A novel sensor node relocation approach to maintain connectivity with a center of interest

Ines El Korbi, Chayma Zidi, Sherali Zeadally, Leila Azouz Saidane

A Novel Sensor Node Relocation Approach to Maintain Connectivity with a Center of Interest

Inès El Korbi
National School of Computer Science, University of Manouba, Tunisia,
ines.korbi@ensi.rnu.tn

Chayma Zidi
National School of Computer Science, University of Manouba, Tunisia,
chayma.zidi@gmail.com

Sherali Zeadally
Department of Computer Science and Information Technology, University of the District of Columbia, Washington DC, USA
szeadally@udc.edu

Leila Azouz Saidane
National School of Computer Science, University of Manouba, Tunisia,
leila.saidane@ensi.rnu.tn

ABSTRACT
We propose a new technique to relocate a set of redundant sensor nodes from their initial deployment area called ROI (region of interest) to a new location outside the ROI called COI (center of interest) where a new event happened (e.g., to measure the soil contamination level in the case of a chemical spill disaster). Relocating mobile sensor nodes from their initial deployment area ROI requires covering that area at a given center of interest while maintaining connectivity with the initial region of interest. Our proposed technique to relocate redundant mobile sensors nodes towards a new target position is a four-phase solution that includes the clustering phase and the cell head election, the evaluation of the number of nodes to be relocated, the propagation of the COI position, and finally the election of the redundant nodes and their relocation. A performance evaluation of our novel relocation approach shows that the consumed energy and the relocation time increase as the distance separating the initial ROI from the COI increases regardless of the size or the density of the initial deployment area.

Categories and Subject Descriptors
C.2.1 [**Network Architecture and Design**]: Wireless communication

General Terms
Algorithms, Performance

Keywords
Energy, networking, performance, sensors, wireless.

1. INTRODUCTION
A wireless sensor network consists of a large number of small, low-cost, low-power and mobile sensor nodes that are able to observe their environment. Due to these attractive characteristics, sensor networks have been deployed in many applications including military applications, smart homes, surveillance, and remote environment control. These sensor nodes are massively deployed in a region of interest (ROI) to collect data and process it [1]. Many applications such as monitoring remote harsh fields or toxic regions where human intervention is unsafe or burdensome require the mobility and self-management of sensor nodes. The motion capability is also required if an event outside the initial deployment area happened and have to be covered. The sensor nodes can then move to ensure the coverage of that new center of interest (COI). For example, if a chemical spill disaster occurs in a chemical plant, the noxious substance may propagate outside of the initial area of the chemical plant already controlled by a wireless sensor network. Hence, if we need to measure the soil contamination degree at a given point outside that initial zone, we would have relocate some sensor nodes to that center of interest to report such an information to the sink node.

The method used to relocate sensor nodes is a critical issue because it affects both the cost and the detection capability of a wireless sensor network. In this paper we investigate the problem of sensor relocation to respond to an event that needs a sensor node to be moved to its location to capture information and send this information back to the sink node. Additional nodes have to be relocated to ensure communication with the initial ROI in order to report information to the sink node. The sensor relocation problem differs slightly from the traditional sensor node deployment problem. In the case of the traditional sensor deployment problem, the initially deployed sensor nodes are moved within the coverage area either to avoid coverage holes, or to minimize the whole network energy consumption or to ensure both coverage and connectivity within the network [14], [19], [4]. In contrast, with the sensor relocation problem, we assume that both the coverage and the connectivity are already satisfied in the sensor network and the challenge is to satisfy the requirements associated with node relocation. The first requirement is the choice of the sensor nodes to be moved because the prospective relocated nodes may be used by some application and their movements may damage the sensing or communication tasks they are performing. The second major requirement aims to minimize the network energy consumption caused by movement because the latter is more expensive in terms of energy than communication and computation.

This rest of the paper is organized as follows. In section 2, we review related works on sensor node relocation. In section 3, we present the fundamental criteria that have to be satisfied with our approach especially the coverage-connectivity criterion [11] and we explain the different assumptions used by our proposed solution. Section 4 presents our proposed relocation approach to cover an event happening outside the ROI. The proposed solution needs to ensure that connectivity is still maintained between the

MSWiM'12, October 21–25, 2012, Paphos, Cyprus.

ROI and the COI where the new event occurred. In section 5, we evaluate the performance of our proposed approach using performance metrics such as the number of exchanged messages, energy consumption, and the relocation time. Finally, we make some concluding remarks in section 6.

2. RELATED WORKS

Several works have considered the relocation of wireless sensor nodes within a wireless sensor network. But most of them have considered sensor nodes relocation either to ensure fault tolerance, to remove coverage holes or for self organization purposes. Benhamed et al. [5] proposed a new technique of self organization in wireless sensor networks to ensure network robustness. This method aims to find the network articulation points that exist in order to ensure coverage and connectivity in the network. Their proposed approach eliminates critical positions by placing additional nodes around the articulation points to replace them if they fail. In [8], Chiasserini et al. considered the relocation of mobile sensor nodes based on virtual forces. Indeed, in the sensor field, each sensor behaves as a "source of force" for all other sensors. If two sensors are placed too close to each other, they exert negative forces on each other. On the other hand, if a pair of sensors is too far apart from each, they exert positive forces on each other. This ensures a global uniform sensor placement in the network [19]. When an event takes place, a potential force is generated by such an event (for example, upon the occurrence of a fire outbreak). Then the sensor nodes detecting such an event use virtual forces to move towards the event while maintaining a security distance that precludes them from destruction.

A grid-quorum based solution was presented in [18] to ensure fault tolerance in wireless sensor networks. Indeed, Wang et al. in [18] split the sensing field into m × m square cells in order to have a grid based topology. Each cell then elects a cell head node and the other nodes within that cell are considered as redundant. Taking advantage of the square sized grid topology, Wang et al. proposed a fault tolerance solution based on relocation of redundant nodes where supplier cells are along columns and demanding cells are along rows. Each cell head broadcasts a message with position information of the redundant nodes to all cell heads in the same column. After this step, all the cells' heads are aware of the redundant nodes within their column. If a cell head needs a redundant node to replace a faulty node within its cell, it first checks if there is an available redundant node along its column. If nothing is available, it then propagates its query to its neighboring cell head on the same row. The receiver performs the same redundant node research procedure as the sender. If it cannot satisfy the sender's query, it propagates the message to its neighbor on the same row until the message reaches a cell head node that can satisfy the initial sender query. Hence, the redundant node found can be relocated to replace the failed node.

In the same way Li et al. [12] proposed an information-mesh structure to ensure fault tolerance in the sensor network. They proposed a Mesh Based Sensor Relocation Protocol (MSRP) to relocate a redundant sensor node to a failed location. Indeed, nodes are initially divided into active nodes (A-nodes) and redundant nodes (R-nodes). Each R-node is associated with a proxy which is the nearest A-node in its vicinity. When the neighboring nodes detect an A-node failure, they cooperate to find the replacement R-node using the proxies' redundant nodes information. The target R-node is then relocated to its final position.

In [10], the authors proposed an optimized algorithm to detect node redundancy within the network by finding if a sensor node is completely covered by its neighborhood using a recursive geometric-based procedure. They proposed a cluster-based redundant node relocation algorithm that reduces the overall network energy consumption by choosing the nearest failed redundant sensor node as a replacement.

All the works presented above have considered sensor node relocation techniques that ensure fault tolerance or for self organization, network robustness and event coverage purposes by making use of an initial deployment area. None of the above proposed solutions consider sensor nodes relocation outside of the initial deployment area. Indeed, when relocating redundant nodes from their initial deployment area, we still need to maintain the continued operation of the initial network while ensuring the coverage and the communication with the initial ROI for the relocated nodes. Hence, we will propose a novel node relocation approach in the next section.

3. ASSUMPTIONS AND SOLUTION REQUIREMENTS

In this paper, we consider the issue of relocating a set of mobile sensor nodes from their initial deployment area (ROI) to cover a target event happening outside of the initial covered zone (COI). For instance our relocation approach aims to maintain both coverage and connectivity within the ROI when nodes relocate from their original positions towards the COI. Indeed, maintaining coverage within the ROI means that each point within that zone is covered at least by one sensor node. Hence only "redundant" nodes can be relocated from the ROI. Moreover, relocating "redundant" nodes from their initial zone to the COI must ensure nodes communications (connectivity) within the ROI and between the ROI and the COI. Coverage and connectivity are closely related because they are strongly related to sensing and communication ranges. According to the literature, as in [9], the relation $R_c \le 2R_s$ (where R_c is the communication range and R_s is the sensing range) is necessary and sufficient for coverage implies connectivity in a sensor network. Therefore, the sensors node relocation mechanism we propose in this work considers the coverage-connectivity criterion first and foremost requirement that must be satisfied.

In the following, we present a few assumptions related to the initial network deployment, sensor nodes characteristics and sensor nodes movement have to be considered.

3.1 Initial sensors deployment

Generally wireless sensor nodes are deployed randomly to cover events in a certain region of interest (ROI). Initial deployment may not ensure the total coverage of the ROI. Therefore, different mechanisms can be used to ensure covering the whole region and avoiding coverage holes [11]. A full discussion of the operations of the mechanisms that ensure the coverage-connectivity criterion is outside of the scope of this work. In this paper, we make the following assumptions:

- **Assumption 1:** The region of interest (ROI) is a square area of side a.
- **Assumption 2:** The wireless sensor network initially deployed satisfies the coverage-connectivity criterion. Indeed, this criterion can be satisfied using self deployment algorithms if the sensor nodes are initially deployed randomly [14], [16].
- **Assumption 3:** The number of sensor nodes within the network is large enough to cover both the ROI and to ensure connectivity with the new COI (center of interest).

The above assumptions can be justified by the fact that the event that has to be covered is in the near vicinity of the initial deployment zone. For example, in the case of a chemical spill disaster, we would like to detect the soil contamination in the surrounding zone (some hundred meters) away from the initial ROI. This could be done by relocating some redundant nodes from the initial zone.

Based on the assumptions above several grid-based techniques using various geometric shapes [1], [18], [12] have been proposed to ensure covering the initial ROI. The grid covering techniques are simple and enable easy coverage measurements. Grid-based techniques employ triangular shaped, rectangular shaped or hexagonal shaped grids [1]. Indeed, the hexagonal grids are optimal in terms of coverage calculations since they increase the maximum area covered by each sensor and therefore increase the number of calculated redundant nodes within each cell. However, in this work we choose a square-based grid because it is much more appropriate to the shape of the ROI given by assumption 1 and such a coverage technique will limit the number of exchanged messages which can be transmitted either on rows or on columns [18].

Moreover, we also consider a side size of each cell in the grid in such a way so as to satisfy both the coverage and the connectivity (as given by assumption 2). We then consider a side of each cell b equal to $\frac{2R_s}{\sqrt{5}}$ (with R_s is the sensing range) which is deduced by considering the fact that the furthest nodes in two adjacent cells can communicate together directly [15]. Therefore, the distance that separates them is at most equal to the communication range R_c (with $R_c = 2R_s$). Moreover, in order for assumption 3 to hold, we consider that initially more than one sensor is deployed within each cell.

3.2 Sensor nodes and sink node characteristics

In a wireless sensor network, sensor nodes detect events in a covered area and transmit them directly or using multi-hop routing to a particular node called a sink which transfers the collected information to the application. Therefore, we assume that:

- **Assumption 4:** The nodes are identical in terms of communication range, sensing range, and initial energy resources E_0. Each node is identified by a unique identifier. This is simplifying assumption and is frequently used in the literature [2], [10].
- **Assumption 5:** Each node *i* can detect its position at a given time using a localization system from [17], [3] and knows the limit of the ROI. It can therefore determine the row and column numbers (r_i, c_i) to which it belongs.
- **Assumption 6:** The sink node *p* is placed at one of the corners of the ROI area at coordinates (x_p, y_p) and is alerted by the application of the COI coordinates (x_{COI}, y_{COI}) when a new event occurred. It will notify the sensor nodes of the position of such an event. The sink node belonging to the cell is identified by row and column numbers (1,1). Generally the sink node is placed at the front of the sensor network to keep connecting with the wired structure.

4. PROPOSED SENSOR NODE RELOCATION APPROACH

In this section, we present our novel approach that relocates wireless sensor nodes from their initial ROI to cover a particular event happening at a COI outside the initial coverage zone and to maintain communication with the initial region of interest in such a way so as to report the captured information to the sink node. Our proposed approach uses four steps: clustering, evaluation of the number of nodes to be relocated, propagation of the COI position and finally redundant nodes election and relocation take place.

4.1 Clustering

When the nodes are initially deployed, all the nodes within the sensor network are in active mode. The clustering phase determines the nodes among the active nodes that will remain active to detect the events within the ROI. All the other nodes within the ROI switch to the passive state and are considered as redundant. Therefore, clustering elects a particular node (called cell head) within each cell of the square grid to cover the area of such a cell. All the other nodes belonging to the same cell switch to the passive state. The cell head also needs to maintain the list of its neighboring nodes belonging to the same cell and their positions. Clustering is one of the classical algorithms in hierarchical-based solutions [13]. The election of the cell head in each cell of the square grid can be made on the basis of multiple criteria (node identifier, energy, communication-coverage range). In our approach we elect the node having the highest remaining energy within a cell as a cell head. Moreover, if two nodes within the same cell have the same energy, the node with the smallest identifier is elected cell head.

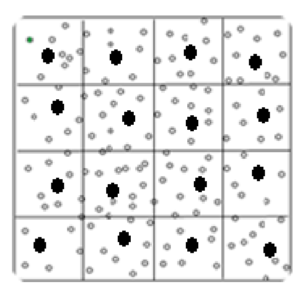

Figure 1: The clustering phase

Therefore, initially each node initializes itself as a *Cell_head* and initializes E_{max} with the maximum value of its remaining energy E_0. Then, each node periodically sends a *HELLO* packet that contains information such as its position, its remaining energy and its identifier. When each node *i* receives a *HELLO* packet from a sender *s*, it checks if the sender *s* is within the same cell (according to assumption 5, each node is aware of the cell row and column to which it belongs). If yes, it compares E_s (the energy received in the *HELLO* packet) with its local value E_{max}. If $(E_s > E_{max})$ or if $(E_s = E_{max})$ and $(s < i)$, then the sender *s* is saved within the node *i* as the *Cell_head*.

After a time α (corresponding to the time required to exchange *HELLO* packets), each node *i*, compares its identifier with the *Cell_head* value: if the *Cell_head* value is equal to *i*, then the node *i* considers that it is a cell head and broadcast a packet *CELL_HEAD_ADV* including its identifier and its position. Each node receiving the *CELL_HEAD_ADV* packet and is within the same cell of the sender acknowledges that packet with a *CELL_HEAD_ACK* including its position and its identifier. Upon reception of *CELL_HEAD_ACK* packets, the cell head node *i* updates its neighboring list with the characteristics of each neighboring node.

As the *HELLO* packets are transmitted periodically by all the nodes, a new cell head election procedure is triggered upon reception of a packet with a remaining energy greater than the one of the cell head node's remaining energy. Then, *CELL_HEAD_ADV* and *CELL_HEAD_ACK* packets are exchanged to update the new cell head's identifier.

4.2 Relocated node number evaluation

When a new event happens outside of the ROI, the application notifies the sink node of the position of such an event. The sink node will then trigger the algorithm to evaluate which nodes have to be relocated towards the COI while minimizing the overall network energy consumption. As mentioned before, since the mobility function is more expensive in terms of energy than communication and computation, the sink has to first determine the nearest cell: the cell whose position is closest to the center of Interest. Then, the sink node determines which cells surrounding the nearest cell may contain other redundant nodes to ensure connectivity between the ROI and the COI. We first proceed to the nearest cell computation.

4.2.1 The COI nearest cell computation

Determining the cell which has the closest position to the COI depends on both the COI and the sink node coordinates (x_p, y_p) and (x_{COI}, y_{COI}). Indeed, according to assumption 6, the sink node can be placed at one of the corners of the square shaped ROI. Different cases have to be discussed to determine r_N and c_N, the row and the column numbers of the COI nearest cell:

- **First case :** The sink node is located at the lower left corner of the grid, then:

$$r_N = \min\left(\max\left(\left\lceil \frac{x_{COI} - x_p}{b} \right\rceil, 1\right), \frac{a}{b}\right) \quad (1)$$

$$c_N = \min\left(\max\left(\left\lceil \frac{y_{COI} - y_p}{b} \right\rceil, 1\right), \frac{a}{b}\right) \quad (2)$$

Where a/b is the number of cells on one side of the ROI area.

- **Second case :** The sink node is located at the upper left corner of the grid, then r_N is given by equation (1) and:

$$c_N = \min\left(\max\left(\left\lceil \frac{y_p - y_{COI}}{b} \right\rceil, 1\right), \frac{a}{b}\right) \quad (3)$$

- **Third case :** The sink node is located at the lower right corner of the grid, then c_N is given by equation (2) and:

$$r_N = \min\left(\max\left(\left\lceil \frac{x_p - x_{COI}}{b} \right\rceil, 1\right), \frac{a}{b}\right) \quad (4)$$

- **Fourth case :** The sink node is located at the upper right corner of the grid, then:

$$r_N = \min\left(\max\left(\left\lceil \frac{x_p - x_{COI}}{b} \right\rceil, 1\right), \frac{a}{b}\right) \quad (5)$$

$$c_N = \min\left(\max\left(\left\lceil \frac{y_p - y_{COI}}{b} \right\rceil, 1\right), \frac{a}{b}\right) \quad (6)$$

Therefore, calculating the nearest cell row and column numbers helps in determining the redundant node within this cell that will be used to cover the event happening in the center of interest. In the next paragraph, we determine the other redundant nodes to be relocated to ensure connectivity between the initial ROI and the COI.

4.2.2 The Virtual sub grid computation

Based on the nearest cell research algorithm presented above and the COI coordinates, the sink node can determine the number of redundant nodes to be relocated to the COI to ensure connectivity with the initial ROI area.

The choice of the number of redundant nodes to be relocated is based on the following conditions:

1) Limiting the energy consumption due to mobility by moving the nodes whose positions are nearer to the COI.
2) Ensuring that the connectivity is maintained when the redundant nodes are relocated.

To satisfy the above conditions, we can choose to select the redundant nodes from a virtual sub-grid belonging to the initial ROI which, when being moved towards the COI, will ensure the coverage and the connectivity with the initial ROI (as shown is figure 2). Indeed, the virtual sub grid is chosen is such a way that:

1) It propagates the square grid topology outside the ROI. This is beneficial because it ensures connectivity with the ROI since the distance that separates two nodes in two adjacent cells is always lower or equal to the communication range. Moreover, this can useful if another event happens outside the ROI and is in the same region of the COI.
2) The upper and the lower boundary of the virtual sub grid are computed based on the distance between the COI and the nearest cell.

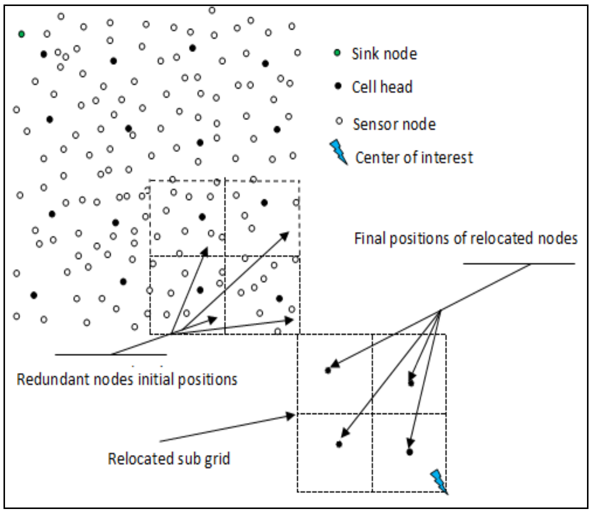

Figure 2: The virtual sub grid to be relocated to the COI

After detecting the virtual sub grid boundaries, the sink node will propagate the position information of the nearest cell, the sub grid boundaries, and the COI coordinates. When the nearest cell cell-head receives the virtual sub grid boundaries (propagated by the sink node), it will trigger the procedure that will relocate redundant sensors belonging to the virtual sub-grid zone to their final positions outside the ROI to capture the event information and to keep connectivity with the center of interest (COI). The virtual sub-grid boundaries propagation and the redundant sensor nodes relocation procedures are presented in the next section.

4.3 Propagation of the COI position

Once the sink node has determined the boundaries of the virtual sub grid that needs to be relocated outside the ROI, it sends a *NEAREST_CELL_ADV* message containing the coordinates of the COI and the coordinates (row and column numbers) of the upper left and the lower right cells of the virtual rectangular sub grid to be moved towards the COI. One of the coordinates sent in the *NEAREST_CELL_ADV* message correspond to the row and column numbers of the COI nearest cell.

As we are in presence of a square grid within the ROI, messages from the sink node can be transmitted along rows and columns. Therefore, the sink node propagates the *NEAREST_CELL_ADV* message to its next neighbor on the same row. When receiving the *NEAREST_CELL_ADV* message, the node *i* verifies if it is the neighbor of the message emitter belonging to the same row. If so, the node *i* then compares r_N, the COI nearest cell row number with r_i, the node *i* line number. If $r_N < r_i$, then the node *i* propagates the *NEAREST_CELL_ADV* message to its next neighbor along the same row until the message reaches a node *j* whose row number is equal to r_N. At that time, the node *j* forwards the *NEAREST_CELL_ADV* message to its neighbor on the same column until the message reaches the COI nearest cell.

4.4 Redundant nodes election and relocation

Once, the COI nearest cell (identified by row/column (r_N, c_N)) cell head receives the sink node message containing the COI position and the size of the virtual sub-grid to be moved to the COI, it computes the $Nearest\_node(r_N, c_N)$ coordinates. Indeed, the $Nearest\_node(r_N, c_N)$ is the sleeping node within the cell (r_N, c_N) which has the closest position to the COI among all its neighbors in the same cell. The coordinates $(x_N^{\min}, y_N^{\min})$ of the $Nearest\_node(r_N, c_N)$ satisfy the following equation:

$$\begin{aligned} d_{\min} &= \sqrt{\left(x_N^{\min} - x_{COI}\right)^2 + \left(y_N^{\min} - y_{COI}\right)^2} \\ &= \min\left\{ d_i = \sqrt{\left(x_N^i - x_{COI}\right)^2 + \left(y_N^i - y_{COI}\right)^2} \forall i \in cell(l_N, C_N) \right\} \end{aligned} \tag{7}$$

Then the $Cell\_Head(r_N, c_N)$ wakes up the node $Nearest\_node(r_N, c_N)$, creates and broadcasts two new messages *FORWARD_ROW* and *FORWARD_COLUMN* containing the same information as in the sink node's *NEAREST_CELL_ADV* message and are identified by a given sequence number.

Upon receipt of a *FORWARD_ROW* packet by the cell head *k* of the cell (r_k, c_k), (where r_k and c_k are row and column numbers of the cell which cell head is node *k*), the cell head *k* performs the following actions:

1) It verifies if it has already processed such a *FORWARD_ROW* message with the same sequence number.
2) If the *FORWARD_ROW* message has not been processed, the cell head compares its row number with the row number of the sender.
3) If they have the same row number, the receiver then verifies if the sender is its direct neighbor, i.e., the sender belongs to a cell whose column number is (c_k +1) or (c_k -1), depending on the COI's nearest cell position. Indeed, since node *k* column number is c_k, its neighboring cells column numbers can be either (c_k +1) or (c_k -1).
4) If condition 3) is satisfied, the receiver node k verifies if its coordinates are inside the virtual sub grid coordinates to be moved towards the COI.
5) If so, the node *k* generates a *FORWARD_ROW* packet and a *FORWARD_COLUMN* packet and then computes the $Nearest\_node(r_k, c_k)$: the node within the cell (r_k, c_k) which is the nearest node to the COI among

its neighbors in the same cell. The cell head k sends a message to wake up the node Nearest_node(r_k, c_k).

In the same way, upon receipt of a *FORWARD_COLUMN* packet by the cell head *m* of the cell(r_m, c_m), the node *m* performs the following actions:

1) It verifies if it has already processed such a *FORWARD_COLUMN* message with the same sequence number.
2) If the *FORWARD_COLUMN* message has not been processed, it compares its column number with the column number of the sender.
3) It they have the same column number, the receiver then verifies if the sender is its direct neighbor, i.e., the sender belongs to a cell whose row number is (r_m +1) or (r_m -1), depending on the COI's nearest cell position.
4) If condition (3) is satisfied, the receiver node m verifies if its coordinates are inside the virtual sub grid coordinates to be moved towards the COI.
5) If so, it generates a *FORWARD_COLUMN* packet and then computes the *Nearest_node*(r_m, c_m): the node within the cell(r_m, c_m) which is the nearest node to the COI. The cell head m sends a message to wake up the *Nearest_node*(r_m, c_m) sensor.

The above procedure is repeated until the *FORWARD_ROW* and the *FORWARD_COLUMN* packets reach a cell whose row and column numbers are outside of the virtual sub grid that should be moved towards the COI. After sending all the *FORWARD_ROW* and *FORWARD_COLUMN* packets and after waking up all redundant nodes, each redundant node in the virtual sub grid computes its trajectory based on the COI coordinates and the nearest cell coordinates transmitted in the *NEAREST_CELL_ADV*, *FORWARD_ROW* and *FORWARD_COLUMN* messages.

Since the trajectory is the same for all the redundant nodes, they move in parallel towards the COI and they all stop at the same time T_s : the time where the COI coordinates are within the *Nearest_node*(r_N, c_N) coverage region. In the next section, we present a performance evaluation of our proposed solution.

5. PERFORMANCE EVALUATION OF PROPOSED APPROACH

To evaluate the performance of our proposed sensor node relocation strategy described earlier, we have implemented our approach under the WSNet [7] simulator, an event driven simulator dedicated to wireless sensor networks. WSNet defines models written in C for the different network layers.

Using the WSNet-replay tool, we depict in figure 3 for a sample scenario, the initial square sized initial deployment zone and the relocated blue colored nodes covering the COI event and maintaining connectivity with the COI.

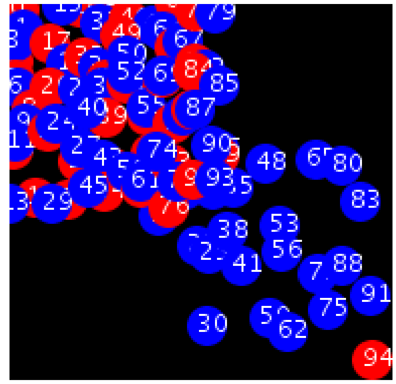

Figure 3: Wsnet-replay screen for the relocation approach

In this section, we evaluate the performance of our relocation approach in terms of the number of relocated nodes, the total energy consumption, and the relocation time. The total network energy consumption is directly affected by the number of relocated nodes. Another important evaluation criterion is the relocation time. In fact, this parameter will indicate how fast is the network reaction when being aware of a hazardous event happening outside the ROI. Hence, the shorter the relocation time is, the faster the network reacts in covering such an event.

Then, we consider three scenarios given by table 1, where we modify for each scenario the side size of the ROI, the number of sensor nodes initially deployed and the position of the COI. For each these configuration scenarios, the sink node is positioned at the upper left corner of the ROI at coordinates (0, 0).

Table 1: Simulation scenarios parameters

	ROI Area	Number of sensors	COI position
Scenario 1	100*100m	95	(180, 180)
Scenario 2	150*150m	200	(200, 190)
Scenario 3	200*200m	353	(220,250)

In figures 4 and 5, we depict the number of relocated sensors and the total energy consumption for the three scenarios above.

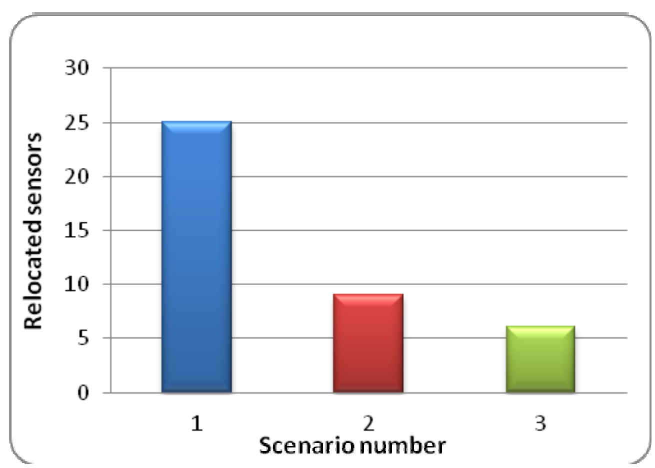

Figure 4: Number of relocated sensors

The figures show that the total consumed energy is proportional to the number of relocated sensors. This shows that mobility is the component that consumes the most energy in wireless sensor networks. Figures 4 and 5 also depict that the results obtained for the third scenario are the lowest in terms of the number of relocated sensors and the network energy consumption. This is because of the fact that the COI position is the closest one to the ROI compared to the two scenarios 1 and 2.

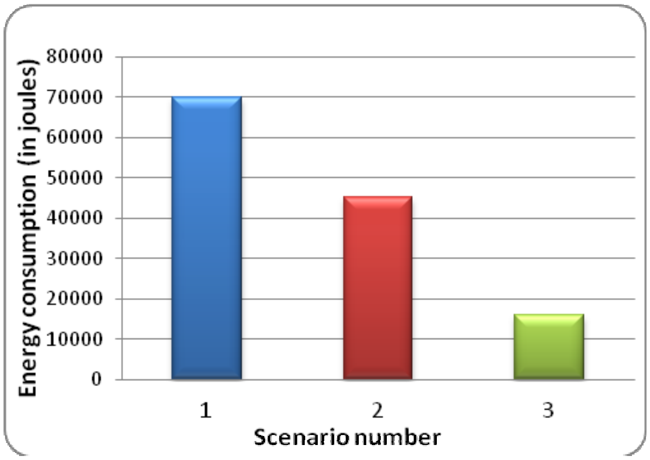

Figure 5: Total energy consumption for scenarios 1, 2, 3.

The close proximity of the COI to the ROI is further illustrated in figure 6 which illustrates the relocation time for the three different scenarios.

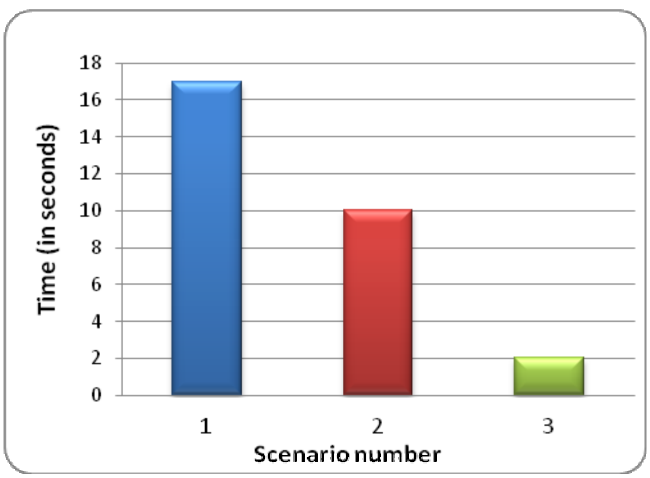

Figure 6: Relocation Time

The figure 6 shows that the relocation time has the highest value for the first scenario. This is due to the fact that for the first scenario, the distance that has to be crossed by the relocated sensors is the longest one among the three scenarios and since the redundant nodes are moving in parallel to their final locations, the relocation time obtained is the highest for that first scenario.

6. CONCLUSION

In this paper, we have proposed a novel approach to relocate wireless sensor nodes towards to center of interest when a new event has happened outside of the initial deployment zone. The relocated sensors have to cover the event happening at the COI while maintaining connectivity with their initial region of interest. To achieve this goal, we proposed a four phase based solution that includes cell heads election, evaluation of the number of nodes to be relocated, propagation of the COI position and redundant nodes election and relocation. We evaluated the performance of our proposed approach using the WSNet simulator. The obtained simulation results showed that the consumed energy and the relocation time increase as the distance separating the initial ROI from the COI increases no matter the size or the density of the initial deployment area are. As part of our future work, we will extend our approach to several centers of interests.

7. REFERENCES

[1] Ab. Aziz, N.A., Ab. Aziz, K. and Zakiah, W., “Coverage Strategies for Wireless Sensor Networks”, World Academy of Science, Engineering and Technology, Issue 50, year 2009.

[2] Ahlswede, R., Cai, N., Li, S.-Y. R, and Yeung, R. W., “Network Information Flow”, IEEE transactions oninformation theory, vol. 46, no. 4, year 2000.

[3] Albowitz, J., Chen, A. and Zhang, L., "Recursive position estimation in sensor networks," ICNP’01, 2001.

[4] Bartolini, N., Calamoneri, T., Fusco, E.G., Massini, A. and Silvestri, S., “Autonomous deployment of self-organizing mobile sensors for a complete coverage”, In proc. of the third International Workshop on Self Organizing Systems IWSOS 2008, Vienna, Austria, December, 2008.

[5] Benahamed, K., Haffaf., H., Merabti, M. and Llewellyn-Jones, D., Khelifa, B., “Monitoring connectivity in wireless sensor networks”, International Journal of Future Generation Communication and Networking, vol. 2, no. 2, year 2009.

[6] Biagioni, E. S. and Bridges, K. W., “The application of remote sensor technology to assist the recovery of rare and endangered species”, The International Journal of High Performance Computing Applications, vol. 16, pp. 112-121, year 2002.

[7] Chelius, G., Fraboulet, A. and Ben Hamida, E., “The WSnet Network Simulator”, http://wsnet.gforge.inria.fr, ver. 9.07, July 2009.

[8] Chiasserini, C.-F., Leonardi, E., Garetto, M. and Gribaudo, M., “A distributed sensor relocation scheme for environmental control”, In proc of the 4th IEEE International Conference on Mobile Ad-hoc and Sensor Systems, MASS 2007, Pisa, Italy, October 2007.

[9] Hou, J. C. and Zhang, H., “Maintaining sensing coverage and connectivity in large sensor networks”, Ad Hoc & Sensor Wireless Networks journal, vol.1, pp. 89-124, year 2005.

[10] Khedr, M. A., and Ramadan, H., “Effective Sensor Relocation Technique in Mobile Sensor Networks”, International Journal of Computer Networks & Communications (IJCNC), vol.3, no.1, year 2011.

[11] Li, J., Lachlan, L.H., Andrew, Foh, C. H., Zukerman, M. and Chen, H.-H., “Connectivity, Coverage and Placement in Wireless Sensor Networks”, Sensors Journal 2009, vol. 9, pp. 7664-7693.

[12] Li, X., Santoro, N. and Stojmenovic, I., “Mesh-Based Sensor Relocation for Coverage Maintenance in Mobile Sensor Networks,” In proc. of the 4th Int. Conf. on Ubiquitous Intelligence and Computing, UIC 2007, Hong Kong, China , July 2007.

[13] Liu, C.-M., Lee, C.-H., “Distributed Algorithms for Energy-Efficient Cluster-head Election in Wireless Mobile Sensor Networks”, In proc. of the 2005 International Conference on Wireless Networks, ICWN 2005, Las Vegas, USA, 2005.

[14] Li, W. and Cassandras, C.G., “A minimum-power wireless sensor network self-deployment scheme”, the 2005 IEEE conference on Wireless Communications and Networking Conference, WCNC, Neaw Orleans, USA, march 2005.

[15] Roychowdhury, S. and Patra, C. “Geographic Adaptive Fidelity and Geographic Energy Aware Routing in Ad Hoc Routing”, Special Issue of International Journal of Computer & Communication Technology (IJCCT) for International Conference on Computer Technology, (ICCT-2010), vol. 2 no. 2,3,4, year 2010.

[16] Sarazin, B., Rizvi, S., “A Self-Deployment Obstacle Avoidance (SOA) Algorithm for Mobile Sensor Networks“, International Journal of Computer Science and Security, year vol. 4, issue 3, year 2010.

[17] Savvides, A., Han, C.-C. and Strivastava., M. B., "Dynamic fine-grained localization in ad-hoc networks of sensors", In proc. of the ACM/IEEE International conference on Mobile Computing and Networking, MobiCom’01, Rome, Italy, July 2001.

[18] Wang, G., Cao, G., Porta, T. and Zhang, W., “Sensor Relocation in Mobile Sensor Networks,” In proc. of the 24th Annual IEEE International Conference on Computer Communications, INFOCOM 2005, Miami, USA, March 2005.

[19] Zou, Y. and Chakrabarty, K., “Sensor Deployment and Target Localization Based on Virtual Forces”, In proc. the 22nd Annual Joint Conference of the IEEE Computer and Communication Societies, INFOCOM 2003, San Francisco, USA, April 2003.