

# A Hierarchical Structure based Coverage Repair in Wireless Sensor Networks \*

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

Jie Wu  
Computer Sci. & Eng. Dept.  
Florida Atlantic University  
Boca Raton, FL 33431, USA  
E-mail: jie@cse.fau.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 area based on a hierarchical structure, one (and only one) snake-like cascading repair 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. The analytical and experimental results show substantial improvements of our approach 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, multifunction sensor devices [7]. 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) [3]. Applications of WSNs range from environmental monitoring to surveillance to target detection. Due to the fact that sensors can very easily fail or misbehave, many nodes could be isolated from the network collaboration [11]. 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 [23], an 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 functions correctly, a complete coverage of its surveillance area must be provided.

Recent attention has been drawn to use mobility for improving the communication performance in WSNs [8, 16, 20, 25]. Rather than preventing the occurrence of holes, a new solution is proposed in this paper to repair the coverage by moving some spare nodes into the vacant area. A spare node is the node that will not cause any loss of connectivity (as well as coverage) upon removing it. Due to critical resource constraints, such a solution must be controlled within a small area in order to avoid too many other nodes involved. We attempt to provide local optimization in reducing the cost of repair process: not only the moving distance of node(s) in each single repair process, but also the total number of repair processes required for any hole occurred in the networks. The challenge to achieve such an optimization is the lack of global information. In this paper, we provide a hierarchical structure information model based on eye theory [4, 13] to maintain the information of coverage and connectivity. Such information will be used to determine the spare node for moving and to synchronize all the repair moving processes. There are three major contributions.

- First, a new information model is proposed for determining where is the hole and where has the spare node to fix the hole, in a localized manner. This is the first time to provide an infrastructure for applications in WSNs to maintain the information of the coverage redundancy and the coverage deficit.
- Second, under the hierarchical information model, a localized solution is provided to quickly locate a spare node nearby the hole and then to move it into the vacant area. Compared with other repair schemes that need a real-time detection to blindly seek the spare

---

\*This work was supported in part by NSF grants CNS 0422762, CNS 0434533, CNS 0531410, and CNS 0626240.

node [19, 26] or a global balancing [22], the new approach is more efficient in solving the hole problem.

- Third, both the analytical and the experimental results are provided to show substantial improvement of using such a hierarchical structure in coverage repair, compared with the best results known to date.

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 [24]. In each grid, one and only one enabled node will be elected as the grid head. The rest of the enabled nodes in the same grid are called spare nodes. As indicated in [24], the connectivity and coverage of networks can be guaranteed if each grid has its own head. A grid head will monitor (a) whether a spare node exists in its grid area (providing redundant coverage), and (b) whether it can connect to a node in neighboring grids (keeping the connectivity).

According to their locations, each 4 neighboring unit grids form a Hamilton cycle [1] (i.e., a level-1 cycle in the hierarchical structure). One of them will be selected as the eye to collect all the coverage and connectivity information along such a cycle. And then, each 4 selected eyes form a higher level (i.e., level-2) Hamilton cycle. This process will continue until all the unit grids are connected. After this construction, each cycle will dominate a certain area and the head node in its eye will maintain all the coverage and connectivity information of that area.

When a grid does not have the head, the directed Hamilton cycle turns into a directed Hamilton path. One (and only one) *repair process* will be initiated along that directed path to move a spare node into the vacant area. In this way, any redundant process that might be initiated nearby the hole can be avoided. Each repair process will always converge at an eye that can locate the existing spare node within its dominated area. Whenever a local area does not have any spare node, the request will be sent along upper level cycle from that dominating eye. This process will continue until a spare node can be located in a larger area. Inspired by the early work for deadlock-free routing around a faulty area that uses the mutual exclusion at only one vertex [21], any request conflict and deadlock can be avoided in the final spare node localization at an eye. Moreover, in such a localized solution, the cost can be minimized in terms of the number of total node movements and the total moving distance. In both analytical and experimental results, we will show the substantial improvement of our approaches compared with the best results known to date.

The remainder of the paper is organized as follows: Section 2 introduces some necessary notions and preliminaries. Our control method and its performance analysis are introduced in Section 3. Its implementation issues are discussed

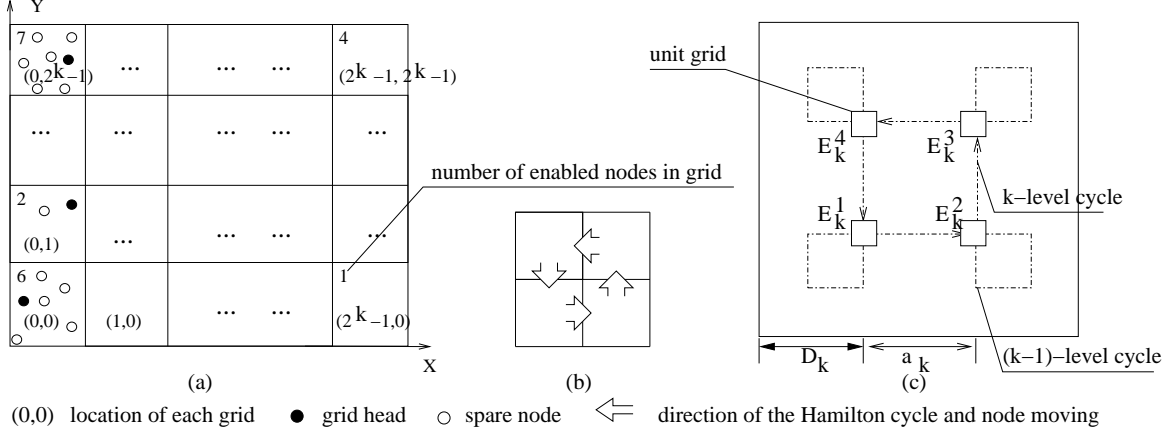
in Section 4. Section 5 shows the simulation results as well as the analytical results. Section 6 concludes this paper and provides ideas for future research.

## 2 Preliminary

We assume that all the nodes have the same sensing range and 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 [9] at some fixed nodes [12] or a mobile beacon node [17], or just by relying on the relative coordinate system [6]. We partition the whole network into an  $m \times n$  2-D grid system (see Figure 1 (a)). To simplify the discussion, assume  $m = n = 2^k$ . Each grid is of a square size  $r \times r$  and is denoted by its relative location in the entire system, say  $(x, y)$  ( $0 \leq x, y \leq 2^k - 1$ ). 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,  $|x_1 - x_2| + |y_1 - y_2| = 1$ . Each grid  $(x, y)$ , except the one 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. In general,  $[x : x', y : y']$  represents a rectangular region with four vertex grids:  $(x, y)$ ,  $(x, y')$ ,  $(x', y')$ , and  $(x', y)$ .

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 [24], 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. 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 (i.e., coverage redundancy). Moreover, each head monitors the area of the neighboring grids and detects the coverage deficit if any (i.e., possible occurrence of the hole). It is also in charge of communication with the corresponding head in neighboring grids to keep the connectivity of the entire network. The role of each head can be rotated within the grid.

It is noted that the grid partition with global information can ensure only one head existing in each grid territory. In a grid partition with only 1-hop neighborhood information, we can guarantee the existence of heads in any  $r \times r$  square territory by using a localized coverage scheduling



**Figure 1. (a) Virtual grid system and grid heads. (b) Directed Hamilton cycle in local unit grids. (c) Hierarchical Hamilton cycle structure.**

algorithm, such as the one presented in [18]. Then, the synchronization between those local heads is needed for any decision making in the repair process. After that, all the schemes presented in this paper can be extended easily under such a partition with local views. To make our movement control schemes clear, we only use the global partition model.

After the grid partition, each head can monitor the status of the neighboring heads. 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  $R = 2\sqrt{2}r (> \sqrt{5}r)$ . It is also noted that when a grid does not have any enabled node, its area may still be covered by its neighboring heads and does not really need a repair. However, in our approach, the repair process will be initiated under such a case to provide a redundancy precaution. This will help to balance the enabled nodes more evenly so that the occurrence of a real hole can be avoided. This kind of redundancy is also a tradeoff for providing a localized solution.

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. We assume that the sensing range and communication range of each node are the same and have a uniform disk of radius. It is noted that the proposed methods can also be supported in other models, such as the exponential model in [27], the staircase model in [2], and the realistic model in [5]. To make the entire system more scalable, all data communication is determined and implemented in the information exchanges between two neighboring nodes.

### 3 Hierarchical Structure and the Corresponding Coverage Repair

This section introduces our control scheme to fill in any vacant grid with enabled nodes. As a result, with a minimum moving cost, each grid will have its own head and the coverage problem will be solved. Obviously, such an optimization relies on the availability of the information of coverage deficit and coverage redundancy. In this section, we first introduce the proposed hierarchical structure and then, the corresponding coverage repair.

#### 3.1 Hierarchical structure information model

In the local area, each four neighboring unit grids constitute a Hamilton cycle, i.e., the level-1 cycle in the hierarchical structure (see Figure 1 (b)). One of the unit grids is selected as the eye of such a cycle based on the eye theory [4]. The head of this eye will collect the information of the existence of spare nodes in these unit grids along the cycle. And then, each four level-1 eyes will form a higher level (i.e., level-2) Hamilton cycle to share the information. This process will continue until a level- $k$  cycle for the entire region  $[0 : 2^k - 1, 0 : 2^k - 1]$  is built on four level- $(k - 1)$  eyes (see Figure 1 (c)).

Obviously, any selected grid on a level- $k$  cycle is also an level- $i$  eye ( $1 \leq i < k$ ). The position of each level- $i$  ( $1 \leq i \leq k$ ) grid can be determined as follows. For the region  $[0 : 2^k - 1, 0 : 2^k - 1]$ , four grids  $E_k^1(D_k, D_k)$ ,  $E_k^2(2^k - 1 - D_k, D_k)$ ,  $E_k^3(2^k - 1 - D_k, 2^k - 1 - D_k)$ , and  $E_k^4(D_k, 2^k - 1 - D_k)$ , constitute the level- $k$  directed

Hamilton cycle, where  $D_k$  is defined in [13] as

$$\begin{cases} 0 & k = 1 \\ 2^{k-1} - 1 - D_{k-1} & k > 1 \end{cases} \quad (1)$$

These grids are connected by four  $a_k$ -hops-distance path [4] where

$$a_k = \frac{1}{3}(2^k - (-1)^k). \quad (2)$$

The entire region is partitioned into four parts with one level- $(k-1)$  eye in each:

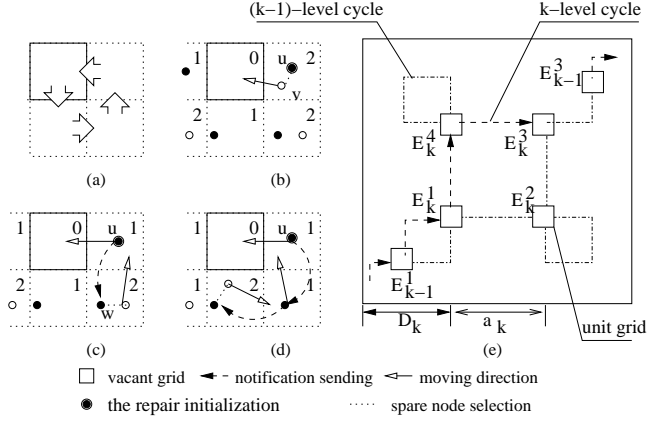
$$\begin{aligned} & [0 : 2^{k-1} - 1, 0 : 2^{k-1} - 1], \\ & [2^{k-1} : 2^k - 1, 0 : 2^{k-1} - 1], \\ & [2^{k-1} : 2^k - 1, 2^{k-1} : 2^k - 1], \\ \text{and} & [0 : 2^{k-1} - 1, 2^{k-1} : 2^k - 1]. \end{aligned}$$

The head node in each selected eye is used to monitor the existence of the spare nodes in the corresponding partitioned sub-region. They can exchange such information via those  $a_k$ -hops-distance paths. Each sub-region and the head of the corresponding grid eye is called the dominated area and the dominating node, respectively. After that, the above procedure can continue and apply to any level- $(k-1)$  cycle in each sub-region. The definition of  $D_k$  ensures that a selected grid eye on level- $k$  cycle is also selected for the level- $(k-1)$  cycle (see Figure 1 (c)). Therefore, the information of the existence of spare nodes to be shared along the level- $k$  cycle can be collected through those level- $(k-1)$  eyes. Such a process will continue until reaching the unit grid. The details are shown in Algorithm 1.

---

**Algorithm 1:** Construction of hierarchical directed Hamilton cycles with the information of the existence of spare nodes.

1. For the region  $[0 : 2^k - 1, 0 : 2^k - 1]$ , four grids  $E_k^1(D_k, D_k)$ ,  $E_k^2(2^k - 1 - D_k, D_k)$ ,  $E_k^3(2^k - 1 - D_k, 2^k - 1 - D_k)$ , and  $E_k^4(D_k, 2^k - 1 - D_k)$ , constitute the level- $k$  directed Hamilton cycle, connected by four  $a_k$ -hops-distance path where  $D_k$  and  $a_k$  are defined in Equation 1 and Equation 2 respectively.
  2. The entire region is partitioned into four parts:  $[0 : 2^{k-1} - 1, 0 : 2^{k-1} - 1]$ ,  $[2^{k-1} : 2^k - 1, 0 : 2^{k-1} - 1]$ ,  $[2^{k-1} : 2^k - 1, 2^{k-1} : 2^k - 1]$ , and  $[0 : 2^{k-1} - 1, 2^{k-1} : 2^k - 1]$ .
  3. Repeat the above Hamilton cycle construction and the partition in each sub-region until reaching unit grids.
  4. The head node in each grid of a level-1 Hamilton cycle maintains the information of the spare nodes in its own grid. The head node of the eye of each level- $i$  ( $1 \leq i \leq k$ ) cycle collects such spare node information along this cycle via those  $a_i$ -hops connecting paths.
- 



**Figure 2. (a) Level-1 directed Hamilton cycle, (b), (c), (d) Samples of the repair along level-1 cycle (intra-level repair), and (e) sample trajectory in an inter-level repair.**

### 3.2 Repair process

With the proposed hierarchical structure, each grid is not only directly threaded by a level-1 directed Hamilton cycle with other 3 neighboring grids, but also indirectly connected to all other grids in the entire network. Each grid head does not only monitor the existence of spare nodes in its grid (i.e., coverage redundancy), but also monitors the existence of a head in each neighboring grid. Whenever such a head detects that a neighboring grid does not have any enabled node inside, a repair process is initiated to fix that vacancy. The proposed repair process has two phases. We summarize the repair process as follows.

In phase one, which is also called intra-level phase, the spare node is quickly located in neighboring grids. First, the repair process is initiated at a grid head, say node  $u$ , only when it detects the vacant neighboring grid in the direction of the level-1 directed Hamilton cycle (see the repair initialization in Figure 2 (b)). Then, 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  $w$  of its preceding grid (see the order in Figure 2 (a)). When the head  $w$  receives such a notification (in the next round), the above selection process will be repeated (see Figure 2 (c)), causing a so-called cascading movement. The whole cascading movement process of nodes is snake-like while each movement is limited within two neighboring grids. For each movement causing the change of spare nodes in a grid, the new spare node information will be collected by the head of unit grid. And then, such information will be updated in the hierarchical

cycles in the way in step 4 in Algorithm 1. The details intra-level repair are shown in the following algorithm.

---

**Algorithm 2:** Mobility control within neighboring grids (intra-level repair).

1. At a head  $u$ , the following repair process will be initiated when  $u$  cannot find the head in the successor grid along the level-1 directed Hamilton cycle; i.e., a vacant grid in such a direction is detected.
2. Find a spare node in the grid of  $u$ , say node  $v$ , to move into that vacant area before the next round starts.
3. If the above step fails, repeat the follows until the notified node  $u$  can find a spare node in the above step: (a) Send the notification to the preceding grid along the level-1 directed cycle to ask for a repair 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 repair.
4. Each head  $u$  detects the spare nodes existing within the same grid. The new information will be updated in the hierarchical structure in the way in step 4 of Algorithm 1.

---

In phase two, whenever the head node of the eye of level-1 cycle, say node  $u$ , realizes the lack of the spare node in all neighboring grids, the second phase is conducted with the information of the spare nodes collected in the hierarchical structure. This phase is also called inter-level repair. First, a notification will be sent along level-2 Hamilton cycle, which also passes through  $u$ . That is, the repair process is applied to an upper level cycle. This process will continue until the spare node is found at the eye of one cycle, say level- $i$  ( $i < k$ ) cycle. That is, a spare node can be provided in the partition region of that eye. After that, that spare node will be localized from that eye down to unit grid as follows. A search for the spare node is initiated at that level- $i$  eye and will go along the corresponding level- $i$  cycle. For each level- $j$  ( $j < i$ ) eye it reaches, the above search process will continue until it reaches the unit grid along level-1 cycle. Then, from that unit grid, a spare node will be selected. An example of such a process is shown in Figure 2 (e). Eventually, the nodes will move in the cascading way along the path that is constituted in the above process. The spare node information will be updated to date as well as the node moving. Algorithm 3 shows the details of the inter-level repair process.

---

**Algorithm 3:** Mobility control with hierarchical directed Hamilton cycles (inter-level repair).

1. Reservation process. The notification for a spare node is sent along the upper level cycle until such a spare node can be provided in the record of an eye (say node  $u$ ) along one cycle. Then, this spare node will be reserved unless there is another request for it coming from higher level cycle.
  2. Spare node localization. From that eye  $u$ , the existence of such a spare node is verified level by level among the cycles down to unit grid.
  3. Cascading movements. The nodes will move in the cascading way along the path constituted in the above, unless a localization process from higher level cycle seizes this cascading movement. In that case, the request for a spare node will re-start from step 1.
  4. Information update. The spare node information will be updated to date in the way in step 4 of Algorithm 1.
- 

### 3.3 Performance Analysis

After the construction of the hierarchical structure, all the unit grids are connected together. Therefore, any repair process is able to find the spare to cover the vacant target grid wherever such a node exists in the network. This will favor the networks with sparse deployment, or in the case when some critical condition disables the most of deployed nodes.

Under the virtual grid model, to detect the vacant neighboring grid, the communication range of each grid head is set large enough to cover all neighboring grids. In other words, each grid can be monitored by as many as 4 grid heads from all its neighboring grids. Without synchronization, the existence of a vacant grid may incur 4 repair processes simultaneously, causing redundant processes and some unnecessary node movements.

To reduce the redundant repair processes that might be initiated for the same vacant grid, the directed path is used to determine one and only one neighboring grid head that conducts a valid detection and then initiates the corresponding repair process. The use of the Hamilton cycle is to guarantee the unique detection direction for each grid. This can ensure the initialization of repair and only the necessary initialization for each vacant grid occurred.

Another synchronization issue occurs when multiple holes appear in the networks, especially when a hole occurs during the ongoing repair process to cover a previously detected hole. To solve the conflict or deadlock of spare node

request, some synchronization schemes must be provided.

First, the detection direction is also used to help the grid head to forward the spare node request. Those requests initiated within the level-1 cycle cannot form any loop in their stretch paths due to the existence of that vacant grid. However, when a spare node exists in the neighbor grid in reverse direction along such a cycle, it cannot be quickly located. Although this inefficiency is reduced to a minimum by connecting only 4 neighboring grids in a cycle, the intra-level repair is still not optimal. This is the tradeoff for reducing spare node request conflict.

Second, the information of the existence of spare nodes will be collected by the head of each unit grid and then will be maintained in the hierarchical structure up to the top level cycle (i.e., level- $k$  cycle). It is noted that the reservation for a spare node is only made in the eye that dominates both that vacant grid and the grid of that spare node. For any two inter-level repairs locating the same spare node, the reservation cannot be made in the same eye. The one made in the eye of a higher level cycle which has started the information update in upper level cycles will have the higher priority. When the localization process of a higher priority repair reaches the eye  $v$  that made the lower priority reservation, it will seizure the localization and the cascading movements of the lower priority repair, including the information update for using that reserved spare node. The lower priority repair will restart from that eye  $v$  to locate another available spare node. It will be treated as a restart of an asynchronous process after a holding period.

It is note that for the purpose of avoiding request conflict, the spare node cannot move directly to that vacant grid and must follow the exact stretch path of request process.

**Theorem 1:** Any vacant grid will be with a new head node in the above control schemes.

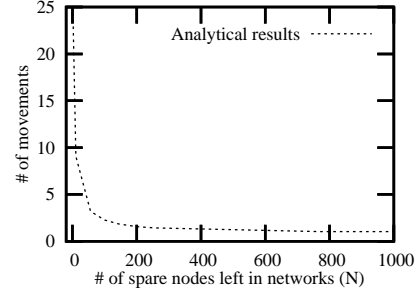
**Proof:** According to the above study on the conflict control in the repair process, it is obvious that each repair will successfully converge at the end. For each initiated repair process, no additional repair is needed.

Because of the use of the hierarchical structure, all the grids are connected. The above repair process can find any spare node if such a node exists in the networks.

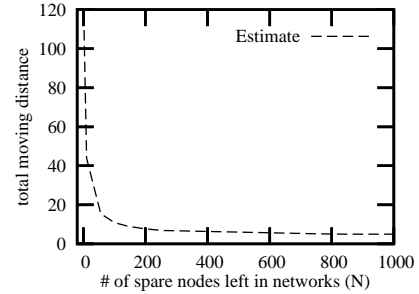
Assume that the number of spare nodes is always larger than the number of vacant grids. We can find at least one vacant grid whose repair can converge first. The other repair processes can continue until all the vacant holes are filled with spare nodes. Therefore, any vacant grid will be filled by a new head node via the above control scheme. ■

**Corollary 1:** In a  $2^k \times 2^k$  grid system under the above hierarchical structure model, the maximum length of the stretch path in the repair process is  $O(2^k)$ .

**Proof:** Under the hierarchical structure model constructed by Algorithm 1, the maximum length of the stretch path in



(a) The number of node movements (analytical results).



(b) The total moving distance when  $R = 10$  (estimated).

**Figure 3. Cost of a single repair in the hierarchical repair process in a  $16 \times 16$  grid system.**

a level- $i$  cycle is  $3a_i$ . Therefore, the maximum length of the entire path  $L = 3a_1 + 3a_2 + \dots + 3a_k + 3a_{k-1} + \dots + 3a_1 = \sum_{j=1}^k 6a_j - 3a_k$ . Because  $a_k = \frac{1}{3}(2^k - (-1)^k)$ ,  $L < 6\frac{1}{3}(2^1 + 2^2 + \dots + 2^k) + 1 = 2 \times (2 \times 2^k - 2) + 1 = 4 \times 2^k - 3$ . Therefore,  $L = O(2^k)$ . ■

The following theorem provides an estimate on the average node movements,  $\overline{M}$ , in any single repair process when nodes are deployed in a uniform distribution, with  $N$  spare nodes still available in the entire network.

**Theorem 2:** If a  $2^k \times 2^k$  grid system still have  $N$  spare nodes available, for any converged repair process,  $\overline{M} = \sum_{i=1}^k \overline{M}_i$ , where  $\overline{M}_i$  is the average length of the stretch path that reaches level- $i$  cycles.  $\overline{M}_i =$

$$\begin{cases} \sum_{j=1}^3 j \times P_k(1, j) & i = 1 \\ (2a_i + \sum_{j=1}^{i-1} 4a_j) \times P_k(i) & 1 < i \leq k \end{cases} \quad (3)$$

where  $P_k(1, j) =$

$$\begin{cases} 1 - (1 - \frac{1}{2^{2k}})^N & j = 1 \\ (1 - (1 - \frac{j}{2^{2k}})^N) \prod_{t=1}^{j-1} (1 - \frac{t}{2^{2k}})^N & \text{otherwise} \end{cases}$$

and  $P_k(i) =$

$$\begin{cases} (\prod_{t=1}^3 (1 - \frac{t}{2^{2k}})^N) (\prod_{t=2}^{i-1} (1 - \frac{2^{2t}-1}{2^{2k}})^N) & i = k \\ (\prod_{t=1}^3 (1 - \frac{t}{2^{2k}})^N) (\prod_{t=2}^{i-1} (1 - \frac{2^{2t}-1}{2^{2k}})^N) & i < k \\ \times (1 - (1 - \frac{2^{2i}-1}{2^{2k}})^N) & \end{cases}$$

**Proof:** With total  $2^{2k}$  grids,  $\overline{M(1)}$  is the average length of stretch path of the repair process that can find a spare node in level-1 cycle, i.e., the average node movements in the internal phase.  $P_k(1, j)$  is the probability that such a spare node is located in the unit grid on level-1 cycle that is  $j$ -hops far from the vacant grid. We have

$$\begin{aligned} \overline{M(1)} &= 1 - (1 - \frac{1}{2^{2k}})^N \\ &\quad + 2 \times (1 - (1 - \frac{2}{2^{2k}})^N) (1 - \frac{1}{2^{2k}})^N \\ &\quad + 3 \times (1 - (1 - \frac{3}{2^{2k}})^N) (1 - \frac{1}{2^{2k}})^N \\ &\quad \times (1 - \frac{2}{2^{2k}})^N \\ &= \sum_{j=1}^3 j \times P_k(1, j) \end{aligned}$$

Along a level- $i$  cycle, the average length of the path from a selected grid to the eye is  $2a_i$ . Therefore, the average length of a path in the notification process that reaches the level- $i$  cycle is  $2a_1 + 2a_2 + \dots + 2a_i + 2a_{i-1} + \dots + 2a_1 = 2a_i + \sum_{j=1}^{i-1} 4a_j$ .  $P_k(i)$  is the probability that at least one spare node exists in the region of the level- $i$  cycle with the vacant grid, but not any lower level cycle with the vacant grid. Therefore, the proof of the above statement is obvious. ■

Figure 3 (a) shows our analytical results for a  $2^4 \times 2^4$  grid system. Even when there is only one spare node available in the entire network, the number of node movements for a single repair process can be controlled to 24.5. Based on the analytical results, Figure 3 (b) shows our estimate on the total moving distance in a single repair process in such a grid system.

## 4 Implementation Issue

As we discussed early, a  $2^k \times 2^k$  grid system is necessary condition for Algorithm 1. Here we extend Algorithm 1 to an  $m \times n$  grid system. The extended algorithm also starts from the localization of 4 selected grids (i.e., eyes) along the top level Hamilton cycle. Then, the dominated sub-region for each selected eye is determined. After that, as the same way in step 3 in Algorithm 1, the cycle construction will be applied to each sub-region again until reaching the unit grid. Although optimality no longer holds, an upper bound of the corresponding length of stretch path is given.

In the region  $[0 : m-1, 0 : n-1]$ , four eyes on top level Hamilton cycle, say level- $k$ , are defined as  $E_K^1 : (D_m, D_n)$ ,  $E_K^2 : (m-1-D_m, D_n)$ ,  $E_K^3 : (m-1-D_m, n-1-D_n)$ ,

and  $E_K^4 : (D_m, n-1-D_n)$ .  $D_m$  and  $D_n$  are defined by

$$D_\alpha = \begin{cases} 0 & \alpha = 1 \\ \lceil \frac{\alpha}{2} \rceil - 1 - D_{\lceil \frac{\alpha}{2} \rceil} & \alpha > 1 \end{cases} \quad (4)$$

where  $\alpha$  is either  $m$  or  $n$ .

**Lemma 1** [13]:  $\lfloor \frac{\alpha-1}{3} \rfloor \leq D_\alpha \leq \lfloor \frac{\alpha+1}{3} \rfloor$  for any  $\alpha \geq 1$ .

**Proof:** Details can be found in [13]. ■

Along this cycle,  $E_K^1$  and  $E_K^2$  (or,  $E_K^3$  and  $E_K^4$ ) are connected by a  $A_m$ -hops-distance path along the  $X$  dimension. Similarly,  $E_K^2$  and  $E_K^3$  (or,  $E_K^4$  and  $E_K^1$ ) are connected by a  $A_n$ -hops-distance path along the  $Y$  dimension.  $A_m$  and  $A_n$  can be calculated by

$$A_\alpha = \lceil \frac{\alpha}{2} \rceil - D_\alpha + D_{\alpha - \lceil \frac{\alpha}{2} \rceil} = 1 + D_{\lfloor \frac{\alpha}{2} \rfloor} + D_{\lceil \frac{\alpha}{2} \rceil}, \quad (5)$$

where  $\alpha$  is either  $m$  or  $n$ .

**Lemma 2:**  $\lfloor \frac{\alpha-1}{3} \rfloor \leq A_\alpha \leq 1 + \lceil \frac{\alpha}{3} \rceil$  for any  $\alpha \geq 1$ .

**Proof:** Based on Lemma 1, the details can be found in [13]. ■

Then, the region  $[0 : m-1, 0 : n-1]$  is partitioned into four sub-regions with an eye in each:

$$\begin{aligned} &[0 : \lceil \frac{m}{2} \rceil - 1, 0 : \lceil \frac{n}{2} \rceil - 1], \\ &[\lceil \frac{m}{2} \rceil : m-1, 0 : \lceil \frac{n}{2} \rceil - 1], \\ &[\lceil \frac{m}{2} \rceil : m-1, \lceil \frac{n}{2} \rceil : n-1], \\ \text{and } &[0 : \lceil \frac{m}{2} \rceil - 1, \lceil \frac{n}{2} \rceil : n-1]. \end{aligned}$$

Obviously, we can ensure that Theorem 1 can still hold in such a hierarchical structure with new partitions. We can also have the following result in corollary.

**Corollary 2:** In a  $m \times n$  grid system under the above hierarchical structure model, the maximum length of the stretch path in the repair process is  $O(\max\{m, n\})$ .

**Proof:** Assume  $\beta = \max\{m, n\}$ . We have  $A_m, A_n \leq A_\beta < \frac{\beta+6}{3}$  (based on Lemma 2). Under the hierarchical structure model constructed by Algorithm 1, the length of the stretch path in the top level cycle is no more than  $3A_\beta$  (i.e.,  $< \beta + 6$ ). Then, based on the partition, the entire path  $L$  has  $\frac{\beta}{3} - 2 < (\beta + 6) + (\frac{\beta}{2} + 6) \dots + (1 + 6) + (1 + \log \beta) < 3\beta + 6 < 4\beta$ . Therefore,  $L = O(\beta) = O(\max\{m, n\})$ . ■

As the result, the average number of node movements in a single repair process in the  $m \times n$  grid system can be controlled as well as it is controlled in the  $2^k \times 2^k$  grid system (see Theorem 2).

## 5 Performance Results

In this section, we verify the improvement of our control scheme based on the hierarchical Hamilton cycles (HR).

We also compare our approaches with other repair solutions, as seen in [14] (*AR*) and [15]. 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 [24] 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 and they constitute the WSN. One of enabled nodes in each grid (if any) will be elected as the head node. After that, we apply schemes *SR*, *HR*, and *AR* to fix the hole problem. At last, we test the performance of different control schemes *AR*, *HR*, 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 repair processes initiated, the total number of node movements and the total moving distance. Some analytical results of our *HR* scheme is compared with the results of *SR* scheme which have been presented in [15]. These analytical results will also be compared with the corresponding experimental results to verify the correctness of our approaches. 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 turnable 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  $16 \times 16$  in the simulation. (2) Number of spare sensors  $N$  in the networks. In [22], 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 4 (a) shows the number of repair processes initiated in schemes *AR*, *HR*, and *SR* in the cases with  $(N + m \times n)$  enabled nodes. Figure 4 (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 all schemes in Figure 5 (a), and the total node movement in meters/distance for all schemes in Figure 6 (a). For the comparison, Figure 5 (b) and Figure 6 (b) show our analysis on the number of movements and the total moving distance, respectively, for both *SR* and *HR*.

Results can be summarized as follows:

1. It is claimed in [15] that among all the existing movement-assisted methods to fix the hole, scheme *SR* has the best performance insofar as the total number of node movements and the total moving distance are concerned. As a result, scheme *HR* achieves the

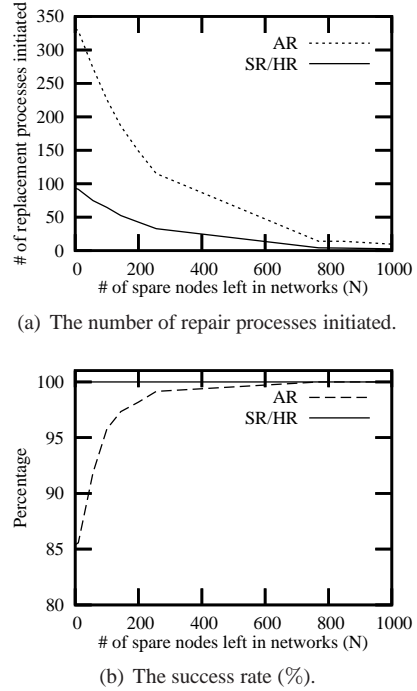


Figure 4. Repair process and its success rate.

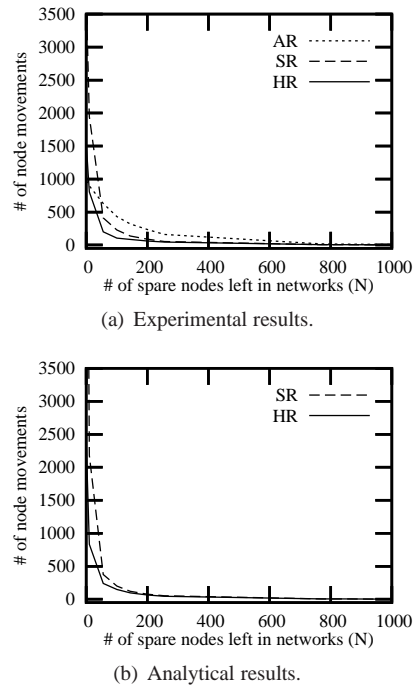
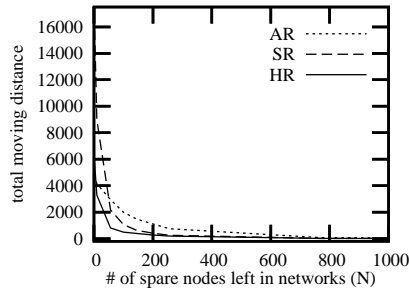
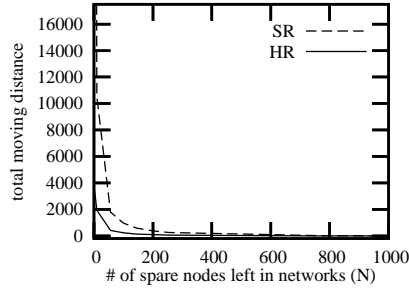


Figure 5. The number of node movements.



(a) Experimental results.



(b) Analytical results.

**Figure 6. The total moving distance of nodes (meters).**

same success as scheme *SR* does. Fewer than 50% processes are needed in *SR* and *HR*, compared with the results of *AR*.

2. When  $N < 55$ , *AR* method has 10% ~ 20% failures in repair processes while the success rate is always 100% in *SR* and *HR* methods; that is, the surveillance coverage is less robust in *AR* method in the networks with lower node density. However, *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. With the hierarchical structure, *HR* can ensure the success of each repair while keeping both the node movements and the total moving distance minimal.
3. When  $N \geq 55$  (i.e.,  $> 1.22$  enabled nodes per grid) which is more common in real applications, *SR* and *HR* requires fewer node movements and less moving distance while keeping the success rate higher than *AR*. As shown in the results, *HR* is the most cost-effective in these cases.
4. The cascading movement is adopted in all schemes. The repair process in *SR* or *HR* is just one of the cases of in *AR* that are along a special path. *SR* and *HR*

have the same bound of converging speed as *AR* which has been presented in [14].

## 6 Related work

Recently, rather than preventing the occurrence of the holes, some extended virtual force methods [10, 19, 26] 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 move slowly towards the relatively sparse region according to each other's repulsive force and head towards a hole in the network. However, as indicated in [22], 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 [22], a more practical balancing method under the virtual grid model [24] 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 the early work [14], a localized control method based on the 1-hop neighborhood is proposed. Whenever a vacant area is detected, a snake-like cascading repair 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 repair processes, causing redundant processes and some unnecessary node movements. In the early work [15], a synchronization based on a single Hamilton cycle connected the entire network is provided. However, due to the length of such cycle, a long stretch path in the repair process is needed even when a spare node nearby the vacant grid is available. A more efficient localized repair solution is needed.

## 7 Conclusion

In this paper, we have presented a more cost-effective, snake-like repair process to cover the surveillance holes of WSNs where some 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 methods, only the 1-hop neighborhood is used, and the adjustment of nodes can be controlled within the local area more efficiently, under a synchronization model based on the hierarchical Hamilton cycles. The analytical and experimental results show the proposed method to be robust and scalable with a minimized cost. This verifies the correctness and the effectiveness of our new information model in coverage repair and its conflict control. In our future work, the energy consumption model will be considered in the node

adjustment so that the lifetime of the complete coverage can be extended.

## References

- [1] Definition of Hamilton path. Document available at [http://en.wikipedia.org/wiki/Hamiltonian\\_path](http://en.wikipedia.org/wiki/Hamiltonian_path).
- [2] N. Ahmed, S. Kanhere, and S. Jha. Probabilistic coverage in wireless sensor networks. *Proc. of IEEE Conference on Local Computer Networks (LCN'05)*, pages 672–681, 2005.
- [3] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: A survey. *Computer Networks*, (38):393–422, 2002.
- [4] S. Cang and J. Wu. Minimizing total communication distance of a broadcast on mesh and torus networks. *Proc. of the 1st Merged International Parallel Processing Symposium and Symposium of Parallel and Distributed Processing (IPPS/SPDP'98)*, pages 10–17, 1998.
- [5] Q. Cao, T. Yan, T. Abdelzaher, and J. Stankovic. Analysis of target detection performance for wireless sensor networks. *Proc. of International Conference on Distributed Computing in Sensor Networks*, pages 276–292, 2005.
- [6] S. Capkun, M. Hamdi, and J. Hubaux. Gps-free positioning in mobile ad hoc networks. *Proc. of the 34<sup>th</sup> Annual Hawaii International Conference on System Sciences*. 2001, pp. 3481–3490.
- [7] J. Carle and D. Simplot-Ryl. Energy-efficient area monitoring for sensor networks. *Computer*, 37(2):40–46, 2004.
- [8] M. Grossglauser and D. Tse. Mobility increases the capacity of ad-hoc wireless networks. *Proc. of the 20th Annual Joint Conference of the IEEE Computer and Communications Societies (IEEE INFOCOM'01)*, pages 1360–1369, 2001.
- [9] B. Hofmann-Wellenhof, H. Lichtenegger, and J. Collins. *Global Positioning System: Theory and Practice, Fourth Edition*. Springer-Verlag, 1997.
- [10] A. Howard, M. Mataric, and G. Sukhatme. An incremental self-deployment algorithm for mobile sensor networks. *Autonomous Robots, Special Issue on Intelligent Embedded Systems*, 13(2):113–126, 2002.
- [11] F. Hu and N. Sharma. Security considerations in ad hoc sensor networks. *Ad Hoc Networks*, 3(1):69–89, 2005.
- [12] A. Jadbabaie. On geographic routing without location information. *Proc. of the 43<sup>th</sup> IEEE conference on Decision and Control*. Dec. 2004, pp. 4764–4769.
- [13] Z. Jiang and J. Wu. Fault-tolerant broadcasting in 2-d wormhole-routed meshes. *Journal of Supercomputing*, 25(3):255–275, 2003.
- [14] Z. Jiang, J. Wu, A. Agah, and B. Lu. Topology control for secured coverage in wireless sensor networks. *Proc. of the 3rd IEEE International Workshop on Wireless and Sensor Networks Security (MASS WSNS'07)*, 2007.
- [15] Z. Jiang, J. Wu, and R. Kline. Mobility control for complete coverage in wireless sensor networks. *accept to appear in the Proc. of the International Workshop on Wireless Ad-Hoc and Sensor Networks (ICDCS WWASN'08)*, 2008.
- [16] B. Liu, P. Brass, O. Dousse, P. Nain, and D. Towsley. Mobility improves coverage of sensor networks. *Proc. of the 6th ACM International Symposium on Mobile Ad Hoc Networking and Computing (ACM MobiHoc'05)*, pages 300–308, 2005.
- [17] M. Sichitiu and V. Ramadurai. Localization of wireless sensor networks with a mobile beacon. *Proc. of the IEEE international conference on Mobile Ad Hoc and Sensor Systems*. Oct. 2004, pp. 174–183.
- [18] D. Tian and N. Georganas. A coverage-preserving node scheduling scheme for large wireless sensor networks. *Proc. of the 1st ACM Workshop on Wireless Sensor Networks and Applications*, 2002.
- [19] G. Wang, G. Cao, and T. Porta. Movement-assisted sensor deployment. *IEEE Transactions on Mobile Computing*, 5(6):640–652, 2006.
- [20] W. Wang, V. Srinivasan, and K. Chua. Using mobile relays to prolong the lifetime of wireless sensor networks. *Proc. of the 6th ACM International Symposium on Mobile Ad Hoc Networking and Computing (ACM MobiHoc'05)*, pages 270–283, 2005.
- [21] J. Wu. A fault-tolerant and deadlock-free routing protocol in 2-d meshes based on odd-even turn model. *IEEE Transactions on Computers*, 52(9):1154–1169, 2003.
- [22] J. Wu and S. Yang. SMART: A scan-based movement-assisted sensor deployment method in wireless sensor networks. *Proc. of the 24th Annual Joint Conference of the IEEE Computer and Communications Societies (IEEE INFOCOM'05)*, pages 2313–2324, 2005.
- [23] W. Xu, K. Ma, W. Trappe, and Y. Zhang. Jamming sensor networks: Attack and defense strategies. *IEEE Network*, 20(3):41–47, 2006.
- [24] Y. Xu, J. Heidemann, and D. Estrin. Geography-informed energy conservation for ad hoc routing. *Proc. of the 7th Annual International Conference on Mobile Computing and Networking (ACM/IEEE MobiCOM'01)*, pages 70–84, 2001.
- [25] W. Zhao, M. Ammar, and E. Zegura. Controlling the mobility of multiple data transport ferries in a delay-tolerant network. *Proc. of the 6th ACM International Symposium on Mobile Ad Hoc Networking and Computing (ACM MobiHoc'05)*, pages 1407–1418, 2005.
- [26] Y. Zou and K. Chakrabarty. Sensor deployment and target localization based on virtual forces. *Proc. of the 22nd Annual Joint Conference of the IEEE Computer and Communications Societies (IEEE INFOCOM'03)*, pages 1293–1303, 2003.
- [27] Y. Zou and K. Chakrabarty. A distributed coverage- and connectivity-centric technique for selecting active nodes in wireless sensor networks. *IEEE Transactions on Computers*, 54(8):978–991, 2005.