Energy-efficient Trust-based Aggregation in Wireless Sensor Networks

Zahra Taghikhaki, Nirvana Meratnia, Paul J.M. Havinga
University of Twente, Pervaisve Syestems
7500 AE, Enschede, Netherlands
(z.taghikhaki, n.meratnia, p.j.m.havinga) @utwente.nl

Abstract— **Wireless sensor networks (WSNs) are often deployed in unattended and noise-prone environments and suffer from energy constraints that limit the quality and quantity of data transmission. Every decision made based on the low quality and low quantity data may have drastic consequence. Therefore, ensuring high quality of sensor data and preserving energy resources of the sensor nodes go hand in hand. In this paper, we propose ETA that is an Energy-efficient Trust based data aggregation aims to achieve reliable and energy-efficient data transmission and aggregation. We use the concept of functional reputation and trust as a means to reach reliability. Functional reputation is used to select nodes that best satisfy the criteria to be an aggregator on the basis of the quality of the node. To find out the best path from every sensor node to the aggregator we take into account the link availability and residual energy of the nodes over the path. Simulation results show that even though ETA introduces some delays, overall it outperforms the other approaches in terms of reliability and lifetime.**

Keywords- Aggregation; Reliabilit; Energy-efficient; Trust; wireless sensor networks.

I. INTRODUCTION

A typical wireless sensor network consists of hundreds to thousands of inexpensive wireless nodes with limited computational capability and energy resources usually deployed in an unattended environment. Most effective utilization of wireless sensor networks requires minimization of energy consumption through the design of energy-efficient network protocols and algorithms to prolong network lifetime. Since sensor nodes are usually inexpensive hardware components, they are highly vulnerable and often malfunction or fail. Non-malicious behavior- such as the malfunctioning of radios or sensors- can result in generation of false data which has detrimental effects on the overall performance of the network.

Several wireless sensor network applications rely on decentralized decision makings. If no reliable data or reliable paths to send data towards the decision nodes exist, the final decision can not be trusted. On the one hand, these incorrect decisions may lead to serious inefficiencies throughout the whole network; while on the other hand, the energy of sensor nodes is wasted by providing such unreliable and false data.

Instead of providing raw dump of sensor data, in-network processing and aggregating algorithms are often used in wireless sensor networks, which in addition to saving energy also provide meaningful results to the end-users. Data aggregation has been considered as a significant primitive in wireless sensor networks that is widely regarded as being sensitive to attack and failures [1]. Since the base station receives an aggregation instead of the raw data, it loses the ability to filter out erroneous reports. Therefore, it is an important challenge for data aggregation process to ensure that an aggregator does not generate faulty data and does correctly send data to the base station. Due to the fact that an aggregator may become a single point of failure, it is better not to have just one special aggregator all the time. Rather, nodes that satisfy best the necessary criteria can be selected and act as an aggregator. Data aggregation techniques are tightly coupled with the routing approaches. If some links fail and do not relay sensor nodes' data for a while, the result of the aggregation may be highly inaccurate, which in turn can have a significant negative impact on the overall network performance.

In this paper, we propose an energy-efficient trust-based data aggregation and data transmission approach for wireless sensor networks. In addition to using trust concepts to have a reliable aggregation and transmission we keep an eye on the residual energy of every sensor node to help achieve an energy efficient reliable data aggregation and transmission.

The rest of this paper is structured as follows. In section II, related work is briefly discussed. In Section III, the model of our system and the definition of the problem are given. Section IV explains the proposed method in detail whereas simulation results are presented in section V. Finally section VI provides the conclusion of our work.

II. RELATED WORK

Wireless sensor networks and their limited resources introduce new design challenges. One of the major concerns in designing WSN algorithms that has been received significant attention is energy-efficiency. Several energy-efficient routing and aggregation protocols have been studied, which aim to minimize the total transmission energy consumption. Heinzelman et al. [2] focus on energy-efficiency by reducing number of nodes that directly communicate with the base station. The proposed approach, called LEACH, assume all sensor nodes have enough power to reach the base station if needed. This assumption makes LEACH unsuitable for large-scale sensor networks. Lindsey et al. [3] propose the PEGASIS algorithm, in which sensor nodes form a chain in such a way that each node can receive from or send to the closer neighbor. The data then is fused along the path to the base station by intermediate nodes and finally one designated node delivers the

results to the base station. In case of having distant nodes in the chain, PEGASIS imposes excessive delays. Moreover having only one aggregator creates a bottleneck. A power efficient data gathering and aggregation method called PEDAP has been presented in [4], which involves constructing a minimal spanning-tree based on energy consumption rooted at the base station. The weight of each edge is the energy required to transmit data from one side of the edge to another. An energy-efficient version of PEDAP, known as PEDAP-PA, tries to balance the load among all the nodes in order to improve network lifetime [4].

All the aforementioned methods suffer from link failures and packet losses. This becomes an important issue as reliability is a top priority in wireless sensor networks and it is important how well data is delivered to the base station.

To overcome the robustness problem, [5][6][7] try to send several copies of data along multiple paths instead of using only one path from every node to the destination. This is to ensure the data delivery in the expense of extra overhead caused by sending duplicate data. ReTrust [8] aims to improve work presented in [7] using some intermediate sinks or source nodes to decrease the packet overhead or multi-paths. PSFQ [9] has been proposed to ensure transmission reliability and to enable each intermediate node to buffer receiving packets destined for other nodes to have a faster retransmission along one of the healthy links in case of packet loss. Similar to PSFQ, [10][11] use acknowledgement or negative acknowledgement (NACK) to guarantee reliable delivery and also employ retransmission to improve data reliability. RDAT [12] is another attempt aiming to have a reliable delivery of aggregated data using functional reputation concepts in different levels. RDAT is divided into two parts. Firstly, every sensor node will find the best aggregator on its own to send its data directly to it using aggregator's reputation. Second, the aggregator splits data into several pieces and selects *n* paths from high reputed ones to send the data along them to the base station using Reed-Solomon error correction codes [13]. RDAT consumes high energy for two reasons. Firstly, sensor nodes send their data directly via only one link to the aggregator and second each sensor node must always overhear all other nodes in its communication range and should also overhear the aggregator. Moreover, splitting and reconstructing the packet is another challenge.

III. System Model and Problem Statemen

A. Network model

In this paper we consider that *N* sensor nodes are randomly scattered in a square field A and the following assumptions are made about the sensor network:

1- Sensor nodes are partitioned into several clusters and each cluster has one cluster head which we call the aggregator.

2- The network has only one powerful base station far away the deployment field A.

3- Every sensor node has a transmission range *R* but it can only choose neighbors located at *R/2* as its upstream relay node.

4- The locations of each sensor node and the base station are fixed and are known a priori.

B. Failure model

We assume that every link can experience transient or permanent failures. But when one link fails temporarily it is very likely that the given link will fail again in the future. In other words, the probability of being failed for a link depends on how many times it has failed before. In fact, there is a temporal correlation between failures of one link/sensor node.

Every packet is lost or successfully received by the destination and we ignore packet corruption and collision during transmission. Collision can be significantly reduced by forwarding packets with some random delay.

We also consider the fact that each sensor node can experience some problems and failures that prevent it from doing its task correctly. However, we do not consider malicious nodes that intentionally inject faulty data into the network.

C. Trust model

Reputation and trust concepts are being recently used in WSNs to diminish the impact of malicious and faulty nodes and links. Having history of the nodes' activities and links' states can give useful information about their situation, based on which the best policy can be chosen to have an overall efficient network. To evaluate the trust, we select Bayesian formulation and to represent reputation, we utilize BETA distribution, which is based on using beta Probability Density Functions (PDF). The advantage of the beta reputation system is its flexibility and simplicity, as well as its foundation on the theory of statistics [14]. The best way to represent reputation is a statistic probability distribution, but to judge the reputation of the nodes and links we must have numerical values. To this end *Trust* can be defined as the probability expectation value of the reputation function as in [14]. The beta PDF can be described using the gamma function as:

$$f(p \mid \alpha, \beta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} p^{\alpha-1}(1-p)^{\beta-1}$$
$$0 \le p \le 1, \alpha \ge 0, \beta \ge 0 \quad (1)$$

where α and β count the number of satisfaction (cooperative) and unsatisfaction (no-cooperative) of a given criteria respectively. Given a reputation metric R_{ij}, we define the trust rate T_{ij} to be expectation of node *i* about future behavior of node *j*. T_{ij} is obtained using the statistical expectation of prediction stated in formula 2:

$$T_{ij} = E(R_{ij}) = E(Beta(\alpha+1, \beta+1)) = \frac{\alpha+1}{\alpha+\beta+2} \quad (2)$$

We use trust and reputation concepts as a means to measure reliability, so each sensor node must maintain two tables: (i) Table about the reputation of its neighboring links (to judge about their availability) which we call Availability Reputation Table (AvRT); and (ii) Table about reputation of its

neighboring nodes (to judge about how well aggregation can be performed) which we call Aggregating Reputation Table (AgRT).

B. *Problem statement*

We consider a static cluster-based wireless sensor network. Every sensor node in a cluster must send its data to its upstream neighbor which is selected by the base. Intermediate nodes along the path to the aggregator fuse the data received from the downstream nodes with their own data and forward the local aggregated value towards the aggregator. The cluster head that we call the aggregator must perform final aggregation on the data received from its neighbors and then forward the result to the sink through the sensor nodes belonging to other clusters (Figure 1). The problem we deal with is to find a routing scheme to deliver data packets gathered by sensor nodes to the aggregator and later on from every aggregator to the base station on the basis of links availability and residual energy of the nodes in the path in order to prolong the lifetime of the network, to have a reliable aggregation as well as reliable data delivery to the destinations under the system model given above. Furthermore, we must find a sensor node, which has enough energy and can also play the role of aggregator better than other nodes.

IV. OUR PROPOSED METHOD

In this section, we present a formal description of our method. Initially, sink node informs every sensor node about its upstream node. Each relay node must relay data towards the aggregator and the base station. Relay nodes are randomly chosen from all upstream neighbors of a sensor node in the initialization phase. Every sensor node beside the sensing and relaying data must perform some extra tasks to judge the state of its neighboring links and its neighboring nodes.

Firstly, we explain sensor's tasks leading to assess links quality. Each sensor node has to send its current sensed data to all of its neighbors every TP_1 time period. When neighboring nodes receive data, they can deduce whether the link between them and the sender is available. If link is available, they increase the α parameter of AvRT for that link, which means these neighboring nodes can be good candidates to relay data from the sender. Otherwise the β parameter will be increased.

Secondly, there are some tasks aiming to find out how well a neighboring relay node can perform aggregation task. Each sensor node has to monitor the local aggregation task of its upstream node. In other words, each sensor node which selects a node as the upstream relay node must monitor the aggregation task of its relay node in every TP_1 time period, and it as well as its relay node have to perform aggregation. Later on, the sensor node overhears the aggregation result of its relay node and will update its AgRT for its relay node based on the difference between its aggregation value and result of its relay node.

Following that, every node sends its reputation table and its residual energy to its cluster's aggregator every TP_2 time period. The aggregator performs final aggregation on the data

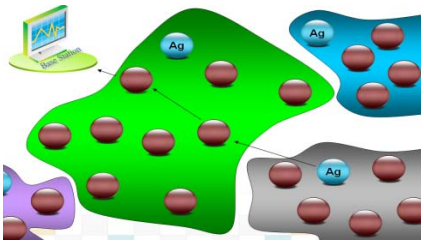

Figure 1. Assumed network structure

received from its cluster members and sends them along with the reputation tables and residual energy of its cluster members to the base station. Upon receiving the reputation tables and residual energy of the sensor nodes, base station will find the best aggregator for the next TP_2 time period for each cluster. It will also find the best path from every sensor node to its cluster's aggregator. For doing so, the base station must have a consensus on the neighbors' viewpoint of every node about the aggregating quality or reputation of a given node by considering AgRTs. The base station applies majority voting to achieve this consensus. To find the best aggregator for every cluster, the base station utilizes the equation 3. Table 1 presents the notation used. Now, the base station will find out the best path from every sensor node to its new aggregator. To do so, it exploits AvRTs and uses Floyd-Warshall algorithm [15] to construct the Shortest-Path Spanning Tree (SPST) for every cluster rooted at the new aggregator for that cluster.

$$C(node) = \frac{Eg^{node} \times T^{node}}{Init - Eg^{node} \times Init - T^{node}}$$
$$Eg^{node} > \theta_{Eg}^{Ag}, T^{node} > \theta_{T}^{Ag} \quad (3)$$

The weight assigned to each edge is obtained by taking the sensor nodes' residual energy and link availability into consideration using the following formula, where $C'(A,B)$ is the cost of each edge between two nodes *A* and *B*.

$$C'(A,B) = \frac{Init - T^{AB} \times Init - Eg^{B}}{T^{AB} \times Eg^{B}}$$
$$Eg^{B} > \theta_{Eg}^{relay-node}, T^{AB} > \theta_{T}^{link} \quad (4)$$

After building the SPST, the base station must revise poor-nodes (if any). Let us assume there is one poor-node that all possible links that it can select to relay its data through them have trust value lower than θ_T^{link}. To make sure that all data from that node will be received by the aggregator, the base station first finds possible neighbors ($L_{poor-node}$) for that node. It then selects some of them having higher weight than the poor-node to relay data from this poor-node as well as the main path selected using SPST in the previous stage. Therefore, one poor-node instead of sending its data through only one link to the aggregator or base station uses multiple links to make sure that its data is more likely to be received at its destination. The number of relay nodes that the base station chooses for the poor-node must satisfy Equation.5, in which m is the number

of selected neighboring links for the poor-node and ω is an application-dependent parameter, which is in direct proportional to needed reliability. Higher ω leads to selection of more neighbors to relay poor-node's data.

In fact, there is a combination of SPST for normal nodes and multipath routing for poor-nodes. When the base station builds these paths it must inform all of the sensor nodes in every cluster about the new path(s) and the new aggregator ID that they must send their data to.

$$L_{poor-node} = \{l_1, l_2, ..., l_n\}$$
$$\sum_{i=1}^{m} T^{l_i} \geq \omega \times \theta_T^{link} \text{ and } \sum_{i=1}^{m-1} T^{l_i} < \omega \times \theta_T^{link}$$
$$m \leq n, \omega > 1 \quad (5)$$

Therefore, base station makes a packet including the best main and emergency paths for each sensor node and sends it to its immediate downstream neighbors which base station itself chooses as the best last nodes for the current TP_2 time period. Each sensor node picks up the ID of its new upstream node(s) and the ID of new aggregator in its cluster from the packet and sends the packet to its downstream neighbor nodes. These operations must be done every TP_2 time period and it is possible that some links will fail during this period. To cope with such a situation, the base station assigns an emergency path to every sensor node; such a path is formed just based on the reputation so the residual energy in this case will be ignored.

TABLE I. NOTATION USED FOR OUR ALGORITHM

Name	Description
Eg^{node}	Node's available energy
$Init \quad Eg^{node}$	Initial energy for the node
T^{node}	Node's reputation (trust value) about its aggregation task
$Init \quad T^{node}$	Initial trust value for the node
T^{link}	Trust value of link about its availability
$Init \quad T^{link}$	Initial trust value for the link
θ_{Eg}^{Ag}	Minimum acceptable value for aggregator's energy
θ_T^{Ag}	Minimum acceptable value for trust value of aggregator
$\theta_{Eg}^{relay-node}$	Minimum acceptable value for relay node's energy
θ_T^{link}	Minimum acceptable value for trust value of link
$L_{poor\ \ node}$	Descending sort of possible neighbor links of poor-node based on their weight.

V. SIMULATION RESULTS

To evaluate the performance of our algorithm, we compare it with PEGASIS [3], PEDAP [4] and RDAT [12].

We use Visualsense [16] as the simulation platform, applying the various parameters in the simulation experiments shown in Table.2. We run the simulations with 10 different 300X300 network topologies and 150 sensor nodes. The base station is located at point (300, 50) and the network topology is single hop mesh. We also assume that 50% link failure and 40% packet loss. During simulation we select 80% of the failed links from the set of already failed links and 20% from the set of healthy and failed links to show temporal correlation between failures.

A. Reliability

To begin with, we start the evaluation looking at the reliability of various methods. To make a good judgment about our method and RDAT we must divide this comparison into two separate parts. This is because sensor nodes in RDAT send their data directly through a single link to their aggregators and later on aggregators send the result to the base station via multi-hop using Reed-Solomon method.

So, first we compare the reliability of our method with RDAT in every cluster till reaching the aggregator. In this part, RDAT sends its data directly to the aggregator while we send the data along the SPST. In the second part, we compare the reliability of data delivery between the aggregators and the base station in RDAT and our algorithm. The reliability metric represents the ratio between the numbers of packets received by the aggregator/base station and the total number of packets transmitted. As it can be seen from Figure 2, because the sensor nodes in RDAT send their data directly to the aggregator, when some links fail the data from the nodes near those links can not be received by the aggregator so the reliability of the algorithm will be decreased drastically when number of failed links increases.

TABLE II. SIMULATION PARAMETERS AND VALUE

Parameter	Value	Parameter	Value
Number of nodes per cluster	25	θ_{Eg}^{Ag}	0.5 J
Number of cluster	6	θ_T^{Ag}	0.9
Initial Energy for nodes	8 J	$\theta_{Eg}^{relay\ \ node}$	0.1 J
Initial trust value for nodes/links	1	θ_T^{link}	0.8
Transmit power	0.660 w	TP_1	10
Receive power	0.395 w	TP_2	50
ω	1.5		

The reliability of RDAT and our algorithm for the second part is shown in Figure 3 RDAT always selects available high reputed edges to send data through and does not consider the residual energy of the nodes over the path, so in the first rounds its reliability is higher than our algorithm. After a while when the nodes in the high reputed paths die it has to select another nodes having lower reputed links so its reliability will decrease. When these nodes also die RDAT has to select the lowest reputed links, so despite the fact that every node sends its data from several path majority of data can not be received by the base station. But our method always keeps an eye on the residual energy of every node beside the reputation (trust value) of the link, so sometimes despite having high reputed links we select multiple lower reputed links whose nodes have high residual energy.

As it can be seen from Figure 2 and Figure 3, reliability of RDAT, PEDAP, and PEGASIS is lower than our algorithm.

B. Life time

There are different definitions for lifetime of a wireless sensor network. In fact, lifetime is an application-dependent concept. There are some applications in which consider lifetime to be the time at which the first node dies, while others consider lifetime to be the time at which last node dies. We express the lifetime of the network in terms of number of dead nodes over the time. Based on the application one can judge the lifetime using different definitions.

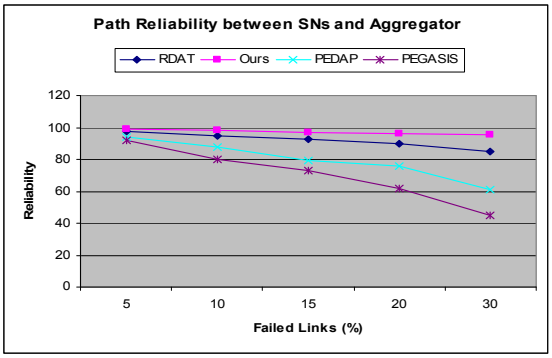

Figure 2. Reliability vs. Time (from sensor nodes to the aggregator)

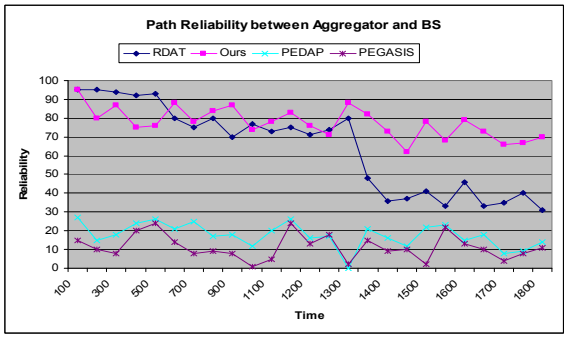

Figure 3. Reliability vs. Time (from the aggregator to the base station)

With respect to the results presented in Figure 4, the lifetime of our method outperforms the one of RDAT, and the lifetime of the first 45 sensor nodes is also longer compare with the other methods. Actually, we keep alive all sensor nodes simultaneously so the difference between death of the first node and death of the last node in our method is lower than the other methods. In some applications, data from all nodes is important and if some nodes die soon, data from those regions will be lost which can influence decision making of the base station. Therefore it is better to keep all nodes alive instead of using some special nodes much more than others that leads to depletion of their energy sooner. Furthermore, most of these methods select their paths ignoring reliability parameter so they have to occasionally retransmit their data several times, which clearly drains their energy.

In Figure 5, the residual energy for five different nodes is plotted against time. Each color shows one node so that one can judge about energy balancing among these five nodes.

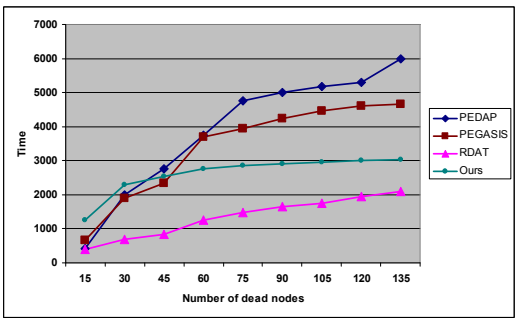

Figure 4. Lifetime of different methods

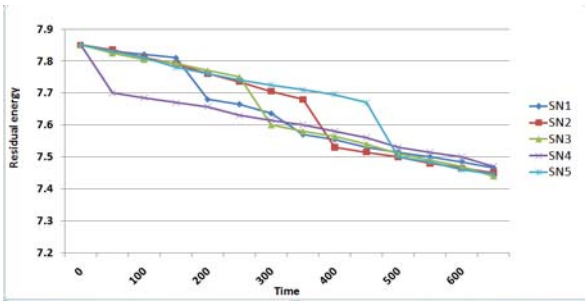

Figure 5. Energy balancing for five sensor nodes

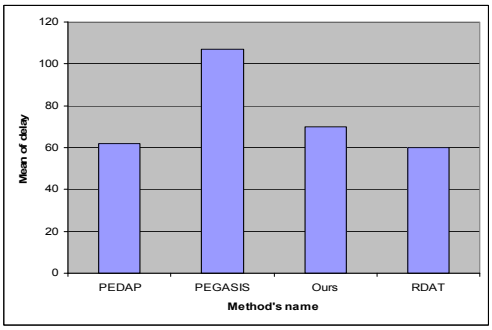

Figure 6. Delay for each method

C. Delay

While our algorithm ensures reliability and long network lifetime, it introduces some delays. Figure 6 shows the delay introduced by our methods compared with the other techniques. As it can be seen, our delay is higher than RDAT and PEDAP but lower than PEGASIS.

VI. CONCLUSION

In this paper, we have proposed a reliable, energy-efficient data aggregation and transmission method via building the shortest-path spanning tree such that the weight of each edge is a combination of link's availability and residual energy of the nodes. Based on the link's availability, we made a decision to either construct multi-path between source and destination nodes or just consider one single path in order to achieve a high reliability. We find the best node based on the residual energy and aggregation reputation to be an aggregator. Role of aggregator changes dynamically between sensor nodes. Furthermore, we have presented experimental results demonstrating that our method outperforms previous works when both network lifetime and reliability are considered.

VII. ACKNOWLEDGEMENT

This work is supported by IST FP7 STREP GENESI: Green sEnsor NEtworks for Structural monItoring project.

REFERENCES

[1] G B. Przydatek, D. Song, and A. Perrig, "Sia: Secure information aggregation in sensor networks," in SenSys, Los Angeles, CA, November 2003.

[2] W. Heinzelman, A. Chandrakasan, H. Balakrishnan, "An applicationspecific protocol architecture for wireless microsensor networks," IEEE Trans. Wireless Comm. vol. 1, pp. 660–670, April 2002.

[3] S. Lindsey and C.S. Raghavendra, "PEGASIS: Power-Efficient Gathering in Sensor Information Systems," Proc. IEEE Aerospace Conf. vol. 3, pp. 1125-1130, 2002.

[4] H. O. Tan, I. Korpeoglu, "Power Efficient Data Gatherting and Aggregation in Wireless Sensor Networks," Proc. ACM Int. Conf. Management of Data (ACM SIGMOD), vol. 32, no. 4, pp. 66-71, Dec. 2003.

[5] S. Bhatnagar, B. Deb and B. Nath., "Service Differentiation in Sensor Networks," Proc. of the 4th International Symposium on Wireless Personal Multimedia Communications, 2001.

[6] B. Deb, S. Bhatnagar, B. Nath., "Information assurance in sensor networks," Proc. of the 2nd ACM International Conference on Wireless Sensor Networks and Applications, 2003.

[7] B. Deb, S. Bhatnagar and B. Nath., "ReInForM: Reliable Information Forwarding using Multiple Paths in Sensor Networks," Proc. of the 28th Annual IEEE Conference on Local Computer Networks, 2003.

[8] Bo-Hyung Lee, Hyung-Wook Yoon, Jongho Park, Min Young Chung and Tae-Jin Lee, "Reliable Transmission Using Intermediate Sink or Source Nodes in Sensor Networks," Lecture Notes in Computer Science, vol. 3746, pp.839-848, 2005.

[9] C.Y. Wan, A.T. Campbell, L. Krishnamurthy, "PSFQ: a reliable transport protocol for wireless sensor networks," Proc. of ACM Workshop on Wireless Sensor Networks and Applications (WSNA), Atlanta, GA, September 2002.

[10] F. Stann, J. Heidemann, "RMST: Reliable data transport in sensor networks," Proc. of IEEE Sensor Network Protocols and Applications (SNPA), pp. 102-112, Anchorage, Alaska, USA, April 2003.

[11] H. Hassanein, J. Luo, "Reliable energy aware routing in wireless sensor networks," Proceedings of the 2nd IEEE Workshop on Dependability and Security in Sensor Networks and Systems, pp. 54–64, 2006.

[12] S. Ozdemir. "Functional reputation based reliable data aggregation and transmission for wireless sensor networks," Computer Communications, vol. 31, pp. 3941-3953, 2008.

[13] I.S. Reed, G. Solomon, "Polynomial codes over certain finite fields," SIAM Journal of Applied Mathematics, vol. 8, pp 300–304, 1960.

[14] A. Josang, R. Ismail, "The beta reputation system," Proc. of the 15th Bled Conf. Electronic Commerce, pp. 41, 2002.

[15] T. H.Cormen, C. E. Leiserson, and R. L.Rivest, "The floydwarshall algorithm," Introduction to Algorithms, first edition, pp. 558–565. MIT Press and McGraw-Hill, 1990.

[16] Visualsens, http://ptolemy.berkeley.edu/visualsense/, 2008.