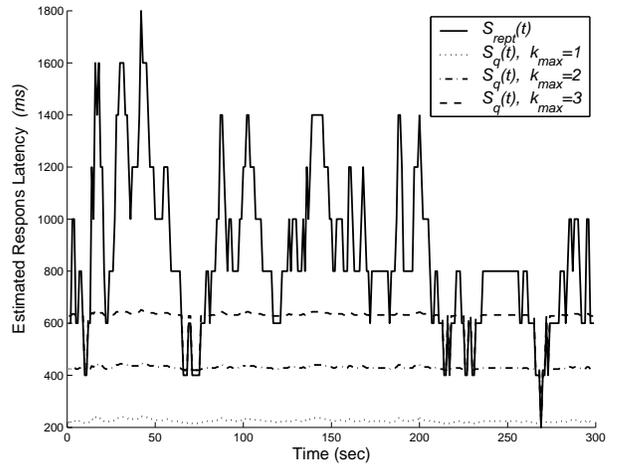


Fig. 9. Latency in the localization of a target by the cluster head.

Since the selection of sensors for querying is based on

both the detection probability table and the distance of sensors from the estimated high-score points, the proposed *a posteriori* approach offers another important advantage. It provides a substantial amount of built-in false-alarm filtering. Fig. 10 illustrate the false-alarm filtering ability of the proposed approach. We manually generated false alarms reported by some malfunctioning sensors, which are during $t \in [18, 22]$ by s_4 , during $t \in [138, 142]$ by s_{16} , and during $t \in [239, 241]$ by s_8 . We calculate the distance d of the target from the sensor in $S_{rep}(t)$ that is farthest from it, as well as the distance d^* of the target from the sensor in $S_q(t)$ that is farthest from it. The difference $d - d^*$ is used as a measure of the built-in filtering ability. Fig. 10 shows the variation of $d - d^*$ with time. Note the fact that prior to querying, the cluster head only knows which sensors have reported the detection of a target, and there is no information available to the cluster head about any detailed information of the target. We find that the proposed approach successfully narrows down the sensors that are the close to the real target location, and selects them for detailed information querying. As shown in Fig. 10, the three spikes present the fact that the false alarms from the sensor (which in this case is the furthest sensor from the actual target location) have been filtered out since the proposed target localization procedure is still able to select the most appropriated sensors to query for detailed target information.

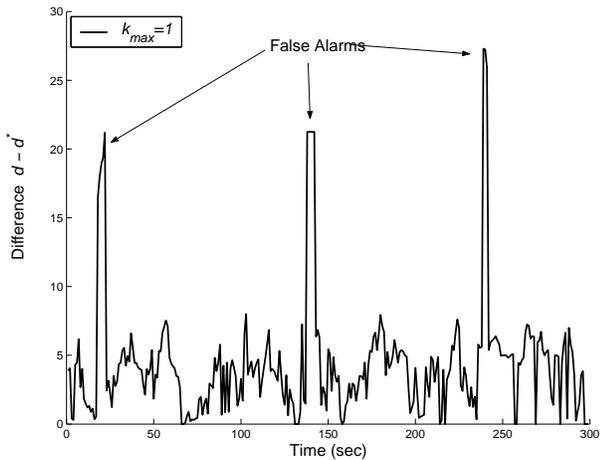


Fig. 10. Results on localization error in the presence of false alarms.

B. Random Target Activity Events

Next, we present simulation results for the case of a target appearing at random locations in the sensor field with a constant event rate. The layout of the sensor field is the same as given by Fig. 6. We generate a sequence of 200 random target locations on the sensor field to simulate a target appearing randomly for 200 discrete time instants. Simulation is carried out for $k_{max} = 1$, $k_{max} = 2$, and $k_{max} = 3$, with different sets of randomly generated target activity events at 200 discrete time instants. Fig. 11 presents the instantaneous energy saving in percentage, and Fig. 12 presents the absolute value of the cumulative energy saving. Fig. 13 presents the latency evaluation results for target localization. The proposed target

localization strategy performs extremely well since energy is saved for the case of random target appearance events, and latency is reduced for target localization.

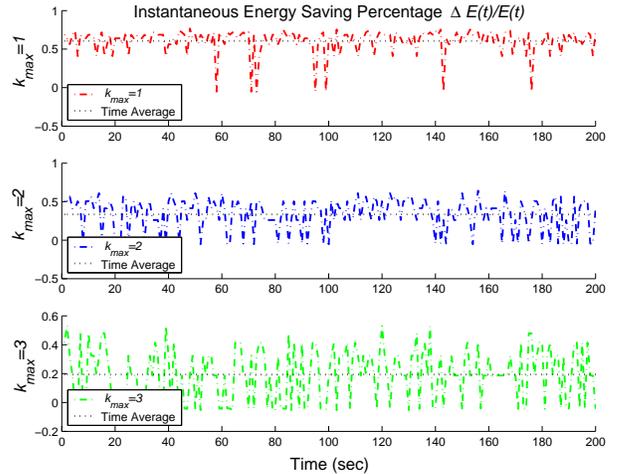


Fig. 11. Instantaneous energy saving percentage during target localization relative to the “always report” one-step base case.

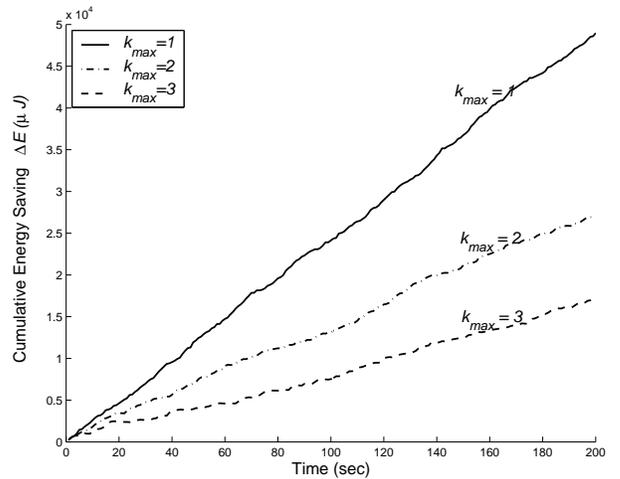


Fig. 12. Cumulative energy saved during target localization relative to the “always report” one-step base case.

V. CONCLUSIONS

We have described an energy-aware target localization procedure for cluster-based wireless sensor networks. The proposed approach is based on the combination of a two-step communication protocol between the cluster head and the sensors in the cluster, and a probabilistic localization algorithm. We have shown that this approach reduces energy consumption, decreases the latency for target localization, and provides a mechanism for filtering false alarms. Our future work will focus on the scalability of these algorithms. We are currently extending this approach for the localization of multiple targets as well as determining an appropriate value of the parameter k_{max} . A larger value of k_{max} is desirable for more accurate target classification, but the need for additional information for classification must be balanced with the need

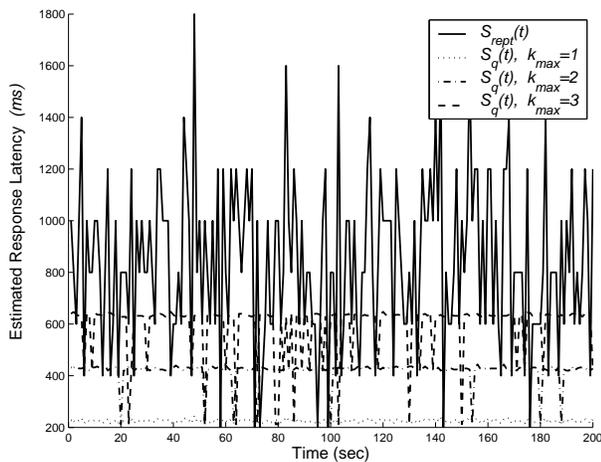


Fig. 13. Latency in the localization of a target by the cluster head.

for low energy, reduced latency, and bandwidth requirements. We are also considering more practical scenarios, e.g. when a sensor is temporarily blocked by external obstacles. These scenarios must be appropriately modeled in the sensor detection probability table.

ACKNOWLEDGMENT

This research was supported in part by ONR under grant no. N66001-00-1-8946. It was also sponsored in part by DARPA, and administered by the Army Research Office under Emergent Surveillance Plexus MURI Award No. DAAD19-01-1-0504. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of the sponsoring agencies.

The authors would also like to thank the anonymous reviewers for their valuable comments.

REFERENCES

- [1] T. Rappaport, *Wireless Communications: Principles & Practice*, New Jersey: Prentice-Hall, Inc., 1996.
- [2] M. Bhardwaj, T. Garnett and A. P. Chandrakasan, "Bounding the Lifetime of Sensor Networks Via Optimal Role Assignments", *Proc. IEEE Infocom*, vol. 3, pp. 1380-1387, 2001.
- [3] P. K. Varshney, *Distributed Detection and Data Fusion*, Springer, New York, NY, 1996.
- [4] A. Sinha and A. Chandrakasan, "Dynamic Power Management in Wireless Sensor Networks", *IEEE Design and Test of Computers*, vol. 18, pp. 62-74, 2001.
- [5] A. Manjeshwar and D. P. Agrawal, "TEEN: A Routing Protocol for Enhanced Efficiency in Wireless Sensor Networks", *Intl. Proc. 15th Parallel and Distributed Processing Symposium*, pp. 2009-2015, 2001.
- [6] W. B. Heinzelman, A. Chandrakasan and H. Balakrishnan, "Energy-efficient Communication Protocol for Wireless Micro Sensor Networks", *Proc. 33rd Annual Hawaii International Conference on System Sciences*, pp. 3005-3014, 2000.
- [7] W. B. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "An Application-Specific Protocol Architecture for Wireless Microsensor Networks", *IEEE Transactions on Wireless Communications*, vol. 1, pp. 660-670, 2002.
- [8] A. Wang, W. B. Heinzelman, A. P. Chandrakasan, "Energy-scalable Protocols for Battery-operated Micro Sensor Networks", *IEEE Workshop on Signal Processing Systems*, pp. 483-490, 1999.
- [9] A. Wang, W. B. Heinzelman, A. P. Chandrakasan, "An Energy-efficient System Partitioning for distributed Wireless Sensor Networks", *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 2, pp. 905-908, 2001.

- [10] S. Lindsey and C. S. Raghavendra, "PEGASIS: Power-Efficient Gathering in Sensor Information Systems", *Proc. IEEE Aerospace Conference*, vol. 3, pp. 1125 -1130, 2002.
- [11] E. Shih, B. H. Calhoun, H. C. Seong, and A. P. Chandrakasan, "An Energy-efficient Link layer for Wireless Micro Sensor Networks", *Proc. IEEE Computer Society Workshop on VLSI*, pp. 16-21, 2001.
- [12] A. Elfes, "Occupancy Grids: A Stochastic Spatial Representation for Active Robot Perception", *Proc. 6th Conference on Uncertainty in AI*, pp. 60-70, July 1990.
- [13] T. Clouqueur, V. Phipatanasuphorn, P. Ramanathan, and K. K. Saluja, "Sensor Deployment Strategy for Target Detection", *Proc. 1st ACM International Workshop on Wireless Sensor Networks and Applications*, pp. 42-48, September 2002.
- [14] J. Liu, P. Cheung, L. Guibas, and F. Zhao, "A Dual-Space Approach to Tracking and Sensor Management in Wireless Sensor Networks", *Proc. 1st ACM International Workshop on Wireless Sensor Networks and Applications*, pp. 131-139, September 2002.