International Journal of Emerging Technology in Computer Science & Electronics (IJETCSE)
ISSN: 0976-1353 Volume 14 Issue 1 –APRIL 2015.

DETECTION AND HEALING OF HOLES IN WIRELESS SENSOR NETWORK

S.KANIMOZHI
PG Student
Salem College of Engineering and Technology
Salem, Tamilnadu, India-636111
dayakanimozhi@gmail.com

DAYALRAJ.R
Professor
Salem College of Engineering and Technology
Salem, Tamilnadu, India-636111
k2daya@gmail.com

Abstract— The basic services provided by a wireless sensor network (WSN) is the monitoring of a specified region of interest (RoI). Knowing the fact that emergence of holes in the RoI is unavoidable due to limited battery, outbreak of fires assuring that the RoI is completely and continuously covered is important. It seeks to address the problem of hole detection and healing in mobile WSNs. we identify four key elements that are critical for ensuring effective coverage in mobile WSNs: 1) defining the boundary of the RoI, 2) detecting holes and analysing their characteristics, 3) determining the best target locations to relocate mobile nodes to repair holes, 4) Moving mobile nodes to the target locations while minimizing the moving and messaging cost. We propose a lightweight and comprehensive solution HEAL which is a distributed and localized algorithm that operates in two distinct phases. The first phase identifies the boundary nodes and discovers holes using a lightweight localized protocol , The second treats the hole healing, with novel concept, hole healing area. A distributed virtual forces-based local healing approach where only the nodes located at an appropriate distance from the hole will be involved in the healing process. We propose a lightweight and comprehensive solution, called Modified Hole Detection, which addresses all of the aforementioned aspects. MHD is an evenly distributed and localized algorithm that operates to (i) identify the boundary nodes and discovers holes within the RoI and (ii) identify the holes with respect to the RoI boundaries. Finally a distributed virtual forces-based local healing approach where only the nodes located at an appropriate distance from the hole will be involved in the healing process.

Index Terms—Hole detection, deployment, hole healing, mobile WSN, obstacle Detection.

1. INTRODUCTION

A wireless sensor network (WSN) consists of small sensors with limited computational and communication power causing depletion of their limited energy resources. This can be regarded as a normal property of the network. Several problems can occur in networks that affect their respected functionalities leading to the creation of different kinds of holes, namely: coverage holes, routing holes, jamming holes, and worm holes. An area where a group of nodes stops working and results in lack of data sensing and communication property.

The most basic services provided by a WSN is the monitoring of a specified region of interest (RoI), where the main role is to sense the environment and transferring the information to the sink. However, the emergence of holes in the RoI is unavoidable due to various reasons such random deployment, limited power. So it act as a barrier for communication. Most of the existing idea uses global operations to calculate the size of a big hole and then relocate a group of mobile sensors to heal the hole.

HEAL is a distributed and localized algorithm which operates in two different phases. The first phase consists of three subtasks; hole identification, hole discovery (HD) and boundary detection. By proposing a distributed and localized hole detection algorithm (DHD) which has very low complexity that operates over the Gabriel graph (GG) of the network. DHD deals with holes of various forms and sizes despite the nodes distribution and density. The second phase treats the hole healing with novel concept, hole healing area (HHA). It consists of two sub-tasks; hole healing area determination and node relocation. A distributed virtual forces-based local healing approach where only the nodes located at an appropriate distance from the hole will be involved in the healing process, in which the forces will be effective. The DHD is proposed to identify the boundary nodes and discover holes.

A sensor network is a collection of many small devices, each with sensing, computation and communication capability. It has many potential applications, such as building surveillance, industrial asset management, and environmental monitoring. However, designing scalable, self-organizing and energy efficient sensor networks faces many challenges. Consequently, sensor networks have attracted researchers from a wide range of disciplines in recent years.

A commonly used assumption in studying sensor networks is that sensors are uniformly densely distributed in the plane. However, in a real system deployment, this assumption does not generally hold. Even if sensors are distributed uniformly at random, there are still regions with sensor density much lower than others. Other factors such as terrain variation and sensor power depletion can also contribute to non-uniform sensor distributions. In practice, sensor networks usually have holes, i.e. regions without enough working sensors. Figure 1 shows an example of a large number of dead sensor nodes, creating a big hole in the network.

The appearance of such holes changes global network topology, and imposes additional difficulties in organizing the network. Many greedy algorithms that assume dense sensor deployment break down most prominently, due to the local minimum phenomenon in geographical greedy forwarding.

Geographical greedy forwarding, a simple, efficient and scalable strategy, is a promising routing scheme for large-scale sensor networks when sensor locations are available. In geographical greedy forwarding, a source node knows the location of the destination node, either by acquiring it from a location service, or by computing it using a hash function in a data-centric storage scheme. A packet is forwarded to a 1-hop neighbour which is closer to the destination than the current node. This process is repeated until the packet reaches the destination, or the packet is stuck at a node whose 1-hop neighbors are all farther away from the destination. The node where a packet may get stuck is called a local minimum, or a stuck node in this paper. The existence of the local minimum phenomenon is due to the existence of communication voids in the sensor network, so that the black nodes failed, leaving a large hole in a network with sensors randomly placed.

There are also small holes due to low sensor density. Red nodes represent stuck nodes that will be explained later.A packet cannot progress towards its destination by greedily examining only its local neighborhood. There are other examples of greedy algorithms that fail when the assumption of a dense sensor distribution no longer holds. For example, in a dense sensor field, we can use a local and greedy path migration algorithm to adjust a communication path between two moving objects to a equivalent path. However, when the routing path is along the boundary of a hole, a local and greedy path migration algorithm faces the same difficulty as that in the greedy forwarding, i.e. there is no local improvement, although bypassing the hole will give a much shorter path.

2.PROBLEM STATEMENT

The basic possible sequence of future events can be described as follows: during normal operation of the network, a average loss of nodes occurs, due to an external attack. For example, the creation of one or several large holes within the network makes it in effective. Our problem is to create a mechanism for detecting and recovering holes by using only the nodes which can move easily. It should be noted that only the holes within the network are considered. The holes on the boundary that are the result of the initial deployment are not addressed.

Most of the hole healing algorithms require coordinate information about the sensor nodes in the RoI and use computation all geometry with tools, such as Voronoi diagrams to detect holes.

The application of WSNs ranges from environmental monitoring to surveillance to coordinated target detection. The efficiency of a sensor network depends on the coverage of the monitoring area. Although, in general, a sufficient number of sensors are used to ensure a certain degree of redundancy in coverage so that sensors can rotate between active and sleep modes, a good sensor deployment is still necessary to balance the workload of sensors. Mobile sensors can be exploited to provide a redistribution.

After an initial random deployment of sensors in the field, movement-assisted sensor deployment can be applied, which uses a potential-field-based approach to move existing sensors by treating sensors as virtual particles subject to virtual forces. Basically, movement-assisted sensor deployment deals with moving sensors from an initial unbalanced state to a balanced state. Therefore, various optimization problems can be defined to minimize different parameters, including total moving distance, total number of moves, communication/computation cost, and convergence rate.

3. EXISTING METHOD COMPARISON:

Mobile sensor nodes (for example, Robomote and iMouse) are equipped with mobile platforms and can move around after deployment. In a mobile WSN, one of the objectives is to maximize area coverage. Several movement strategies have been developed for improving network coverage. Algorithms that belong to the category include in coverage pattern-based movement , the target locations for mobile nodes are computed based on a predefined coverage pattern, while mobile nodes are out the resemblance as electromagnetic particles in virtual forces-based movement. In grid-quorum based movement, the mobility-assisted network redeployment problem is viewed as a load-balancing problem in traditional parallel processing systems. The RoI is partitioned into many small grid cells, and the number of nodes in each cell is considered.

The point coverage problem with novel evaluation metric, coverage radius that proposed three different deployment protocols that relocate mobile sensors once coverage holes are detected using Voronoi diagrams. Another proposed scheme called Co-Fi that relocates mobile nodes to replace low energy nodes.The three hole-movement strategies for moving an existing big hole in a way that either the total energy consumption is minimized or the power consumption of sensors is balanced.

In a hybrid network consisting of both stationary and mobile sensor nodes, one objective for using mobile sensornodes is to minimize coverage holes created by those stationary nodes.They describe a tracking mechanism and a robot repairingalgorithm. The tracking mechanism leaves the robot's footmark on sensors so that they can learn better routes for sending repairing requests to the robot while the repairing algorithm constructs an efficient path thatpasses through all failure regions.

Also by proposing family of localized robot- sensing holes. In this work we propose HEAL, a localized and distributed hole healing algorithm as opposed to the centralized approach employed. The Co-Fi protocol fails if the coverage loss is due to physical damage of the nodes. Our work explains the problem of coverage loss due to both energy depletion and physical damage of the nodes. In addition, we do not assume that the WSN boundary is known a priori as done. In HEAL the hole detection and healing process can be triggered by any node in the network, where the sink is used for starting the hole detection and healing process. HEAL is based on a localized healing process that:

1) Reduces the WSNs' resources consumption;
2) Increasesthe healing process;
3) Preserves as much as possible

the initial WSN topology. The number of nodes solicited in the healing process depends on the holes characteristics. Inaddition, only the nodes located at an appropriate distance from the hole will be involved in the healing process.

4.Hole Detection:

The stuck nodes where packetscan possibly get stuck in greedy multi-hop forwarding. A node is a stuck node if there exists a location outside transmission range developed a local rule, the TEN1T rule, for each node in the network to test if it is a stuck node. The TENT rule specifies that a node is not a stuck node when there is no angle spanned by a pair of its angularly adjacent neighbors greater than $2\pi/3$.

we have to calculate the existence of a hole, which is done by identifying stuck nodes. Each node in the networkexecutes the TENT rule to check if it is a stuck node as follow. First, it orders all its 1-hop neighbors counterclockwise.

All the nodes that are marked as stuck nodes by the TENT rule trigger the discovery of holes.It aims to find the boundary of the hole and the computation of the holes characteristics (center and radius). A stuck node creates a new hole-discovery packet, marked with its ID (the hole will take the same ID as that of the node), whose mission is to collect location information of hole boundary nodes, and forwards it to the next boundary node by right-hand rule over the GG of the network. Node inserts its location information into the received HD packet and forwards it to the node by right-hand rule over the GG and so on. This process is repeated until the HD packet has traveled around the hole and eventually been received by the initiator node. Node extracts from the received HD packet the locations of the boundary nodes.

It selects two nodes 0 and n so that the distance between them is thelongest between any two nodes in the set of boundary nodes.Then, it calculates the hole center, which is the mid point.It can be noted that each stuck node transfers a HD packet withoutany coordination between stuck nodes. Thus, therewill be discharge in the discovery process. This willcreate unnecessary traffic and more packet crash;the situation may become worse especially for largeholes. To avoid this problem, we can use a mechanism toprevent error in the discovery process. The basicidea is to remove noticed HD packets.The criterion for judging whether a HD packet isdismissed is as follows: at each node, if a HD packetarrives and discovers that the packet has a Hole-IDgreater than a Hole-ID carried by a packet alreadypassed, the packet will be considered redundant and itwill be deleted.

At the end of this step the node that has the smallestHole-ID removes the HD packet and names itself as 'HoleManager (HM): it will be responsible for the hole-healing process. In another hole manager selection strategy,one can pick the node with the largest residual.

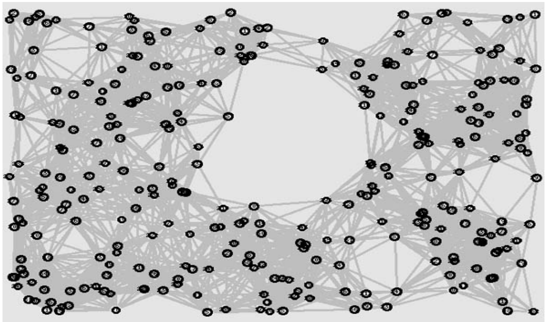

Figure 1: Principle of Healing Process

4.1 The Tent Rule and Bound Hole Identification:

Assume there are n wireless sensor nodes S in the plane, each with a communication range as a disk with radius 1.The communication graph is thus modeled by the Unit Disk Graph (UDG). We try to identify the holes in the regions without enough sensors, in the sensor network. We do this in two steps. First we want to identify that there exists a hole. This is done by identifying stuck nodes, where packets may possibly get stuck. With the stuck nodes identified which helps to identify the boundary of the holes. We discuss two types of stuck nodes: weakly stuck nodes and strongly stuck nodes, as well as their hole-surrounding routes respectively, in each of the following subsections.

Weakly Stuck Nodes and Holes

4.1.1 Weakly Stuck Nodes:

A node p 2 S is called a weakly stuck node if there exists a node q 2 S outside p's transmission range so that none of the 1-hop neighbors of p is closer to q than p itself. This de_nition of stuck node suits applications where routing destinations are always some nodes in the network. In such case, a packet can only get stuck at a stuck node. For weakly stuck nodes, we de_ne a hole to be a face with at least 4 vertices in the Delaunay triangulation with all the edges longer than 1 removed.

4.1.2 Finding the Holes

For a set of nodes S in the plane, the Voronoi diagram partitions the plane into convex regions, called Voronoi cells, such that all the points inside one cell are closest to only one node. The Delaunay triangulation isthe dual graph of the Voronoi diagram, by connecting the nodes whose corresponding cells are adjacent in
the Voronoi diagram. The Delaunay triangulation enjoys an empty-circleproperty: the circumcircle of a Delaunay triangle contains no nodes of S inside. It's known that geographical forwarding doesn't get stuck in a Delaunay triangulation when the destination is also a node in the point set. In particular, one ofthe Delaunay neighbors of u must be closer to the destination than u. However, a Delaunay triangulation may contain edges longer than 1 which are not available in a unit disk graph. If all the Delaunay edgesattached at a node u are no longer than 1, u is not a weakly stuck node.

5. PROPOSED WORK:

In this, we enhance HEAL to cope with obstacles by including DVFA algorithm. Obstacles such as walls or buildings might exist in the environment where sensors are deployed. DVFA is based on a periodic exchange of Hello messages between sensors. It assumes that any sensor node knows its position. These Hello messages enablewireless sensor nodes to detect the arrival and departure of sensors in the vicinity. Our idea is to keep the simplicity of the DVFA algorithm to ensure the full coverage and network connectivity in the presence of obstacles. To achieve this, we add a repulsive force exerted by the obstacle on any node whose new position goes through or penetrates the obstacle. Consequently, at the end of deployment, no mobile sensor node is within the obstacle.

More recently, some extended virtual force methods,
such as those, which are based on disk packing theory and the virtual force field concept from robotics, are proposed. These methods simulate the attractive and repulsive forces between particles. Sensors in a relatively dense region will explode slowly according to each other's repulsive force and head toward a sparse region. In this way, the whole monitoring area can achieve an even distribution of sensors.

However, these methods may have long deployment times since sensors move independently, and they may even fail if all the sensors can achieve force balance but not load balance.We assume that sensors are deployed randomly into the square monitoring area without consideration of any physical obstacles. Then, if we partition the monitoring area into many small regions and use the number of sensors in a region as its load, the sensor deployment problem can be viewed as a load balance problem in traditional parallel processing, where each region corresponds to a processor and the number of sensors in a region corresponds to the load.Unlike the optimal solution, SMART is a hybrid of the local and global approaches.

5.1 HOLE FORMATION

The existence of a hole, which is done by identifying stuck nodes which are border of hole. Stuck nodes & Border Nodes are similar to this Algorithm. That is why we need to carry out the network boundary-node identification to avoid that the hole discovery process be launched by those nodes.

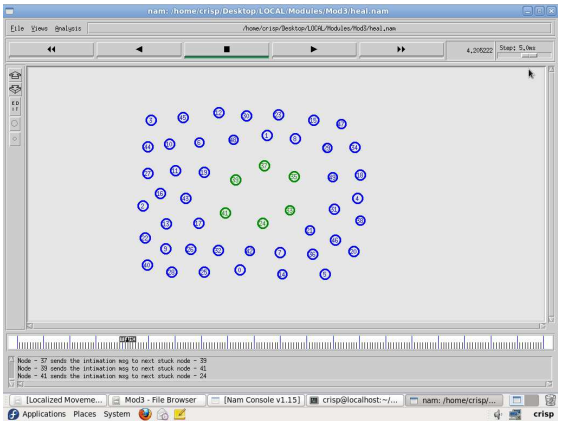

Figure 2: Hole Formation process Detect

5.2 OBSTACLE DETECTION AND HEALING PROCESS

HEAL to cope with obstacles by including DVFA algorithm. Obstacles such as walls or buildings might exist in the environment where sensors are deployed. DVFA is based on a periodic exchange of Hello messages between sensors. It assumes that any sensor node knows its position. These Hello messages enable wireless sensor nodes to detect the arrival and departure of sensors in the vicinity. Our idea is to keep the simplicity of the DVFA algorithm to ensure the full coverage and network connectivity in the presence of obstacles. To achieve this, we add a repulsive force exerted by the obstacle on any node whose new position goes through or penetrates the obstacle. Consequently, at the end of deployment, no mobile sensor node is within the obstacle.

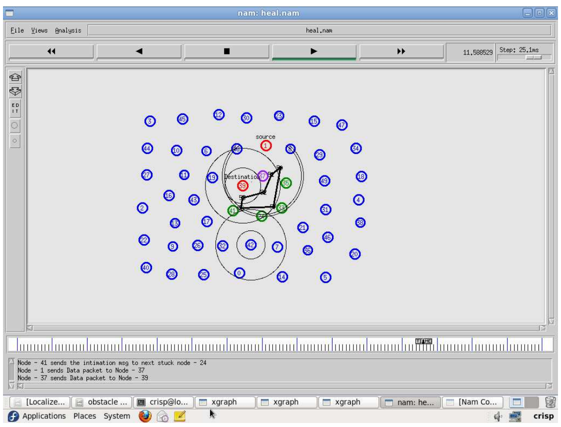

Figure 3: Obstacle Detection And Healing Process

5.3. IMPROVING NETWORK LIFETIME

To reduce energy consumption while maintaining good connectivity between sentinel nodes, we compose our solution on two main concepts, node adaptation and link adaptation. The first algorithm uses node adaptation technique and permits to distributively schedule nodes activities and select a minimum subset of active nodes (sentry) to monitor the interest region. And secondly, we introduce a link control algorithm to ensure better connectivity between sentinel nodes while avoiding outliers appearance. Without increasing control messages overhead, performances evaluations show that our solution is scalable with a steady energy consumption

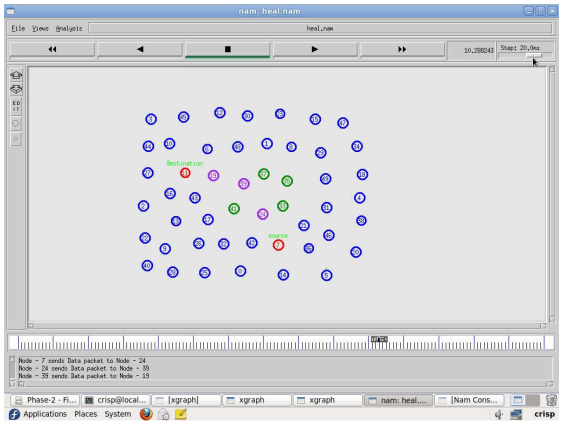

Figure 4: Improving Network Lifetime

5.4 COMPARISON BASED ON ALGORITHMS:

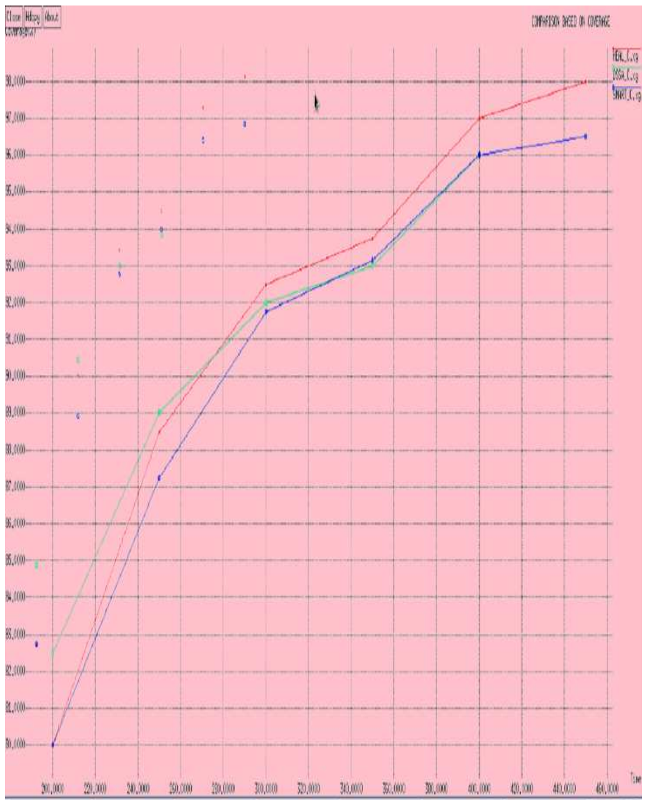

Figure 5. Comparison Based on Coverage

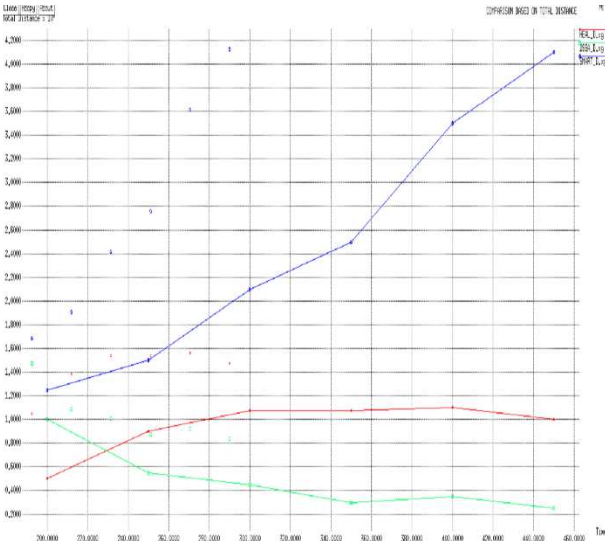

Figure 6. Comparison Based on Distance

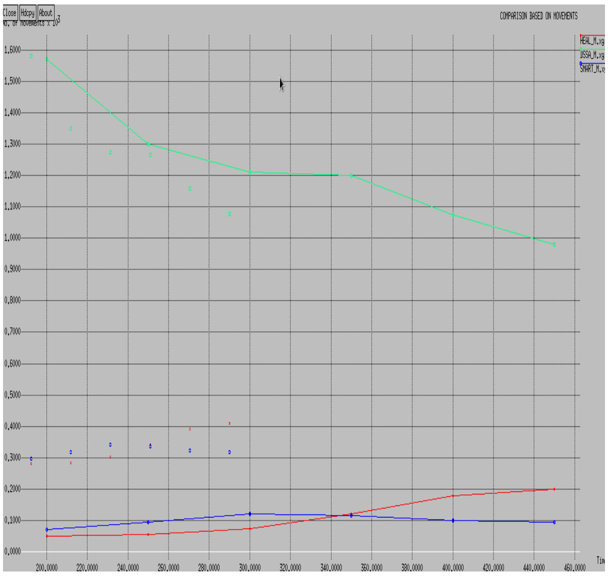

Figure 7. Comparison Based on Movements

Our boundary detection algorithm is motivated by an observation that holes in a sensor field create irregularities in hop count distances. Simply, in a shortest path tree rooted at one node, each hole is "hugged" by the paths in a shortest path tree. We identify the "cut", the set of nodes where shortest paths of distinct homotopy types1 terminate and touch each other, trapping the holes between them. The nodes in a cut can be easily identified, since they have the property that their common ancestor in the shortest path tree is fairly far away, at the other side of the hole. Thedetection of nodes in a cut can be performed independently and locally at each pair of adjacent nodes. When there are multiple holes in the network (indicated by multiple branches of the cut), we can explicitly remove all of the nodes on cut branches except one, thereby connecting multiple holes into one. Our algorithm then focuses

on finding the inner and outer boundaries of the network, which, with the cut nodes put back, will give the correct boundary cycles. In a network with only one hole (and one

cut branch), one can easily find a hole-enclosing cycle. Indeed, for a pair of nodes that are neighbors across a cut (a "cut pair"), the concatenation of the paths from each node in a cut pair to their common ancestor gives such a cycle. This "coarse" boundary cycle is then refined to boundtightly both the inner boundary and outer boundary.

CONCLUSION:

This paper has proposed and implemented a lightweight and comprehensive two-phase protocol, HEAL, for ensuring area coverage employing a mobile WSN. The protocol uses a distributed DHD to detect holes in the network. The computation complexity of DHD is Oðv^{2}Þ, where v is the average number of 1-hop neighbours. Compared to the existing schemes, DHD has a very low complexity and deals with holes of various forms and sizes despite the nodes dis-tribution and density. By exploiting the virtual forces concept, our approach relocates only the adequate nodes within the shortest time and at the lowest cost.

Through the performance evaluation, we validated HEAL, using different criteria and showed that it detects and heals the holes despite their number or size with less mobility in various situations. The evaluation results demonstrate that HEAL provides a cost-effective and an accurate solution for hole detection and healing in mobile WSNs.

In the future, we plan to investigate the interaction between HEAL and the network layer for on-demand hole detection and healing. We are currently working on open holes located at the network boundary. We also plan to investigate a special case of holes with obstacles.

REFERENCES:

1. N. Ahmed, S.S. Kanhere, and S. Jha, "The Holes Problem in Wire-less Sensor Networks: A Survey," SIGMOBILE Mobile Computing Comm. Rev., vol. 9, no. 2, pp. 4-18, 2005.
2. F. Stefan, "Topological Hole Detection in Wireless Sensor Net-works and its Applications," Proc. Joint Workshop on Foundations of Mobile Computing, p. 44-53, 2005.
3. Q. Fang, J. Gao, and L.J. Guibas, "Locating and Bypassing Holes in Sensor Networks," Mobile Networks and Applications, vol. 11, no. 2, 187-200, 2006.
4. S. Yangy, M. Liz, and J. Wu, "Scan-Based Movement-Assisted Sensor Deployment Methods in

Wireless Sensor Networks," IEEE Trans. Parallel and Distributed Systems, vol. 18, no. 8, pp. 1108-1121, Aug. 2007.

5. G. Wang, G. Cao, and T.F.L. La Porta, "Movement-Assisted Sen-sor Deployment," IEEE Trans. Mobile Computing, vol. 5, no. 6, pp 640-652, June 2006.

6. C.Y. Chang, C.Y. Lin, G.J. Yu, and C.H. Kuo, "An energy-Efficient Hole-Healing Mechanism for Wireless Sensor Networks with Obstacles," Wireless Comm. and Mobile Computing, vol. 13, no. 4, 377-392, 2013.