World Academy of Science, Engineering and Technology
International Journal of Electronics and Communication Engineering
Vol:12, No:3, 2018

Relay Node Placement for Connectivity Restoration in Wireless Sensor Networks Using Genetic Algorithms

Hanieh Tarbiat Khosrowshahi, Mojtaba Shakeri

Open Science Index, Electronics and Communication Engineering Vol:12, No:3, 2018 publications.waset.org/10008613/pdf

***Abstract*—**Wireless Sensor Networks (WSNs) consist of a set of sensor nodes with limited capability. WSNs may suffer from multiple node failures when they are exposed to harsh environments such as military zones or disaster locations and lose connectivity by getting partitioned into disjoint segments. Relay nodes (RNs) are alternatively introduced to restore connectivity. They cost more than sensors as they benefit from mobility, more power and more transmission range, enforcing a minimum number of them to be used. This paper addresses the problem of RN placement in a multiple disjoint network by developing a genetic algorithm (GA). The problem is reintroduced as the Steiner tree problem (which is known to be an NP-hard problem) by the aim of finding the minimum number of Steiner points where RNs are to be placed for restoring connectivity. An upper bound to the number of RNs is first computed to set up the length of initial chromosomes. The GA algorithm then iteratively reduces the number of RNs and determines their location at the same time. Experimental results indicate that the proposed GA is capable of establishing network connectivity using a reasonable number of RNs compared to the best existing work.

***Keywords*—**Connectivity restoration, genetic algorithms, multiple-node failure, relay nodes, wireless sensor networks.

I. Introduction

WSNs are one of the most important technologies in the 21st century [1]. A WSN comprises a set of tiny sensor nodes that sense and gather information from the physical world and, according to some local processing, send the sensed data to a base station controlled by the user [2]. WSNs are used in various areas of human life, especially in remote and harsh environments in which human work is risky or impractical [3]. Some of the main application areas are health, military, environment and security. The military applications of sensor networks, for example, include monitoring friendly forces, equipment and ammunition, battlefield surveillance, reconnaissance of opposing forces and terrain; targeting, battle damage assessment, and nuclear, biological and chemical (NBC) attack detection and reconnaissance [4].

The employment of WSNs in harsh environments incurs a number of challenges. One problem is node/s failure. Node failure can be categorized into two classes: single and multiple-node failures [5]. The single node failure is the failure of one node at a time. The node is called critical if its failure partitions the corresponding graph into at least two disjoints subgraphs. This type of nodes is referred to as cut-vertices. For example, nodes M_1, M_2, M_6, M_7, M_9 and M_{10} are cut-vertices in Fig. 1 [5]. The multiple-node failure, on the other hand, is the failure of more than one node at a time. Fig. 2 shows a multiple-node failure that partitions the network into five disjoint segments [5]. Both single and multiple-node failures result in the problem of network partitioning. In this paper, we consider the disjointedness caused by multiple-node failures.

The network partitioning problem seriously affects the network reliability mainly due to the collection of incomplete or perhaps incorrect information. It is thus very important to employ some strategies to restore network connectivity.

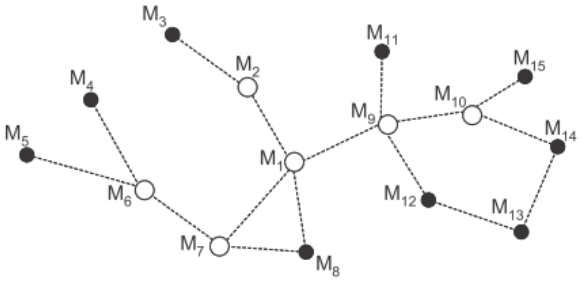


Fig. 1 Example of a single node failure; white nodes are cut-vertices whose failure leads to network disjointedness [5]

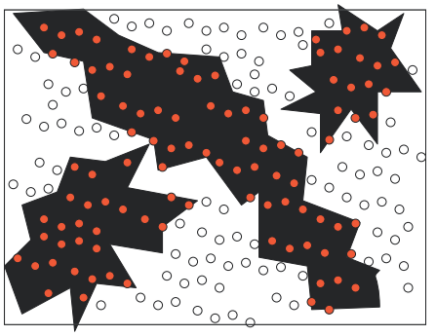


Fig. 2 Example of a multiple-node failure; solid circles indicate failed sensors whereas white ones are operational sensors [5]

Two main classes of connectivity restoration strategies have been introduced in the literature to address the network partitioning problem: pre-treated active recovery and passive recovery [6]. Pre-treated active recovery techniques run before

Hanieh Tarbiat Khosrowshahi is with the Faculty of Computer and Information Technology Engineering, Qazvin Branch, Islamic Azad University, Qazvin, Iran 34185-1416.

Mojtaba Shakeri is with the Computer Engineering Department, Faculty of Engineering, University of Guilan, Rasht, Iran 4199613776 (e-mail: shakeri@guilan.ac.ir).

Open Science Index, Electronics and Communication Engineering Vol:12, No:3, 2018 publications.waset.org/10008613/pdf

any node failure happens in the network to immediately restore connectivity after the failure. Passive recovery methods, however, run after the disjointedness and restore connectivity by using the neighboring information. The latter is usually used for recovering multiple-node failures as the number and location of failed nodes are unknown and it is not possible to employ active recovering strategies.

One widely-used passive technique to restore connectivity in the multiple-node failure is the placement of stationary relays [5]. Relays are more powerful sensing nodes used as data forwarders for prolonging the lifetime of a WSN. To find the location of RNs, the network partitioning problem is normally redefined as the Steiner tree problem (STP) by representing the segmented areas as terminals [7], [8]. The STP objective is to identity a number of non-terminals nodes called Steiner points (SPs) to form a minimum weighted spanning tree to cover all terminals nodes. The location of RNs is thus identified by finding the SPs. In the other words, a solution to the STP determines the number and position of SPs for the placement of relays to restore connectivity. The objective is to find a minimum number of SPs.

In this paper, we address the network partitioning problem where the network is split into a few disjoint segments when multiple nodes fail due to a catastrophic incident like an explosion. The connectivity is restored by introducing a number of relays. The problem is re-formulated as a STP where the objective is to find the minimum number of SPs for the placement of RNs. Having confronted with an NP-hard optimization problem, we develop a GA to iteratively reduce the number of relays and determine their location simultaneously. We can summarize the main contributions of our work in three directions as follows:

i. A new methodology based on metaheuristic algorithms (here, GA) is proposed for the first time for connectivity restoration in multiple-node failures.
ii. The superiority of the proposed GA in reducing the number of relays is demonstrated against the best existing work, especially for large scale damage.
iii. Having belonged to the class of metaheuristic algorithms, the proposed GA is capable of orienting the search towards optimizing some quality of service (QoS) indicators (other than minimizing the number of RNs) if being supplied with appropriate objective functions.

The rest of the paper is organized as follows. Section II reviews the related work on connectivity restoration algorithms specifically for multiple-node failures. Section III describes our solution approach. It starts from the statement of the problem under study and then continues by describing our proposed GA. Section IV presents our experimental results and the discussions about the accomplishments and drawbacks of proposed approach. Finally, Section V concludes the paper and highlights our future research directions.

II. Related Work

It is nearly a decade since the network partitioning problem has attracted the WSN research community, and numerous centralized and distributed algorithms have been developed accordingly. Most of the studies in this area, however, addressed the single node failure problem. Famous algorithms include DARA[1] [9], RIM[2] [10], PCR[3] [11], and DCR[4] [12] all of which employed a distributed policy.

Regarding the studies for connectivity restoration in multiple-node failures, most of the proposed algorithms adopt a centralized approach. This is mainly because sensors are normally stationary due to coverage and task requirements. Moreover, the final topology formed often results in poor coverage [8]. DORMS[5] is one of the rarest distributed algorithms developed by Lee and Younis [7] to address the multiple-node failure problem for large scale damage in the network. The objective is to restore a segmented WSN using a minimum number of mobile RNs. The approach establishes a topology that looks like a Steiner tree. SPs specify the location of RNs. DORMS has two advantages. First, the recovery is faster as compared to its counterparts and second, message complexity is low due to its distributed nature. However, it suffers from two drawbacks. Redundant RNs are required for the federation of the partitioned network, yet it is possible that a failure of one RN splits the restored connected network into at least two partitions.

Lee and Younis [13] addressed the multiple-node failure problem for scenarios where the distance between each pair of segments is more than twice the communication range of a node. It is also required that some inter-segment QoS indicators to be satisfied. They developed a restoration algorithm called QRP[6]. Their proposed approach operates in two phases. The first phase ensures inter-segment connectivity by specifying the positions to place the fewest number of RNs. In the second phase, QRP deploys RNs in the selected positions to meet the corresponding QoS value. The drawback of QRP is that critical paths may be created during connectivity restoration in which the failure of a RN can cause a new partition in the network. Senel et al. [14] addressed the same problem by developing a bio-inspired approach aiming at reestablishing connectivity using the fewest number of relays while ensuring a certain quality in the formed topology. The approach establishes a topology that resembles a spider web. It consists of two algorithms: 1C-SpiderWeb and 2C-SpiderWeb. 1C-SpiderWeb deploys the relays inwards to yield better network connectivity and coverage. Similar to DORMS, the algorithm unites the network significantly fast, but it uses many RNs and does not consider obstacles in the network. 1C-SpiderWeb does not benefit from a strong topology and lacks minimized average path length, coverage, and balanced traffic load. This led to the development of 2C-SpiderWeb to provide 2-vertex connectivity which is a stronger connectivity.

One of the efficient algorithms to restore connectivity for multiple-node failures is ORC[7] developed by Lee and Younis

[1] Distributed Actor Recovery Algorithm
[2] Recovery through Inward Motion
[3] Partitioning detection and Connectivity Restoration
[4] Distributed partitioning detection and Connectivity Restoration
[5] Distributed Optimized Relay node placement using Minimum Steiner trees
[6] QoS-aware Relay node Placement
[7] Optimized Relay node Placement

Open Science Index, Electronics and Communication Engineering Vol:12, No:3, 2018 publications.waset.org/10008613/pdf

[8]. The approach forms a topology that looks like a Steiner tree on the convex hull. ORC tries to identify SPs on which relays are deployed such that the segments are connected with the fewest number of Relays. ORC is able to significantly reduce the number of populated RNs and generate more efficient topology in terms of the average length of inter-segments data paths and balanced traffic load. Nonetheless, it cannot handle inter-segment QoS requirements, and generate a bi-connected topology that can tolerate the failure of a node and enable enforcement of disjoint data distribution paths as a means of balancing the load [3]. Lee et al. [15] later aimed at improving ORC to ensure inter-segment bi-connectivity by proposing a connectivity restoration with assured fault tolerance algorithm called CRAFT for large-scale damage in WSN. Another failure-tolerant algorithm proposed is OBiC[8] that establishes 2-vertex connectivity between each pair of segmented parts of the network while minimizing the number of the deployed RNs [16].

The most recent work carried out for multiple-node failure connectivity restoration belongs to Bouyahia and Benchaïba [17] who developed a connectivity repairing protocol called CRVR[9]. The algorithm restores the inter-segment connectivity by creating strengthening paths to attain the following three objectives: failure toleration, latency reduction and energy consumption minimization. A strengthening path allows the network to have a longer connectivity lifetime for different segments, but it requires more redundant nodes. It can be concluded from our review that different, mostly conflicting, objectives have been considered for connectivity restoration in the existing work including the restoration cost (in terms of the number of deployed relays) versus the restoration time and reliability. Given the restoration cost, ORC proves to be the most efficient algorithm among others [17]. This paper also focuses on minimizing the cost of restoring the segmented network by proposing a GA. The experimental results indicate that the proposed GA is capable of establishing network connectivity using a reasonable number of RNs compared to ORC.

III. The Proposed Approach

A. Problem Statement

We can describe our failure scenario as follows. Suppose a WSN is formed by a number of stationary sensor with identical communication ranges populated randomly in a flat environment with no obstacle (Fig 3 (a)). The network is fully operational where a destructive event like an explosion occurs at a random time that damages a number of nodes. The failure splits the network into a few disjoint partitions (Fig. 3 (b)). We adopt the same procedure used in [8] to form a convex hull from which a STP is introduced. For each segment, a representative node is chosen among the surviving nodes in its vicinity to connect its corresponding segment to the resulting topology. The problem objective is then to connect the representative nodes by populating RNs such that the disconnected network is restored using the fewest number of relays as possible.

The representative nodes to each segment are shaped as follows [8]. First, the scope of failures is detected by a number of surviving nodes by observing an unexpected major drop in communication traffic and/or by noticing that a certain number of remote nodes cannot be reached. These nodes then become the border nodes of the damage area and specify the boundaries of segments by broadcasting a message on active links to notify all reachable nodes, which will obviously lie in their segment. The formation of segments is illustrated in Fig. 3 (b) where the active links are shown by yellow lines. Finally, the representative node to each segment would be the border node that has lost connection to the most number of neighbors. This is shown by solid green squares in the same figure.

Similar to [8], we use a centralized control policy to identify the location of network partitions. It is assumed that the network has a few mobile robots equipped with a GPS that scout the area on behalf of the individual segments. It is important to note that an additional relay is placed next to each representative node to take over forwarding inter-segment traffic since the surviving stationary nodes are normally of insufficient capacity and range. Accordingly, the final number of deployed relays is equal to the number of representative nodes plus the number of RNs required for connectivity restoration.

To limit the area for finding the placement of RNs, we find a convex hull bounded by the location of representative nodes using the Graham scan algorithm [18]. The algorithm has been described in [8] as follows. It starts from the left-most (least y-coordinate) segment Seg_L, and then sorts the segments in the increasing order of an angle with x-axis relative to Seg_L and finally selecting segments from the sorted list so that right-turns are maintained when moving from Seg_L until ending at Seg_L. Fig. 4 illustrates the convex hull formed by applying the Graham scan algorithm based on five segments. To address the network partitioning problem as the STP, it is required to calculate an upper bound to the number of SPs within the convex hull such that inter-segment connectivity is guaranteed. The number is used to determine the length of GA's chromosomes and generate the initial population accordingly. The proposed GA is then employed to minimize the number of SPs while maintaining inter-segment connectivity. Note that SPs indicate the location where RNs should be placed.

B. SPs Upper Bound Calculation

To acquire an appropriate estimation of the number of SPs, we first produce the smallest rectangle that can embrace the convex hull. The width of the rectangle is the longest internode vertical distance (i.e., maximum difference in y-coordinates) while the length is the longest internode horizontal distance (i.e., maximum difference in x-coordinates). This is illustrated in Fig. 5. Next, we calculate the maximum number of SPs by developing a simple heuristic. The rectangle area is filled with identical squares whose diameter is two times the communication radius of RNs (see

[8] Optimized Bi-Connected federation of multiple sensor network segments
[9] Connectivity Repairing in WSN with Void Region

Fig. 6). The total number of squares should thus determine the number of SPs required to establish inter-segment connectivity. However, this is only valid if relays are placed on the center of squares. As shown in Fig. 7, if relays are randomly placed near to the corners of squares, it is possible that inter-segment connectivity is not maintained before the GA starts iteratively to reduce the number of required relays. We thus require considering more SPs.

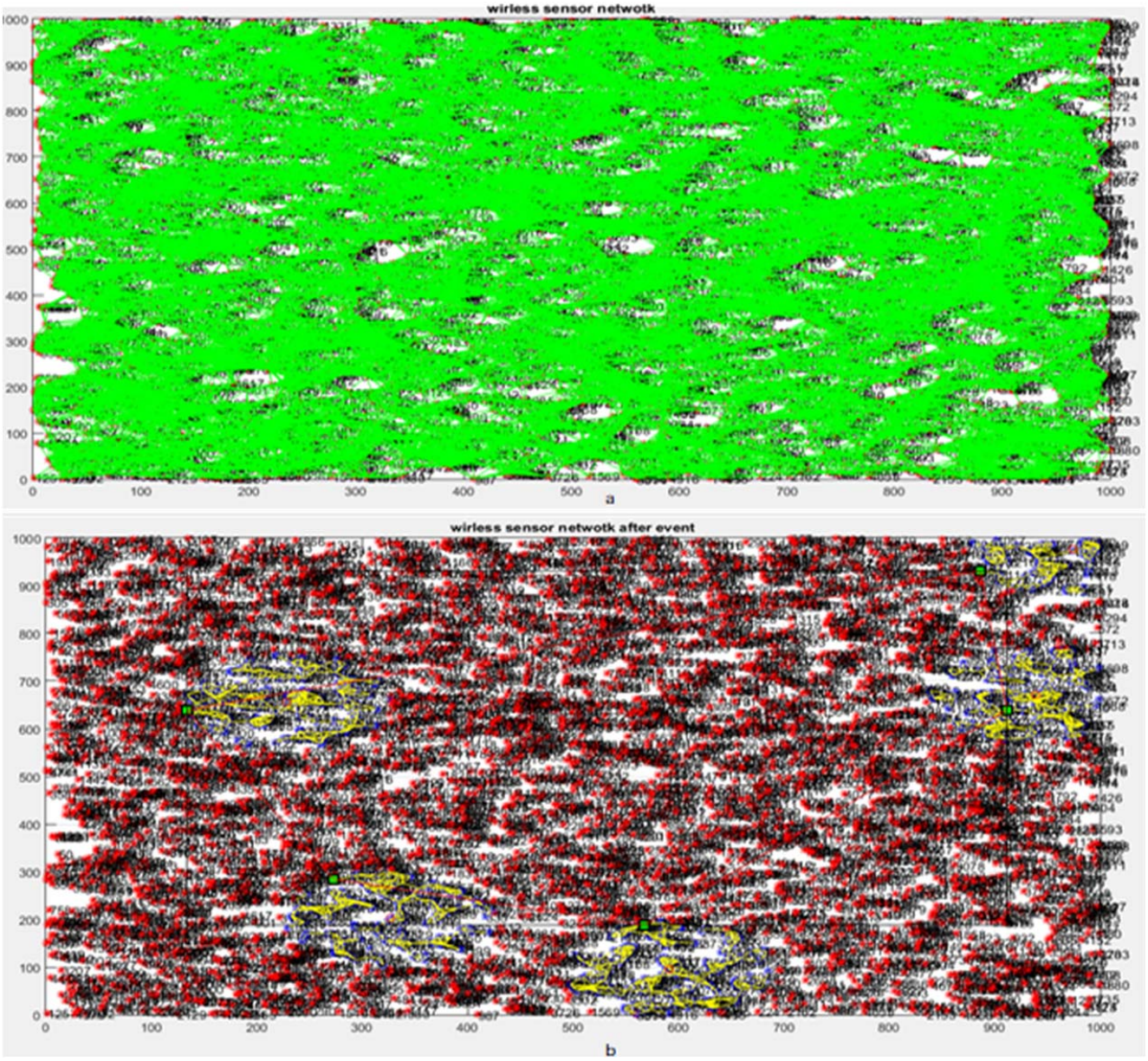


Fig. 3 The status of a WSN before and after a destructive event where the network has been split into five disjoint partitions formed by active links shown in yellow; to each segment, one representative node depicted in solid green squares is identified

Open Science Index, Electronics and Communication Engineering Vol:12, No:3, 2018 publications.waset.org/10008613/pdf

Open Science Index, Electronics and Communication Engineering Vol:12, No:3, 2018 publications.waset.org/10008613/pdf

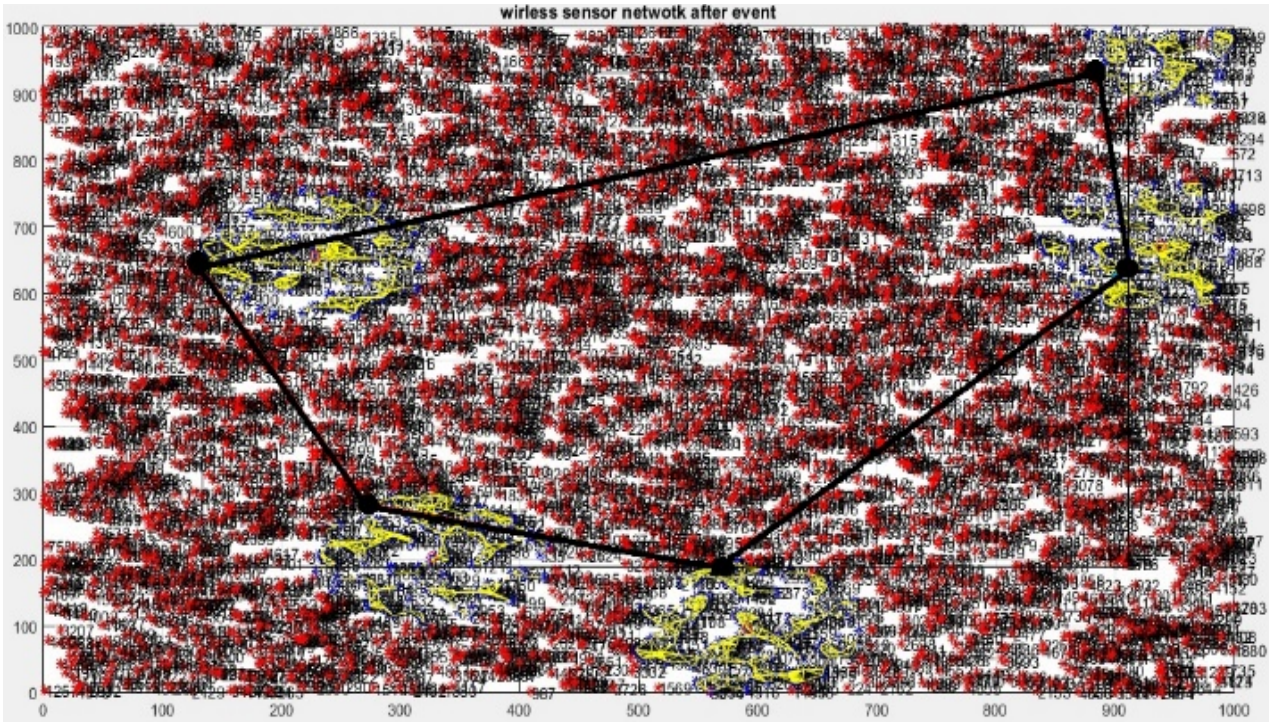


Fig. 4 The convex hull formed out of five segments

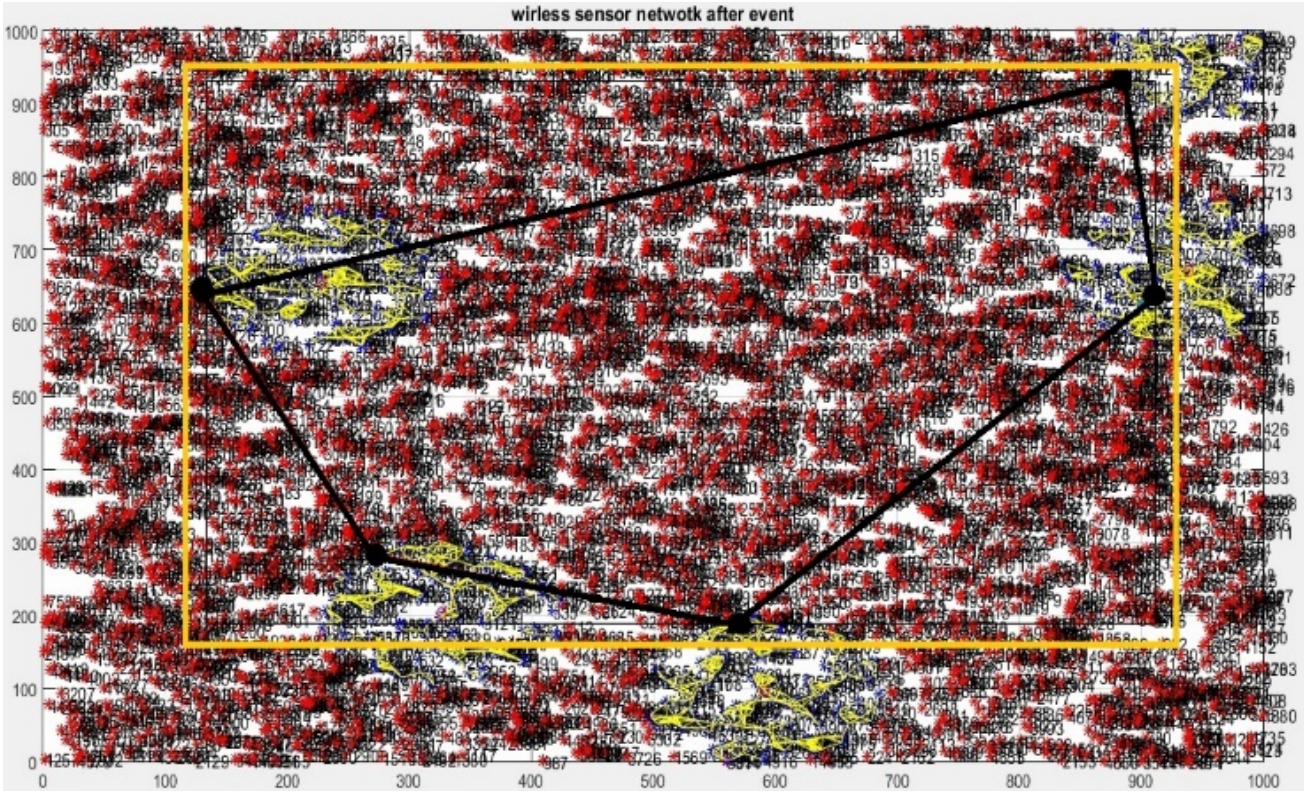


Fig. 5 The light orange rectangle specifies the area where the RNs should be placed

Open Science Index, Electronics and Communication Engineering Vol:12, No:3, 2018 publications.waset.org/10008613/pdf

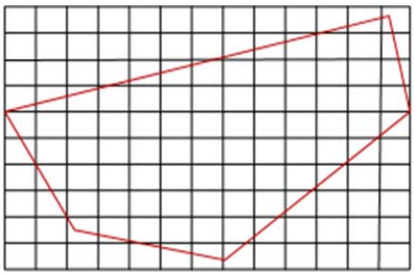


Fig. 6 The rectangle embracing the convex hull has been filled with identical squares whose diameter is two times the communication radius of RNs

Having inspired from Younis and Akkaya [19], we insert one square between two successive squares both along the x- and y-axes (see Figs. 8 (a) and (b)). This makes the total number of SPs be twice the original one. We can thus calculate the upper bound *UB* according to:

$$UB = \frac{2\times W\times L}{r_{RN}^2} \quad (1)$$

where r_{RN} is the communication radius of RNs and L and W are the length and width of the rectangle, respectively.

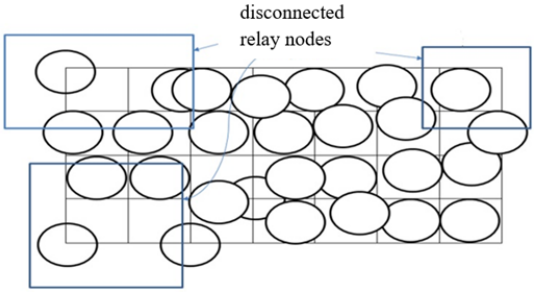


Fig. 7 The condition where deployed RNs become disconnected

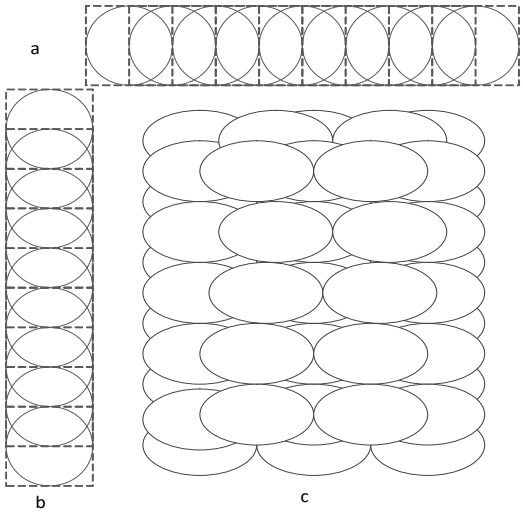


Fig. 8 Insertion of additional relays between two successive squares along (a) the x-coordinate and (b) the y-coordinate; (c) the resulting initial placement

This number is then given as the input to the GA to set up the chromosome representation and generation of initial population.

C. Genetic Algorithm

1) Chromosome Representation

In this paper, we introduce a novel representation of chromosomes such that GA is capable of simultaneously determining the number and location of RNs in the segmented network. Fig. 9 depicts the structure of a chromosome. It consists of two parts. The first one is an array of binary values whose length is equal to the maximum number of SPs calculated in the previous section. The value of one indicates that the corresponding SP has an active RN whereas the value of zero means no relay is considered. The second part is a vector of real values specifying the x-y coordinates where relays, if selected by the first part, should be placed. This implies that the length of the second part is twice the number of maximum relays. Consider the chromosome shown in Fig. 9. The first part comprises a binary vector of length five, meaning that the maximum number of relays to ensure connectivity is five. Accordingly, we need a real array of length 2×5=10 to accommodate the x-y coordinates of the five potential relays. This chromosome, however, chooses three relays to be placed according to their corresponding position specified in the second part. For instance, the position of the first relay is 24 and 25 for x- and y-coordinates, respectively.

1	2	3	4	5	X1	Y1	X2	Y2	X3	Y3	X4	Y4	X5	Y5
1	0	1	1	0	24	25	1	63	14	35	78	33	52	70

Fig. 9 An example of chromosomes for five RNs

2) Initial Population

The initial population of our proposed GA algorithm is set up by generating 100 chromosomes by deploying the maximum number of relays. This ensures that the full network connectivity has been maintained for each of the initial chromosomes before GA sets out its optimization process. The initial location of each relay is randomly chosen within its corresponding SP's area which is determined by the communication radius of RNs. Fig. 10 illustrates an example where the maximum number of SPs is eight and given that the communication radius of relays is 50, 16 real numbers representing x- and y-coordinates for the placement of eight relays are randomly chosen within their corresponding domains.

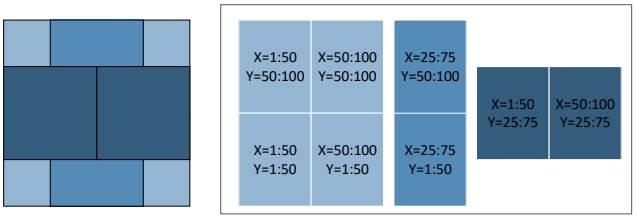


Fig. 10 The sampling domains for placing RNs onto their corresponding SPs, given that eight SPs have been considered

Open Science Index, Electronics and Communication Engineering Vol:12, No:3, 2018 publications.waset.org/10008613/pdf

3) Fitness Evaluation

The procedure to evaluate the fitness of GA's chromosomes is made of two phases. The first phase checks whether the decision that the given chromosome has made on the selection and positioning of relays ensures network connectivity. If the connectivity holds then the second assessment phase is invoked; otherwise, the chromosome is not considered for the evaluation. The second phase evaluates each chromosome by re-considering it as an instance to the STP (STP). Recall that STP aims at finding a minimum spanning tree to cover all terminal nodes (here, representative nodes to each segment) by choosing an appropriate number of SPs (here, the RNs). Given that the number of deployed relays and their positions have been specified in the chromosome, the evaluation mechanism employs the Kruskal algorithm [20] to construct the minimum spanning tree. The final fitness of the chromosome is then measured by dividing the total number of deployed relays by the total weight of the constructed spanning tree:

$$Fitness\ function = \frac{\sum_{i=1}^{UB} gene(i)}{tw} \quad (2)$$

where *UB* is the upper bound, *gene*(i) is the value of the i^{th} gene of the first part of the chromosome that indicates the deployment or nondeployment of the i^{th} relay and tw is the total weight of the produced minimum spanning tree. Note that the weight to each edge is equal to the Euclidean distance between its two end nodes.

4) Reproduction

Unlike traditional GA algorithms where the reproduction process consists of recombination and mutation phases, in this paper we only apply mutation to each individual per generation. The reason why recombination was not used in the structure of our proposed GA was that the generated offspring is highly susceptible to violate network connectivity if n-point crossover operators are to be used.

The mutation operator is applied to both parts of the chromosomes. Subject to a given mutation probability, those elements whose value is equal to 1 in the first part are reset. This is because of GA's objective is to reduce the number of active relays per generation. For the second part, the mutation operator is described as follows. Again, subject to a probability, each x- or y- coordinate of active relays is mutated by being added to a random variable uniformly chosen from [-10, +10]. This can lead to a slight movement of relays in each of vertical, horizontal, or diagonal directions as shown in Fig. 11.

It is important to note that in case a mutated individual does not maintain network connectivity, it is replaced with the original one to ensure that connectivity is not violated after mutation; otherwise, its fitness is calculated according to (2). In either case, the individual is added to the mutated population.

5) Replacement

Finally, the next population of GA is formed by merging the current and mutated populations. The individuals of both populations are sorted in descending order of their fitness, and the first half is selected as the next population. Both reproduction and replacement procedures continue for a predetermined number of iterations until GA converges. After GA's termination, the number and location of active relays are decoded from the best chromomere. As mentioned earlier in the paper, this number is added to the number of representative nodes to produce the final number of deployed relays. Fig. 12 depicts the overall flowchart of our proposed GA algorithm.

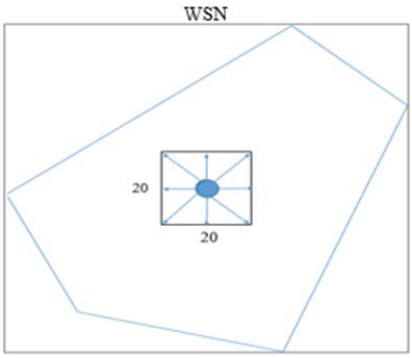


Fig. 11 Possible movements of a relay in vertical, horizontal and diagonal directions within the convex hull

IV. Experimental Results and Discussion

Our proposed solution approach was coded in MATLAB R2015a programming environment and ran on a system equipped with a Core™ i7 processor and 4GB RAM. As our focus was on connectivity restoration and to make the network operational again, we ignored the dynamics of packet exchanges and energy consumption in our evaluation scenarios. This justifies why MATLAB was used for the implementation rather than those well-known network simulators like ns-2 or OMNeT++.

Table I lists the value of parameters required for the experimental setup. According to the table, the network area was considered to be 1000×1000 m^2. 5000 stationary sensors with the communication radius of 40 m were randomly populated in the network. We carefully checked that full connectivity had been attained and the network was entirely operational before any failure scenario was applied. We assumed two experimental parameters to conduct our experiments: the communication radius of RNs denoted as r_{RN} and number of disjoint segments denoted as n_{Seg}. As Table I indicates, r_{RN} is supplied with three different values (including 200, 250 and 300 m) and n_{Seg} is given with two values (including 5 and 7) leading to totally six test configurations. We expect that our GA would identify fewer number of RNs with respect to higher values of r_{RN} and smaller values of n_{Seg}. It should be noted that disjoint segments were randomly located at least r_{RN} units away from each other to ensure that the network was split into the prespecified number of disjoint segments.

Open Science Index, Electronics and Communication Engineering Vol:12, No:3, 2018 publications.waset.org/10008613/pdf

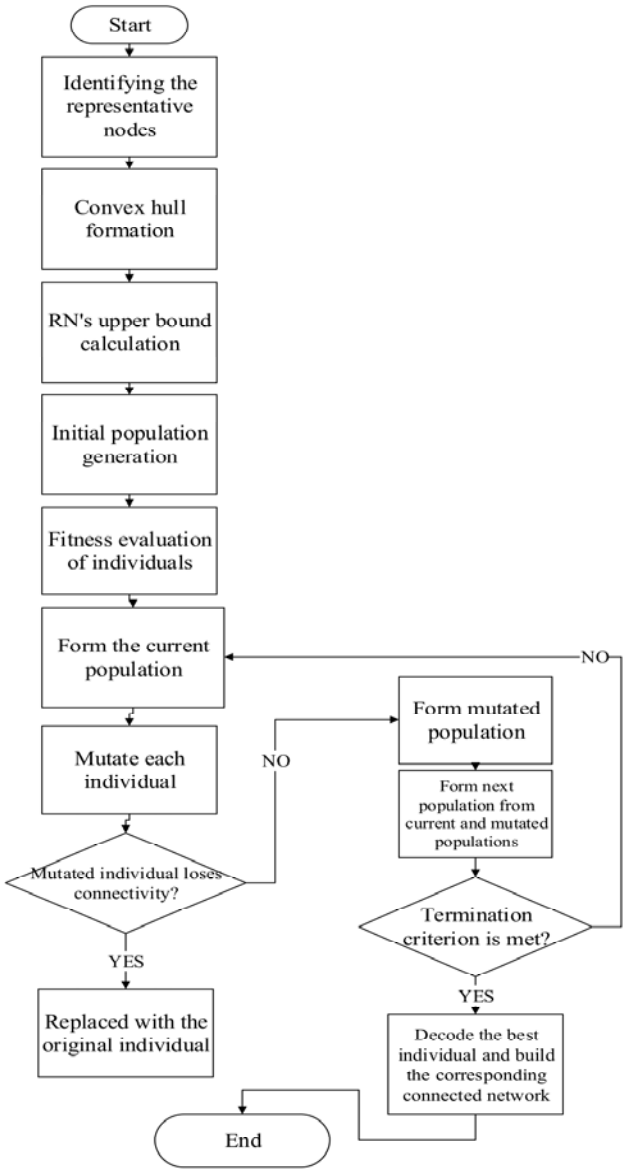


Fig. 12 The flowchart of our proposed GA algorithm

TABLE I
EXPERIMENTAL SETUP

Parameter	value
Network area	1000×1000 m^2
Number of sensors	5000 nodes
Communication radius of sensors	40 m
Communication radius of RNs	200-300 m
Number of disjoint segments	5 and 7
GA's population size	100
GA's mutation probability	0.1
GA's replacement strategy	Elitism
GA's termination criterion	100 generations

Our approach was compared to ORC [8], which, as stated in Section II, proves to be the best restoration algorithm in minimizing acquired restoration costs represented as the number of deployed relays. Two evaluation measures were defined for the comparison: number of deployed RNs and average hop count. The latter is an indicator of the connectivity of the resulting topology in terms of delivery latency [8]. It is equal to the total number of links along the shortest path between each pair of two segments divided by n_{Seg}. We assume that all communication links are bidirectional; thus, we count both directions between two segments as is done in [8]. Similar to ORC, our GA was run 30 times per test configuration.

Figs. 13 and 14 present the number of deployed RNs for both GA and ORC relative to different values of r_{RN} and n_{Seg}. According to Fig. 13, our proposed GA outperforms ORC by deploying fewer relays as the communication range of RNs increases. This is in line with future generations of RNs which naturally benefit from broader communication ranges. On the other hand, when the number of disjoint segments increases from 5 to 7, as shown in Fig. 14, GA is capable of deploying fewer RNs than ORC when the communication range of relays is at its lowest value (here, 200 m). This implies that our proposed GA exhibits a better performance than ORC for large scale damage. Lastly, according to Fig. 15, GA not only deploys fewer relays than ORC for long communication ranges but also benefits from shorter delivery latency as indicated by the value of the average hop count of the resulting topology. This is shown for five disjoint segments in the figure.

Although restoration time was not considered as the focus of this study, we should mention that the major drawback of our proposed GA is its high convergence time which was nearly a couple of hours on average. There are two key reasons for such a long termination time. First, we used a fixed termination criterion for all evaluation scenarios. In the other words, GA stopped searching exactly after 100 generations. Alternatively, if we had employed an adaptive termination criterion where GA could stop searching if no improvement would achieve, the computational time would definitely decrease significantly. Second, the time complexity of our proposed GA is severely dependent on the upper bound calculated for the number of RNs. The smaller is the upper bound, the faster is the GA. This could be readily observed when the communication range of RNs increased to 300 as the upper bound computed in this scenario was considerably lower than the case where r_{RN}=200. Accordingly, combining our GA with fast heuristics that estimate a tighter upper bound to the number of RNs, would substantially improve our GA's computational burden.

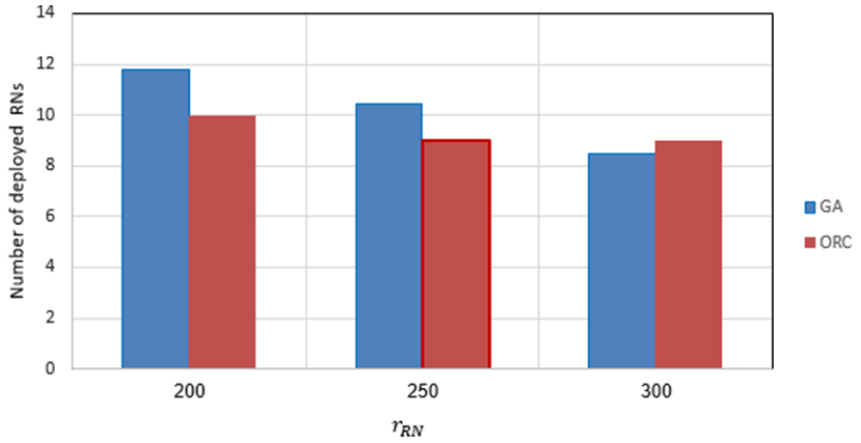


Fig. 13 The average number of deployed RNs as r_{RN} increases from 200 to 300 for n_{Seg}=5; GA is able to deploy fewer relays for r_{RN}=300

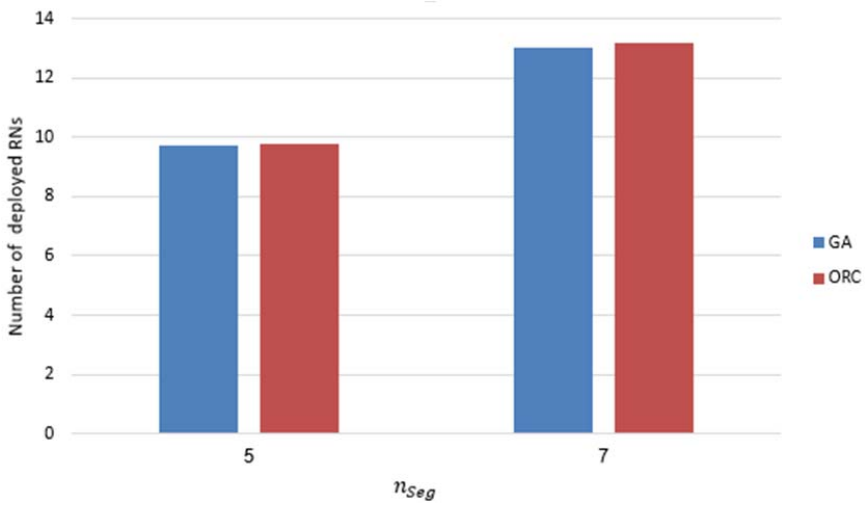


Fig. 14 The average number of deployed RNs relative to the number of disjoint segments when r_{RN}=200; GA is able to deploy fewer relays when n_{Seg}=7

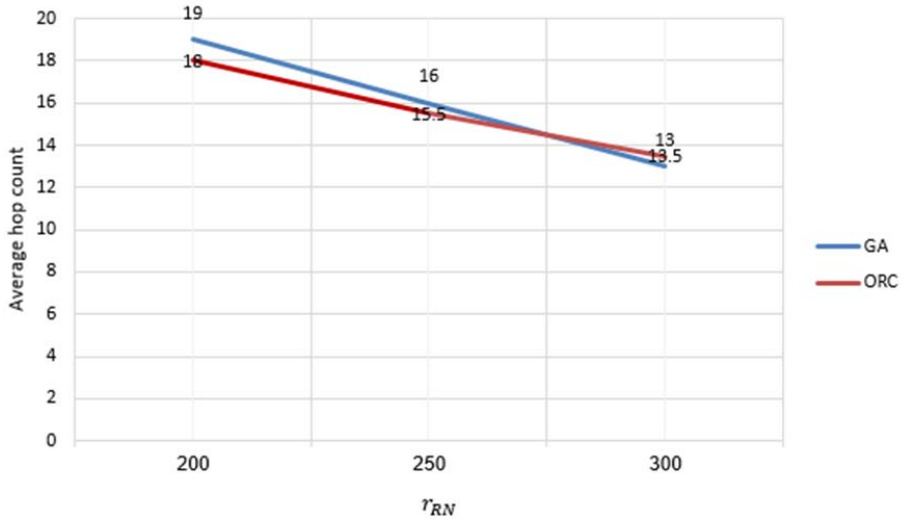


Fig. 15 The average hop count of the resulting topology of GA and ORC when n_{Seg}=5; GA produces relatively a smaller value for r_{RN}=300

V. Concluding Remarks and Future Work

This paper addressed the network partitioning problem when the network is split into a number of disjoint segments after multiple nodes fail as a result of a catastrophic incident

Open Science Index, Electronics and Communication Engineering Vol:12, No:3, 2018 publications.waset.org/10008613/pdf

like an explosion. Having redefined the problem as the STP (which is known to be an NP-hard optimization problem), a GA was developed to find the fewest number of RNs required for connectivity restoration. To the best of our knowledge, no related study has exploited the powerful searching strategy of metaheuristic algorithms like GA for connectivity restoration in multiple-node failures. The proposed GA was able to outperform the ORC algorithm (known to be the best existing algorithm for restoration cost minimization) as the communication range of relays increases and the number of disjoint segments grows.

As our future work, we can extend our GA by considering other QoS indicators like restoration reliability. Given that GA belongs to the class of metaheuristic algorithms whose search behavior is systematically oriented by defining different objective functions, we strongly believe that the realization of the above claim would not be intractable.

REFERENCES

[1] C. Y. Chong, S. P. Kumar," Sensor networks: evolution, opportunities, and challenges," *Proceedings of the IEEE*, vol. 91, no. 8, pp. 1247–1256, Aug. 2003.

[2] J. Yick, B. Mukherjee, D. Ghosal, "Wireless sensor network survey," Computer networks, vol. 52, no. 12, pp. 2292-2330, Apr. 2008.

[3] V. Ranga, M. Dave, A. K. Verma, "Network partitioning recovery mechanisms in WSANs: a survey", *Wireless personal communications*, vol.72, no. 2, pp. 857-917, Feb. 2013.

[4] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, "Wireless sensor networks: a survey," Computer networks, vol. 38, no. 4, pp. 393-422, Dec. 2002.

[5] M. Younis, I. F. Senturk, K. Akkaya, S. Lee, F. Senel, "Topology management techniques for tolerating node failures in wireless sensor networks: A survey," Computer Networks, vol. 58, pp. 254-283, Sep. 2014.

[6] C. Wu, H. Chen, J. Liu, "A survey of connectivity restoration in wireless sensor networks," in 3rd International Conference. Consumer Electronics, Communications and Networks (CECNet), IEEE, 2013, pp. 65-67.

[7] S. Lee, M. Younis, "Recovery from multiple simultaneous failures in wireless sensor networks using minimum Steiner tree," Journal of Parallel and Distributed Computing, vol.70, no. 5, pp. 525-536, Jun 2010.

[8] S. Lee, M. Younis, "Optimized relay node placement for connecting disjoint wireless sensor networks," Computer Networks, vol. 56, no. 12, pp. 2788-2804, May 2012.

[9] A. A. Abbasi, K. Akkaya, M. Younis, "A distributed connectivity restoration algorithm in wireless sensor and actor networks", in 32nd IEEE conference, on Local computer networks, IEEE, 2007, pp. 496-503.

[10] M. Younis, S. Lee, S. Gupta, K. Fisher, "A localized self-healing algorithm for networks of moveable sensor nodes". In Global Telecommunications Conference, Nov, 2008. IEEE GLOBECOM 2008, IEEE pp. 1-5.

[11] M. Imran, M. Younis, A. M. Said, H. Hasbullah, "Partitioning detection and connectivity restoration algorithm for wireless sensor and actor networks", IFIP International Conference on Embedded and Ubiquitous Computing, IEEE,2010.

[12] M. Imran, M. Younis, A. M. Said, H. Hasbullah, "Localized motion-based connectivity restoration algorithms for wireless sensor and actor networks," Journal of Network and Computer Applications, vol. 35, no. 2, pp. 844-856, Dec 2012.

[13] S. Lee, M. Younis," QoS-aware relay node placement for connecting disjoint segments in wireless sensor networks," In Distributed Computing in Sensor Systems Workshops (DCOSSW), 2010 6th IEEE International Conference on (pp. 1-6). IEEE, June 2010.

[14] F. Senel, M. F. Younis, K. Akkaya, "Bio-inspired relay node placement heuristics for repairing damaged wireless sensor networks," IEEE Transactions on Vehicular Technology, vol. 60, no. 4, pp. 1835-1848, May 2011.

[15] S. Lee, M. Younis, M. Lee," Connectivity restoration in a partitioned wireless sensor network with assured fault tolerance", Ad Hoc Networks, vol. 24, pp. 1-19, 2015.

[16] S. Lee, M. Younis, M. Lee, "Optimized bi-connected federation of multiple sensor network segments," Ad Hoc Networks, vol. 38, pp. 1-18, 2016.

[17] K. Bouyahia, M. Benchaïba, "CRVR: Connectivity Repairing in Wireless Sensor Networks with Void Regions," Journal of Network and Systems Management, vol. 25, no. 3, pp. 536-557, 2016.

[18] R. L. Graham," An efficient algorithm for determining the convex hull of a finite planar set", Information Processing Letters 1, 1972, pp. 132–133,1972.

[19] M. Younis, K. Akkaya," Strategies and techniques for node placement in wireless sensor networks: A survey," Ad Hoc Networks, vol. 6, no. 4, pp. 621-655, 2008.

[20] J. B. Kruskal,"On the shortest spanning subtree of a graph and the traveling salesman problem," Proceedings of the American Mathematical society, vol. 7, no. 1, pp. 48-50, 1956.

Open Science Index, Electronics and Communication Engineering Vol:12, No:3, 2018 publications.waset.org/10008613/pdf