Recovery of Faulty Cluster Head Sensors by Using Genetic Algorithm (RFGA)

Elmira Moghaddami Khalilzad[1] , Sanam Hosseini[2]

[1]Department of Technical and Engineering, Tabriz Branch, Islamic Azad University, Tabriz, Iran

[2]Department of Technical and Engineering, Tabriz Branch, Islamic Azad University, Tabriz, Iran

Abstract

One of the most important challenges in the field of wireless sensor network is energy consumption that generally clustering is used to reduce energy consumption, increase scalability and facilitate routing. Fault in cluster heads makes disturbance in packets transmission so that their members become unavailable. So, re-clustering should be done and this action results in energy and time waste in system. In order to reduce negative effects of this approach, a method has been presented to recover member nodes of faulty cluster head by using genetic algorithm based recovery algorithm which acts multi-purposely. The proposed algorithm considers recovery of each node based on the distance of each node from cluster heads, remained energy of cluster heads and member number of each cluster head and selects the best cluster head to recovery. The proposed algorithm has been simulated by Matlab and obtained results show that this algorithm has better efficiency in case of considering three parameters of distance of each node from cluster heads, remained energy of cluster heads and member number of each cluster head compared with a case of considering only one of the parameters.

Keywords: *Wireless Sensor Network, Cluster-Head, Recovery, Genetic Algorithm.*

1. Introduction

Wireless sensor networks are high level distributed networks which have so many light, inexpensive and small wireless nodes used to maintain environment or a complex system. Each node includes three sub-systems. First one is wireless sub-system which senses the environment. The second one is processing sub-system which performs calculations and the third one is communication sub-system which transmits data among neighbor nodes [1]. Sensor nodes have limitation in the sensing area and processing capability. The location of sensor nodes has been predefined and it is not known that whether such property could make it possible to leave them in risky and unavailable locations. Although each node has low capability separately, the combination of hundreds small sensors provides new possibilities [2]. In fact, the power of wireless sensor networks is in their capability to apply many small nodes which are able to organize by themselves and they are used in various situations like simultaneous routing, maintaining environmental conditions, maintaining the safety of structures or equipments of a system. All nowadays applications and technologies converge to ad hoc and wireless sensor networks which have effect on our life, so that most of the researches are in the field of improving the application of wireless sensor networks [3]. Energy is one of the most important factors in routing. So, most of the routing protocols are based on reducing consuming power. Designing an efficient routing protocol and quality of services is one of the main challenges in wireless sensor networks resulted from the combination of wireless sensor networks' features [4, 5]. In order to save consuming energy, clustering is performed. In wireless sensor networks, clustering is an efficient and useful mechanism to control topology and data collection. However, rough areas, resource restriction of nodes and unbalanced workload among nodes make clustered wireless networks vulnerable while encountering with communicative faults and it reduces the reliability of network. Therefore, fault tolerant mechanisms are necessary in real applications of sensors. Wireless sensor networks consist of many sensor nodes which spread in a particular range in order to maintain environmental phenomena and sense the environment. The sensed data are transmitted by each node in a broadcasting way in order to reach central node called sink [6, 7].

In order to save consuming energy to communicate with central node, several routing techniques have been suggested which are multi-hop and aware of energy. These techniques have overhead due to route detection and finding the number of optimum hops to communicate with the central node. In order to prevent this overhead and unbalanced consumption of energy, some nodes with high energy are selected as head-cluster and clustering is done [8] in order to solve the problem of re-clustering which results in high energy and time dissipation [9]. In this article, we try to improve the network efficiency and reduce energy consumption by an algorithm to recover faulty cluster head nodes and assign nodes to neighbor cluster heads. The structure of this article is as follows: in section 2 the previous algorithms are presented and in section 3 the proposed algorithms are described completely. In section 4 the proposed algorithm has been simulated. In section 5 the proposed algorithm has been evaluated and in section 6 the general conclusion has been presented.

2. Related Works

The sensors assigned to the faulty cluster-head are reorganized on the fly without bringing the system to a complete shutdown. The recovery information is created during clustering which facilitates the recovery process. Various types of observed communicative fault scenarios are solved during recovery. This method makes significant improvement in system stability and reduces the overhead of re-clustering and re-configuration. To place a node like S_j in a cluster-head, the following equation is used:

$$s_j \in RSet_{Gi} \leftrightarrow [(R_{Gi} > d_{sj-Gi}) \land (R_{sj,max} > d_{sj-Gi})] \quad (1)$$

Where, s_j is a node must be recovered, R_{Gi} is the communication range of cluster-head G_i, $R_{sj,max}$ is the maximum range of node s_j and d_{sj-Gi} is the distance between node s_j and cluster-head G_i.[10]

For solving this problem in [11], in choosing a cluster-head both distance from cluster-head, and remained energy of cluster-head considered. For a node to be a member of cluster-head, the nod should be satisfied in the following conditions.

$$R_{(ch_i)} > d_{(s_j,ch_j)} \Lambda \left(R_{(s_j)} > d_{(s_j,ch_j)}\right) \quad (2)$$

For which $R_{(ch_i)}$ is the range of cluster-head that recovered node connect to it, $d(S_j, ch_i)$ is the distance of nod S_j and cluster-head ch_i and ch_i is $cluster - head_i$ that node S_j is added to it. Among cluster-heads that satisfy in above conditions, a cluster-head is chosen that has the highest energy and the least distance.

$$MAX\left(\frac{E_{(ch_i)}}{d_{(s_j,ch_j)}}\right) \quad (3)$$

$E_{(ch_i)}$ is the energy of $cluster - head_i$.

3. Proposed Algorithm

If a fault occurs in cluster head, its members should be recovered to other cluster heads. We have done recovery of sensors of faulty cluster head with genetic algorithm. So, the chromosome which represents a solution should show that each member of cluster assigned to which cluster head. A chromosome in the proposed algorithm is a vector which has a size equals to the number of cluster members and the contents of each gene is one of the live cluster heads which could be obtained by generating a random number between 1 and the number of cluster heads. Figure 1 shows a sample chromosome in which for example the node 1 and 2 are recovered to cluster heads 8 and 6 respectively.

1	2	3	4	5	6	7	8
8	6	7	12	15	4	12	7

Fig. 1 A sample chromosome

After fault occurring in cluster head, its nodes should be recovered. The number of faulty cluster head nodes determines the length of each chromosome and the population size that is specified at the first number, chromosomes are produced whose gene contents is a random number between 1 and the number of cluster heads. After producing the primary population, the quality of each chromosome should be evaluated, this is done by fitness function. Fitness function considers three parameters to recover each node to cluster heads where these parameters include the distance of a node being recovered to given cluster head, remained energy of cluster head and the member number of a cluster head for which the current node would be recovered. Figure 2 shows an example of cluster head failure and its recovery by using distance, energy and members parameters.

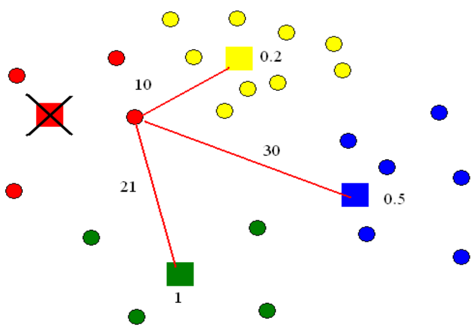

Fig. 2 Cluster head failure and the way of evaluating the fitness of cluster heads

So the primary fitness function could be presented as equation (4).

Fitness (I) = distance (I, Ch) + energy (Ch) + members (Ch) (4)

Assuming that the yellow, blue and green cluster heads have number 1, 2 and 3 respectively, by using equation (4) we have:

Fitness (1) = 10 + 0.2 + 9 = 19.2
Fitness (2) = 30 + 0.5 + 6 = 36.5
Fitness (3) = 21 + 1 + 4 = 26

As it can be seen, those cluster heads have more distance and members, obtain more fitness so that in order to prevent this we enter distance and members' number into equation inversely and equation (4) is converted to equation (5).

Fitness (I) = (1/distance (I, Ch)) + energy (Ch) + (1/members (Ch)) (5)

Another time, we calculate values with equation (5) in order to determine the impact of inversing parameters.

Fitness (1) = 0.1 + 0.2 + 0.12 = 0.42
Fitness (2) = 0.034 + 0.5 + 0.167 = 0.701
Fitness (3) = 0.047 + 1 + 0.25 = 1.297

The used selection mechanism in this algorithm is Tournament selection mechanism in which k chromosomes are selected randomly and among them a chromosome with the highest value of fitness is selected and transmitted to crossover pool where in this algorithm the value of k has been set to 5.

The crossover operator is one of the genetic operators combine two chromosomes in order to produce two better offspring from combination of two good parents. There are different types of crossover operators like one-point crossover, two-point crossover and multi-point crossover where in this algorithm two-point crossover with crossover possibility of 0.8 has been used which means that crossover is done in 80 percent of events and it is not done in 20 percent of event and it is implemented in a way that first a random number between 0 and 1 is produced, if this value is less than crossover possibility, crossover is done, otherwise the crossover operator is not done on chromosome. Two point crossover acts such that first it selects two random numbers which are representative of a cut point in the chromosome. Then the genes of this range are displaced in each two chromosomes as shown in figure 3.

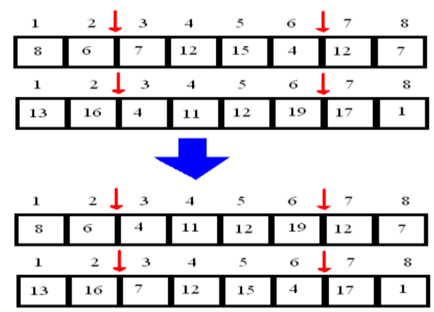

Fig. 3 Crossover operator

Mutation is done with 0.5 possibilities in a way that random number between 0 and 1 is produced, and then this value is compared with mutation possibility, if it is less than 0.5, mutation is done and otherwise, it is not done. Mutation acts such that first a random number between 1 and the number of cluster heads is produced, then this value is placed in a random cell within chromosome as shown in figure 4.

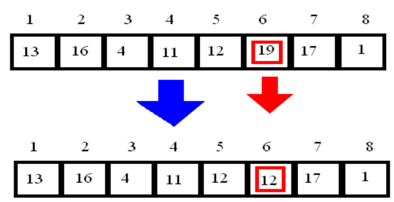

Fig. 4 Mutation operator

There are various methods to determine the termination condition. Another method is that iterations continue until reaching to stagnation which means that if there is not any improvement after some subsequent iteration, iterations are stopped, and otherwise they would continue. In the presented algorithm this method has been used and if in 25 subsequent iterations there is not any improvement, it stops and the best chromosome is presented as an output of algorithm.

4. Simulation and Results

Matlab software is used for simulation and a 200x200 m area has been considered where 400 nodes are spread randomly. Each node can communicate with all the other nodes and it has 1 Joule energy which can sense its two meter radius. Also, 20 cluster heads are spread in the area which has 2 Joule energy. In this simulation it has been assumed that in a random period one of the cluster heads which is selected randomly stops, then the restoration process of nodes of that cluster is done by using the presented algorithm. Figure 5 shows the given area, the placing way of nodes and cluster heads and also faulty cluster head and its members which should be recovered.

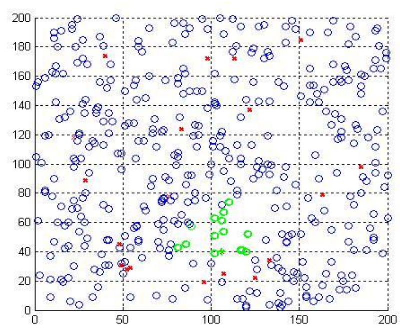

Fig. 5 The placement way of nodes, faulty cluster head and its members in the area

The presented genetic algorithm could improve the fitness amount, as figure 22 shows at first the produced solutions by primary population have fitness about 34 and over time and performing iterations this value has been increased and reached to 39 which obtained better improvement.

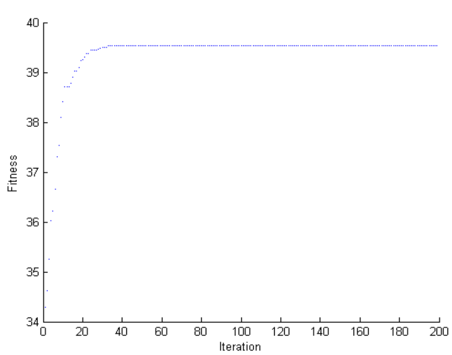

Fig . 6 Fitness amount over 200 iterations

Efficiency evaluation: the average energy of nodes in the presented genetic algorithm has been compared with Gupta algorithm (Gupta, 2003) so that the result of this comparison is shown in figure 7. As figure shown the presented genetic algorithm has better efficiency and energy consumption is more balanced and less energy is wasted.

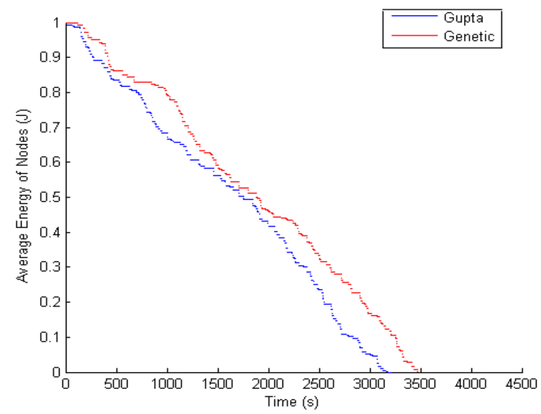

Fig. 7 Comparison of average energy of nodes in Gupta algorithm and genetic algorithm

5. Conclusions

Usually, nodes in wireless sensor networks are classified to classes called cluster so that each cluster has one cluster head. The task of cluster head is to taking data from member nodes, collecting and sending them to base station and if this cluster head collapses all the cluster members encounter with problem, so a mechanism to recover faulty cluster head nodes should be considered. In this paper, a cluster nodes recovery algorithm based on genetic algorithm has been presented which works multi-purposely and it can consider recovery of each node based on distance of each node from cluster heads, remained energy of cluster heads and member number of each cluster head and selects the best cluster head to recovery. Simulation results show that this algorithm has better efficiency than a state considered only one of the parameters. Also, this method presents 9.81 percent improvement in energy consumption compared with previous works.

References

[1] Therence.Houngbadji,Samuel9Pierre" QoSNET: An integrated QoS network for routing protocols in large scale3 wireless sensor networks"Computer Communications (2010)

[2] A.Habibizad Navin, B. Asadi, S. Hassan pour, M. Mirnia"Solving Coverage Problem in Wireless Camera-based Sensor Networks by using Genetic Algorithm"International Conference on Computational Intelligence and Communication Networks, 978-0-7695-4254-6/10 $26.00 © IEEE, 2010

[3] Yanjun Li, Chung Shue Chen, *Member, IEEE*, Ye-Qiong Song, Zhi Wang, *Member, IEEE*, and Youxian Sun" Enhancing Real-Time Delivery in Wireless Sensor Networks With Two-Hop Information" IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, VOL. 5, NO. 2, MAY 2009

[4] Dechene.D. J, El Jardali.A, Luccini. M and Sauer. A, A Survey of Clustering Algorithms for Wireless Sensor Networks, Department of Electrical and Computer Engineering the University Of Western Ontario London, Ontario, Canada.

[5] J.Yick, B.Mukherjee,D.Ghosal, "Wireless sensor network survey",coputer networks 52(2008) 2292-23300.

[6] X.Zhang,F.Georg, "An On-Demand Bluetooth Scartter net Formation and routing protocol for wireless sensor network",Department of ECE Georgia Institute of Technology Atlanta, GA 30250-30332, 23-25 May 2005 IEEE.

[7] F.Gao, M.E, " A hybrid topological-stochastic partitioning method for scaling QoS routing algorithms", Department of Computering, university of Bradford,Rchmond Road,2009.

[8] Heinzelman, W., Chandrakasan, A., & Balakrishnan H.(2000). Energy Efficien Communication protocol for Wireless Microsensor network, *Proceedings of the 33rd Annual Hawaill International Conferenc on System Sciences*, USA: IEEE Computer Society.

[9] Khadivi, A., Shiva, M. (2006).A Fault Tolerant Power Aware Protocol With Static Clustering for Wireless Sensor Networks, *IEEE*.

[10] Gupta, G., & Younis, M., (2003). Fault–Tolerant Clustering of Wireless Sensor Networks, *IEEE*.

[11] Moghaddami K, E., Nematy, F., & Haghparast M. (2011). Increasing the reliability in wireless sensor networks by node's recovery mechanism in defective clusters, *Australian Journal Of Basic and Applied Sciences ISSN*.