Secure Load Balancing for Wireless Sensor Networks via Inter Cluster Relaying

Suat Özdemir

Abstract— **One of the limitations of wireless sensor nodes is their inherent limited energy resource. In order to distribute the energy dissipated throughout the wireless sensor network, data load of the sensor nodes must be balanced. Clustering is one of the key mechanisms for load balancing. Clustering algorithms may result in some clusters that have more members than other clusters in the network and uneven cluster sizes negatively affect the load balancing in the network. In addition, crowded clusters suffer from congestion and data loss which negatively affect the accuracy of the collected data. This paper presents Secure Load Balancing (SLB) protocol and introduces pseudo-sinks in order to improve data accuracy and lifetime of wireless sensor networks while still providing secure communication.**

***Index Terms*—Sensor networks, security, load balancing**

I. Introduction

THE emergence of sensor architectures with special capabilities and the developments in low-power computational components will bring wireless sensor network applications into reality in either controlled environments (such as home, office, warehouse, etc.) or uncontrolled environments (such as disaster areas, toxic regions, etc.) [1]. In a wireless sensor network, data of individual sensor nodes is aggregated by intermediate sensor nodes to eliminate redundant data and/or combine unreliable sensor measurements [2]. As many neighboring sensor nodes often produce the same or similar data, data aggregation is essential for reducing the unnecessary data transmissions. Clustering is the key mechanism for implementing data aggregation protocols in which cluster heads take the responsibility to coordinate their cluster sensor nodes and aggregate their data. However, clustering algorithms, such as lowest-id clustering [3], are originally designed for Mobile Ad-Hoc Networks (MANETs) and they do not consider the sensing-driven communication nature of wireless sensor networks. Therefore, it is possible that some clusters have more members than others in the network due to the clustering scheme being used. Uneven cluster sizes negatively affect the network in the following two ways: (i) Cluster heads of the crowded clusters perform more computation and communication than other cluster heads. Hence, those cluster heads tend to die sooner than cluster heads with less cluster members, thereby shortening the network's lifetime. (ii) Sensor nodes in crowded clusters have more delay in their data transmission to the cluster head. Even in Time-Division based Medium Access (TDMA), which provides bounded transmission delay, data in memory of some sensor nodes will be overwritten by new measurements because of long waiting time for channel access, limited storage and transmission capacity. Therefore, accuracy of the collected information, which is vital for surveillance networks, is reduced on regions covered by crowded clusters because measurements of all sensor nodes are not reflected in the aggregated data. In this paper, we will refer this issue as *accuracy problem*.

Security is another key requirement for many wireless sensor network applications (e.g., surveillance, monitoring, and battlefield). The widespread deployment of these networks can be curtailed without proper security because a sensor network should not leak sensor readings to outsiders [4]. Especially in military applications, the data stored and transmitted among sensor nodes may be highly sensitive. Preventing unauthorized parties from discovering the transmitted data is typically accomplished by setting up an encrypted communication channel. In wireless sensor networks, to setup an encrypted channel, a sensor node pair needs to share a secret key so that the sender node encrypts its data using the secret key before transmitting it and the receiver node decrypts the encrypted data using the same key. Therefore, several random key predistribution protocols are proposed for wireless sensor networks [5,6,7]. In random key predistribution protocols, sensor nodes are provided a set of keys from a key pool before the deployment and expected to have shared keys with their neighboring nodes.

To prolong the network lifetime and to mitigate the accuracy problem while also providing security and data aggregation, this paper proposes Secure Load Balancing (SLB) protocol that employs *pseudo-sinks* that are a small number of special, tamper-proof sensor nodes with more computational, storage, and energy resources. The novel idea behind SLB protocol is to mitigate accuracy problem by securely relaying data from congested clusters to nearby free clusters or pseudo-sinks. Moreover, data are aggregated at

Suat Özdemir is with Gazi University, Faculty of Engineering and Architecture, Computer Engineering Department, Ankara, 06570, TURKEY (corresponding author to provide phone: 312-231-7400 (ext. 2123); fax: 312-230-8434; e-mail: suatozdemir@gazi.edu.tr).

cluster heads and pseudo sinks to reduce the data transmission overhead. In order to securely transmit data and perform data aggregation, the existence of shared keys between cluster heads and sensor nodes is required. However, it may not be possible for every sensor node to share a secret key with its cluster head. Although random key predistribution protocols offer path key establishment methods for sensor nodes that do not have a shared key, in the presence of malicious sensor nodes path key establishment is not secure [8]. In SLB protocol, in addition to data of congested clusters, sensor data that cannot be aggregated due to lack of shared keys are also relayed to pseudo-sinks for data aggregation. Thanks to the increased storage capability of pseudo-sinks, pseudo-sinks store much more secret keys than ordinary sensor nodes and each sensor node is able to share a key with one of the nearby pseudo-sinks. Therefore, SLB protocol not only mitigates the accuracy problem but also improves the data aggregation efficiency of the network by enabling pseudo-sinks to aggregate sensor data that cannot be aggregated at cluster heads due to absence of shared keys. Fig. 1 shows the reference network architecture for SLB protocol. Simulation results show that, in comparison with traditional cluster based networks, SLB protocol improves the accuracy of the information gathered in the network and increases the data delivery rate in the presence of security constraints. The rest of the paper is organized as follows. Section 2 presents the related work whereas Section 3 describes the network model and assumptions. Section 4 introduces SLB protocol. Simulation results are given in Section 5 and concluding remarks are made in Section 6.

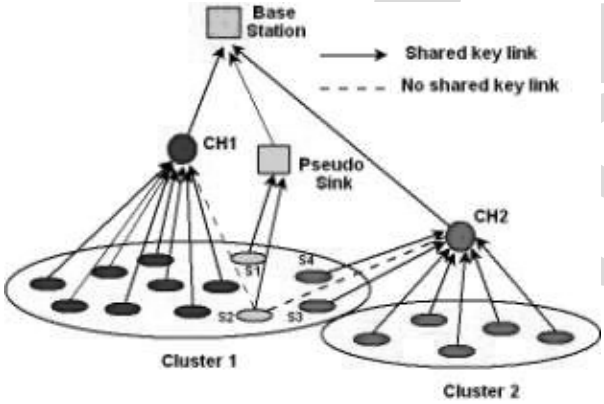

Fig. 1. The reference network architecture for SLB protocol. Cluster 1 is the most crowded cluster in this example and suffers from congestion. Hence, sensor nodes S1 sends its data to the pseudo sink whereas S3 and S4 send their data to a neighboring free cluster (Cluster 2). Also, sensor node S2 in Cluster 1 does not share a key with its cluster head CH1 or CH2, and therefore sends its data to the pseudo sink.

II. Related Work

Heterogeneous sensor networks are already introduced in [9,10,11]. Actors or actuators that perform appropriate actions based on the data collected from sensor nodes are introduced in [9]. Authors explore sensor-actor and actor-actor coordination and describe research challenges for coordination and communication problems. Authors of [10] evaluate the effect of heterogeneous sensor deployments on sensing coverage and the trade-off between initial coverage and the duration of network sensing operations. Intel also has some work on heterogeneous sensor network where they overlay an 802.11 mesh network over a sensor network to reduce the energy consumption of sensor nodes due to data forwarding [11].

The concept of relaying has been used in the context of cellular wireless networks [12]. In [13], relays can operate using the channels already available for the cell. iCAR [14] uses an additional air interface for relaying operation. Improvements due to relaying such as improved coverage, increased capacity due to multiple simultaneous short range transmissions also hold for wireless sensor networks. However, there are several key differences between relaying in cellular networks and relaying in wireless sensor networks considered in this paper. In contrast to the fixed base stations of cellular networks, cluster-heads in wireless sensor networks may change in time making deployment of fixed relay nodes as in [14] impossible. Also, the nature of communication request arrivals is different for cellular and sensor networks.

The security needs of wireless sensor networks along with data aggregation requirement have led many researchers [15,16,17] to study secure data aggregation problem. The security protocol presented in [15] proposes security mechanisms to detect node misbehaviors such as dropping or forging messages and transmitting false aggregate values. In this work, instead of aggregating messages at the immediate next hop, messages are forwarded unchanged over the first hop and then aggregated at the second hop. In [16], it is assumed that there are certain nodes in the sensor network, called aggregators, that aggregate information requested by a query. The main idea of this work is to employ sampling mechanisms to detect the injected false data. In [17], an energy efficient secure data aggregation protocol that uses small data representatives called data patterns is proposed. Data patterns may not be applicable if high precision is required for aggregated data.

III. Network Model and Assumptions

We consider a static cluster based wireless sensor network where each sensor node is battery powered and composed of a small computation unit, a sensing unit, and a short range radio. Data collection is done at a powerful base station located nearby the sensor network, which queries the network through cluster heads or pseudo-sinks. Pseudo-sinks are special tamper-proof sensor nodes that have more memory space, computational power and battery life. Due to low cost requirement of wireless sensor networks [1], sensor nodes are inexpensive nodes such as Mica2 [18] motes from Crossbow.

On the other hand, Intel's STARGATE [11] mote with some modification is the ideal candidate for pseudo-sinks. As pseudo-sinks are costly compared to ordinary sensor nodes, the ratio of number of pseudo-sinks to regular sensor nodes is usually small, such as 1 to 50. Sensor nodes are vulnerable to physical node compromise attacks whereas pseudo-sinks are made tamper-proof.

Cluster heads collect raw data from sensor nodes and perform data aggregation. Sensor nodes encrypt the sensed data prior to transmission. Hence, cluster heads must have shared keys with the sender of any incoming data to be able to decrypt it for data aggregation. It is assumed that a random key predistribution scheme is in use [5,6,7]. Random key predistribution schemes are realistic key distribution schemes for wireless sensor networks as sensor nodes are deployed randomly and do not have their location information before the deployment. The downside of random key predistribution schemes is that they may result in neighboring sensor node pairs that do not have a shared key [5]. For example, it may not be possible for a cluster head to have shared keys with some sensor nodes in its cluster. Keys are distributed to pseudo-sinks in such a way that each sensor node is able to find at least one shared key with one of its nearby pseudo-sinks.

IV. SECURE LOAD BALANCING

SLB protocol enables sensor nodes in congested clusters to relay their data securely to neighboring free clusters or pseudo-sinks to prolong the lifetime of the network and to mitigate the accuracy problem. In addition, a predetermined data rate threshold ($DataRate_{Threshold}$) is defined for sensor nodes. If a sensor node's data rate is below the threshold then the sensor node is required to send its data to a pseudo-sink rather than its cluster head. Hence, SLB protocol reduces the variance among data rates of sensor nodes, thereby ensuring that each node in the sensor network is able to transmit its measurements on time. In addition to improving data accuracy, SLB protocol improves data aggregation efficiency by aggregating the data that cannot be aggregated at cluster heads due to the lack of shared keys. Sensor nodes take turns to be a cluster head to balance the energy consumption among sensor nodes. Hence it may not be possible to have a shared key between every sensor node and cluster head pair. Such sensor nodes that do not share a secret key with the data aggregator send their data to nearest pseudo-sink. Figure 2 presents the pseudo code of SLB protocol.

Before proceeding to explain SLB protocol in detail, it should be noted that pseudo-sinks periodically broadcast their IDs so that sensor nodes can discover the nearby pseudo-sinks. Each pseudo-sink ID broadcast message also indicates the ID's of the secret keys of the pseudo-sink [5]. This key ID information enables sensor node S_i to select the pseudo-sinks that have a shared key with S_i. ID broadcasting is performed periodically to provide an adaptation mechanism against link failures. In addition, in order to prevent network from flooding, pseudo-sink IDs are forwarded by sensor nodes up to a certain number of hops depending on the network's diameter.

SLB Protocol - [Relay Node Discovery]

Input: Sensor nodes S_i, cluster head of S_i (CH_k), neighboring node set of S_i $[S_1, \cdots S_j]$ and their cluster heads $[CH_1, \cdots CH_j]$.
Output: $OFFERLIST_{S_i}$ of neighboring clusters that are available for relaying.

1: Broadcast $S_i.RelayReq$.
2: *// Collect RelayReq from other sensors and check conditions for relaying*
3: **for all** received $S_j.RelayReq$ message **do**
4: **if** $CH_i \neq CH_j$ **and** $S_i.chsize < S_j.chsize$ **then**
5: Forward $S_j.RelayReq$ to CH_i and send $CH_i.RelayOffer$ to S_j.
6: **end if**
7: **end for**
8: *// Collect all offers*
9: **for all** received $CH_k.RelayOffer$ message **do**
10: **if** a shared key between CH_k and S_i exists; **then**
11: Add $CH_k.RelayOffer$ to $OFFERLIST_{S_i}$
12: **end if**
13: **end for**

Fig. 2: Relay node discovery part of SLB Protocol

SLB protocol consists of two parts: (i) Relay node discovery (Fig. 2) and (ii) Data relaying (Fig. 3). Sensor node S_i implements SLB protocol at each data transmission session. In Relay Node Discovery part, cluster heads assign time slots to their cluster members for data transmission and each sensor node S_i obtains knowledge of its cluster head (CH_i) and cluster size (*chsize*). Each sensor node S_i broadcasts a *RelayReq* packet to its neighbors which includes its identifier (S_i), identifier of its cluster head (CH_i), the cluster size of CH_i, and its distance to CH_i ($|S_i, CH_i|$). Then, S_i starts waiting for *RelayReq* packets from other sensor nodes. When, sensor node S_i receives the $S_j.RelayReq$ packet from S_j, it checks the cluster head identifier in the packet. If S_i and S_j belong to the same cluster, the request packet is ignored; otherwise the following condition is checked. The size of S_i's cluster should be smaller than the cluster size of S_j so that S_j will be able to increase its data rate. If this condition is met, S_i forwards the request to its cluster head CH_i and CH_i allocates a new time slot for S_j. Information including cluster head identifier of CH_i, cluster size, time slot assignment, and CH_i's list of secret key identifiers is sent to S_j in a *RelayOffer* packet. To find the cluster head that will provide the best data rate, S_i also collects the relay offers from cluster heads that have a shared key with S_i. The collected relay offers are added to *OFFERLIST* of S_i.

In Data Relaying part of SLB protocol, S_i selects the relay offer with the minimum cluster size ($CH_k.RelayOffer_{min}$) that will yield the best improvement in its data rate from its *OFFERLIST*. S_i verifies the offer since the cluster head CH_k can send the same offer to multiple requesting nodes. On relay link establishment, S_i notifies its previous cluster head CH_i to release its resources. Both CH_i and CH_k update their member count and notify their members on current cluster status. After the relay link establishment, S_i encrypts its data D_i using the key that it shares with CH_k and sends the encrypted D_i to CH_k.

SLB Protocol - [Data Relaying]

Input: Sensor node S_i, cluster head of S_i (CH_i), neighboring node set of S_i $[S_1, \cdots S_j]$, pseudo-sink of S_i ($pseudosink_i$), $OFFERLIST_{S_i}$, data D_i of sensor node S_i, and $DataRate_{Threshold}$.
Output: Data D_i is securely transmitted to a congestion free cluster head or a pseudo-sink.

```
// Select the best relay offer
if OFFERLIST_Si not empty then
    Select CH_k.RelayOffer_min, the offer with the minimum choice;
    Send ACK to CH_k.RelayOffer_min and notify CH_i to release its resources;
    Encrypt the data D_i using the shared key between CH_k.RelayOffer_min and S_i ;
    Send encrypted D_i to CH_k.RelayOffer_min;
else
    if S.DataRate > DataRate_Threshold and a shared key between S_i and CH_i exists then
        Encrypt the data using the shared key between S_i and CH_i;
        Send encrypted data to the data CH_i ;
    end if
else
    Encrypt the data using a key and send it to PseudoSink_i;
end if
```

Fig. 3: Data relaying part of SLB Protocol

If S_i does not have any relay offer that can improve its data rate, then S_i first checks if its date rate is above the predetermined threshold. Then, it checks whether it shares a secret key with its own current cluster head CH_i. If both conditions are met, S_i sends its encrypted data to CH_i for data aggregation. However, due to congestion, S_i's data rate may be too low. In addition, as sensor nodes take turns to be a cluster head, it may not be possible for S_i to have shared key with its current cluster head CH_i [5]. Hence, CH_i cannot decrypt and aggregate the data of S_i. In such situations, SLB protocol takes advantage of pseudo-sinks. If S_i does not share a key with its current cluster head, it encrypts the data using one of the keys that it shares with the nearby pseudo-sink ($pseudosink_i$) and sends the encrypted data to $pseudosink_i$. $pseudosink_i$ decrypts the encrypted data and aggregate it. Hence, in addition to mitigating accuracy problem, pseudo-sinks improve the data aggregation efficiency of the network as well.

V. Simulation Results

We have evaluated SLB protocol in terms of information accuracy, average data rate per sensor node and network life time by generating random network instances with 200 nodes and various numbers of pseudo-sinks (2-8). QualNet [19], a parsec based commercial sensor network simulator, is used for simulations. Lowest-id clustering algorithm is used to form the clusters in the network. The channel access scheme is chosen as TDMA, therefore data rate of nodes depend on the size of the cluster they belong to. The placement of pseudo-sinks in the network is an important issue in SLB protocol. Hence, simulations are performed for both uniform and random distribution of pseudo-sinks over the network. The base station is located at one corner of the network. Simulations are performed using SNR of 1.5dB to adapt the high packet loss rate of wireless sensor networks and the packet retransmission limit is set to 3.

Figure 4 illustrates the effect of relaying on the variance of sensor node data rates when 2.4Kbps radio with 20m communication range is used. As seen from the figure, as number pseudo-sinks increases, SLB protocol decreases the variance of sensor node data rates. Hence, load balancing and fairness among sensor nodes are provided. In addition, the load balancing prevents quick exhaustion of some cluster heads due to excessive data transmission, and therefore positively affects the lifetime of the network. It should also be noted that, uniform distribution of pseudo-sinks always yields better load balancing and fairness among sensor nodes.

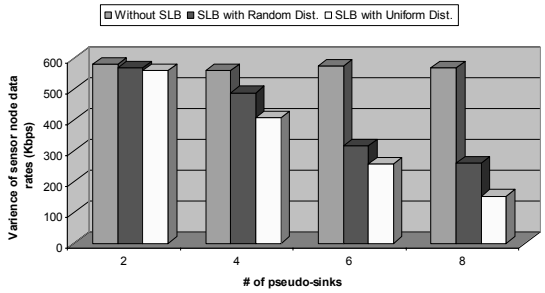

Fig. 4: Variance of sensor node data rates for various numbers of pseudo-sinks

Since data aggregation reduces the amount of data transmission in the network [2], the energy consumption of sensor nodes is also reduced and lifetime of each sensor node is increased. As a result, the lifetime of the network is significantly prolonged due to load balancing and improved data aggregation efficiency. In this simulation, we refer the network lifetime as the maximum time limit that nodes in the network remain alive until one or more nodes drain up their energy. Figure 5 shows the increment in lifetime of the network due to SLB protocol. As seen from the figure, SLB protocol improves the network lifetime up to 50% when eight pseudo-sinks are uniformly distributed over the network.

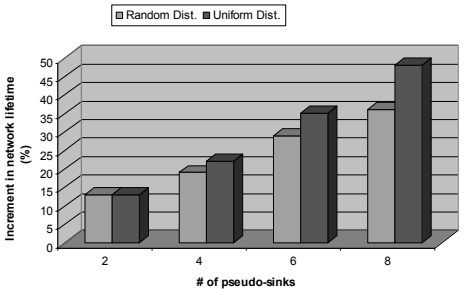

Fig. 5: Increment in lifetime of the network

Figure 6 shows how SLB protocol mitigates the accuracy problem. As all sensor nodes are able to send their data to cluster heads and pseudo-sinks, SLB protocol reduces the deviation from the correct information collected at the base station. In this simulation, sensor nodes measure temperature values from the environment and the difference of the average temperature collected by the base station and the actual average temperature values of sensor nodes is computed. Figure 6 clearly shows that pseudo sinks reduces the deviation from the actual average temperature value and therefore

mitigates the accuracy problem.

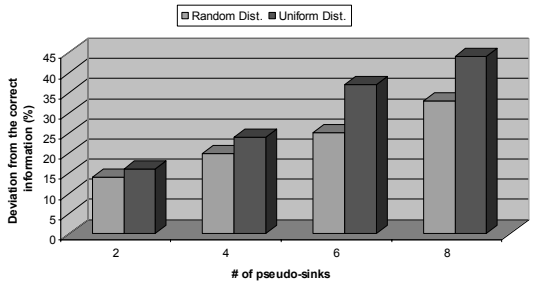

Fig.6: Deviation from the correct information

VI. CONCLUSIONS

This paper has presented Secure Load Balancing (SLB) protocol for heterogeneous sensor networks to securely prolong the network lifetime and to mitigate the accuracy problem that occurs in congested clusters. By employing pseudo-sinks, SLB protocol balances the data transfer rates of sensor nodes to enable every sensor node to transmit its data, thereby improving the accuracy of the information gathered at the base station. In addition, SLB protocol increases the data aggregation efficiency of the network as pseudo-sinks aggregate the data that cannot be aggregated at cluster heads. The performance results show that SLB protocol ensures the data accuracy and improves the network lifetime due to balanced data rates and increased data aggregation ability of the network.

REFERENCES

[1] Akyildiz, I.F., Su, W., Sankarasubramaniam, Y., and Cayirci, E., A survey on sensor networks, IEEE Communications Magazine, 40(8), 102-114, 2002.
[2] Intanagonwiwat, C., Estrin, D., Govindan, R., and Heidemann, J. "Impact of network density on Data Aggregation in wireless sensor networks", Proc. of the 22nd International Conference on Distributed Computing Systems, pp. 575-578, July 2002.
[3] Ephremides, A., Wieselthier, J.E., and Baker, D. J. "A Design concept for Reliable Mobile Radio Networks with Frequency Hopping Signaling", Proceeding of IEEE, Vol. 75, No. 1, pp. 56-73, 1987.
[4] Perrig, A., Szewczyk, R., Tygar, J. D., Wen, V., and Culler, D. E. "Spins: security protocols for sensor Networks", Wireless Networking, 8(5), 521–534, 2002.
[5] Eschenauer, L. and Gligor, V. D., "A key-management scheme for distributed sensor networks", 9th ACM conference on Computer and communications security, Washington, DC, 41-47, 2002.
[6] Chan, H., Perrig, A., and Song, D., "Random key predistribution schemes for sensor networks, IEEE Symposium on Security and Privacy", Berkeley, California, 197-213, 2003.
[7] Du, W., Deng, J. and Varshney, P. K., "A pairwise key pre-distribution scheme for wireless sensor networks", 10th ACM Conference on Computer and Communications Security, Washington, USA, 42-51, 2003.
[8] G. Li, H. Ling, and T. Znati, "Path key establishment using multiple secured paths in wireless sensor networks", Proc. of ACM Conference on Emerging Network Experiment and Technology, Toulouse, France, October 24 - 27, 2005.
[9] Akyildiz, I.F. and Kasimoglu, I.H. "Wireless sensor and actor networks: research challenges", Ad Hoc Networks, No. 2, pp. 351367, 2004.
[10] Lee, J.-J., Krishnamachari, B., and Kuo, C.-C. J. "Impact of heterogeneous deployment on lifetime sensing coverage in sensor networks", First IEEE International Conference on Sensor and Ad hoc Communications and Networks (SECON) Santa Clara, CA, October 4-7, 2004.
[11] www.intel.com/research/exploratory/heterogeneous.htm (2007)
[12] Aggelou, G.N. and Tafazolli, R. "On the Relaying Capability of Next-Generation GSM Cellular Networks", IEEE Pers.Commun., vol. 8, no. 1, pp. 40-47, Feb. 2001.
[13] Sreng, V., Yanikomeroglu, H., and Falconer, D.D. "Capacity enhancement through two-hop relaying in cellular radio systems", IEEE Wireless Communications and Networking Conference (WCNC'02), 17- 21 March, 2002, Orlando, FL, USA.
[14] De, S., Tonguz, O., Wu, H., and Qiao, C. "Integrated cellular and ad hoc relay (iCAR) systems: pushing the performance limits of conventional wireless networks", Proc. of the 35th Annual Hawaii International Conference on System Sciences HICSS, 7-10 January 2002, pp. 3931-3938.
[15] Hu, L. and Evans, D. "Secure aggregation for wireless networks", Proc. of Workshop on Security and Assurance in Ad hoc Networks, Jan 28, Orlando, FL, 2003.
[16] Przydatek, B., Song, D., and Perrig, A. "SIA : Secure information aggregation in sensor networks", Proc. of SenSys'03, Nov 5-7, Los Angeles, CA, 2003.
[17] Cam, H., Ozdemir, S., Nair, P., Muthuavinashiappan, D., and Sanli, H.O. "Energy-Efficient and secure pattern based data aggregation for wireless sensor networks", Special Issue of Computer Communications on Sensor Networks' pp. 446-455, Feb. 2006.
[18] Crossbow Technologies Inc., www.xbow.com (2007)
[19] QualNet, www.scalable-networks.com/ (2007)

Suat Özdemir Suat Ozdemir has been with the Department of Computer Engineering at Gazi University, Ankara, Turkey since March 2007. He received his MSc degree in Computer Science from Syracuse University (August 2001) and PhD degree in Computer Science from Arizona State University (December 2006). His main research interests include broad areas of wireless networks and network security.