FNR Algorithm for Wireless Sensor Networks using Particle Swarm Optimization Algorithm

Tathappa Rachannanavar[1] Prof. Hanumanthappa S.N[2]
[1]P.G. Student [2]Assistant Professor
[1,2]Department of Electronics & Communication Engineering
[1,2]UBDT College of Engineering Davanagere, Karnataka, India

Abstract— Wireless Sensor Networks are the wireless networks contain spatially speckled autonomous devices (sensors) to monitor the physical conditions and ecological parameters. Nodes in WSNs (Wireless Sensor Networks) are very prone to failures due to energy depletion, defective hardware and environmental conditions. Fault tolerance is one of major issue in WSNs. This paper proposes a fault node recovery algorithm, which will improve the life time of WSN when nodes shut down. The FNR (fault node recovery) algorithm proposed here is the combination of Grade Diffusion algorithm and Particle Swan Optimization algorithm which will detect and recover the fault node in most efficient way and replace the sensor nodes with more reused routing paths and also increases the number of active nodes. Reduce the rate of data loss with reduced energy consumption.

Key words: Wireless Sensor Networks, Grade Diffusion algorithm, Dynamic Source Routing (DSR) Protocol, Particle Swarm Optimization algorithm

I. INTRODUCTION

Wireless Sensor Network (WSN) is a wireless network which got significant and wide range of applications so that gained global attention in recent years. Contemporary improvements in micro processing, wireless technology and smart sensors, particularly the explosion in Micro Electro Mechanical Systems (MEMS) technology enhanced data processing.

A WSN often comprises nodes equipped with sensors, processors, and communication devices such as short-range transceivers over wireless channels. These nodes may be distributed over a large area. These nodes interact with the environment by sensing or controlling physical parameters. Which have to cooperate with the neighbor nodes by using wireless communication to achieve their tasks, because a single node cannot do the task alone. The network without such cooperative environment performs least computation, wireless communication, and sensing or control functionalities.

Hence a node with the sufficient energy has to relay information to the destination, and if relay repeats involving same node results in shut down after reaching its operational threshold. And also here WSN has restricted amount of energy which cannot be replaced. Hence the energy optimization got much importance so here I am going to use some techniques to optimize the energy usage of WSNs.

Application areas for the paper are Remote Environmental Monitoring, Biomedical Health Monitoring, Military Target Tracking, Surveillance, and Hazardous Environment Exploration and Seismic Sensing.

II. EXISTING SYSTEM

Many techniques have been proposed till now for fault node detection and recovery.

Muhammad Asim et al. [1] proposed systematic structure properly distributed fault management jobs between sensor nodes by introducing additional 'self-managing' functions. This could carry out failure detection and recovery much quicker than other present schemes and consumes significantly lower energy.

Abolfazl et al. [2] Presented clustering algorithm which creates the routing for each sensor node but also identifies a set of neighbor nodes to reduce the transmission loading. Each sensor node can select a sensor node from the set of neighbor nodes when its grade table lacks a node able to perform the relay.

Ehsan et al.[3] proposed an efficient method based on genetic algorithm to solve optimization problem using clustering protocols for ex LEACH, TEEN.

Song Jia et al. [4] Proposed a Recovery Algorithm based on Minimum Distance Redundant Nodes (MDRN). By using redundant nodes wisely, the recovery algorithm is developed on the sink node with free energy consumption.

Hong-chi et al. [5] projected a fault node recovery (FNR) algorithm which is based on grade diffusion algorithm and genetic algorithm. This FNR algorithm creates grade values and payload values along with mapping of routing table and neighbor nodes. By using this information calculates fitness value of the node which describes the active and inactive nodes. Genetic algorithm replaces those inactive nodes resulting in the recovery of fault node.

In wireless sensor networks (WSN), every sensor node has restricted wireless computational power to process and to transfer the live data to the destination and also the battery cannot be replaced. When the energy of a sensor node exhausts WSN trickles will occur due to the failed sensor node. Hence there will be no transmitting data to the other nodes. So sink node finds alternative path for transmission, that path used repeatedly by other nodes. Thus, those sensor nodes will be suffered with increased transmission load and processing.

III. PROPOSED SYSTEM

This paper proposes a fault node recovery (FNR) algorithm to improve the lifespan of a wireless sensor network (WSN) when certain sensor nodes shut down, either because they no longer have battery energy or they have reached their operational threshold. FNR algorithm replaces fewer sensor nodes and result in more reused routing paths.

The algorithm projected here is based on the grade diffusion algorithm combined with the particle swarm optimization algorithm which is shown in figure.3.1.

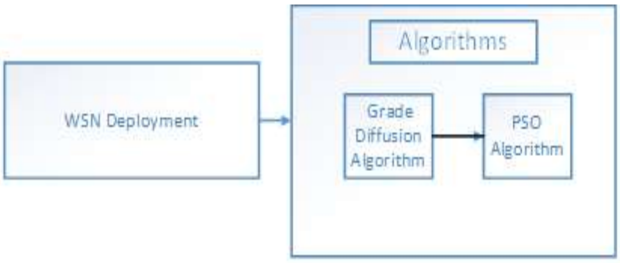

Fig. 3.1: General Block diagram

Construction of WSN is nothing but developing nodes. Which has to meet the requirements of applications hence called application oriented and node should be small, cheap and energy efficient.

IV. SYSTEM IMPLEMENTATION

System Implementation is important stage where the theoretical design is turned into a proper working system. Thus it is considered to be the most critical stage in achieving a successful new system and giving confidence to the user that the new system works effectively.

Designing of system here includes an algorithm for WSNs based on the Grade Diffusion Algorithm combined with the particle swarm optimization algorithm.

A. *Grade Diffusion (GD) Algorithm:*

The GD algorithm presented by H. C. Shih et al.[8] is the enhancement for Ladder Diffusion Ant Colony Optimization Algorithm (LD-ACO) for wireless sensor networks[7].

1) *Dynamic Source Routing (DSR) Protocol:*

When the GD algorithm is applied, it invokes DSR protocol, this broadcasts query packets to the neighbor nodes finding the grade values which is the collection of node information. Then, the sensor nodes transfer this grade information (Event data) to the sink node.

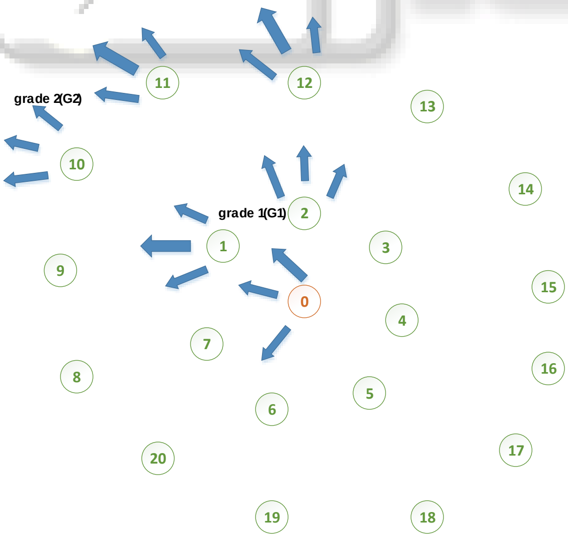

Fig. 4.2: Broadcasting to find Grade Values.

Grade values gives the information about which node is functioning and non-functioning. Here the functional node is the node which is in perfect working condition with sufficient energy and non-functional node is the node which has been reached their operational threshold.

By using these grade values one has to make the decision of which node has to be replaced.

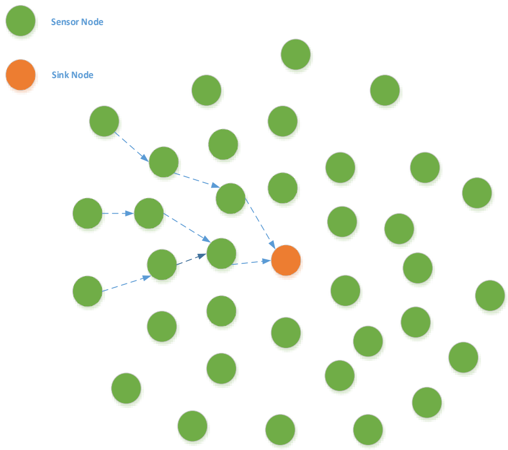

Fig. 4.2: Operation with functional node

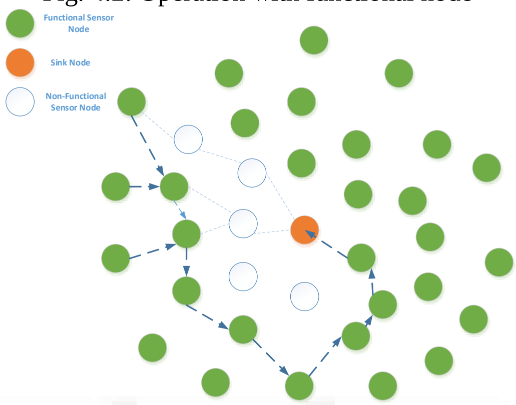

Fig. 4.3: Operation with non-functional node

The node parameters such as grade values, routing table, set of neighbor nodes and payload value for each sensor node recoded by GD algorithm are encoded in binary string and serve as the chromosomes for the further process. The structure of encoded chromosome is shown in figure.

Index values of nodes

8	15	20	17	3	14	20	7	
0	0	1	1	1	1	1	0	G1

0 Represents non-functional node

1 Represents functional node

Fig. 4.4: Encoded Binary String

Here by considering the output from GD algorithm decision has to be made. Here B_{th} is calculated according to (4.1). If B_{th} is larger than zero, the algorithm is invoked and replaces nonfunctioning sensor nodes by functional nodes.

$$B_{th} = \sum_{i=1}^{\max\{Grade\}} T_i$$

$$T_i = \begin{cases} 1, & \dfrac{N_i^{now}}{N_i^{original}} < \beta \\ 0, & \text{otherwise.} \end{cases} \qquad \text{........ 4.1}$$

Where

Grade - sensor node's grade value.

$N_{original}$ - number of sensor nodes with the grade value i.

N_{now} - number of sensor nodes still functioning at the current time with grade value i.

The factor β is set by the user and must have a value between *0* and *1*.

If the number of sensor nodes that function for each grade is less than β, T_i will become 1, and B_{th} will be larger than zero. Then, the algorithm will calculate the sensor nodes to replace.

B. Particle Swarm Optimization (PSO) Algorithm:

Particle swarm optimization is an experimental universal optimization technique originally developed by Doctor Kennedy and E Berhart [9].

In PSO algorithm each particle keeps track of its coordinates in the solution space which are associated with the best solution (fitness) that has achieved so far by that particle. This value is called personal best, ***pbest.*** Another best value that is traced by the PSO is the best value found so far by any particle in the neighborhood of that particle. This value is called ***gbest.*** The basic concept of PSO lies in accelerating each particle toward its pbest and the gbest locations with a random weighted acceleration at each time step as shown in Figure 3.6.

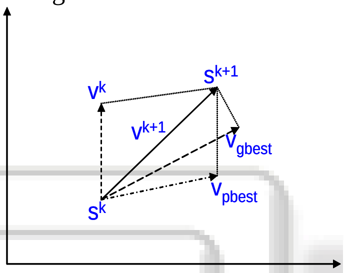

Fig. 4.5: Concept of modification of a searching point by PSO

Where

s^k - Current searching point.

s^{k+1} - Modified searching point.

v^k - Current velocity.

v^{k+1} - Modified velocity.

v_{pbest} - Velocity based on pbest.

v_{gbest} - Velocity based on gbest.

Each particle tries to modify its position using the following information:

- The current positions,
- The current velocities,
- The distance between the current position and pbest,
- The distance between the current position and the gbest.

The modification of the particle's position can be mathematically modeled according the following equation:

$$V_i^{k+1} = wV_i^k + c_1\,rand_1(\ldots) \times (pbest_i - s_i^k) + c_2\,rand_2(\ldots) \times (gbest - s_i^k) \ldots\ldots \; 4.2$$

Where

v_i^k - velocity of agent i at iteration k.

w - Weighting function.

c_j - Weighting factor.

r and -Uniformly distributed random number between 0 and 1.

s_i^k - Current position of agent i at iteration k,

pbest$_i$ - pbest of agent i.

gbest - gbest of the group.

The following weighting function is usually utilized in 3.1

$$w = w_{Max} - [(w_{Max} - w_{Min}) \times iter]/maxIter \ldots\ldots \; 4.3$$

Where

w_{Max} - Initial weight.

w_{Min} - Final weight.

maxIter - Maximum iteration number.

iter - Current iteration number.

$$s_i^{k+1} = s_i^k + V_i^{k+1} \ldots\ldots \; 4.4$$

A large inertia weight (w) facilitates a global search while a small inertia weight facilitates a local search. By linearly decreasing the inertia weight from a relatively large value to a small value through the course of the PSO run gives the best PSO performance compared with fixed inertia weight settings.

- Larger ***w*** means greater global search ability.
- Smaller ***w*** means greater local search ability.
  - Unlike in Genetic Algorithms, there is no selection operation in PSO.
  - PSO is the only algorithm that does not implement the survival of the fitness.
  - Equation 4.3 resembles mutation.
  - PSO algorithm here used to recover the fault node of WSN.

There are 2 steps in the PSO algorithm crossover and mutation.

C. Crossover:

The crossover step is used in the PSO algorithm given by equation 1.1 is to change the individual chromosome. In this algorithm, two distinct chromosomes are selected from the mating pools shown in fig 3.7. Which are encoded by the GD algorithm of having comparably higher grade values to produce two new offspring. A crossover point is selected between the first and last genes of the maternal individuals this is shown in figure 3.8. Then, the portion of each individual on either side of the crossover point is swapped and concatenated i.e. shown by fig 3.9.

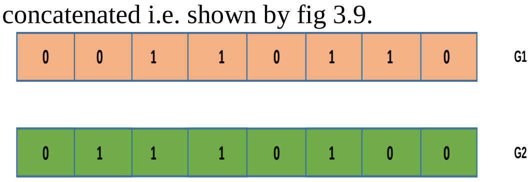

Fig. 4.6: strings selected from mating pool

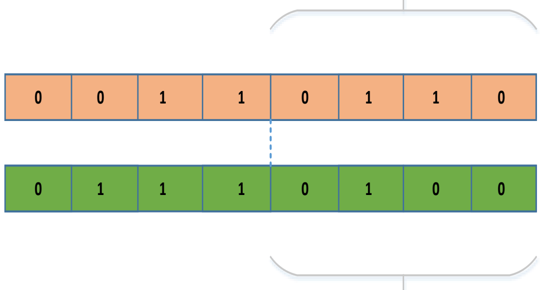

Fig. 4.7: Portion selected for Crossover

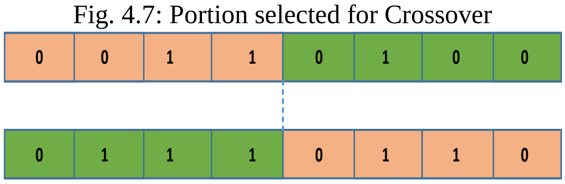

Fig. 4.8: Crossover

After crossover will get two new offspring's, among them one chromosome is selected which is having highest grade value. This implies that numbers of functional nodes are more.

0	1	1	1	0	1	1	0

Fig. 4.9: New chromosome generated after Crossover

New chromosome generated by iterations i.e. figure 3.10 shows that the resultant node contains more number of functional nodes compared to string which is taken as maternal.

D. *Mutation:*

The mutation step can introduce traits not found in the original entities by flipping gene arbitrarily in the chromosome, as shown in Fig.

0	1	1	1	0	1	1	0

0	1	1	1	1	1	1	0

Fig. 4.10: Mutation

The chromosome with the best fitness value is obtained as the solution after this iteration.

0	1	1	1	1	1	1	0

Fig.4.11. Node Replaced by Mutation

1) *Flow chart:*

The flow chart is depicted in Fig.4.12 which shows the flow of the entire system.

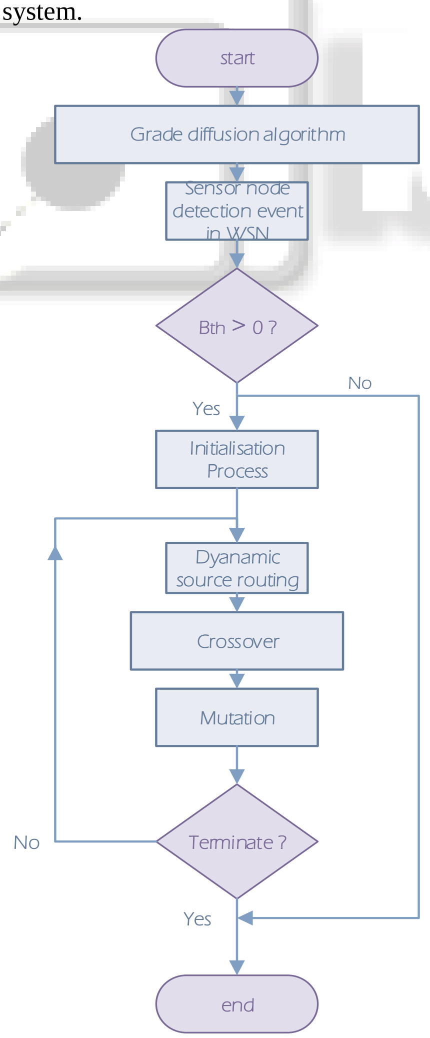

Fig. 4.12: Flow Chart

2) *Simulation:*

ns2 is used for simulation purpose which is a diverse event driven simulator developed at UC Berkeley. It is part of the VINT project. Supports networking research and education. It is suitable for designing new protocols, comparing different protocols and traffic estimations. . It is open source.

V. RESULTS AND DISCUSSIONS

The constructed FNR algorithm results in the recovery of faulty node. Each iteration of operation is shown in the following screen shots. Which are taken from the nam window and trace graph of network simulator.

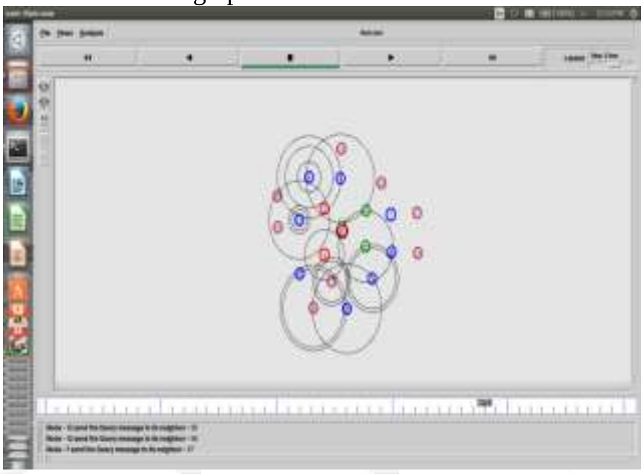

Fig. 5.1: Communication in Presence of Fault Node

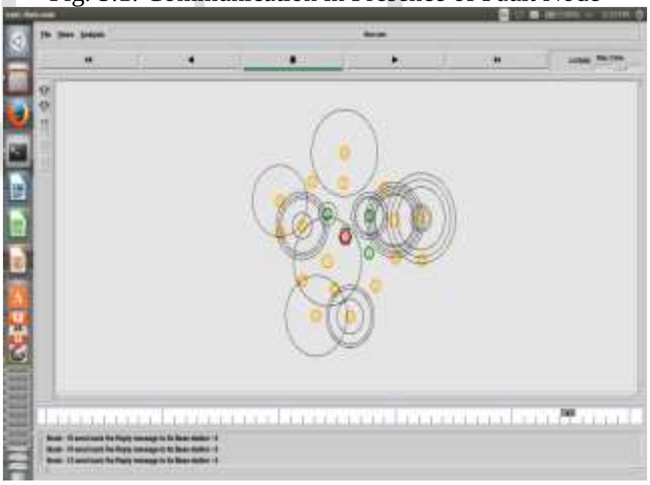

Fig. 5.2: Fault Nodes Recovered

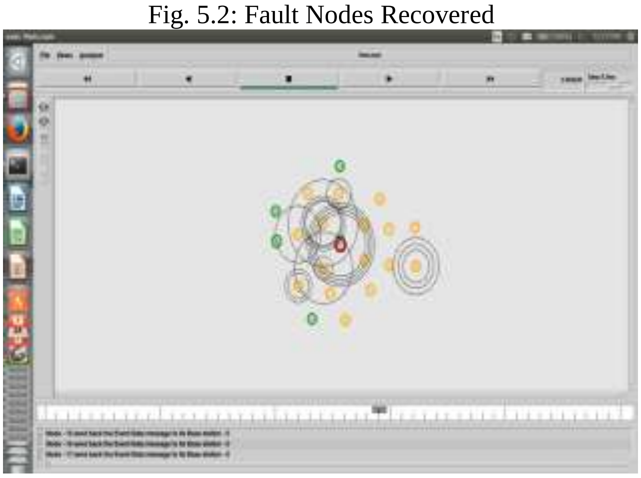

Fig. 5.3: Normal Communication of Nodes after Recovery

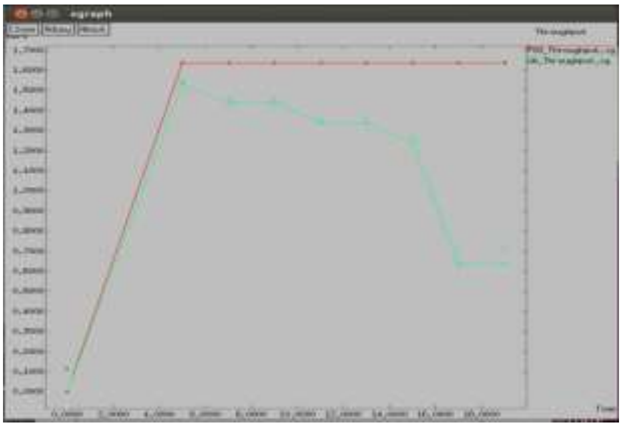

Fig. 5.4: Comparison of Throughput

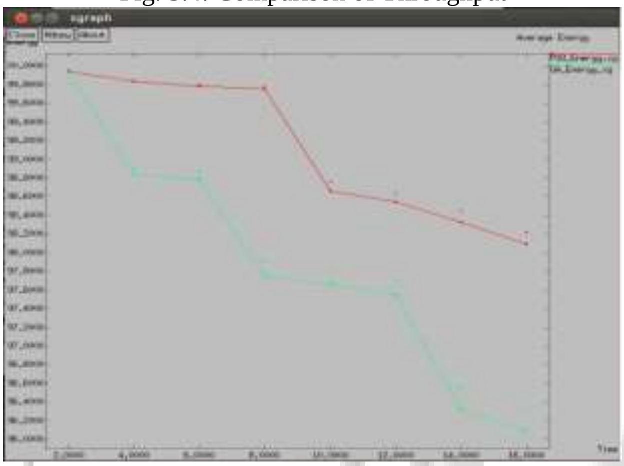

Fig. 5.5: average Energy of PSO algorithm compared with GA algorithm

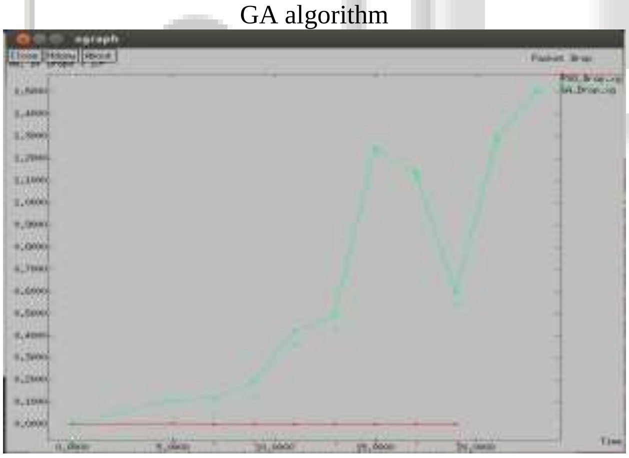

Fig. 5.6: Packet Drop

VI. CONCLUSION

This paper proposed an FNR algorithm for WSN based on the grade diffusion algorithm combined with a PSO algorithm. This optimized the replacement cost, energy consumption, throughput and packet drop of sensor nodes also replaced non-functional sensor nodes and reuses the most routing paths, resulting in increased WSN lifespan.

VII. FUTURE SCOPE

In WSN the life span of the network is also increased by employing Enhanced Grade Diffusion algorithm or Ant Colony Optimization (ACO) algorithm and recharging the battery can be also tried with energy gathered from the environment, solar cells or vibration based power generation. This possible option is known by the name "Energy Scavenging".

REFERENCES

[1] Muhammad Asim, Hala Mokhtar and Madjid Merabti,"A self-managing fault management mechanism for wireless sensor networks" in proc of IJWMN Vol.2, No.4, November 2010.
[2] Abolfazl Akbari1, Arash Dana, Ahmad Khademzadeh and Neda Beikmahdavi," Fault Detection and Recovery in Wireless Sensor Network Using Clustering" in proc of IJWMN Vol. 3, No. 1, February 2011.
[3] Ehsan Heidari and Ali Movaghar, "An efficient method based on genetic algorithm to solve sensor network optimization problem" in proc of International journal on applications of graph theory in wireless ad hoc networks and sensor networks, Vol.3, No.1, March 2011.
[4] Song Jia,Wang Bailing, Peng Xiyuan, Li Jianfeng and Zhong Cheng, "A Recovery Based on Minimum Distance redundant Nodes Fault Management in WSNs" in proc of International Journal of Control and Automation, Vol. 6, No. 2, April, 2013.
[5] Hong-Chi Shih, Jiun-Huei Ho, Bin-Yih Liao and Jeng-Shyang Pan, "Fault Node Recovery Algorithm for Wireless Sensor Networks" in proc of IEEE Sensors Journal Vol:13 No 7 , 2013.
[6] A.Manikandan, S.Rathinagowri, "An Efficient Detection and Recovery of Fault node in Wireless Sensor Networks" in proc of IJIRSET, Volume 3, Special Issue 1, February 2014.
[7] Jiun-Huei Ho, Hong-Chi Shih, Bin-Yih Liao, Shu-Chuan Chu, "A ladder diffusion algorithm using ant colony optimization for wireless sensor networks," in proc of Information Sciences 192, 204–212, April 2011.
[8] Jiun-Huei Ho, Hong-Chi Shih, Bin-Yih Liao, jeng-shyang pan, "Grade diffusion algorithm", in proc of applied mechanics and materials, vol 284-287, 2013.
[9] Qinghai Bai," Analysis of Particle Swarm Optimization Algorithm", in proc of computer and information science,vol 3,no 1,feb 2010.
[10] Mani Subramanian, network management principles and practice, Pearson, 2nd edition, 2011.
[11] Holger Karl and Andreas Willig, Protocols and Architectures for Wireless Sensor Networks, Wiley, 2005.
[12] Kazem Sohraby, Daniel Minoli and Taieb Znati, Wireless Sensor Networks Technology, Protocols and Applications, Wiley, 2007.

