Computer Science

College of Arts & Sciences

2008

Mobility Control for Complete Coverage in Wireless Sensor Networks

Zhen Jiang
West Chester University of Pennsylvania, zjiang@wcupa.edu

Jie Wu
Florida Atlantic University

Robert Kline
West Chester University of Pennsylvania, rkline@wcupa.edu

Jennifer Krantz
West Chester University of Pennsylvania

Mobility Control for Complete Coverage in Wireless Sensor Networks [*]

Zhen Jiang
Computer Sci. Dept.
West Chester University
West Chester, PA 19383, USA
zjiang@wcupa.edu

Jie Wu
Computer Sci. & Eng. Dept.
Florida Atlantic University
Boca Raton, FL 33431, USA
jie@cse.fau.edu

Robert Kline, Jennifer Krantz
Computer Sci. Dept.
West Chester University
West Chester, PA 19383, USA
{rkline|jk599336}@wcupa.edu

Abstract

In this paper, we propose a new control method to cover the "holes" in wireless sensor networks. Many applications often face the problem of holes when some sensor nodes are disabled from the collaboration due to their failures and misbehavior. These holes may occur dynamically, and such a problem cannot be solved completely by simply deploying more redundant sensors. With a synchronization around each hole using the directed Hamilton cycle, one (and only one) snake-like cascading replacement process will be initiated in the local area in order to fill in that vacant area with a spare node. In this way, network connectivity and coverage can be guaranteed. Our analytical and experimental results show substantial improvements of such a replacement compared with the best result known to date.

1 Introduction

Recent advances in micro-electromechanical systems, digital electronics, and wireless communications have enabled the development of low-cost, low-power, multi-function sensor devices. These devices can operate autonomously to gather, process, and communicate information about their environments. When a large number of sensor devices collaborate using wireless communications, they constitute a wireless sensor network (WSN) [1]. Applications of WSNs range from environmental monitoring to surveillance to target detection [7]. Due to the fact that sensors can very easily fail or misbehave, many nodes can be isolated from the network collaboration [2]. Thus, a "hole" in the surveillance area may occur in the deployed area, and such an occurrence may be dynamic. For instance, as indicated in [8], the attacker can cause the nodes to move and deplete their battery power, which might reduce node density in certain areas. The holes in the surveillance area can occur even when many redundant sensors are deployed. Many applications often face the problem of such holes in surveillance areas, causing incomplete coverage. To ensure that the entire network works correctly, complete coverage of its surveillance area must be provided.

Recently, rather than preventing the occurrence of the holes, some extended virtual force methods [5, 10] that simulate the attractive and repulsive forces between sensor nodes have been proposed to fix the hole problem. In these methods, sensors in a relatively dense region will slowly move towards the relatively sparse regions according to each other's repulsive force and head towards a hole in the network. However, as indicated in [6], without global information, these methods may take a long time to converge and are not practical for real applications due to the cost in total moving distance, total number of movements, and communication/computation. Then, in [6], a more practical balancing method under the virtual grid model [9] is discussed. This method allows for quick convergence but requires node adjustments in the entire grid network, causing many unnecessary node movements just for providing the coverage for a single hole. In an early work [3], a localized control method based on 1-hop neighborhood is proposed. Whenever a vacant area is detected, a snake-like cascading replacement process is initiated to move nodes to cover the hole area. However, due to the lack of synchronization, the existence of a hole will incur multiple replacement processes, causing redundant processes and some unnecessary node movements.

To reduce the redundant replacement processes initiated by the method in [3], a new control scheme is proposed in this paper. There are three major contributions.

- Whenever a hole area is detected, the snake-like cascading replacement will be initiated under our synchronization. As a result, one and only one replacement node will move in from neighbors.

- The cost of our new replacement process is analyzed in

terms of the number of node movements and the total moving distance.

- A simulation is developed to illustrate the correctness of our analytical results. Compared with the best results known to date, the experimental results show the improvement of our new method in terms of the number of node movements, the total moving distance, and the success rate of hole recovering.

A short summary of our approach follows. First, we build a surveillance model to detect the occurrence of a hole. We partition the entire surveillance area into many small squares (of size $r \times r$) in a virtual grid model [9]. After many faulty sensors and misbehaving sensors are disabled, the rest of the nodes (also called enabled nodes) constitute the WSN. In each grid, one and only one enabled node will be elected as the grid head to monitor the neighborhood. The rest of the enabled nodes in the same grid are called spare nodes. As indicated in [9], the connectivity and coverage of networks can be guaranteed if each grid has its own head. According to their locations, all the grids are threaded along a directed Hamilton cycle. When a grid does not have the head, the directed Hamilton cycle turns into a directed Hamilton path. One (and only one) *replacement process* will be initiated at the preceding grid along that directed Hamilton path to move a spare node into this vacant area. In this way, the neighboring grids around the vacant grid are synchronized to avoid any redundant replacement process that might be initiated. As a result, each grid will be filled by at least one enabled node which will become the head, allowing the whole network to keep its coverage and connectivity, even when many nodes are disabled and the network is disconnected. It is noted that the movement of a node during the replacement process may trigger another replacement for that particular node, causing a snake-like cascading movement. In our analytical and experimental results presented in this paper, we will show that the cost of such a moving process is limited in terms of the number of total node movements and the total moving distance. Throughout the paper, proofs to theorems are shown in [4].

2 Preliminary

We assume that all the nodes have the same communication range R. The nodes inside the communication range are called neighbors and two neighboring nodes are directly connected. Each node u has its location, which is simply denoted by $L(u)$. The location information can be discovered by having Global Positioning System (GPS) receivers at some fixed nodes or a mobile beacon node, or just by relying on the relative coordinate system. We partition the whole network into an $n \times m$ 2-D grid system (see Figure 1 (a)). Each grid is of a square size $r \times r$ and is de-
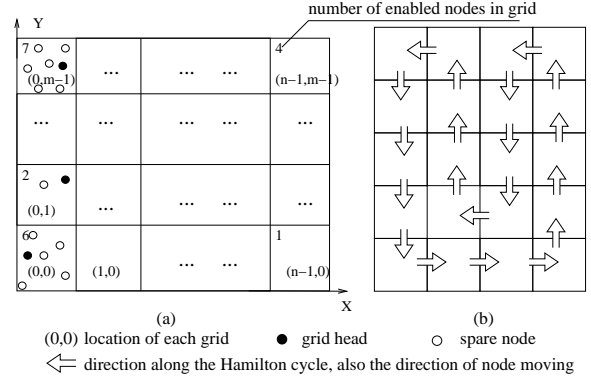

Figure 1. (a) Virtual grid and grid heads. (b) Hamilton cycle in a 4×5 grid system.

noted by its relative location in the entire system, say (x, y) ($0 \leq x \leq n - 1$, $0 \leq y \leq m - 1$). Two grids (x_1, y_1) and (x_2, y_2) are called neighboring grids if their location addresses differ in one (and only one) dimension, say X. Moreover, $\mid x_1 - x_2 \mid + \mid y_1 - y_2 \mid = 1$. Each grid (x, y), except the ones at the edge of grid system, has four neighbors $(x, y + 1)$, $(x - 1, y)$, $(x, y - 1)$, and $(x + 1, y)$, with one in each of four directions: north, south, east, and west.

After many faulty nodes and misbehaving nodes are disabled from the collaboration, the rest of the nodes, also called enabled nodes, will constitute the WSN. According to the results presented in [9], when $R = \sqrt{5}r$, each enabled node can communicate with nodes in the neighboring grids. In each grid, one of the enabled nodes will be elected as the grid head. The rest of the enabled nodes in the same grid are called spare enabled nodes, or simply spare nodes. In this way, when each grid has its own head, the connectivity of all the heads and the coverage of the entire network can be guaranteed. Each head can monitor the status of the heads in neighboring grids. To minimize the coverage overlaps between the heads, we do not pursue the surveillance of diagonal neighboring grids for each head, which requires a larger communication range $R = 2\sqrt{2}r$ ($> \sqrt{5}r$). As a result, each movement monitored by a head will be limited within two neighboring grids. Each head knows the following information: (1) its grid location, and (2) the number of enabled nodes in the grid and their locations. Moreover, all the grids are connected in a Hamilton cycle in one direction; i.e., a directed Hamilton cycle (see the sample in Figure 1 (b)). Each head also monitors the area of the successor grid along such a directed Hamilton cycle and is in charge of communication with the corresponding head. The role of each head can be rotated within the grid.

It is noted that the grid partition with global information can only ensure one head existing in each grid territory.
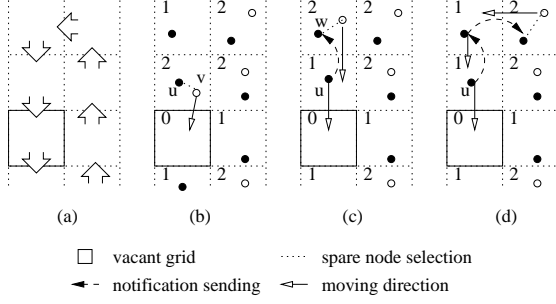
(a)   (b)   (c)   (d)

□   vacant grid   ⋯⋯   spare node selection

◂- - notification sending   ◂— moving direction

Figure 2. Vacant grid, its location along the Hamilton cycle, and the corresponding movement control.

By only using the 1-hop neighborhood information, we can guarantee the existence of heads in any $r \times r$ square territory with a localized coverage scheduling algorithm. After that, all the schemes presented in this paper can be extended easily under such a local view model. To make our movement control schemes clear, we only use the global partition model. Moreover, we describe the schemes in a round-based system. All the schemes presented in this paper can be extended easily to an asynchronous system. However, to simplify the discussion, we do not pursue the relaxation.

3 Replacement Process and its Analysis

This section introduces our control scheme to fill in any vacant grid with enabled nodes. As a result, each grid will have its own head and the coverage problem will be solved.

In our approach, each grid is monitored by not only its head, but also the head in the preceding grid along the directed Hamilton cycle, say node u. Whenever this grid becomes vacant, i.e., no head exists, the replacement process will be initiated immediately at u. In this way, the vacant grid can be detected and covered. It is noted that only the 1-hop neighborhood is used in our approaches and the control scheme is implemented in a fully distributed manner to support any dynamic change, which makes the entire system more scalable.

We summarize the replacement process as follows. First, that node u will select one spare node v in its grid to move to the vacant grid (see Figure 2 (b)). If such a spare node cannot be found, u itself will move to the vacant grid. Before the movement, u will send a notification to the head of its preceding grid (see the order in Figure 2 (a)). When the corresponding head, say node w, receives such a notification (in the next round), the above selection process will be repeated (see Figure 2 (c)) which then causes a so-called cascading movement (see Figure 2 (d)). The whole cascad-

ing movement process of nodes is snake-like. The details are shown in the following algorithm.

Algorithm 1: Mobility control scheme based on the directed Hamilton cycle

1. At a head u, the following replacement process will be initiated when u cannot find the head in the successor grid along the directed Hamilton cycle; i.e., a vacant grid in such a direction is detected.

2. Find a spare node in the grid of u, node v, to move into that vacant area before the next round starts.

3. If the above step fails, repeat the following steps until the notified node u can find a spare node in the above step: (a) Send the notification to the preceding grid to ask for a replacement for u itself. (b) Wait until the corresponding head w receives this notification. (c) Move u to the vacant grid before the next round starts; i.e., leaving the current grid vacant for cascading replacement.

Theorem 1: *Any vacant grid will gain a new head node in the above control scheme.*

Because of the use of the Hamilton cycle, the above replacement processes are able to cover the holes whenever spare nodes exist in the network. This will favor the networks with sparse deployment, or in the case when some critical condition disables the most of deployed node. This distinguishes our new approach from the existing ones [3, 6] which require at least $4 \times m \times n$ deployed nodes for a $m \times n$ grid system. Because such a Hamilton cycle is directed, the above replacement process is conflict-free when multiple holes occur in the networks simultaneously. As shown in the experimental results, this improves the success rate of hole-recovery. Moreover, due to the use of the directed Hamilton cycle, only one replacement will be initiated for a single vacant grid. That is, there is no overreaction [3], and each replacement is necessary. The following theorem provides an estimate on the average node movements, $\overline{M}$, in any replacement when nodes are deployed in a uniform distribution.

Theorem 2: *For any converged replacement process, $\overline{M} = \sum_{i=1}^{L} i \times P(i)$, where L (> 1) is the length (in hops) of the directed Hamilton path which is deduced from the directed Hamilton cycle, N is the number of spare nodes left in the entire network, and $P(i) =$*

$$
\begin{cases}
1 - (\frac{L-1}{L})^N & i = 1 \\
\prod_{k=1}^{i-1}(\frac{L-k}{L-k+1})^N & i = L \\
(1 - (\frac{L-i}{L-i+1})^N)\prod_{k=1}^{i-1}(\frac{L-k}{L-k+1})^N & otherwise
\end{cases}
\tag{1}
$$

Figure 3 shows our analytical results for a small-sized grid system (4×5) as well as a medium-sized grid system (16×16). For instance, when 12 spare nodes exist in the

(a) 4×5 grid system (L=19).　(b) 16×16 grid system (L=255).
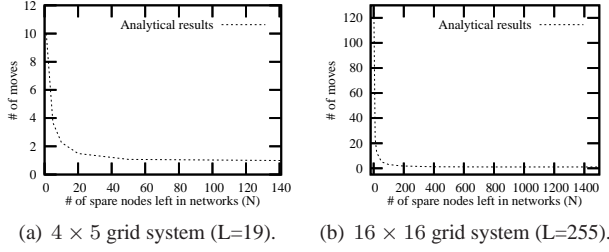
Figure 3. Analytical results of the number of node movements for a single replacement.

4×5 grid system, the replacement takes 2.0139 movements on average. That is, in most cases, the replacement process will converge within 2 movements. When the size of grid system is increased, it will take longer to approach the spare node along the Hamilton path. However, when the density of enabled nodes is kept above 1.68 per grid, the number of node movements can still be controlled to 2 in the 16×16 grid system. Compared with the minimum density of 4 per grid required in [3, 6], the improvement of our new approach is obvious.

4　Implementation Issue

In our approach, the replacement is initiated at a head node only when such a node cannot detect any enabled node in its successor grid. We connect all the grids along a directed Hamilton cycle to ensure that each vacant grid will have its own replacement initiated at one and only one of its neighboring grids.

For an $m \times n$ grid system, when either m or n is even, the Hamilton cycle, as shown in a sample 4×5 grid system in Figure 1 (b), can be built easily. When both m and n are odd, we constitute an $(m \times n - 1)$-hops Hamilton cycle with directed dual-paths. These two directed Hamilton paths share $(m \times n - 2)$ girds. For the remaining two grids, denoted by A and B, path one starts from the first and ends at the second, while path two starts from the second and ends at the first. Figure 4 shows a sample of this construction in a 5×5 grid system. Denote C as the common preceding grid of A and B. Denote D as the common successor grid of A and B. We have the following extended control scheme to ensure that Theorem 1 can still hold in such a Hamilton cycle with dual-paths.

Algorithm 2: Mobility control scheme for a grid system with the dual-path Hamilton cycle

1. Determine the grids A, B, C, and D.

2. When grid A becomes vacant, C will initiate a replacement process. Such a replacement will stretch along path two with
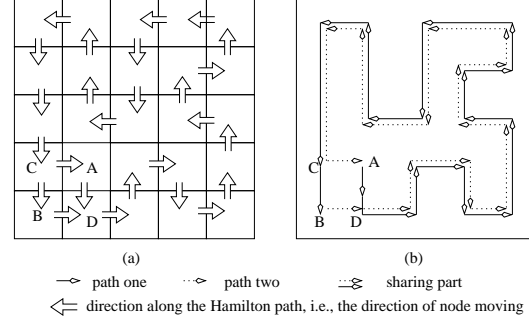

(a)　　　　　　　(b)

⊸ path one　　⋯⊸ path two　　⇒ sharing part
⇐ direction along the Hamilton path, i.e., the direction of node moving

Figure 4. Dual paths in construction of the Hamilton cycle in a 5×5 grid system.

Algorithm 1. Similarly, whenever grid B becomes vacant, its replacement initiated at C will stretch along path one.

3. When grid D becomes vacant, only B will initiate the replacement process and such a replacement will stretch along path one. However, at grid C, grid A with spare nodes is always preferred before the replacement continues to stretch along path one.

4. When any other grid becomes vacant, its replacement will follow the direction of the Hamilton cycle until it reaches grid D. From D, either A or B will be notified when any of them has at least one spare node.

Simply, the extended control scheme has three cases. In case one, when grid A becomes vacant, the replacement initiated at grid C will stretch along path two with Algorithm 1. Similarly, when grid B becomes vacant, the replacement initiated at grid C will force the node to move along path one. The replacement initiated for these two vacant grids can stretch as far as $(m \times n - 1)$ hops away along a Hamilton path; i.e., $L = m \times n - 1$. In case two, when grid D becomes vacant, only grid B will initiate the replacement process. After that, the process will stretch along path one. However, grid A with spare node(s) is always preferred for grid C to send out the replacement notification. After that, the replacement (if any) will continue to stretch along path one. In case three, when any other grid becomes vacant, the replacement process will stretch along the sharing part of both paths until it reaches grid D. From grid D, either A or B will be notified when any of them has at least one spare node. In both case two and case three, the replacement processes can stretch as far as $(m \times n - 2)$ hops away while it does not miss any chance to find a spare node in the entire network.

Corollary 1: *In a system with the dual-path Hamilton cycle, a vacant grid can be filled whenever at least one spare node exists in the network.*
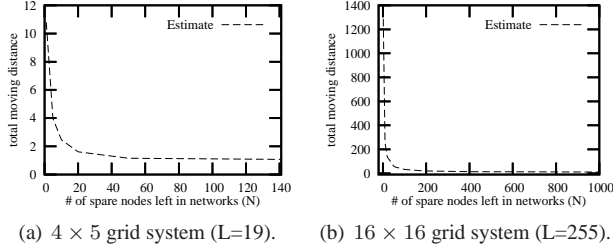
(a) 4×5 grid system (L=19).   (b) 16×16 grid system (L=255).

Figure 5. Estimates of the total moving distance for a single replacement ($r = 10$).

(a) The number of replacement processes initiated.
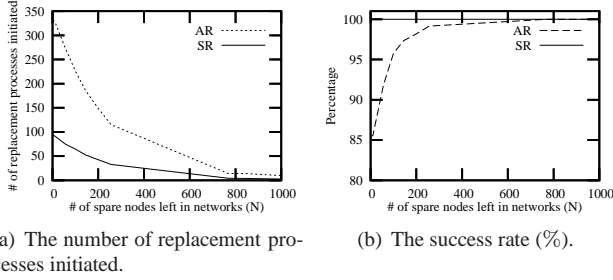
(b) The success rate (%).

Figure 6. Replacement process and its success rate.

Corollary 2: *For any converged replacement process in a $m \times n$ grid system with the dual-path Hamilton cycle,*

$$\overline{M} \quad \cong \quad \overline{M(m \times n - 2)} \tag{2}$$

where $\overline{M(k)} = \sum_{i=1}^{k} i \times P(i)$.

In Corollary 1, we prove that Theorem 1 can still hold in the dual-path Hamilton cycle. In Corollary 2, we provide the analysis on the number of node movement in each replacement process in the grid system with the dual-path Hamilton cycle.

Another important issue in our implementation is the mobility control in each node movement. To control the moving distance, each spare node moves straightforward to the central area of the target grid. The minimum distance is $\frac{1}{4}r$ and the maximum distance is $\frac{\sqrt{58}}{4}r$. In this paper, we use the average, $1.08 \times r$, to estimate the total moving distance. Figure 5 shows our estimates based on the results in Figure 3 when $r = 10$.

5 Experimental Results

In this section, we verify the improvement of our control scheme based on the Hamilton cycle (SR), and compared it with the best result known to date, as seen in [3]

(AR). The results show that our snake-like cascading movement will successfully cover any hole while substantially lowering the cost. For the deployed sensors with communication range $R = 10m$, we determine the grid size $4.4721m \times 4.4721m$ and then form the virtual grid system [9] in the target surveillance area. After deploying all the nodes in the uniform distribution, we randomly disable some nodes from the collaboration and create the holes. Then, the rest of the nodes are enabled nodes and they constitute the WSN. One of enabled nodes in each grid (if any) will be elected as the head. After that, we apply schemes SR and AR to fix the hole problem. Finally, we test the performance of different control schemes AR and SR in terms of their success in finding a spare node to fill the hole. We also test the cost of these schemes in terms of the number of replacement processes initiated, the total number of node movements and the total moving distance. The experimental results of our SR scheme are also compared with the analytical results to verify the correctness of our approach. It is noted that each movement of node u from one grid to its neighbor will randomly select the destination location in the central area of the target grid.

The variable parameters in our simulation are as follows. (1) Number of grids $m \times n$. Once the size of each grid has been decided, the surveillance area of security applications will determine the size of grid system needed. We use 16×16 in the simulation. (2) Number of spare sensors N in the networks. In [6], it has been mentioned that the control scheme can guarantee the coverage with at least $3m \times n$ spare nodes. Therefore, we deploy 5000 sensors and select those cases when N's value is in the range from 10 to 1000 ($\simeq 4m \times n$).

Figure 6 (a) shows the number of replacement processes initiated in schemes AR and SR in the cases with $(N + m \times n)$ enabled nodes. Figure 6 (b) shows how many of them (percentage-wise) will approach a spare node in the networks and converge successfully. We also show the number of node movements in both schemes in Figure 7 (a), and the total node movements in meters/distance for both schemes in Figure 8 (a). For the comparison, Figure 7 (b) and Figure 8 (b) show our analysis on the number of movements and the total moving distance in SR, respectively.

Results and observations can be summarized as follows:

- [3] claims that among all the existing movement-assisted methods to fix the hole, scheme AR has the best performance insofar as the total number of node movements and the total moving distance are concerned. However, some redundant procedures and node movements are required in this method. Due to the use of the directed Hamilton path, the replacement processes initiated around the vacant grid can be synchronized. As a result, fewer than 50% replacement processes are needed in SR. The total number of node
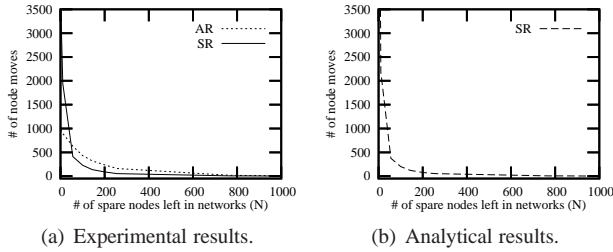
(a) Experimental results.   (b) Analytical results.

Figure 7. The number of node movements.

(a) Experimental results.   (b) Analytical results.

Figure 8. The total moving distance of nodes (meters).

movements and the corresponding total moving distance can be reduced substantially.

- When $N < 55$, SR requires a long path along the Hamilton cycle to approach the spare node. More node movements and moving distance are required in SR method. However, the AR method has $10\% \sim 20\%$ failures in replacement processes while the success rate is always 100% in SR method; that is, the surveillance coverage is less robust in AR method in the networks with lower node density.

- When $N \geq 55$ (i.e., > 1.22 enabled node per grid) which is more common in real applications, SR requires fewer node movements and less moving distance while keeping the success rate higher than AR. As shown in the results, our new approach SR is more cost-effective than AR in usual cases.

- The cascading movement is adopted in both AR and SR. The SR replacement process is just one of the cases of the SR replacement processes that are along a special path. SR has the same bound of converging speed as AR, which has been presented in [3].

- A short-cut along the Hamilton cycle can reduce the length of the path for replacement process to approach a spare node. The construction of such a short-cut will be our future work to further increase the convergence speed of SR. As a result, the cost of SR will be reduced greatly in the cases when $N < 55$.

6 Conclusion

In this paper, we have presented a more cost-effective, snake-like replacement process to cover the surveillance holes of WSNs where all sensors deployed in certain sensing areas are disabled from the collaboration. As a result, the connectivity and coverage of WSNs can be guaranteed, even when the working status of nodes changes dynamically. In our meth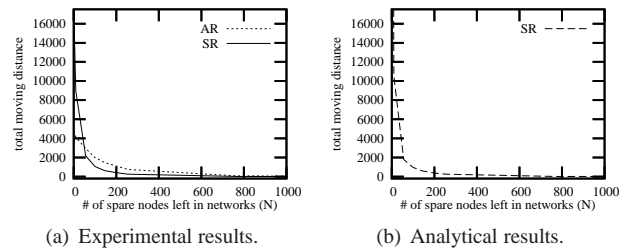ods, only the 1-hop neighborhood is used, and the adjustment of nodes can be controlled within the local area under a synchronization model based on the Hamilton cycle. The analytical and experimental results show the proposed method to be robust and scalable with minimized costs. In our future work, a more effective synchronization model will be considered to further reduce the length of stretch path in each replacement.

References

[1] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: A survey. *Computer Networks*. Vol. 38, 2002, pp. 393-422.

[2] F. Hu and N. Sharma. Security considerations in ad hoc sensor networks. *Ad Hoc Networks*. Vol. 3, No. 1, 2005, pp. 69-89.

[3] Z. Jiang, J. Wu, A. Agah, and B. Lu. Topology control for secured coverage in wireless sensor networks. *Proc. of 3rd Workshop on Wireless Sensor Networks Security (WSNS'07), in conjunct with MASS'07*. 2007, CD-ROM.

[4] Z. Jiang, J. Wu, and R. Kline. Mobility control for complete coverage in wireless sensor networks. *Technical Report, West Chester University*. 2007, document available at http://www.cs.wcupa.edu/ zjiang/hole_icdcs_TR.pdf.

[5] G. Wang, G. Cao, and T. Porta. Movement-assisted sensor deployment. *IEEE Transactions on Mobile Computing*. Vol. 5, No. 6, June. 2006, pp. 640-652.

[6] J. Wu and S. Yang. SMART: A scan-based movement-assisted sensor deployment method in wireless sensor networks. *Proc. of INFOCOM 2005*. Vol. 4, March 2005, pp. 2313-2324.

[7] N. Xu. A survey of sensor network applications. *Technical report, University of Sourthern California*. available at http://courses.cs.tamu.edu/rabi/cpsc617/resources/sensor%20 nw-survey.pdf.

[8] W. Xu, K. Ma, W. Trappe, and Y. Zhang. Jamming sensor networks: Attack and defense strategies. *IEEE Network*. Vol. 20, No. 3, 2006, pp. 41-47.

[9] Y. Xu and J. Heidemann. Geography-informed energy conservation for ad hoc routing. *Proc. of the ACM/IEEE International Conference on Mobile Computing and Networking (MOBICOM'01)*. 2001, pp. 70-84.

[10] Y. Zou and K. Chakrabarty. Energy-aware target localization in wireless sensor networks. *Proc. of the First IEEE International Conference on Pervasive Computing and Communications (PerCom'03)*. 2003, pp. 60.