Redundancy Effect on Fault Tolerance in Wireless Sensor Networks

A. Mojoodi[α], M. Mehrani[Ω], F. Forootan[β], R.Farshidi[Ψ]

Abstract- there are several usages for wireless sensor networks in military, supervisory, region considering, physiology, etc. Sensor nodes are usually equipped with some non chargeable batteries having limitations in lifetime while they cannot be replaced with other sensors when they fail. Each sensor has a failure probability which is affected from some factors like electrical energy, hardware failures, communication error, undesired environment situations, etc. Thus, fault tolerant is a very important and critical factor in such networks. Hardware redundancies like having redundant nodes and paths are suitable echniques used for increasing fault tolerant factor. On the other hand, using hardware redundancies leads increasing in overall network consumption because of using more number of sensors. In this paper we estimate the affect of redundancy on the number of correct responses that wireless sensor networks have on the received queries and also show the level of redundancy needed in different network conditions like having different fault probabilities and needed clusters to response the received queries. Keywords: wireless sensor network, fault tolerance, redundancy, reliability

I. Introduction

Recent advancements in integrated circuits have fostered the emergence of a new generation of tiny, inexpensive and low power sensors. Due to their economic and computational feasibility, a network of hundreds and thousands of sensors has the potential for numerous applications in both military and civil applications such as combat field surveillance, security and disaster management. These sensing devices are capable to monitor a wide variety of ambient conditions such as: temperature, pressure, motion etc. The sheer number of these devices and their ad-hoc deployment in the area of interest brings numerous challenges in networking and management of these systems.

Sensors are typically disposable and expected to last until their energy drains. Therefore, energy is a very scarce resource for such sensor systems and has to be managed wisely in order to extend the life of the sensors for the duration of a particular mission. Sensors send the received data to a base station. This can be done periodically or according to an event. The base station is located faraway from the area where the sensors are usually deployed. In order to conserve energy consumed in communication with the command node various multi-hop and energy aware routing techniques have been suggested in the literature [5][6]. Considering the application of sensor networks, fault tolerance is very important for them. This importance can be observed in some applications such as military environments and nuclear experiment obviously.

Fault tolerant is a characteristic of the network which leads the network to continue its functionality when an error occurs by covering the affect of occurred error, such that we can make sure about the service presented by the network. Since in some sensor network applications receiving correct data by base station is a very important factor and decision making based on non correct data leads huge damages, fault tolerant in sensor networks has an extreme importance. Fault tolerant is the ability of maintaining the proper usefulness of the sensor network without having any lacks made by network problems. [4]

Using hardware redundancies like applying more sensor nods and more data transmitting paths for recognizing and covering occurred faults are some techniques used to achieve at a fault tolerant wireless sensor network. [5-8]

Using hardware redundancy increases overall network consumption because of applying more number of sensor nodes to send and receive data packets, but in other hand, it increases network fault tolerant and reliability.

By paying attention to mentioned notes, implementing hardware redundancy can be useful against sensors faults and transmitting errors but it should be used just in critical situations to avoid network overall energy consumption.

II. Faults and Hardware Redundancies

The occurred faults in wireless sensor networks have two reasons. First, since the sensor networks are dispersed in a region such that replacing the sensors is very hard or even impossible, losing energy by sensor nodes is one of the main reasons of faults. The network consumes its energy to do its duty.

About [α, β, Ψ] - Islamic Azad University, Dezful Branch, Dezful Iran
E-mail- mojoodi@iaud.ac.ir, forootan@iaud.ac.ir, farshidi@iaud.ac.ir
About [Ω]-Islamic Azad University, Shoushtar Branch, Shoushtar, Iran
E-mail- m.mehrani@iau-shoushtar.ac.ir

April 2011

Global Journal of Computer Science and Technology Volume XI Issue VI Version I

The second main reason is the sensor problems like hardware lacks, software lacks or data transmitting lacks made by noises or having interferes in the network. Also, having changes in binary data packets during processing leads data inconsistency.

There are some different hardware redundancies to avoid the mentioned possible faults. Using the following strategies increases fault tolerant factor for WSN.

The first strategy is path redundancy. This technique instead of using a single path to connect a source cluster to the processing center, utilizes m disjoint paths between them. Node redundancy is the second strategy. In this strategy, instead of having one sensor node in charge of returning requested data, ms sensor nodes within a source cluster are used. This is done to cope with sensor or transmission faults.

In the next part we estimate using of the mentioned hardware redundancies in a given sensor network and try to calculate the fault tolerance degree, network energy consumption and the number of correct reactions to the number of queries.

III. How to Apply Redundancy in WSN

Both of the average energy consumption of each request (Eq) and its average safety factor (Rq) are dependent on the redundancy level presented by the network to reply a request. The magnitude of Eq and Rq largely depends on the redundancy level used by the system to reply the incoming request.

Without any redundancy, both E_q and R_q are decreased. But using excess redundancy leads the same increment in both E_q and R_q. So, a trade off is required between these two parameters.

Like [1] we consider a wireless sensor network with a certain number of nodes dispersed in a non accessible and dense region. All the sensor nodes have same initial energy. Also a clustering algorithm like LEACH [10] or HEED [9] is used for partitioning the nodes in different groups. Obviously the clustering method denotes a cluster head for each cluster to manage the cluster activities during each round of network functioning.

Usually, the network nodes are equipped by low power batteries without chance if recharging, so they use multi hop data transmission to save energy.

Let q_t to be the failure rate of data transmission caused by noise or interfere while q_s represents the sensor fault.

Assume that the user has a query. Therefore, the processing center is called to send this request to cluster head. Each user query may involve one cluster, some of them or even all the network clusters in responding the issued query. When a query calls for a cluster, one sensor is sufficient to answer the query. It can transmit its collected information from environment to processing center through a path.

To achieve at fault tolerance, the chosen sensor nodes for receiving region information can send information to data processing center via m separate paths, instead of sending them just to the cluster head. Based on this data transmission technique, considering the probability of data transmission faults and also by paying attention to probability of sensor faults we can make sure about calculating the reliability and energy consumption of the network.

IV. Reliability and Energy Consumption

In this section we calculate the network energy consumption and network reliability factors for the network which is equipped by some redundant ensors to achieve at hardware redundancy.

Assume d as a random variable which determines distance between a sensor and processing center. Thus, the number of hops between the processing center and the source sensor, denoted by h, is given by: h=d/r-1 [1]. Like [1] let the source cluster head be randomly located at (Xi, Yi) in the square shape sensor area when -A/2 ≤ Xi ≤ A/2 and- A/2 ≤ Yi ≤ A/2 and the processing center is located at the center of the network area and its coordinates is (0, 0). Then, E[h] which is the expected value of *h* is given by:

$$E[h] = \int_{-A/2}^{A/2}\int_{-A/2}^{A/2}\left(\frac{\sqrt{(X_i^2+Y_i^2)}}{r}-1\right)\left(\frac{1}{A}\right)\left(\frac{1}{A}\right)dX_i\,dY_i = \frac{0.3825A}{r}-1 \tag{1}$$

For notational convenience, let .Here N_h represents the average number of hops (or sensors) to forward sensor data from a source sensor to the processing center.

A sensor will fail to return its reading to the cluster head when any hop fails, so the failure probability of that source cluster in delivering data to the processing center due to sensors fault is given by:

$$P_s = 1-(1-q_s)^{N_h+1} \tag{2}$$

Also, the failure probability of that source cluster in delivering data to the processing center due to transmissions fault is given by:

$$P_t = 1-(1-q_t)^{N_h} \tag{3}$$

Now, suppose there is just one simple path from the source sensor to the processing center and because of sensor and transmission failures the process of sending data from the source cluster to the processing center may encounter with a failure. Therefore, the probability of the mentioned possible failure is calculated as following:

$$P_f = 1-(1-p_t)\ (1-p_s) \qquad (4)$$

If a sensor uses m separated paths to return its reading hen the failure probability is given by:

$$P_{fm} = (p_f)^m \qquad (5)$$

Suppose that the application demands k source clusters to return sensor data to reply a query, then the failure probability is given by:

$$P_{fkm} = 1-(1-p_{fm})^k \qquad (6)$$

Therefore, the reliability of a query that requires k clusters to respond is given by:

$$R_{qm}(k) = 1 - P_{fkm} \qquad (7)$$

Now energy consumption in this process is to be estimated and calculated as following. The energy used for communication is denoted as Eelec per bit. Thus, the energy spent by a sensor node to sense (or to receive) and transmit a data packet of length nb bits is given by:

$$E_{packet} = 2 * n_b E_{elec} \qquad (8)$$

m redundant paths transmit received data from a sensor to the data processing center, separately. In this situation, if a request needs k clusters and each one uses m redundant paths to reply the request then the overall needed energy for the sensors placed in these K clusters to send data to the processing center can be calculated as $E_{packet} *N_h *k *m$. Thus, the amount of energy spent by the system, Eq(k), to answer a query that demands k clusters to respond is given by:

$$Eq(k) = E_{packet} * N_h * k * m \qquad (9)$$

The average number of queries that the system is able to sustain before running out its energy is given by:

$$N_q = \frac{E_{initial} - E_{threshold}}{Eq} \qquad (10)$$

Let the average reliability of a query be Rq , given as:

$$R_q = \sum_{k=1}^{np} R_q(k) P_q(k) \qquad (11)$$

If the system is able to reply Nq requests with reliability factor Rq then the number of the requests that can be replied correctly before system loses its energy can be used as a scale to measure the network life time.

V. Numeric Results

Assume a wireless sensor network characteristics: n = 1000 nodes, r = 1, J = 10 nodes/square unit, A = 10 units, nb = 50 bytes, Eelec = E0 = 2J, Ethreshold = 0. We want to count the number of requests that network can reply correctly in different conditions.

In the first phase with assumption that network needs just one cluster to reply each request and has different fault probabilities while there isn't any redundancy in the network we calculate the number of correct replies. Then we use two, three and four redundant paths to send data and calculate the number of correct replies in each mentioned situations. The results are shown in figure 1.

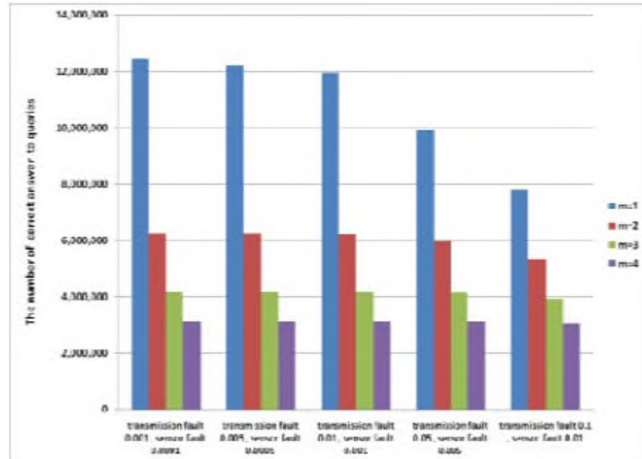

Figure 1. The number of correct answers when network requires one cluster to answer

In the next phases we assume the data transmission fault rate between 0.001 and 0.1, also we suppose the sensor faults rate between 0.0001 and 0.001 (less than data transmission fault rate [1]), then we calculate the number of correct replies of the system.

The number of needed clusters to reply the requests is chosen from k=1 to k=50 and then the calculations for the situations that the number of redundant paths are zero, two, three or four will be repeated. The results are shown in figure 2 to figure 6.

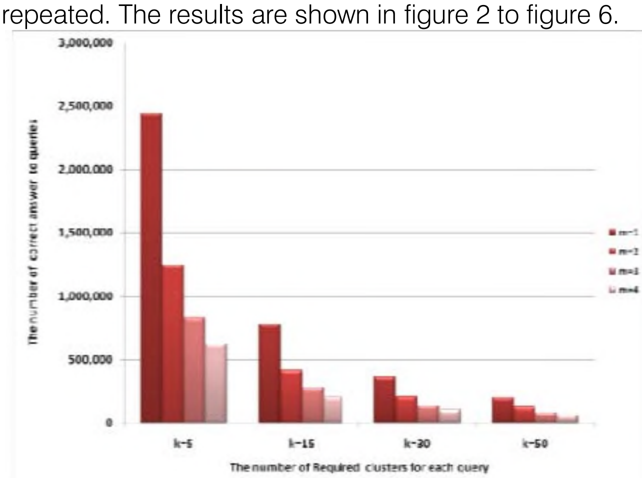

Figure 2. The number of correct answers when transmission fault is 0.001 and sensor fault is 0.0001

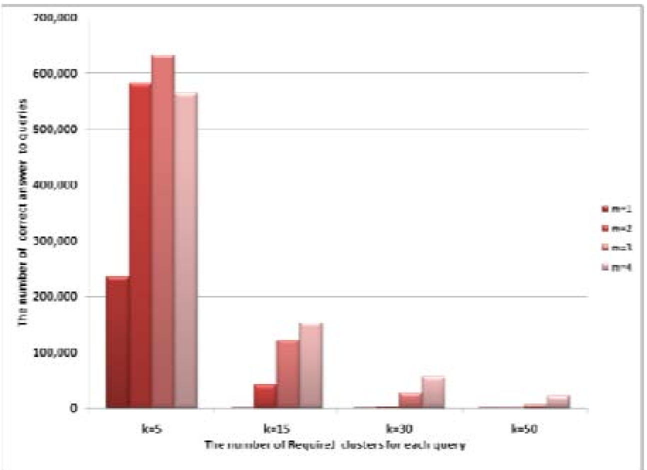

Figure 3. The number of correct answers when transmission fault is 0.005 and sensor fault is 0.0005

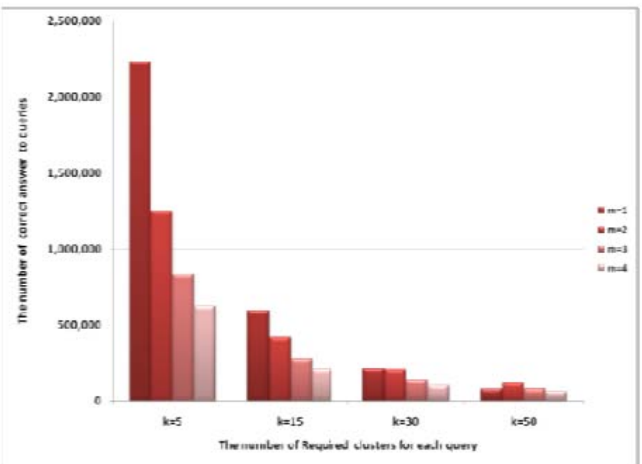

Figure 4. The number of correct answers when transmission fault is 0.01 and sensor fault is 0.001

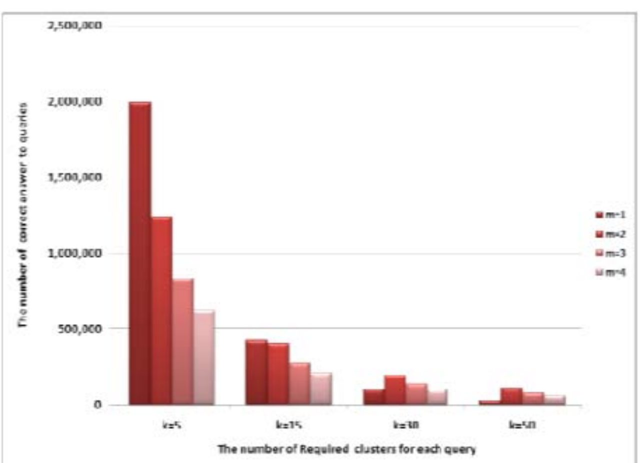

Figure 5. The number of correct answers when transmission fault is 0.05 and sensor fault is 0.005

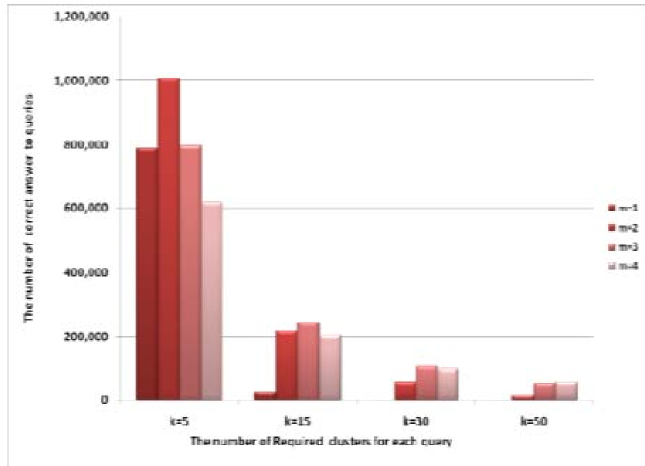

Figure 6. The number of correct answers when transmission fault is 0.1 and sensor fault is 0.01

VI. Results

According to the charts presented in the previous section the following results have been seen.

1. If the average number of clusters needed to response a request is equal to one then the use of redundancy at any level despiting the high probability of transmission and sensor errors, reduces the number of correct responses to the request in the network lifetime. The reason is clear. Due to the use of only one cluster, the reliability of each request will not greatly reduce, however if the redundant path is used the energy of sensors in the redundant paths are wasted and based on the charts if the redundant path is not to be used then the average network lifetime will be longer.

2. When the probability of transmission errors and sensor errors is negligible, the use of redundancy reduces the number of correct responses during network lifetime. Because the error probability is low, so there is high reliability. Therefore, the use of redundancy increases energy consumption and reduces the number of correct responses in network lifetime.

3. Whatever the number of used clusters for responding the request increases and also whatever the possibility of errors raises, then the affect of redundancy is going to be more visible. In this case, using the different levels of redundancy will have different affects such that whatever the number of used paths increases the number of correct responses during network lifetime will increases until using more redundancy will have reverse affect. This means that the use of excessive number of redundant paths reduces the number of correct responses because it causes network energy wasting in excessive paths.

So in this paper, by estimating the affect of redundancy on the number of correct responses that an on- demand network can have during its lifetime and before losing its total energy, we found out that the effect of redundancy on different scenarios of the network is different.

When the number of clusters needed to response to the request is low or little chance of error exists in the network, using the redundancy decreases the number of correct responses. Against, when the number of clusters is high or there is a big chance of error then more paths are needed to maximize the reactions. In other hand, these redundant paths have an optimum number in evernetwork situation such that using more paths reduces the correct reactions.

References Références Referencias

1. Anh Phan Speer and Ing-Ray Chen , "Effect of Redundancy on the Mean Time to Failure of Wireless Sensor Networks" , Concurrency Computat. 2007. Pract. Exper. 19:1119- 1128.
2. S. Singh, M. Woo and C. S. Raghavendra, "PowerAwareRouting in Mobile Ad Hoc Networks", Proc. of ACM MOBICOM', Dallas, Texas, October 2004
3. D. Estrin, R. Govindan, J. Heidemann, and S. umar. "Scalable coordination in sensor networks". Proc. of ACM/IEEE MobiCom , Seattle, Washington, August 2005.
4. Koushanfar F., Potkonjak M., and Sangiovanni A., "Fault tolerance Techniques for sensor networks", in: Proceedings of the IEEE Sensors Mag, Vol. 2, pp. 1491–1496, 2002.
5. Ding M., Chen D., Thaeler A. and Cheng X., "Fault-Tolerant Target Detection in Sensor Networks", in: Proceedings of the IEEE WCNC 2005, Vol. 4, pp. 2362 2362-2368, 13-17 March 2005.
6. Ding M., Chen D., Xing K. and Cheng X., "Localized Fault- Tolerant Event Boundary Detection in Sensor Networks", in: Proceedings of the IEEE INFOCOM 2005, Miami, March 2005.
7. Luo X., Dong M. and Huang Y., "On Distributed Fault Y., "On Distributed Fault-Tolerant Detection in Wireless Sensor Networks", in: Proceedings of the IEEE Transactions on Computers, Vol. 55, no. 1, pp. 58 ters, Vol. 55, no. 1, pp. 58-70, January 2006.
8. Huang X. M., Deng J. and Ma J., "Fault Tolerant Routing For Wireless Sensor Grid Networks", in: Proceedings of the IEEE s", in: Proceedings of the IEEE Sensors Applications Symposium, Houston, TX, February 2006 m, Houston, TX, February 2006
9. O. Younis and S. Fahmy, "HEED: A Hybrid Energy Efficient, Distributed Clustering Approach for Ad Hoc Sensor Network", IEEE Transaction on Mobile Computing, Vol. 3, No. 3, October- December 2004,pp. 366-379.
10. W. Heinzelman, C. Chandrakasan and H. Balakrishnan, "An Application-Specific Protocol Architecture for Wireless Microsensor Networks," IEEE Transactions on Wireless Communication, Vol.1, No. 4, 2002, pp. 660 02, pp. 660-670.