Sensor Network Surveillance and Counter-surveillance with Power Constraint

Yu Hu
hu@ee.ucla.edu
Electrical Engineering Department
University of California, Los Angeles

ABSTRACT

Targeting power constrained surveillance and counter-surveillance for a given sensor network, this work first design a power efficient surveillance system by sensor nodes coordination and cooperation, and then solve the power constrained sensor node covering and scheduling (PCSCS) problem by generalized maximal network flow. The counter-surveillance problem is formulated as a hide and seek game, which can be solved efficiently and optimally by game theory. The simulation results show that our surveillance scheduling algorithm is adapted to a large range of different G values and C node velocities. The typical event detection rate is over 50% given G=10. Compared to random scheduling approach of counter-surveillance system, our proposed algorithm achieves up to 30% increase of non-detected travelling distance.

1. INTRODUCTION

When large-scale sensor networks were first developed, as early as in the 1960s, their use was intended for military surveillance, among other applications. To this day, surveillance remains a primary problem and the challenges it offers to research remain central to the leading experimental and theoretical research activities [1, 2, 3]. However, while these systems are primarily designed to operate against passive opponent targets, it must be recognized that opponents of a surveillance system may be called upon to implement autonomous, distributed methods that are intended to reduce the effectiveness of the surveillance system. We will refer such systems as counter-surveillance networks [4].

Most of the existing work on the surveillance sensor network [1, 2, 3] considers 2-D sensor network planning and there is no much literature considering counter-surveillance [4]. Note that 2-D sensor network offers more design freedom and therefore more flexible solutions are open to the designers. In this work, we attack 1-D sensor network for both surveillance and counter surveillance with sensoring obstacles under power constraints.

We consider two types of nodes in this system, surveillance, S, and counter-surveillance, C. There will also be obstacles, O, and a barrier, B. The barrier will include a series of narrow portals, P, enabling egress of nodes C. For simplicity, the node deployment, and other features will be linear and regular in geometry. Figure 1 shows the system placement. There are 100 S nodes, 100 portals and 100 C nodes, respectively. S nodes and portals are placed uniformly in a $1000m$ span.

For the surveillance system, due to the presence of obstacles, uncertainty exhibits during S nodes detection and identification. A successful identification requires at least two S nodes identify a C node in the same time. A user specified parameter G is used to describe the life time requirement of this system, where $G = Pwr_I/Pwr_A$, Pwr_A and Pwr_I are the average and maximal power dissipation of the system, respectively. The objective is to maximize the probability of the identification of C nodes given the power constraint G.

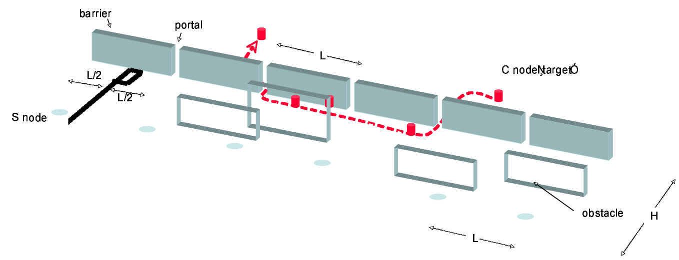


Figure 1: System Placement

For the counter-surveillance system, a C node has the capability to detect the RF energy from S nodes. Similarly to S nodes, uncertainty exhibits during such detection due to the random obstacles. A movement of a C node starts from a portal and ends in the other portal. The objective is to maximize the distance of movement without being detected and satisfy the power constraint specified by user-defined parameter G.

Our solution for the surveillance system is based on the coordination of multiple S nodes. Given G, we formulate the power constrained sensor node covering and scheduling (PCSCS) problem as a generalized maximal network flow [5, 6]. To tackle the high complexity (degree) of the flow network, we propose a simulated annealing based algorithm to obtain a near-optimal solution of the PCSCS problem. The simulation results show our surveillance scheduling algorithm is adapted to a large range of different G values and C node velocities. The typical event detection rate is over 50% given G=10.

Furthermore, we formulate the counter-surveillance problem as a hide and seek game [7] and can be solved efficiently and optimally by game theory [8]. Compared to random scheduling approach of counter-surveillance system, our proposed algorithm achieves up to 30% increase of non-detected travelling distance.

The organization of this paper is as follows. Section 2 presents our overall system design and single S node operating state machine for surveillance system. Section 3 proposes algorithms to solve power constrained sensor covering and scheduling problem, which is the key problem in our surveillance system. Section 4 describes a game theory based approach for counter-surveillance. The simulation methodology and results are given in Section 5. The paper is concluded in Section 6.

2. DESIGN FOR SURVEILLANCE SYSTEM

2.1 Overall System Design

The system level operation diagram of our surveillance system is shown in Figure 2. Initially (after synchronization), all S nodes are in state V and a subset of S nodes are waken up by timeout T. After waken up from V state , each S node S_i will enter C state. In C state, a node keeps detecting the presence of events by sensor A and waiting for messages from other nodes for t time. If no event is detected during t time, these S nodes will go back to state V. Otherwise, once an S node S_i detects an event, it will enter I state after sending out a broadcast message which encodes its node ID and the time-stamp that the event occurs. When the other S node S_j receives such a message, it will decide whether it should enter I state according to its location and the information in the message. If an event is identified successfully (scanned by at least two I sensors), S_j will send out a broadcast message to inform other active S nodes to re-calculate the issue rate of the event sequence and update the timeout period T accordingly.

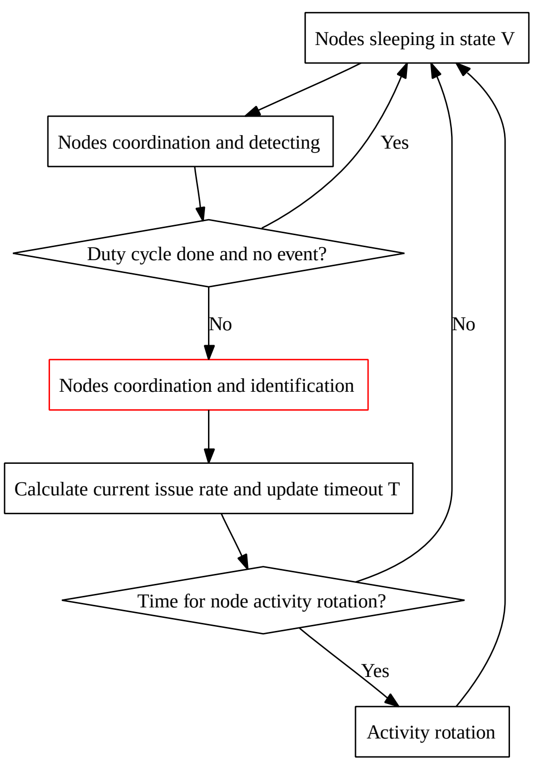


Figure 2: System level operation diagram

In our surveillance system, the subset of S nodes which will be activated periodically is decided statically. As all S nodes are geometrically symmetric, we pre-calculate several optimal patterns of S nodes and the corresponding duty cycles which can maximize identification probability under the power constraint (see subsection 3.3). To uniformly distribute energy consumption to all S nodes, we periodically rotate these patterns on all S nodes, making one hop foreword in each period. For example, suppose the optimal pattern is $(0, 1)$, where 1 (0) means the S node is active (inactive), and we have four S nodes in the system. As shown in Figure 3, the initial assignment of this pattern is $S_1 = 0, S_2 = 1, S_3 = 0, S_4 = 1$ and the assignment after one step rotation is $S_1 = 1, S_2 = 0, S_3 = 1, S_4 = 0$. Since the rotation is always in clock-wise direction, the utilization of each S node is identical given the long enough simulation time.

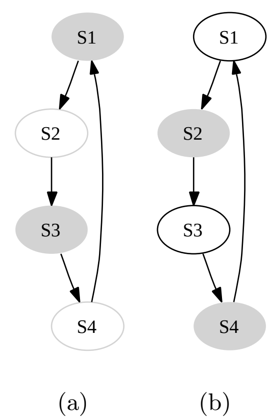


Figure 3: Illustration for pattern rotation, (a) Initial assignment, (b) After one step rotation.

Figure 4 shows the state machine to describe the node level operation. The states and transitions in the figure are explained in Table 1.

2.2 Design Limitation and Assumption

We assume that the issue rate of a C node is a fixed value, which is known by S node. Note that S node still cannot guarantee matching all C node events since we preserve the uncertainty brought from the Poisson distribution of a C node. Suppose the power dissipations of an S node under state I and C are Pwr_I and Pwr_C, respectively, the average operating power of an S node is Pwr_A^O, the duty cycle is D, the number of active (operating) S node is N_O, the C node event issue rate is $1/T$ [1] and the energy consumption during time T is E. We have the following claim.

Claim 1. *Our surveillance system is limited to the following relation between C node event issue rate $1/T$ and power constraint G.*

$$\frac{T}{G} \geq (T_I^B + T_C^B) \cdot \frac{N_O \cdot Pwr_C}{N_S \cdot Pwr_I} \tag{1}$$

[1]Note that T is equal to the issue rate of the C node events in our algorithm. T is therefore the timeout period to wake up a S node from state V to state C.

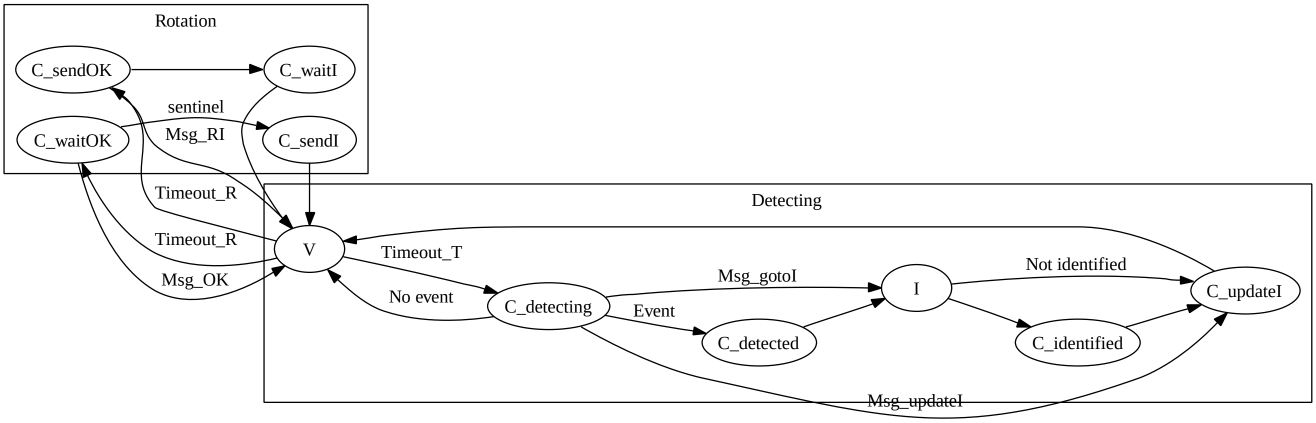


Figure 4: Node level operation state machine

name	state	description
V	State V	Waiting for timeout T for waking up to detection or timeout R for waking up to rotate the activity
C_detecting	State C	Waiting for events and trying to receive messages from other S nodes. i) When an event is coming and detected by sensor A, the node will go to state C_detected. ii) Otherwise, if a message (Msg_gotoI) with the location and time of the detected event is received and the event can be covered by its I sensor, it will go to state I. iii) Otherwise, the node will not attend the identification and wait for a message from the sentinel to update the timeout period T.
C_detected	State C	An event is detected by A sensor. The node will send out a broadcast message (Msg_gotoI) to other active S nodes to tell them the location and time of the detected event
I	State I	An event is identified by sensor I. The node will go to C_updateI state if no event is identified during the duty cycle t.
C_identified	State C	i) Event identified by I sensor. ii) Sending out a message to sentinel node S_s to tell it the location and time of the identification.
C_updateI	State C	i) Sentinel node receives up to two messages from other active S nodes, if two messages are received, then update issue rate and timeout period T based on the current issue rate. Finally it broadcasts the latest T to other other nodes and go to state V. ii) Other S nodes receive the message from sentinel, update T and go to state V
C_sendOK	State C	Each newly activated node s_i sends an acknowledgment message (Msg_OK) to the current active node s_{i-1} for its availability. After sending this message, it will go to state C_waitI.
C_waitI	State C	The newly activated node is waiting for the message from the sentinel node for the newest timeout period T.
C_waitOK	State C	The current active node is waiting for an acknowledgment message (Msg_OK) from its neighbor node, which is about to be activated. It will go to state C_sendI if it is a sentinel node, and go to state V otherwise.
C_sendI	State C	The sentinel node broadcast the current timeout period T for detection. It will go to state V.

Table 1: Description for node level operation states

PROOF.

$$D = \frac{T_I + T_C}{T} \quad (2)$$

$$E = T \cdot \frac{N_S \cdot Pwr_I}{G} = (T_I + T_C) \cdot N_O \cdot Pwr_A^O \quad (3)$$

$$Pwr_A^O = \frac{Pwr_I \cdot T_I + Pwr_C \cdot T_C}{T_I + T_C} \geq Pwr_C \quad (4)$$

$$\Rightarrow \quad \frac{T}{T_I + T_C} \cdot \frac{N_S \cdot Pwr_I}{G \cdot N_O} \geq Pwr_C \quad (5)$$

$$\Rightarrow \quad \frac{T}{G} \geq (T_I + T_C) \cdot \frac{N_O \cdot Pwr_C}{N_S \cdot Pwr_I} \quad (6)$$

$$\Rightarrow \quad \frac{T}{G} \geq (T_I^B + T_C^B) \cdot \frac{N_O \cdot Pwr_C}{N_S \cdot Pwr_I} \quad (7)$$

where T_I and T_C are operating time within on period in state I and C, respectively. Taking state transition delay into consideration, we have the following delay constraints, $T_C \geq T_C^B$ and $T_I \geq T_I^B$. $\square$

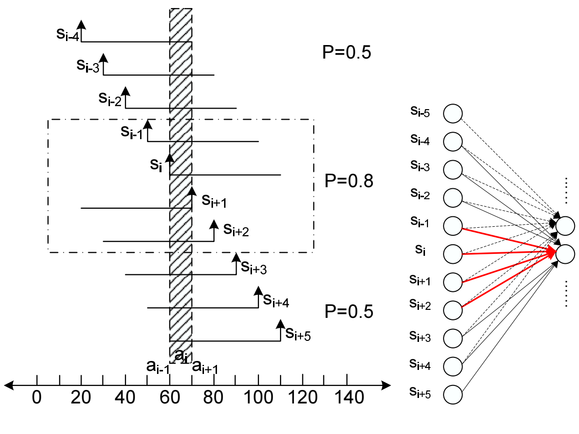
2.3 Single Node Sensoring Range Analysis

Due to the presence of random obstacles, events corresponding to the presence of a C node target are identified by identification sensor I with a range r dependent probability P_I as follows.

$$P_I = \begin{cases} 0, & \text{if } r > 50m \\ 0.5, & \text{if } 20m < r < 50m \\ 0.8, & \text{if } r < 20m \end{cases} \quad (8)$$

According to the given sensors and barrier placement, we can calculate that the identification range of an identification sensor as follows. Consider the distance $d_{p \to s}$ between point p on the barrier, which is opposite to an S node s. If $d_{p \to s} \leq R_{near} = \sqrt{20^2 - 5^2} \approx 20m$, the probability that a C node on p is identified by sensor s is 0.8. If $R_{near} < d_{p \to s} \leq R_{far} = \sqrt{50^2 - 5^2} \approx 50m$, the identification probability is 0.5.

We divide the barrier uniformly into 100 sub-areas, each of which is $10m$ as shown in Figure 5. To ignore edge effects, we assume that the ends of the barrier are joined (or wrap) and retain the assumption of a linear geometry, so that each sub-area is geometrically symmetric. Figure 5 shows that each sub-area a_i is covered by ten I sensors, i.e., $s_{i-4}, \cdots, s_{i+5}$.



(a) (b)

Figure 5: Area coverage by identification sensor (a) Physical view (b) Graph representation

To make the representation clear, we use a bipartite graph as shown in Figure 5 (b) to represent the geometrical coverage relationship between S node and barrier sub-areas. For each area a_i, we divide the S nodes that can cover it into the following two groups, S_i^N and S_i^F, based on their coverage probability for area a_i. In Figure 5 (b) the red (bold) edges show the connection from nodes in S_i^N to area a_i.

$$S_i^N = \{s_{i-1}, s_i, s_{i+1}, s_{i+2}\} \quad (9)$$
$$S_i^F = \{s_{i-4}, s_{i-3}, s_{i-2}, s_{i+3}, s_{i+4}, s_{i+5}\} \quad (10)$$

A successful identification requires that two I sensors capture a C node in the same time. If we associate a {0, 1} variable s_i with each S node S_i to indicate if its identification sensor is turned on ($s_i = 1$ means on), the probability P_{a_i} that area a_i is identified in a particular time instance is

$$P_{a_i} = f(a, b) \quad (11)$$
$$f(a,b) = 1 - (1-0.8)^a(1-0.5)^b - [a \cdot 0.8(1-0.8)^{(a-1)}(1-0.5)^b + b \cdot 0.5(1-0.5)^{(b-1)}(1-0.8)^a] \quad (12)$$
$$a = s_{i-1} + s_i + s_{i+1} + s_{i+2} \quad (13)$$
$$b = s_{i-4} + s_{i-3} + s_{i-2} + s_{i+3} + s_{i+4} + s_{i+5} \quad (14)$$

where $f(a,b)$ is the probability that area a_i is identified by two S nodes given the number of active nodes in S_i^N (a) and S_i^F (b).

3. POWER CONSTRAINED SENSOR COVERING AND SCHEDULING

For the surveillance system design shown in subsection 2.1, we focus on the optimization in "nodes coordination and identification" procedure in Figure 2. In this stage, a sub-set of S nodes should be selected and scheduled for identification. We define sub-set $F_i = \{S_i^1, \cdots, S_i^k\}$ of S nodes as a *group*.

If we represent the coverage range of each I sensor in X-axis and the operation time in Y-axis, each S node forms one or more rectangles as shown in Figure 6. The contribution of each S node for identification is related to the area of its corresponding rectangles. As the sensor coverage for certain range is an non-deterministic value, i.e., a probability, the **power constrained sensor covering and scheduling (PCSCS)** problem is to find an optimal floor-planning of these rectangle so that the covering probability of the X-Y plane is maximized and the total area of the rectangles corresponding to each I sensor should not excess a bound (i.e., the covering and scheduling for each I sensor is $feasible$). For example, both floor-planning (a) and (b) in Figure 6 have the same covering probability, but the area rectangle S3 in (a) excesses the given upper bound. Therefore the optimal solution is (b).

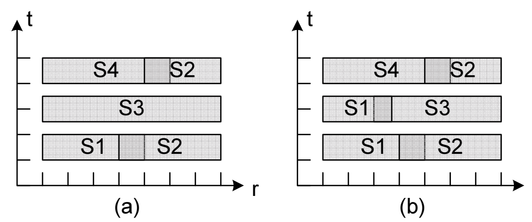


Figure 6: Illustration of power constrained sensor scheduling problem. (a) A scheduling option with one power constrain violation, (b) an optimal feasible scheduling

3.1 Problem Formulation of PCSCS

We use a {0,1} variable a_{ij} to describe group F_i. S node S_j is in group F_i iff $a_{ij} = 1$. Since the sensoring range and covering probability of each individual I sensor is given (see subsection 2.3, the covering capability, $P(F_i)$, of group F_i can be calculated as follow.

$$P(F_i) = \sum_{j=1}^{100} P_{a_j}, s_k = a_{ik} \quad (15)$$

where P_{a_j} is calculated by (11).

Each S node can belong to multiple groups. For each group F_i, the operation time of each S nodes $S_i^k \in F_i$ is T_i. Our objective is to find a set of groups of S nodes and their scheduling (operation time) to maximize the identification probability of C nodes under the given power constraint (i.e., G value specified by user).

Formally, suppose all possible groups for the given S nodes are $F_1, \cdots, F_{N_F}$ and the covering probability of group F_i is $P(F_i)$. The overall covering probability of group F_i is $P(F_i) \cdot T_i/T_s$, where T_s is the total simulation time. Therefore, the objective function in our PCSCS problem is shown in (16). The energy constraint (17) is that the total energy consumption of each S node cannot excess the given battery reservoir capacity. If we divide Ts in both sides of (17) and perform substitution $D_i = T_i/T_s$, we obtain (18), where D_i is the duty cycle of group F_i. Finally, PCSCS problem is formulated as the objective (16) under constraint (18) and the variables are D_i, duty cycle of group F_i, $i = 1, \cdots, N_F$.

$$\max \quad \sum_{i=1}^{N_F} P(F_i) \cdot T_i/T_s = \sum_{i}^{N_F} P(F_i) \cdot D_i \tag{16}$$

$$s.t.$$

$$\sum_{i=1}^{N_F} a_{ij} \cdot T_i \cdot Pwr_i \leq E_{max} \tag{17}$$

$$\Rightarrow \quad \sum_{i=1}^{N_F} a_{ij} \cdot T_i \cdot Pwr_i/T_s \leq E_{max}/T_s$$

$$\Rightarrow \quad \sum_{i=1}^{N_F} a_{ij} \cdot D_i \cdot Pwr_i \leq Pwr_A = \frac{Pwr_T}{G} \tag{18}$$

where Pwr_i, Pwr_A and Pwr_T are the operating power, average power and limit power (operating in state I), respectively.

3.2 Optimal Sensor Covering and Scheduling Based on Network Flow

Motivated by [9], which maximizes the sensor life time under performance and energy constraint, we transfer the PCSCS problem into a generalized max-flow problem [5], which can be solved in $O(m^{1.5}n^2 \log B)$ time [6]. Figure 7 shows the network flow diagram for the example given in Figure 6. The supply of source node S is infinite. Node $S_1, \cdots, S_4$ denote the S nodes and $F_1, \cdots, F_3$ denote the groups of S nodes. In the generalized max-flow problem, a multiplier λ is associated with each edge. For inter-medium each node i, supply-demand balance condition of the generalized network flow is that

$$\sum_j x_{ij} - \sum_j \lambda_{ji} x_{ji} = 0, \forall i \in S, F \tag{19}$$

For edge (S, S_i), we set a capacity Pwr_A to reflect the power constraint (18). For edge (S_i, F_j), we set

$$\lambda(S_i \rightarrow F_j) = 1/Pwr_i \tag{20}$$

to transfer the unit of power (in S_i node) to duty cycle (in F_j node), where Pwr_i is the average power of node S_i when it is operating. For edge (F_j, D), we set

$$\lambda(F_j \rightarrow D) = P(F_j)/\sum_{k=1}^{N_{F_j}} a_{kj} \tag{21}$$

to calculate the product (16) of covering probability and the duty cycle of group F_j, i.e., the duty cycle of each S node. To guarantee that the identical operating time of the elements within the same group, the flow from S_i nodes to the same F_j node should be the same and therefore we have an additional constraint

$$x_{S_p,F_j}/Pwr_p = x_{S_q,F_j}/Pwr_q \tag{22}$$

The objective is to maximize the flow enters sink node D as following

$$\max \sum x_j \cdot \lambda(F_j \rightarrow D) \tag{23}$$

where x_j is the flow amount from F_j to D. It is easy to see that $x_j = D_j$.

Although the generalized max-flow problem can be solved efficiently, the number of groups may be prohibitively large. In our case, $N_F = 2^{100}$. In fact, the general sensor covering and scheduling problem is NP-hard even in deterministic case [10]. In the next subsection, we present a heuristic to reduce N_F without loss much optimality.

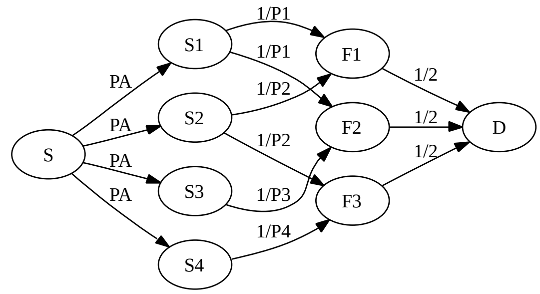


Figure 7: Generalized max-flow formulation of the power constrained sensor scheduling problem

3.3 Sensor Coverage Optimization

Instead of enumerating all possible groups (combinations of S nodes), we can only consider certain groups that have potential to lead to optimal covering. To break down the complexity of the PCSCS problem, we consider a sub-problem, *power constraint sensor covering (PCSC)*, which finds a placement of active S nodes (i.e., an assignment of a_{ij} for group F_j) to maximize the covering probability of F_j under the given duty cycle, D_u. Formally, this problem can be formulated as follow.

$$\max \quad D_u \cdot P(F_i) \tag{24}$$

$$s.t.$$

$$D_u \cdot Pwr_j \cdot \sum_{i=1}^{100} a_{ij} = Pwr_A$$

$$\Rightarrow \quad D_u \cdot Pwr_j \cdot N_{F_j} = Pwr_A \tag{25}$$

Note that $\sum_{i=1}^{100} a_{ij} = N_{F_j}$ is the number of S nodes in F_j. We have the following theorem which indicates the optimal solution for PCSC problem.

THEOREM 1. *Given power constraint Pwr_A, the optimal covering probability $P_I(N)$ for a fixed N_F, the optimal number of S nodes in group F is N_F^* which maximizes $\frac{P_I(N_F)}{N_F}$.*

PROOF. Eliminate D_u in (24) based on (25) and obtain the following new formulation.

$$\max \quad \frac{P_A}{Pwr} \cdot \frac{P_I(N)}{N} \tag{26}$$

Suppose the scheduling algorithm for each individual S node is given, then Pwr is a constant in (26) and the objective is to maximize $\frac{P_I(N)}{N}$. □

Following Theorem 1, we can find optimal solution for PCSC problem by enumerating $\frac{P_I(N)}{N}$ values for all possible values of N_F, i.e., $2, \cdots, 100$, and select those with large values as the candidates for F_j. In sub-subsection *3.3.1* and *3.3.2*, we will show how to find the optimal placement of the sensors given N_F and the results for the optimal solution for PCSC, i.e., the candidates for groups F_j.

3.3.1 Optimal Sensor Placement Under Fixed N_F

As shown in subsection 2.3, the probability that sub-area a_i is covered (identified by at least two S nodes) is related to the S nodes placement around it. Given the total number of active (working in I state) S node, we solve the following sub-problem of PCSC.

Definition 1. *Suppose that the probability, $P(C \text{ in } a_i)$, that a C node presents in any sub-area a_i is identical in a time instance and the number of active S nodes is N. Find a placement (assignment) of these N nodes to maximize the probability of identification.*

Formally, this problem can be formulated as follows.

$$\max \quad \sum_{i=1}^{100} P(C \text{ in } a_i) \cdot P(a_i) \tag{27}$$

$$s.t.$$

$$\sum_{i=1}^{100} s_i = N \tag{28}$$

$$s_i = \{0, 1\} \tag{29}$$

where $P(a_i)$ is defined in (11).

Note that $P(a_i) = f(a, b)$ is non-polynomial and even not convex, and also note that the total number of different placement for N active S nodes are C_{100}^N, which is prohibitively large for most of N values. Instead of solving this problem directly, we use simulated annealing (SA) to obtain near-optimal solution.

The novelty of our SA based algorithm is that it changes the patterns of the S nodes placement in each movement. We encode the placement result of N active S nodes as a 100-bit $\{0, 1\}$ sequence Q, where 1 (0) in i bit means that s_i is activated (unactivated). Due to the symmetric property of all S nodes, we only need to fix the pattern (sub-sequence of the placement sequence Q) and repeat it in a periodic fashion. The number of symmetric patterns is far less than the number of different placement when C_{100}^N is large.

In cooling process, we first checks different patterns with a fixed pattern size N_P. After δ iterations for N_P, the pattern size N_P is increased by one and enter the next iteration. In each move of SA, one of N_P bits in the pattern selected randomly and turned on. We maintain a list to record all tested placements so that avoid redundant calculation. The cost of each placement is the coverage probability that can be calculated by (27). To make SA process more efficient, we pre-calculate the values of function $f(a, b)$ and store them into a 5×7 look-up table[2].

We set $N = 2, \cdots, 100$ and run the SA algorithm. The maximal coverage probabilities under each N values are shown in Figure 8. It shows that the increase of coverage probability by adding more S nodes becomes less substantial when the N is large enough, e.g., $N > 50$. This observation leads us to find a better trade-off between N and the coverage probability.

3.3.2 Sensor Group Candidates

Since $N \in [2, \cdots, 100]$, we can enumerate all $\frac{P_I(N)}{N}$, where $P_I(N)$ is obtained by (27), and pick the N value leading to the maximal $\frac{P_I(N)}{N}$. Figure 9 shows the all $\frac{P_I(N)}{N}$ values obtained by the SA based algorithm presented in sub subsection 3.3.1. In our implementation, we take five group, i.e., $N = 30, \cdots, 35$, which give the largest $\frac{P_I(N)}{N}$ values.

[2]The possible values of a and b are $a = \{0, 1, 2, 3, 4\}$ and $b = \{0, 1, 2, 3, 4, 5, 6\}$

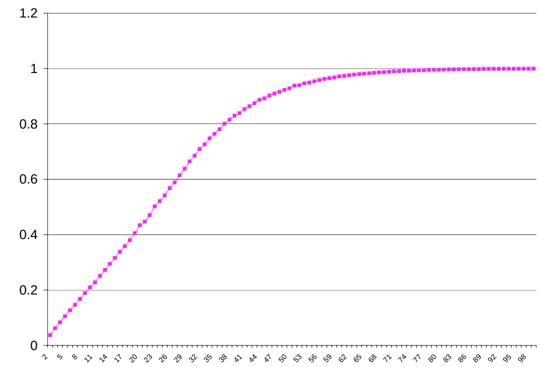


Figure 8: The maximal coverage under given N

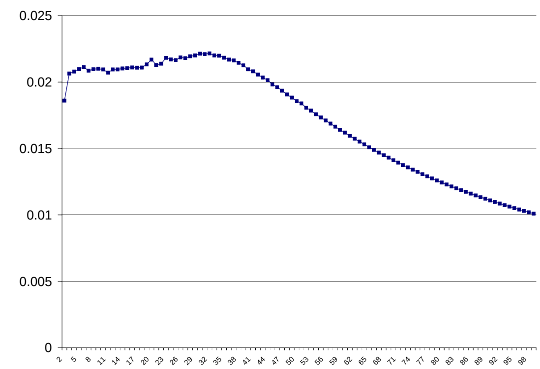


Figure 9: $\frac{P_I(N)}{N}$ values ($N = 2, \cdots, 100$)

4. DESIGN FOR COUNTER-SURVEILLANCE SYSTEM

The mission of counter-surveillance system is to avoid being identified. C nodes can check the density of S nodes by detecting the RF energy of S nodes' wireless communication, hence C nodes can take the advantage to avoid detecting. Although the C nodes have no knowledge about the detail of the S nodes' algorithm, we can still assume that C nodes are aware of the following strategies of S nodes' behavior.

1. S nodes are operating periodically but the period T is unknown.

2. A subset of S nodes are operating for identification and the placement of the active nodes is stable in a very long time. Therefore the C nodes have the capability to learn all the possible groups (see Section 2) of S nodes. However, the scheduling of different groups of the active S nodes is unknown.

The design freedom of the counter-surveillance system includes deciding the velocity and direction of a move for an individual C node, and also the scheduling of the starting and ending time of one move.

Figure 10 shows the system level diagram for counter-surveillance system operation. Initially, all C nodes stay behind the barrier and try to learn the S nodes operation groups. After that, a game theory based approach is used to calculate, P_i^C, the probability of escaping from portal

$Port_i$ (will be introduced in subsection 4.1). A fixed timeout period T_C is set for re-learning the S node groups. During T_C time, a C node moves from a portal $Port_i$ selected based on probability P_i^C. As the decision of the starting portal corresponds to a S node group F_i, the C node which is going to move will choose the direction and moving distance based on F_i. When the timeout T_C passes, the C nodes system will begin learning S node patterns again.

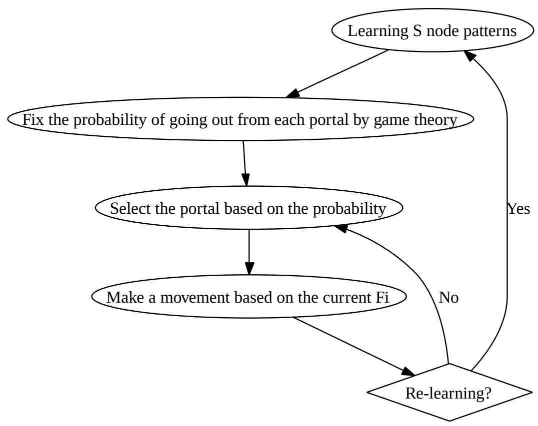


Figure 10: System operation diagram of counter-surveillance system

In the process for learning S node groups, the coordination and group of C nodes can be design similarly as for S nodes (see subsection 3.3). The moving velocity is decided by the user specified parameter G. In this section, we are only concerned with the decision about the starting and ending point of a moving. In the next subsection, we will solve this problem by a game theory.

4.1 Formulation Based on Game Theory

To make this paper self-contained, we briefly introduce the hide and seek game theory proposed by Von Neumann in [7]. There are two players, a hider and a seeker, in the game. Given a $n \times n$ values of a square array $\|g_{ij}\|$ of positive rational numbers. The hider chooses a cell c_{ij} and the seeker chooses a line (row or column index), each in ignorance of the choice made by the other player. If the seeker chooses a line that includes the cell chosen by the hider then the hider pays the seeker the amount for that cell, otherwise he pays 0. Thus, if the hider chooses cell c_{ij} he pays g_{ij} to the seeker if and only if the seeker chooses row i or column j as his line. This completes one play of the game.

Formally, the Von Neumann's hide and seek game is formulated as follows. The hider has n^2 pure strategies corresponding to the n^2 cells. Let his mixed strategy be $h = (p_{ij})$ where $\sum_{ij} p_{ij} = 1$ and where p_{ij} denotes the probability that he hides in cell c_{ij}. The seeker has $2n$ pure strategies corresponding to the $2n$ lines. We let his mixed strategy be $s = (p_i, q_i)$, where $\sum_i (p_i + q_i) = 1$, and where p_i denotes the probability that he seeks in row i and q_i in column i. The expected value of the payoff from one play of the game, for the seeker, is

$$V(h, s) = \sum_{ij} p_{ij} g_{ij} (p_i + q_j) \qquad (30)$$

The objective is to exhibit specific values $p_{ij}^*, p_i^*, q*_j, V^*$ that satisfy the relation

$$\max_s V(h^*, s) = \min_h V(h, s^*) = V^* \qquad (31)$$

[8] presents a simple but optimal solution for Von Neumann's game based on Hungarian method [11].

Essentially, in the hide and seek game, one player wins by matching the other's decision and the other wins by mismatching. It is very natural to model the surveillance sensors as the seekers and the counter-surveillance sensors as the hiders. Furthermore, as mentioned in Section 2, the surveillance and counter-surveillance problem can be represented as a two-dimension ($time \times area$) covering problem. We can formulate it as a two-person hide and seek problem as shown in Figure 11. Suppose C nodes have learned that S nodes exhibit five different operating groups $F_1, \cdots, F_5$ as shown in Figure 11 (a). We can map them into the ($time \times area$) plane in Figure 11 (b). The gain g_{ij} associated with cell $c_{i,j}$ in (b) denotes the covering probability in area a_j by S node group F_i. We need to calculate the probability p_{ij} for a C node (hider) to choose cell $c_{i,j}$ in (b) to minimize the expected value of covering probability by S nodes. Having solved the hide and seek game, the probability for a C node to choose area a_j as the moving start point can be calculated as $\sum_i p_{ij}$.

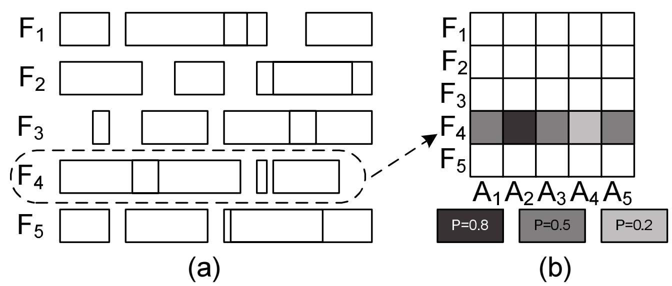


Figure 11: Illustration of formulation of counter-surveillance by hide-seek game

5. SIMULATION AND EXPERIMENTAL RESULTS

We conduct two groups of simulations to test the effectiveness of our surveillance and counter-surveillance algorithm, respectively. In our simulator for surveillance system, we perform cycle accurate simulation for $T_{sim} = 100ks$ with precision ms. Given a G value, the energy reservoir of the system is $T_{sim} \cdot Pwr_I / G$. In each clock cycle, S nodes are scheduled based on the algorithm presented in the previous sections and their energy dissipations are recorded according to their operating states. The movement of C nodes is generated as a poisson process with fixed issue rate and the distance of each movement can be any number from one to ten hops, which will be randomly chosen in the simulator. To highlight our power aware covering and scheduling algorithm, we do not consider the nodes transition during the simulation. As suggested in the problem statement, we set $Pwr_I = 10W$ and $Pwr_C = 1W$, respectively. We use $T_C^B = 10s$ and $T_I^B = 1s$ to model the transition delay from state V to C and the required time for staying in state I for identification. A $2s$-delay for receiving a message is used to model the communication delay. We test the following combinations of the system level the parameters, i.e., Velocity=$\{0.01m/s, 0.1m/s, 1m/s, 10m/s\}$, event issue period=$\{100s, 100s\}$, $G = \{10, 100, 1000\}$. Figure 12 shows the event identification rate (probability) for the surveillance system. The results under two different issue rates are compared side by side. The main observation is that our algorithm offers stable identification rate for the fixed G value under the substantial changes of velocities. We also note that the degradation of the system perform is non-linear to the G values. When G is increased from 10 to 100, the identification rate is decreased only two times. However, the system will almost fail if G value is too large (e.g. 1000). Figure 13 shows the energy consumption during the whole life time for a surveillance system with $G = 10$, Velocity=$0.1m/s$, Issue period=$1000s$. The points for detection and identification of two events are labeled in the figure. We can see that the energy consumption increases stably during the simulation time.

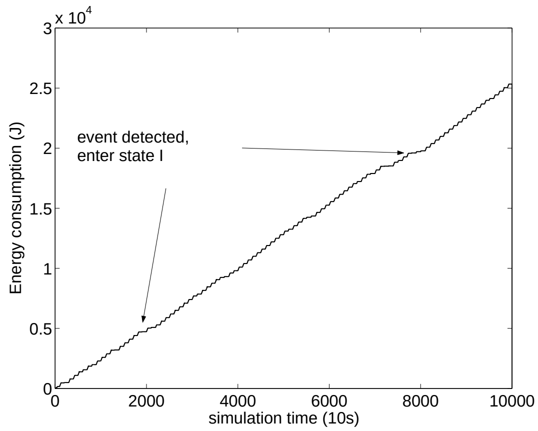


Figure 13: Energy consumption during surveillance (G=10, Vel=0.1, IR=1000)

For the counter-surveillance system simulation, we aim to highlight the effectiveness of our game theory based algorithm. We assume that C nodes have the information of S nodes activity patterns but not the scheduling algorithm of S nodes. For each movement, the start portal of a C node is selected based on the algorithm given in subsection 4.1, and it moves one hop and enter its neighbor portal. The surveillance system behavior is exactly the same as the first group of simulation aforementioned. For comparison, we schedule C node movements by randomly selecting the start portal of a C node and keep other settings unchanged. Table 2 compares these two strategies for the total distance Dis_{game} and Dis_{rand} of C nodes movements without being detected during simulation time. Note that the G value in this table is for surveillance system. Table 2 shows that the improvement of our game theory based algorithm increase up to 32% non-detected traveling distance of C nodes compared to the random algorithm. The improvement is decreased when G value of the surveillance system is decreased since the performance of the surveillance system is degraded accordingly.

velocity	issue period	G	Dis_{game}	Dis_{rand}	impv(%)
0.01	1000	10	740	564	31%
0.01	100	10	7720	5851	32%
0.1	1000	10	738	565	31%
0.1	100	10	7975	6700	19%
1	1000	10	822	662	24%
1	100	10	7828	6730	16%
10	1000	10	897	830	8%
10	100	10	8332	7538	11%
0.01	1000	100	970	869	12%
0.01	100	100	8590	8046	7%
0.1	1000	100	990	945	5%
0.1	100	100	9000	8019	12%
1	1000	100	930	846	10%
1	100	100	8470	8276	2%
10	1000	100	900	863	4%
10	100	100	9480	9302	2%
0.01	1000	1000	1000	1013	-1%
0.01	100	1000	9600	9540	1%
0.1	1000	1000	1110	1100	1%
0.1	100	1000	10400	10480	-1%
1	1000	1000	900	885	2%
1	100	1000	9480	9610	-1%
10	1000	1000	910	906	0%
10	100	1000	9990	10120	-1%

Table 2: Comparison of moving distance between game theory based counter-surveillance strategy and random strategy

6. CONCLUSIONS AND FUTURE WORK

In this paper, we have studied sensor network surveillance and counter-surveillance problem under power constraints and sensoring obstacles. this work first designs a power efficient surveillance system by sensor nodes coordination and cooperation, and then solve the power constrained sensor node covering and scheduling (PCSCS) problem by generalized maximal network flow. The counter-surveillance problem is formulated as a hide and seek game, which can be solved efficiently and optimally by game theory. The simulation results show that our surveillance scheduling algorithm is adapted to a large range of different G values and C node velocities. The typical event detection rate is over 50% given G=10. Compared to random scheduling approach of counter-surveillance system, our proposed algorithm achieves up to 30% increase of non-detected travelling distance. Our future work is to integrate and simulate our algorithms with the real sensor network.

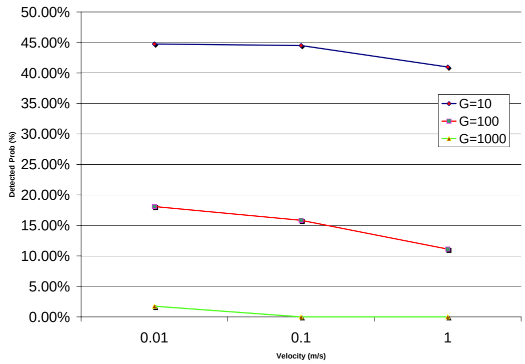
7. REFERENCES

[1] T. He, S. Krishnamurthy, L. Luo, T. Yan, L. Gu, R. Stoleru, G. Zhou, Q. Cao, P. Vicaire, J. A. Stankovic, T. F. Abdelzaher, J. Hui, and B. Krogh, "Vigilnet: An integrated sensor network system for energy-efficient surveillance," *ACM Trans. Sen. Netw.*, vol. 2, no. 1, pp. 1–38, 2006.

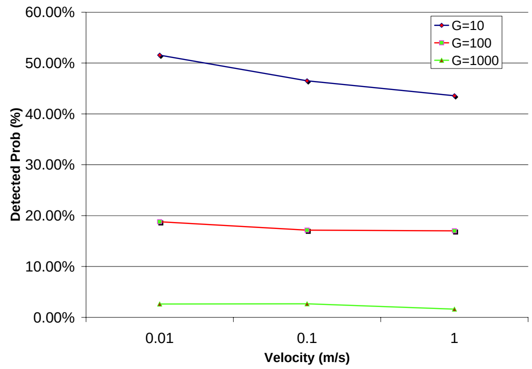
[2] Q. Cao, T. Abdelzaher, T. He, and J. Stankovic, "Towards optimal sleep scheduling in sensor networks for rare-event detection," in *IPSN '05: Proceedings of the 4th international symposium on Information processing in sensor networks*, (Piscataway, NJ, USA), p. 4, IEEE Press, 2005.

[3] F. Y. S. Lin and P. L. Chiu, "A simulated annealing algorithm for energy-efficient sensor network design," in *WIOPT '05: Proceedings of the Third International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks*, (Washington, DC, USA), pp. 183–189, IEEE Computer Society, 2005.

[4] S. Meguerdichian, F. Koushanfar, G. Qu, and M. Potkonjak, "Exposure in wireless ad-hoc sensor networks," in *MobiCom '01: Proceedings of the 7th*



(a) Issue Period=100s



(b) Issue Period=1000s

Figure 12: Surveillance system simulation results

annual international conference on Mobile computing and networking, (New York, NY, USA), pp. 139–150, ACM Press, 2001.

[5] E. Tardos and K. D. Wayne, "Simple generalized maximum flow algorithms," in *Proceedings of the 6th International IPCO Conference on Integer Programming and Combinatorial Optimization*, (London, UK), pp. 310–324, Springer-Verlag, 1998.

[6] P. M. Vaidya, "Speeding-up linear programming using fast matrix multiplication (extended abstract)," in *FOCS*, pp. 332–337, 1989.

[7] J. von Neumann, "A certain zero-sum two-person game equivalent to the optimal assignment problem," *Ann. Math. Studies*, vol. 28, pp. 5–12, 1953.

[8] M. M. Fllod, "The hide and seek game of von neumann," *Management Science*, vol. 18, no. 5, pp. 107–109, 1972.

[9] M. Perillo and W. Heinzelman, "Optimal sensor management under energy and reliability constraints," in *IEEE Wireless Communications and Networking Conference*, 2003.

[10] A. L. Buchsbaum, A. Efrat, S. Jain, S. Venkatasubramanian, and K. Yi, "Restricted strip covering and the sensor cover problem," in *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms (SODA '07)*, 2007.

[11] H. W. Kuhn, "The hungarian method for the assignment problem," *Naval Research Logistics Quarterly*, vol. 2, pp. 83–97, 1955.