Island-Based Sensor Relocation in Wireless Sensor Network to Improve Connectivity

Sahla Masmoudi Mnif
National School of Computer Science
University of Manouba
Tunisia 2010
e-mail: sahla.masmoudi@ensi-uma.tn

Leila Azouz Saidane
National School of Computer Science
University of Manouba
Tunisia 2010
e-mail: leila.saidane@ensi.rnu.tn

***Abstract*—Wireless sensor networks applications require connectivity between deployed sensors over the region of interest (ROI). We note that the random deployment of sensors, leads to dividing the network in different partitions and makes the communication impossible between deployed nodes. In this paper, we propose to use a mobile robot which will travel through the supervised area and will relocate redundant sensor nodes in order to ensure connectivity. We assume in our work that the mobile robot is not aware of the network topology, so that the robot has to discover the network topology and to enhance the connectivity in the WSN. We propose for this purpose two Island-Based strategies. In the first strategy, the robot walk is made in a random manner; this strategy is called Island-Based Random Walk (IBRW). In the second strategy, called Island-Based Walk with Memorization (IBWM), the robot memorizes the collected information and tries to improve the connectivity in the WSN. Through simulation we evaluate and compare the performances of these strategies.**

Keywords- Sensor; Wireless Sensor network; Connectivity; Mobile Robot; Relocation.

I. INTRODUCTION

Ensuring connectivity in Wireless Sensor Networks (WSNs) is a challenging issue, especially in hazardous areas. Many applications of WSN require an important level of connectivity in the network to detect any abnormal event (e.g., fire, seism, intrusion detection, etc.) and forward an alert to the "sink" node in order to inform users. As examples we can mention, detection instruction application in military fields, the survey of frontiers zones and the control of mountains and forest occupied by terrorists. In fact, to survey this kind of environment, sensors are generally deployed, in a random manner (e.g., dropped from an aircraft). Sensor nodes are expected to detect any given events (intrusion, fire, etc.) and communicate together in order to survey a Region Of Interest (ROI). But nevertheless, the total connectivity between sensors is not guaranteed with random deployment of sensor nodes, which leads generally, in partition of the network.

Upon an initial deployment, the sensors should communicate and maintain this communication between nodes in order to stay reachable to each other.

A. Motivation

In recent years, the mobility has been introduced in WSNs to ensure and improve connectivity and coverage of the ROI. The sensor node using a mobile platform will have the possibility to move and to relocate its position. In our work, we are essentially interested by hazardous areas like inaccessible mountains, forests and deserts or harsh frontiers. Our goal is to survey these areas against any attacks (terrorist attacks). For this kind of areas, the deterministic deployment of sensors is not easy. For this purpose, in this paper we deal with a random deployment of sensors.

In general case, with random deployment the total connectivity is not guaranteed. Sensors need to be relocated to achieve total connectivity. Some existing sensor self-deployment algorithms [4][12][15] are adaptive to node failure and may actually be employed to solve the sensor relocation problem regarding sensing hole healing. A sensor relocation algorithm is proposed in [2]; this algorithm relocates redundant nodes in a cascading manner. However, its assumption, i.e., pre-knowledge about sensor field, makes it less practical in real-world scenario. Noting that, using mobile sensor nodes is expensive, we envision using, in our work, static sensor nodes and a mobile robot to relocate static nodes if necessary. The robot is assumed to be equipped with sensors and moves through the ROI according the given strategies. The robot has to relocate sensors to obtain a connected and covered network. The main task of the mobile robot is to discover disconnected nodes and to try to ensure total connectivity over the network. Different scenarios are envisaged where robot is in short of static sensors and should pick up redundant sensors and relocate them in the ROI. We propose to exploit the redundant nodes resulting in random deployment of sensors rather than adding some new sensors for economic purpose.

B. Problem Statement

We consider a given ROI equipped with a large number of wireless sensors which are deployed randomly in this area. By this deployment of nodes, the connectivity between nodes is not guaranteed. In our work, we propose to use a mobile robot to relocate redundant sensors in order to ensure communication between sensors.

 ISBN: 978-1-61208-482-4

Assumptions: we are assuming the following:

- The deployed nodes are wireless static sensors.
- The number of deployed sensor nodes is very important.
- Each sensor node is characterized by its unique Identity which is computed from its address.
- Each sensor node is able to compute its location by mean of localization technique like the Global Positioning System (GPS).
- Each sensor node has two independent components: sensing and communication units. Both parts are powered from the same limited source of power (battery).
- Each sensor node is able to compute its residual energy.
- The transmission (communication) range of each node is denoted r_c. We suppose that all nodes have the same communication range.
- The sensing range of each node is denoted r_s. We suppose that all nodes have the same sensing range.
- After a random deployment of sensor nodes in the area, they will be redundant nodes in some zones and they will be non- covered zones.
- We use a mobile robot which is a mobile sensor and have a very important storage and computation capacities.
- The robot has a communication range noted R_c and a sensing range R_s larger than the other sensor nodes.
- Initially, the mobile robot is equipped with sensor nodes (which we will call them “Reserve Nodes”) that it can use them to enhance connectivity in the network.

The mobile robot should run through the controlled area and decides the appropriate action to do: the robot can discover the network topology (discover the position of deployed sensors), can pick up redundant sensors or can continue its travel in the ROI.

Our paper is structured as follows: We start with a brief review of the existing solutions for the redeployment of sensors in WSN. Then we present our solution and we validate it by means of different experiments and simulations and finally, we close this paper by a conclusion.

II. RELATED WORK

In recent years, sensor relocation has been a challenging matter that was studied by many researchers. Several solutions have been proposed to solve the redeployment issue. One relevant solution was to provide motion capability to all sensors. This way, the sensors can move and relocate themselves in order to adjust the topology and achieve the connectivity and/or the coverage.

The sensors must synchronize their movement to enhance the network topology.

Among the proposed solutions, we mention particularly the cascade motion which is detailed in [2]: instead of moving directly to the target, the sensor nodes adopt a cascade movement which means that the nearest node to the target point will move there, and the location of nearest node is replaced by moving another sensor and so on.

Virtual Forces Aspect has been also proposed as a solution for sensor relocation. In this way, deployed sensors communicate together and compute their new locations in order to ensure connectivity and/or coverage. Then these sensors exercise a repulsive or an attractive force to move to their estimated locations. This strategy was studied and presented in [18].

The mobility of nodes is very efficient and improves the network topology, but it requires an important energy consumption which causes the node depletion and decreases the network lifetime.

Other solutions consist of the use of fixed sensor nodes and the network is assisted by some "actors" like mobile robots.

Some studies proposed to use the robot to carry data between disconnected sensors so that the robot collects the detected event from nodes and then delivers these information to the other nodes. This approach is presented by Zhao et al. [19]. In this way, the event is delayed and a latency time is introduced which can be considered as a shortcoming for critical applications.

Another set of related works include algorithms using DATA MULES [14][15], which are wireless devices integrated on mobile entities (e.g., animals, vehicles, etc.) A DATA MULE is a data collector; it picks up data from nodes and relays it to other nodes, so that, data would not be relayed on long routes and the network lifetime is increased

In other proposed solutions, the actors are mobile sensors that exploit the redundant nodes and relocate them to achieve better connectivity and/or coverage trying to preserve the network lifetime as long as possible. Most of the proposed solutions are grid-based ones like the solution proposed in [2]. For sensor relocation in mobile sensors networks we mention for example ZONER proposed in [20]. This solution presented a distributed zone-based sensor relocation protocol for mobile sensors on the basis of restricted flooding technique. When mobile sensors are cost effective and have critical energy constraints, we try in our work to propose a sensor relocation strategy for static sensor networks using a mobile robot.

III. PROPOSED SOLUTIONS

In this section we present our approach to relocate the sensors using a mobile robot. We start this section by presenting the network modeling and then we define the robot algorithms to relocate the sensors.

We assume that the region of interest is unreachable, making the deterministic deployment impossible. Hence, we consider an initial random deployment of sensor nodes and we scatter a large amount of sensors within the region of interest.

Each node in the network knows its own position by an attached GPS (Global Positioning System) or any other equipment of localization. Sensors have the same communication range r_c and the same sensing range r_s, we note that $r_c >= r_s$.

Using this kind of sensor deployment, connectivity in the resulted network is not guaranteed. Furthermore, the random deployment leads to the creation of disconnected islands.

1) ***Definitions:***

Island: An isolated set of connected sensor nodes.

MainIsland: The Island containing the "Sink" node is called "Main Island".

Redundant sensor: a sensor is said to be redundant if his perception zone is covered by the perception zones of other sensors.

In each island, the connectivity is ensured but the islands are not able to communicate between each other. Generally, each island contains redundant sensor nodes. Figure 1 shows an example of an Island-based network.

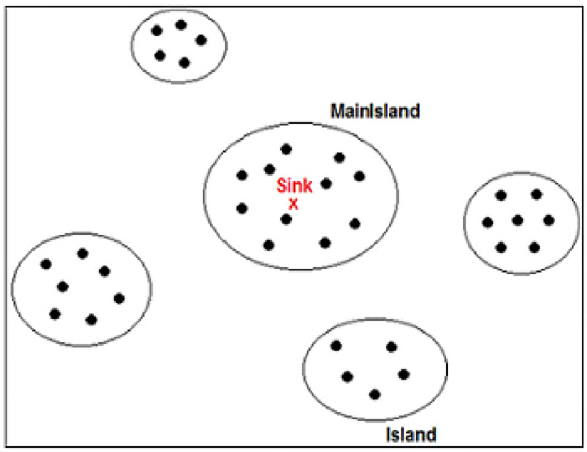

Figure 1. Example of an Island-Based WSN.

2) *Redundant Sensor Identification:* the mobile robot has to identify the redundant sensors in order to use them to connect islands. We use a hexagonal partition of region in order to identify and locate redundant sensors. The structure of hexagon cell is chosen in a manner that sensors belonging to two adjacent cells are able to communicate. A sensor is said to be redundant if its cells (perception zone) is covered by other cells. Tasks of redundant sensors can be made by the neighboring sensors and so that the redundat sensor can be in a passive mode in order to save energy.

3) *Island-Head Identification :* For each Island, a chief is elected, called Island-Head. This island-Head collects all the information about the island (positions of redundant nodes, positions of nodes in the islands, etc.). The Island-Head is elected as the node with the highest level of residual energy and having the largest set of neighbors. In order to select the Island-Head, an election factor noted *f* is defined by (1):

$$f = \frac{1}{2} * \frac{E_{res}}{E_{\max}} + \frac{1}{2} * \frac{Nb_n}{Nb_{nodes}} \qquad (1)$$

where E_{res} and E_{max}, represent respectively the residual energy and the maximum level of energy for a given node. Nb_n and Nb_{nodes} refer to the number of neighbors of a sensor node and the number of nodes in a given Island.

The node with the highest value of *f* is elected as an Island-Head. In case of multiple candidates, the node with higher Identity is elected. A backup Island-Head is chosen to replace Island-Head in case of its depletion.

When the Island-Head is elected, it collects the positions and all information concerning the redundant_nodes. After that, the Island-Head orders the redundant sensor nodes go to the passive mode (sleeping mode) to save energy of the whole network.

We assume that the robot knows the position of the sink node. Therefore, it is not aware of the network topology. Hence, the main role of the mobile robot is to discover the topology of the network and simultaneously, it tries to redeploy redundant sensors in order to enhance the network topology and to ensure connectivity between each Island of the network and the "MainIsland" to obtain a connected network.

We notice that, the mobile robot is considered as a sophisticated entity with an important computational capability and a large amount of energy. We suppose also that the robot can be recharged as needed. The robot has also sensing and communication capabilities, we note R_c the communication range of the robot and R_s its sensing range; $R_c >= R_s$. We assume also that the robot is equipped by a number of sensors Nb_{res} that can be used connect disconnected Islands.

Each couple of nodes (whether sensor node or robot) can communicate directly when they are within each other communication range.

In our solution, we will exploit sensor redundancy to enhance the connectivity over the network. We mention that the mobile robot can be in one of these states:

- Discovering topology: it has to discover the position of deployed sensors, Islands, etc.
- Collecting redundant sensors: when encountering redundant nodes, the robot can pick them up.
- Connecting Island: the robot places sensors in order to connect the Islands
- Free: the robot has no task to do.

We propose two strategies for sensor relocation: the first strategy is called Island-Based Random Walk (IBRW) in which the robot walk is made completely in a random manner and the second strategy is called Island-Based Walk with Memorization (IBWM) in which the robot walk is made based on the recently discovered information (about topology, position of redundant nodes, disconnected islands, etc.)

A. *Island-Based Random Walk*

Our first proposed solution is called Island-Based Random Walk (IBRW). In this solution the robot walks in the ROI in a random manner. Periodically, the robot stops (after a distance of 2* R_c) and sends a Hello-Robot Message. Each sensor receiving a "Hello-Robot", forwards this message to its "Island-Head" and the "Island-Head" replies with "Island-information" containing all the information concerning this island (position of nodes, positions of redundant sensors, sensors identities, number of redundant nodes, etc.).

- If the robot does not receive any reply (after a prefixed duration), it continues its walk in a random direction.
- If the robot receives an "Island-information", it computes the position of the nearest node of the "sink" and then it calculates the number of needed sensors to connect the island to the “MainIsland”.
  - If this requested number of sensors is available on the robot, it relocates them (the nodes will be relocated according to hexagonal pavement).
  - If this requested number of sensors is not available on the robot, it continues its walk in a random manner.

Figure 2 shows the way that two Islands should be connected using the hexagonal pavement. Green cells are used to connect an Island to the “MainIsland”.

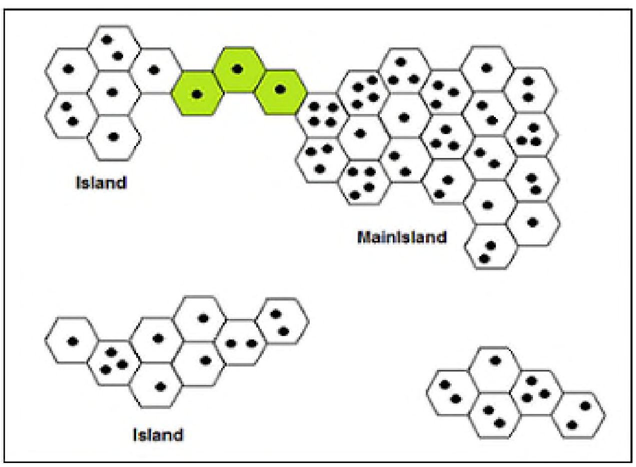

Figure 2. Connecting two Islands

B. *Island-Based Random Walk with Memorization*

In this strategy, the robot adopts the same functioning as the IBRW with some other amelioration. The robot is initially in a free state. The robot starts its travel in Discovering Topology state and it moves through a random direction. The robot, during this state, sends periodically a Hello-Robot Message. When the robot encounters an Island, it receives a reply from the Island-Head. This message contains all information about the considered Island. All received information is saved in the robot and the robot updates its information about the network topology.

- When a robot encounters an Island, it memorizes all the information concerning this Island mainly the locations of redundant nodes.
- When they are no carried sensors on the robot, the robot returns back to the nearest redundant nodes, picks them up and relocates them like in the IBRW algorithm. Then the robot continues its travel in a random direction.

IV. PERFORMANCE EVALUATION

Our proposed solution is implemented under NS2 simulator. Several simulations were established with different scenarios. For all simulations we use a large number of deployed sensors to ensure full connectivity and enhance coverage over the network.

The sensors are initially deployed randomly through a square ROI; we set r_c=25m, r_s=25m and R_c=45m. We set the dimension of the ROI to 500*500. The Initial load of the robot 5Reserve nodes) is fixed to 60 sensors. The number of deployed sensor nodes is set to 200 sensors in the first time. In a second step, we will vary the number of deployed sensors from 100 to 600 sensors.

The number of created islands over the network is an important factor which gives us an idea on the total connectivity in the network. Figure 3 shows that the number of created islands increases when the number of deployed sensors decreases. As a result, the connectivity between nodes increases when the number of deployed sensors increases.

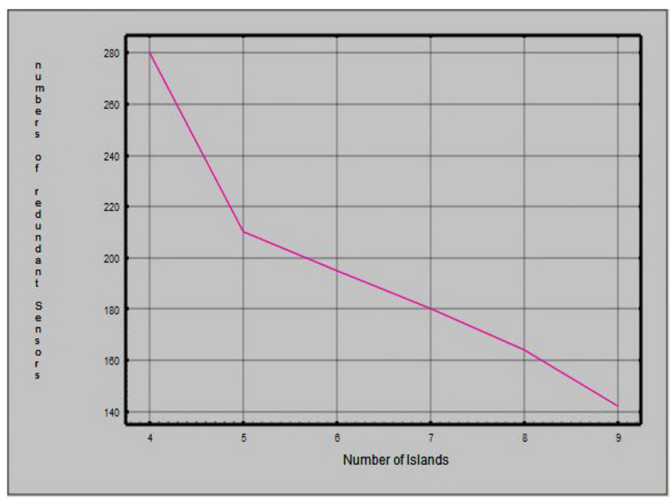

Figure 3. Number of formed Islands

In our work we try to attend a tradeoff between the number of deployed sensors and the connectivity rate. In other terms, we try to determine the optimal number of deployed sensors to achieve a desired connectivity level.

To evaluate our proposed solutions we fixed some metrics like connectivity rate, connectivity Time, the total

 ISBN: 978-1-61208-482-4

travelled distance by robot and the average consumed energy by static sensors.

A. *Connectivity Time*

Connectivity time (CT) is the time needed by the robot to ensure connectivity over the entire network. This metric should to be minimized.

Figure 4 shows that when the number of deployed sensors increases, the connectivity time decreases for the two proposed strategies IBRW and IBWM. This can be explained by the important number of redundant sensors when the number ofdeployed sensors number increases. In this case the number of Islands to connect to the “MainIsland” decreases.

We remark also that IBWM outperforms IBRW in terms of connectivity time which can be explained by the optimization of functioning of robot for IBWM compared to IBRW.

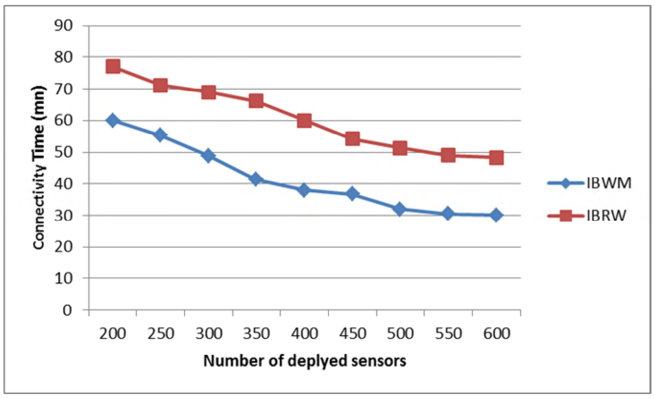

Figure 4. Connectivity Time

B. *Connectivity Rate*

The connectivity rate (CR) is the rate of connected sensors in the network; this metric can be given by (2)

$$\frac{\text{The number of connected sensors}}{\text{The number of deployed sensors}} \quad (2)$$

This metric should be maximized to enhance the performance of the tested algorithms. We modify the number of deployed sensors and we compute the CR to show the impact of the numbers of the deployed sensors on the connectivity rate.

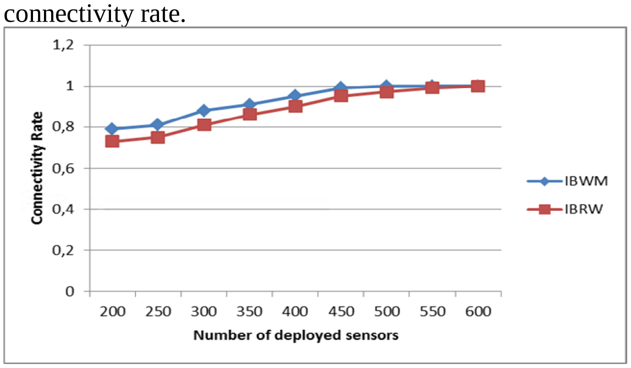

Figure 5. Connectivity Rate

Figure 5 shows that CR increases with the number of deployed sensors. Figure 5 illustrates also that IBWM algorithm outperforms IBRW in terms of connectivity rate. In fact, the walk of robot is more optimized in IBWM strategy making the connectivity process easier.

C. *Total Travelled Distance*

We compute for each proposed algorithm the total travelled distance by the robot. Figure 6 shows that the travelled distance decreases when the number of deployed sensors increases. In fact, in this case, the robot had to connect more Islands to the “MainIsland” and is obliged to travel more long distances. Figure 6 shows also that IBWM outperforms the IBRW in terms of the total travelled distance. In fact, compared to IBRW, the IBWM algorithm exploits the nearest discovered redundant sensors.

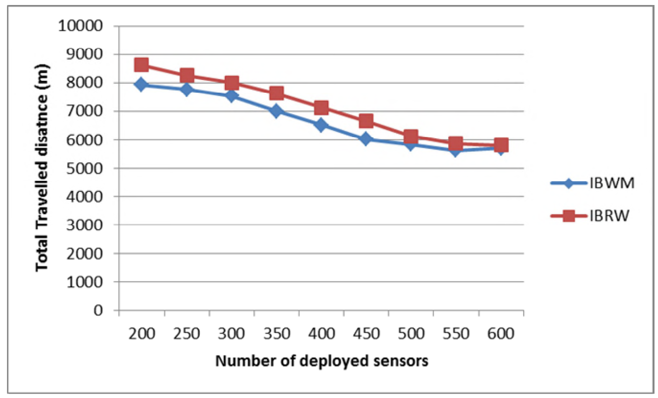

Figure 6. Total Travelled Distance

D. *Energy Consumption*

The high energy consumption driven by mobile sensors is an important criterion which justifies the use of static sensors and a mobile robot to redeploy them. We compute for our proposed solutions the average consumed energy by all static sensors. Figure 7 represents the mean consumed energy according to the number of deployed sensors.

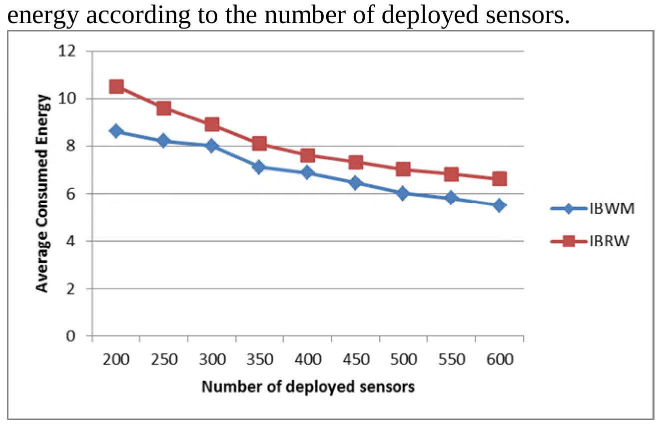

Figure 7. Average Consumed energy by static sensors

We note that the energy consumption decreases with the number of deployed sensors. In fact, when the number of deployed sensors increases, the number of resulted islands decreases and so the number of redundant sensor nodes

 ISBN: 978-1-61208-482-4

increases within each island; the redundant nodes will be in sleeping mode. We note also that IBWM outperforms IBRW strategy in terms of consumed energy. In fact, in IBRW strategy, the robot does not memorize any information about the topology. Each time the robot is obliged to communicate with encountered sensors.

E. An illustrative Example:

Our proposed solutions can be used to insure connectivity in WSN applications. We mention mainly agriculture precision, where a set of sensor nodes is randomly deployed on a zone and a mobile robot can be used to ensure connectivity in this network. From Figure 4, we can know the minimum number of needed deployed sensors to ensure connectivity in a given time. We can also determine the minimum number of needed sensors to have connectivity lower than a given threshold. For example in our example the CR is greater than 0.96 when the number of islands exceeds 7.

Detection Intrusion in hazardous areas is an example which can use our work. A robot can be used to redeploy sensors in order to achieve total connectivity. The optimal number of deployed sensors to achieve a given level of connectivity in a given time can be determined from curves and figures resulting from our simulations.

V. CONCLUSION AND FUTURE WORK

In this paper we proposed a robot-based sensor relocation to ensure connectivity in the Wireless sensors networks. We proposed to model our network by a set of disconnected islands that are formed due to a random deployment of nodes. We proposed also to use a mobile robot to relocate redundant nodes in order to connect the islands of the networks.

We defined two strategies; in the first one IBRW, the mobile robot makes a random travel. In the second one the robot memorizes the locations of encountered redundant nodes and uses the nearest ones when needed. Through several simulations we validated our work.

We show that our work can be used to determine a tradeoff between the required connectivity rate or time and the number of deployed sensors.

As a further work we propose to enhance these solutions by the use of a large number of robots and we propose also to compare our proposed solutions to other relevant proposed solutions in literature.

REFERENCES

[1] R. Kershner, The number of circles covering a set, American Journal of Mathematics, 1939,p665-671,

[2] T.La Porta, G.Wang, G.Cao and W.Wang, Sensor Relocation In Mobile Sensor Networks, Infocom 2005.

[3] F.Wang, M.Thai and D.Du, on the construction of 2-connected virtual backbone in wireless networks, IEEE Transactions on Wireless Communications 2009.

[4] A.Abbasi, M.Younis and K.Akkaya, Movement-assisted connectivity restoration in wireless sensor and actuator networks. IEEE Transactions on Parallel Distributed Systems 2009.

[5] G.Wang, G.Cao and T.Laporta, A Bidding Protocol for Deploying Mobile Sensors, The 11th IEEE International Conference on Network Protocols (ICNP) 2003.

[6] G.Wang, G.Cao and T.Laporta, Movement-assisted sensor deployment, Infocom 2004.

[7] G. Fletcher, X.. Li, A. Nayak and I.Stojmenovic, Carrier-based sensor deployment by a robot team, IEEE SECON, 2010.

[8] Y. Mei, C. Xian, S .Das, Y. C.Hu and Y. H.Lu, sensor replacement using mobile robots, computer communication 30(13) 2007.

[9] D.Xuan, Z.Yun, X.Bai, S.Kumar and Ten H. Lai, Deploying Wireless sensors to achieve both coverage and connectivity, Mobile Ad Hoc Networking and Computing,2006.

[10] X.Li and N.Santoro, ZONER: a zone-based sensor relocation protcol for mobile sensor networks, IEEE WLN, 2006.interface," IEEE Transl. J. Magn. Japan, vol. 2, August 1987, pp. 740–741, [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].

[11] X.Li, N.Santoro and I.Stojmenovic, Mesh-based sensor relocation for coverage maintenance in mobile sensor networks, UIC 2007.

[12] A.Howard, M.J.Mataric and G.S.Sukhatme, Mobile Sensor Networks Deployment Using Potential Fields: A Distributed, Scalable Solution to the Area Coverage Problem, the 6th International Symposium on Distributed Autonomous Robotics Systems, 2002.

[13] E.Egea-López, J.Vales-Alonso, A.S.Martínez-Sala, P. Pavon-Mariño and J.García-Haro, Simulation Tools for Wireless Sensor Networks, Summer Simulation Multiconference - SPECTS,2005.

[14] I.F.Akyildiz, W.Su, Y.Sankarasubramaniam and E.Cayirci, A Survey On Sensor Networks, IEEE Communications Magazine,2002.

[15] T.Watteyne, Using Existing Network Simulators for Power-Aware Self-Organizing Wireless Sensor Network Protocols, INRIA,2006.

[16] G. Chalhoub, Réseaux de capteurs sans fil, Clermont Université,2009.

[17] A.Gallais, J.Carle and D.Simplot-Ryl , La k-couverture de surface dans les réseaux de capteurs, AlgoTel, 2007.

[18] X.Wang, Sh.Wang andD.Bi, Virtual force-directed particle swarm optimization for dynamic deployment in wireless sensor networks, ICIC, 2007.

[19] W. Zhao, M.Ammar, and E. Zegura, A message Ferrying Approach for data deliveryin sparse mobile adhoc networks, Mobihoc 2004.

[20] A.M. Khedr and H. Ramadan, Effective Sensor Relocation Technique in Mobile Sensor Networks, IJCNC Vol3, No.1, january 2011.

ISBN: 978-1-61208-482-4