

Secure Load Balancing via Hierarchical Data Aggregation in Heterogeneous Sensor Networks

SUAT OZDEMIR

Computer Engineering Department

Gazi University

Ankara, 06570 Turkey

In wireless sensor networks, sensing driven nature of data generation and uneven cluster sizes result in unbalanced data traffic load among clusters. Clusters in which the event generation rate is high and/or clusters that have more members than others suffer from congestion and data loss which negatively affect the accuracy of the collected data. In addition, the cluster heads of such clusters exhaust their energy earlier than others, thereby reducing the network lifetime. Hence, the data load among clusters must be balanced to preserve the data accuracy and prolong the network lifetime. This paper presents Secure Load Balancing (SLB) protocol and introduces pseudo-sinks in order to improve data accuracy and bandwidth utilization of wireless sensor networks while still providing secure communication. Simulation results show that, in comparison with traditional cluster based networks, SLB protocol improves the data accuracy and increases the data delivery rate in the presence of security constraints.

Keywords: sensor networks, congestion, load balancing, data aggregation, security

1. 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 office, warehouse, *etc.*) or uncontrolled environments (such as disaster areas, *etc.*) [1]. In a wireless sensor network, data of individual sensor nodes are aggregated by intermediate sensor nodes to eliminate redundant data and/or combine unreliable sensor measurements [2]. As many neighboring sensor nodes often have overlapping sensing ranges and produce correlated data, data aggregation is essential for reducing unnecessary data transmissions and prolonging the network lifetime. Clustering is one of the key mechanisms for implementing data aggregation protocols in which cluster heads take the responsibility to coordinate their cluster sensor nodes and aggregate their data. However, many clustering algorithms, such as lowest-id clustering [3, 24], are originally designed for Mobile Ad-Hoc Networks (MANETs) and they do not consider the sensing-driven communication nature of wireless sensor networks [25]. Therefore, it is possible that some clusters have more members than others in the network. Moreover, due to sensing-driven event generation, some clusters may have higher data generation rate than others [25]. Such clusters that have more members or have higher data rates compared to other clusters are referred to as *congested clusters*. Sensor nodes in congested clusters have more delay in their data transmission to the cluster head. Even in time-division based medium access, which provides bounded transmission delay, data in memory of some sensor nodes will be overwritten by new

Received March 17, 2008; revised July 16 & August 29, 2008; accepted November 12, 2008.

Communicated by Ten-Hwang Lai.

measurements because of long waiting time for channel access, limited storage and transmission capacity. Hence, the accuracy of the collected information, which is vital for surveillance networks, is reduced on regions covered by congested clusters because not all measurements of sensors are reflected in the aggregated data. In this paper, we refer this issue as *accuracy problem*.

Security is another key requirement for many wireless sensor network applications. The widespread deployment of these networks can be curtailed without proper security because a sensor network should not leak sensor readings to outsiders [4]. Hence, like any other wireless sensor network protocol, data aggregation protocols must work in conjunction with security policies of the network. Preventing unauthorized parties from discovering the transmitted data is typically accomplished by setting up an encrypted communication channel which requires a shared secret key between communicating parties. Therefore, several random secret key predistribution protocols are proposed for wireless sensor networks [5-7, 23] in which 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. Due to their probabilistic nature, however, random secret key predistribution protocols may result in sensor node pairs that do not have a shared key but reside in the same cluster. Although there are path key establishment methods [5] for sensor node pairs that do not have a shared key, in the presence of compromised sensor nodes path key establishment is susceptible to security attacks, such as man in the middle [8]. Despite their probabilistic nature, random secret key predistribution protocols are widely employed in wireless sensor network research.

To mitigate the accuracy problem while also providing security and data aggregation, this paper proposes Secure Load Balancing (SLB) protocol that employs pseudo-sinks which 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 both cluster heads and pseudo-sinks to reduce the data transmission overhead and improve the network lifetime. In order to securely transmit data and perform data aggregation, the existence of shared keys between cluster heads and sensor nodes is required. However, due to the nature of random key predistribution protocols, it may not be possible for every sensor node to share a secret key with its cluster head. 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 their increased storage capability, pseudo-sinks store much more secret keys than ordinary sensor nodes and sensor nodes are able to share a key with the pseudo-sink of its deployment group. Therefore, SLB protocol not only mitigates the accuracy problem but also improves the security and 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 data aggregation efficiency and network lifetime in the presence of security constraints.

Our contribution in this paper is to mitigate the load balancing problem by considering security and data aggregation requirements of wireless sensor networks. To the best

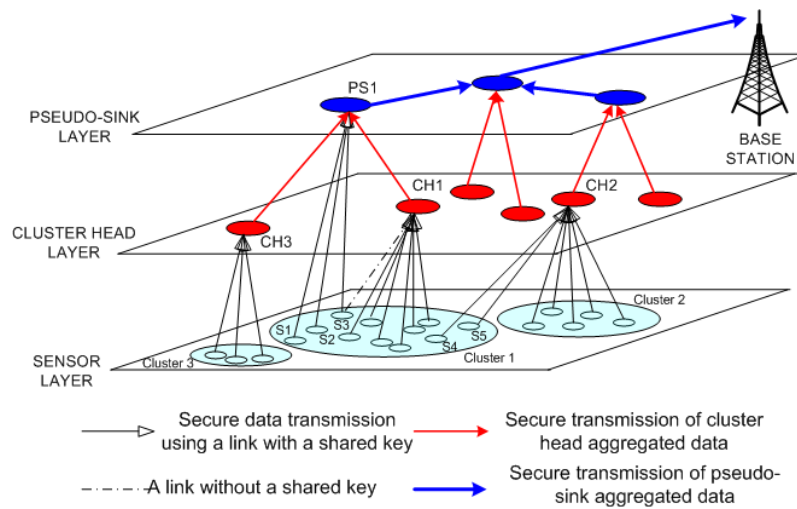


Fig. 1. The reference network architecture for SLB protocol. Cluster 1 is congested and hence sensor nodes S1 and S2 send their data to the pseudo sink PS1 whereas S4 and S5 send their data to the cluster head CH2 of a neighboring free cluster (Cluster 2). Moreover, sensor node S3 in Cluster 1 does not share a key with its cluster head CH1 and therefore it sends its data to PS1. In addition, CH3 sends its aggregated data to PS1 as it is closer to CH1. Note that some of the clusters are not shown in the figure for the sake of simplicity.

of our knowledge, this is the first paper that considers *security* and *load balancing* concepts together in wireless sensor networks. In addition, our proposed SLB protocol achieves longer network lifetime by securely balancing the data load among clusters and hierarchically aggregating the balanced data at pseudo-sinks and cluster heads. The rest of the paper is organized as follows. Section 2 presents the related work whereas network model and assumptions are given in section 3. SLB protocol is introduced in section 4. In section 5, the benefit of hierarchical data aggregation is analyzed. The simulation results and security analysis are given in section 6 and concluding remarks are made in section 7.

2. RELATED WORK

Heterogeneous sensor networks are already introduced in [9-11]. Actors or actuators that perform appropriate actions based on the data collected from sensor nodes are introduced in [9]. The authors explore sensor-actor and actor-actor coordination and describe research challenges for coordination and communication problems. The 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 [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, such as mobility and communication pattern.

The security needs of wireless sensor networks along with data aggregation requirement have led many researchers to study secure data aggregation problem [15-18]. 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. In [18], a functional reputation based secure and reliable data aggregation protocol is proposed. The proposed protocol takes advantage of reputation and trust concepts to improve reliability of the aggregated data.

3. 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 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. Sensor nodes are inexpensive nodes such as Mica2 [20]. On the other hand, STARGATE [21] or Imote2 [20] can be used as pseudo-sinks after being made tamper proof [26]. Since tamper proof 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 100. Sensor nodes estimate their distance to other nodes based on signal strength [27]. Cluster heads collect raw data from sensor nodes and perform data aggregation. Sensor nodes encrypt and authenticate 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.

3.1 Key Distribution

Random key predistribution schemes [5-7, 23] are realistic key distribution schemes for wireless sensor networks. However, these schemes 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. To increase the probability of key sharing between two sensor nodes, the number of key in sensor nodes must be increased. However, increasing the number of keys in each sensor node weakens the security [5, 6]. Hence, to support secure data transmission for all sensors without increasing the number keys stored in sensor nodes, SLB protocol employs pseudo-sinks

and a group based key predistribution scheme [23] which requires sensor nodes to be deployed in groups. It is shown that group based key predistribution reduces the number of keys in each sensor node and increases both the resiliency against node compromise attacks and the probability of having a shared key among sensor nodes [7, 23]. As described in [23], SLB protocol deploys sensor nodes into the network area as deployment groups where each group has a separate key pool. In addition, a pseudo-sink is deployed along with each sensor group. The keys stored in a pseudo-sink are selected from the key pool of the pseudo sink's deployment group. Hence, the number of keys stored in a pseudo-sink is reduced and the probability of having a shared key between a pseudo-sink and sensor nodes is increased. Interested readers are referred to [7] and [23] for group based random key predistribution methods.

3.2 Attack Model and Security Goal

We consider a dual operational mode adversary (passive and active) who is interested in revealing network data secrecy and forging the integrity of transmitted data. In our model, we assume that adversaries can compromise sensor nodes. When a sensor node is physically compromised, it means that an adversary gained control over the sensor node's operation, having access to its memory, keys, and resources, and is capable to reprogram such a compromised node with malicious code. Therefore, a compromised node can always generate and send false data and the proposed protocol cannot prevent that. Detecting compromised nodes that inject false data requires intensive monitoring mechanisms [28], and therefore it falls out of scope of this paper. In addition to false data injection, compromised sensor nodes can perform a wide range of attacks that disrupt the regular operation of a wireless sensor network such as denial-of-service attacks, Sybil attacks, or underlying routing protocol attacks. However, in this paper, we are only interested in preventing attacks that aim data confidentiality and authentication. Therefore, in SLB protocol a secret key is shared between any communicating party and all messages are encrypted and authenticated to provide data confidentiality and integrity, respectively.

4. SLB PROTOCOL

SLB protocol enables sensor nodes in congested clusters to relay their data securely to neighboring free clusters or pseudo-sinks to mitigate the accuracy problem. Sensor data are hierarchically aggregated at pseudo-sinks and cluster heads as much as possible to reduce the data transmission overhead of the network. In addition, before the deployment, an application depended data rate threshold ($DataRate_{Threshold}$) is defined for sensor nodes to balance their data transmission load. 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. Moreover, SLB protocol improves data aggregation efficiency by aggregating the data that cannot be aggregated at cluster heads due to the lack of shared keys. As sensor nodes take turns to be a cluster head to balance the energy consumption among sensor nodes, it may not be possible to have a shared key between every sensor node and

cluster head pair. In SLB protocol, such sensor nodes that do not share a secret key with the data aggregator send their data to the pseudo-sink of its deployment group. Fig. 2 presents the pseudo code of SLB protocol.

SLB Protocol

Input: Sensor node S_i , cluster head of S_i (CH_i), neighboring node set of S_i $\{S_1, \dots, S_j\}$, pseudo-sink of S_i 's deployment group ($Pseudosink_i$), 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.

```

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 offer list;
12:  end if
13: end for
14: // Select the best relay offer
15: if offer list not empty then
16:  Select  $CH_k.RelayOffer_{min}$ , the offer with the minimum  $chsize$ ;
17:  Send ACK to  $CH_k.RelayOffer_{min}$  and notify  $CH_i$  to release its resources;
18:  Encrypt the data  $D_i$  using the shared key between  $CH_k.RelayOffer_{min}$  and  $S_i$ ;
19:  Send encrypted  $D_i$  to  $CH_k.RelayOffer_{min}$ ;
20: else
21:  if  $S_i.DataRate > DataRate_{Threshold}$  and a shared key between  $S_i$  and  $CH_i$  exists
  then
22:    Encrypt the data using the shared key between  $S_i$  and  $CH_i$ ;
23:    Send encrypted data to the data  $CH_i$ ;
24:  end if
25: else
26:  Encrypt  $D_i$  using a shared key and send it to  $PseudoSink_i$ ;
27: end if

```

Fig. 2. SLB protocol.

SLB protocol is implemented in each data transmission session. Cluster heads assign time slots to their cluster members for data transmission and each sensor node S_i obtains the knowledge of its cluster head (CH_i) and the 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 re-

ceives 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 (Line 4 of SLB protocol). 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 all the relay offers from cluster heads that have a shared key with S_i . Then, 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 (Lines 16-17 of SLB protocol). 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. Once the relay link is established, S_i encrypts its data D_i using the key that it shares with CH_k . In addition, S_i also computes the Message Authentication Code (MAC) of D_i using the key that it shares with CH_k so that message integrity and source authentication is provided. S_i appends computed MAC(D_i) to encrypted D_i and sends this packet to CH_k .

If S_i does not have any relay offer that can improve its data rate, then S_i first checks if its data rate is above the predetermined threshold $DataRate_{Threshold}$. It also 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 D_i along with MAC(D_i) to CH_i for data aggregation (Lines 22-23 of SLB protocol). However, due to congestion, S_i 's data rate may be lower than $DataRate_{Threshold}$. In addition, as sensor nodes take turns to be a cluster head, it may not be possible for S_i to have a shared key with its current cluster head CH_i [5]. Hence, CH_i cannot decrypt and aggregate the data of S_i . In either of these cases, SLB protocol takes advantage of pseudo-sinks. If S_i 's data rate is below $DataRate_{Threshold}$ or S_i does not share a key with its current cluster head, it encrypts and authenticates the data using one of the keys that it shares with the pseudo-sink of its deployment group ($Pseudosink_i$) and sends the encrypted and authenticated data to $Pseudosink_i$ (Line 26 of SLB protocol). Note that after the network deployment, each pseudo-sink broadcasts its ID and the list of its key ID's to its deployment group. Hence, each sensor node is able to find the pseudo-sink of its deployment group and the key that it shares with its pseudo-sink. $Pseudosink_i$ 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.

Pseudo-sinks not only mitigate the accuracy problem but also improve the data aggregation efficiency by hierarchically aggregating cluster heads' data. As seen from Fig. 1, after the data collection, data aggregators aggregate the collected data and securely send this data to their pseudo-sinks for data aggregation thereby increasing data aggregation efficiency by hierarchical data aggregation. In the next section, we give a detailed analysis on how hierarchical data aggregation reduces the data transmission overhead of the network.

5. ANALYSIS OF HIERARCHICAL DATA AGGREGATION

In traditional data aggregation schemes, cluster heads aggregate the sensor data and

forward it to the base station over long distance communication links. SLB protocol reduces energy consumption between cluster heads and base station by employing the pseudo-sinks over the network. As pseudo-sinks are able to aggregate the sensor data which cannot be aggregated at the cluster heads, the number of data packets from sensor nodes to base station is reduced. The efficiency of the pseudo-sinks increases as the redundancy rate of the sensor data increases which results in an increment in energy efficiency and bandwidth utilization of the sensor network. In order to show energy efficiency and bandwidth utilization of SLB, in what follows, we propose an energy model for hierarchical data aggregation process in wireless sensor networks. For the sake of easiness, we do not consider the data relaying to neighboring clusters and assume that a sensor node's data is always aggregated by its original cluster head. This is a reasonable assumption because the relayed data is also aggregated at the neighboring cluster head and therefore it does not change the data aggregation efficiency. We also assume that data aggregators receive the sensor data which they cannot aggregate due to lack of shared keys and forward this data to the pseudo-sink without aggregating it. Hence, all data produced in a cluster C_j is forwarded to a pseudo-sink by the cluster head of C_j .

In wireless sensor networks communication among sensor nodes or cluster heads is over multi hop wireless links. Therefore the total data aggregation cost is proportional to number of packets that need to be transmitted from source to destination, usually the base station. Assuming that the sensor nodes are homogenously distributed over the network and packet size is fixed, we can reasonably assume that transmission cost of each packet/hop is the same. Suppose that cluster C_j is composed of c sensor nodes and each sensor i generates n_i packets per second, then X , the total number of packets per second received by CH_j is

$$X = \sum_{i=1}^c n_i \text{ packets/sec.}$$

If sensor nodes reach the cluster head over multi hop paths, then we can represent the number of hops from a sensor node i to cluster head CH_j as $H_{i,j}$. In this case, similar to the formula in [20], the total energy consumed by the sensor nodes to transmit raw data to cluster head would be proportional to

$$\sum_{i=1}^c n_i H_{i,j} \text{ packet-hops/sec.}$$

After receiving X packets per second, we assume that cluster head can aggregate pX ($0 \leq p \leq 1$) packets because it can decrypt them. The remaining data $(1-p)X$ packets cannot be decrypted as cluster head does not have decryption keys for them. The data aggregation process is assumed to lead the pX packets to be reduced to rX packets. So, the benefit of doing data aggregation is to eliminate the redundant data which is $(p-r)X$ packets. Since cluster heads forwards all non-aggregated data (*i.e.*, $(1-p)X$ packets) and the resultant data of data aggregation (*i.e.*, rX packets), cluster head forwards these packets at a rate:

$$r \sum_{i=1}^c n_i + (1-p) \sum_{i=1}^c n_i \text{ packets/sec.}$$

Note that the ratio r/p shows what percentage of those data that can be aggregated need to be forwarded by the cluster head. The ratio r/p is referred as the *aggregation rate* in this paper. Let $H_{j,ps}$ denote the distance between CH_j and pseudo-sink PS_{ps} , and $H_{j,BS}$ denote the distance between CH_j and the base station (BS). Assuming that CH_j 's aggregated data is aggregated at each cluster head on the way to the base station. The total energy that is consumed by all packets transmitted by CH_j is proportional to

$$\sum_{t=1}^{H_{j,BS}} r^t \sum_{i=1}^c n_i + (1-p) \sum_{i=1}^c n_i \text{ packet-hops/sec.}$$

Similarly, let m denote the aggregation rate of PS_{ps} and $H_{ps,BS}$ denote the distance between PS_{ps} and the base station. If there are s cluster heads sending non-aggregated data to the pseudo-sink PS_{ps} , then, assuming that PS_{ps} 's aggregated data will be aggregated at each cluster head on the way to the base station, the total energy cost of transmitting aggregated data from PS_{ps} to the base station can be shown as

$$m(1-p) \sum_{t=0}^{H_{ps,BS}} r^t \sum_{j=1}^s \sum_{i=1}^c n_i \text{ packet-hops/sec.}$$

Since there are s cluster heads sending data to pseudo-sink PS_{ps} , the total aggregation energy cost of PS_{ps} would be proportional to

$$\sum_{j=1}^s \left(\sum_{i=1}^c n_i H_{i,j} + \sum_{t=1}^{H_{j,BS}} r^t \sum_{i=1}^c n_i + (1-p) \sum_{i=1}^c n_i H_{j,ps} \right) + m(1-p) \sum_{t=0}^{H_{ps,BS}} r^t \sum_{j=1}^s \sum_{i=1}^c n_i$$

The final formula shows that the energy efficiency of the network is improved by pseudo-sinks and the benefit of pseudo-sinks increases as the amount of data that cannot be aggregated at cluster heads increases.

6. SIMULATION RESULTS AND SECURITY ANALYSIS

We have evaluated SLB protocol in terms of data accuracy, average data rate per sensor node, and data aggregation efficiency by generating random network instances with 400 nodes and various numbers of pseudo-sinks (2, 4, 6, 8). In the simulations, sensor nodes are assumed to be deployed as 4 deployment groups. A pseudo-sink is deployed along with a deployment group. When 2 pseudo-sinks used, only 2 of the deployment groups have a pseudo-sink. Each deployment group is assigned a key pool of size 1000. Each pseudo-sink and sensor node receives 100 and 30 keys from its deployment group's key pool, respectively. The key sharing probability of among sensor nodes

is defined in [5-7, 23] as follows:

$$p = 1 - \frac{((S-k)!)^2}{(S-2k)!S!}$$

where p : probability of key sharing, S : key pool size, and k : number of keys in a node. Following above formula, in SLB protocol, the probability of key sharing among sensor nodes is 61.345% and the probability of key sharing between a sensor node and pseudo sink is 98.749%. It should be noted that 98.749% key sharing probability between a sensor node and a pseudo-sink means that for every 100 sensor nodes only 2 sensor nodes will not be able to share a key with its pseudo-sink on average. Considering that wireless sensor networks are deployed with high node and coverage redundancy, a wireless sensor network can correctly perform its task even if 2 nodes out of every 100 nodes cannot communicate with its pseudo-sink. QualNet [22], a parsec based commercial sensor network simulator, is used for simulations. Lowest-id clustering algorithm is used to form the clusters in the network. Each cluster consists of 19 sensor nodes and a cluster head. The average neighboring degree of a sensor node is 7. The channel access scheme is chosen as TDMA and therefore data rate of nodes depend on the size of the cluster they belong to. As the placement of pseudo-sinks in the network is an important issue in SLB protocol, 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 1.5 dB to adapt the high packet loss rate of wireless sensor networks and the packet retransmission limit is set to 3.

First, the effect of SLB protocol over the sensor node and cluster head data rates is measured when 250 Kbps radio with 20 m communication range is used. The results are illustrated in Figs. 3 (a) and (b). As seen from Fig. 3 (a), as the number of 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, Fig. 3 (b) shows the variance of cluster head data rates. Like sensor node data rates, the variance of cluster head data rates reduces as the number of pseudo-sinks increases. The load balancing among sensor nodes prevents quick exhaustion of some cluster heads due to excessive data transmission and therefore positively affects the lifetime of the network as explained later in this section. It can be seen from Figs. 3 (a) and (b) that the initial variance of cluster head data rates is less than the initial variance of sensor node data rates. This is due to the sensor nodes in congested clusters that are not able to send their data to cluster heads. Also, Figs. 3 (a) and (b) show that uniform distribution of pseudo-sinks always yields better load balancing and fairness among sensor nodes. We have also evaluated the effect of sensor node communication range on SLB protocol. In the simulation, eight pseudo-sinks are uniformly placed in the network and communication range is varied from 15 m to 30m. Fig. 3 (c) shows that as the transmission range increases, fairness among sensor nodes increases as well. This improvement is due to the increased ability of sensor nodes to reach available cluster heads because with higher communication range sensor nodes are able to reach more cluster heads. However, it must be noted that increasing the transmission power significantly increases the energy consumption of sensor nodes [1].

The performance of SLB protocol in terms of mitigation of the accuracy problem is also evaluated and the results are presented in Figs. 4 (a) and (b). In the simulation sce-

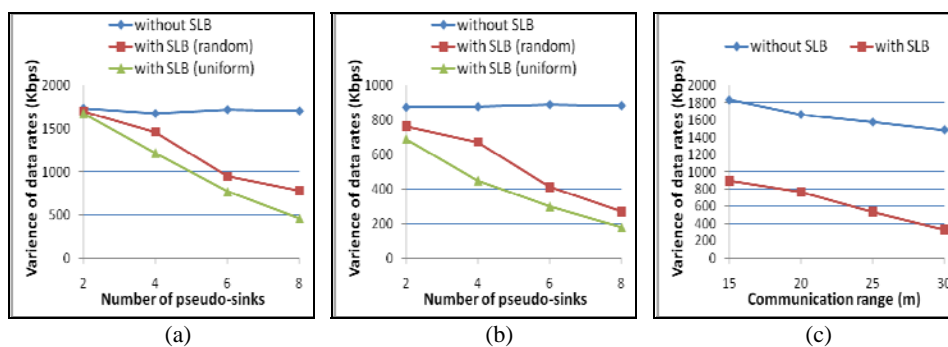


Fig. 3. (a) Variance of sensor node data rates for various numbers of pseudo-sinks; (b) Variance of cluster head data rates for various numbers of pseudo-sinks; (c) Variance of sensor node data rates versus communication range sensor nodes.

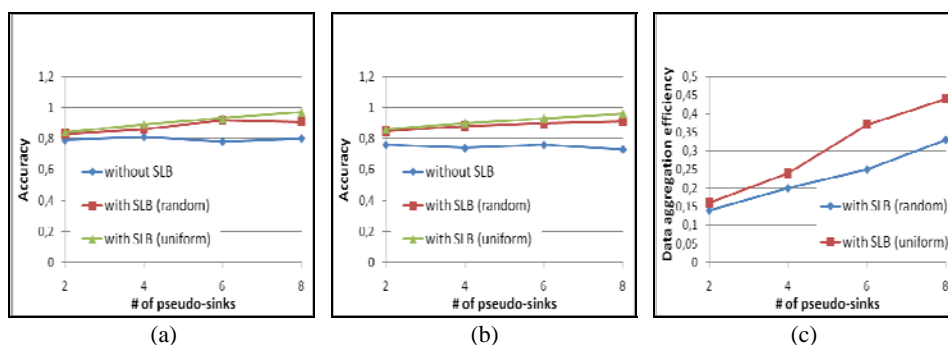


Fig. 4. (a) Accuracy of the data collected at the base station (data rate is 250 Kbps); (b) Accuracy of the data collected at the base station when sensor node (data rate is 400 Kbps); (c) Data aggregation efficiency of SLB protocol.

nario, sensor nodes measure temperature values from the environment; cluster heads collect this information and send it to the base station over pseudo-sinks. We measured the normalized difference (error) of the average temperature collected by the base station and the actual average temperature value of sensor nodes. The data accuracy of the network is defined as [1-Error in the collected data]. Fig. 4 (a) shows the data accuracy of the network when sensor nodes transmit data with 250 Kbps data rate. As seen from Fig. 4 (a), since all sensor nodes are able to send their data to cluster heads and pseudo-sinks, SLB protocol reduces the deviation from the correct aggregation results and increases the accuracy of the data collected at the base station. Similarly, Fig. 4 (b) shows the data accuracy of the network when sensor nodes transmit data with 400 Kbps data rate. It can be observed from Fig. 4 (b) that, without SLB protocol, if the sensor node data rate is high, the accuracy of the network is low. This is due to the high number of sensor nodes that cannot transmit their data because of the congestion in clusters. However, if SLB protocol is employed, the accuracy of the network is not affected by the data rate of sensor nodes, because pseudo-sinks balance the data load of cluster heads and all sensor nodes are able to send their data.

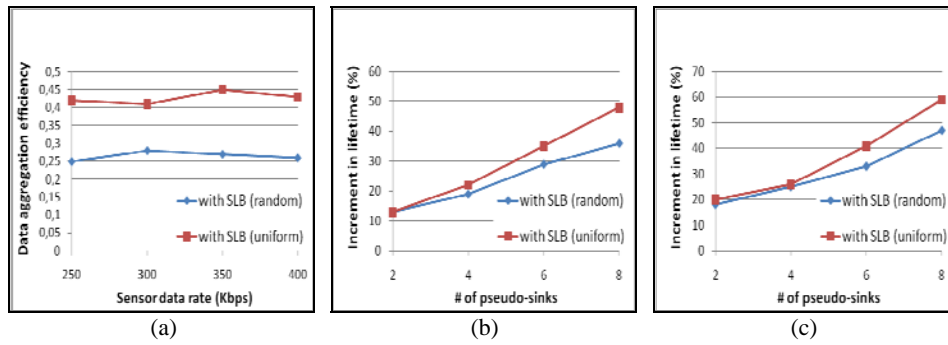


Fig. 5. (a) Effect of sensor node data rate on data aggregation efficiency; (b) Increment in network lifetime when sensor node data rate is 250 Kbps; (c) Increment in network lifetime when sensor node data rate is 400 Kbps.

The data aggregation efficiency of SLB protocol is also evaluated. The data aggregation efficiency of the network is defined as the ratio of the amount of aggregated data to the total amount of data that needs to be transmitted. Fig. 4 (c) shows the data aggregation efficiency of the network with respect to various numbers of pseudo-sinks. As seen from Fig. 4 (c), SLB protocol achieves high security and data aggregation efficiency together. The reason is that pseudo-sinks are able to decrypt and aggregate the data that cannot be aggregated at cluster heads due to non-existence of shared keys. Therefore, SLB protocol is suitable for sensor network applications that need both security and energy efficiency. In Fig. 5 (a), we show how the data aggregation efficiency of SLB protocol is affected by sensor node data rates. 6 pseudo-sinks are used in the simulation. Since pseudo-sinks collect and aggregate the data of sensor nodes that cannot transmit their data due to congestion, the data aggregation efficiency of the network is not affected by sensor node data rates as shown in Fig. 5 (a).

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. Figs. 5 (b) and (c) shows the increment in lifetime of the network due to SLB protocol. As seen from Fig. 5 (b), SLB protocol improves the network lifetime up to 50% when eight pseudo-sinks are uniformly distributed over the network and sensor nodes have 250 Kbps data rate. In Fig. 5 (c), we show the results of same simulation scenario with 400 Kbps sensor node data rate. As seen from Fig. 5 (c), SLB protocol improve the network life time up to 60% when sensor nodes have 400 Kbps data rate. The reason behind this improvement is that as the data rate of sensor nodes increases, pseudo-sinks collect and aggregate more data and hence positively affect the network lifetime.

The security of SLB protocol against compromised sensor nodes is analyzed to show how SLB protocol provides data confidentiality and authentication under the existence of compromised nodes. The key property of SLB protocol is to ensure that each sensor node shares a key with its cluster head and/or pseudo-sink to protect data confi-

dentiality and authentication. However, due to random key predistribution, each key in the network is possessed by more than one sensor node [7]. Therefore, we want to find the answer for the following question. Let k be a communication key used by sensor nodes A and B which are not compromised, what is the probability that the attacker can have the key k in the subset of the keys recovered from the compromised nodes? Assuming that each sensor node carries x keys in its key ring and there is only one compromised node, then the probability that k is not among the compromised node's key ring is $1 - (x/nS)$ where n is the number of deployment groups and S is the number of keys in each group key pool. If there are α compromised nodes, then the probability that k is not among the any one of compromised nodes' key ring is $[1 - (x/nS)]^\alpha$. Since this is the probability that k is not in the key rings of the compromised nodes, the expected fraction of keys being compromised is calculated as $(1 - [1 - (x/nS)]^\alpha)$. As seen from the last formula, as the number of keys in each sensor node increases SLB protocol's resiliency against node compromise reduces. However, as shown in [7], due to group deployment of sensor nodes and employment of small group key pools, SLB protocol results in better resilience against node compromise attacks compared to existing random key predistribution methods.

7. CONCLUSION

This paper has presented Secure Load Balancing (SLB) protocol for wireless sensor networks that mitigates the *accuracy problem* due to unbalanced data traffic load among clusters and increases the data aggregation efficiency under the security constraints. The novelty of SLB protocol comes from considering the load balancing problem along with security and data aggregation requirements of wireless sensor networks. To the best of our knowledge, there is no existing work that considers *security* and *load balancing* concepts together. SLB protocol employs pseudo-sinks to balance sensor data rates securely, thereby improving data accuracy. In addition, pseudo-sinks help increase the data aggregation efficiency of the network. The performance analysis and simulation results show that SLB protocol ensures the data accuracy and improves the network lifetime by increasing the energy efficiency and bandwidth utilization while still providing secure communication.

ACKNOWLEDGEMENT

The author is grateful to editors of the journal and the anonymous reviewers for their cooperation and comments.

REFERENCES

1. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Communications Magazine*, Vol. 40, 2002, pp. 102-114.
2. C. Intanagonwiwat, D. Estrin, R. Govindan, and J. Heidemann, "Impact of network density on data aggregation in wireless sensor networks," in *Proceedings of the 22nd*

- International Conference on Distributed Computing Systems*, 2002, pp. 575-578.
3. A. Ephremides, J. E. Wieselthier, and D. J. Baker, "A design concept for reliable mobile radio networks with frequency hopping signaling," *Proceedings of IEEE*, Vol. 75, 1987, pp. 56-73.
 4. A. Perrig, R. Szewczyk, J. D. Tygar, V. Wen, and D. E. Culler, "Spins: security protocols for sensor networks," *Wireless Networking*, Vol. 8, 2002, pp. 521-534.
 5. L. Eschenauer and V. D. Gligor, "A key-management scheme for distributed sensor networks," in *Proceedings of the 9th ACM Conference on Computer and Communications Security*, 2002, pp. 41-47.
 6. H. Chan, A. Perrig, and D. Song, "Random key predistribution schemes for sensor networks," in *Proceedings of IEEE Symposium on Security and Privacy*, 2003, pp. 197-213.
 7. W. Du, J. Deng, Y. S. Han, and P. Varshney, "A key predistribution scheme for sensor networks using deployment knowledge," *IEEE Transactions on Dependable and Secure Computer*, Vol. 3, 2006, pp. 62-77.
 8. G. Li, H. Ling, and T. Znati, "Path key establishment using multiple secured paths in wireless sensor networks," in *Proceedings of ACM Conference on Emerging Network Experiment and Technology*, 2005, pp. 43-49.
 9. I. F. Akyildiz and I. H. Kasimoglu, "Wireless sensor and actor networks: research challenges," *Ad Hoc Networks*, Vol. 2, 2004, pp. 351-367.
 10. J. J. Lee, B. Krishnamachari, and C. C. J. Kuo, "Impact of heterogeneous deployment on lifetime sensing coverage in sensor networks," in *Proceedings of the 1st IEEE International Conference on Sensor and Ad Hoc Communications and Networks*, 2004, pp. 367-376.
 11. <http://www.intel.com/research/exploratory/heterogeneous.htm>, 2007.
 12. G. N. Aggelou and R. Tafazolli, "On the relaying capability of next-generation GSM cellular networks," *IEEE Personal Communications*, Vol. 8, 2001, pp. 40-47.
 13. V. Sreng, H. Yanikomeroglu, and D. D. Falconer, "Coverage enhancement through two-hop relaying in cellular radio systems," in *Proceedings of IEEE Wireless Communications and Networking Conference*, Vol. 3, 2002, pp. 880-884.
 14. S. De, O. Tonguz, H. Wu, and C. Qiao, "Integrated cellular and ad hoc relay (iCAR) systems: Pushing the performance limits of conventional wireless networks," in *Proceedings of the 35th Hawaii International Conference on System Sciences*, 2002, pp. 3931-3938.
 15. L. Hu and D. Evans, "Secure aggregation for wireless networks," in *Proceedings of Workshop on Security and Assurance in Ad Hoc Networks*, 2003, pp. 384-394.
 16. B. Przydatek, D. Song, and A. Perrig, "SIA: secure information aggregation in sensor networks," in *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems*, 2003, pp. 255-265.
 17. H. Cam, S. Ozdemir, P. Nair, D. Muthuavinashiappan, and H. O. Sanli, "Energy-efficient and secure pattern based data aggregation for wireless sensor networks," *Computer Communications*, 2006, pp. 446-455.
 18. S. Ozdemir, "Functional reputation based reliable data aggregation and transmission for wireless sensor networks," *Computer Communications*, Vol. 31, 2008, pp. 3941-3953.
 19. S. J. Seung, G. de Veciana, and X. Su, "Minimizing energy consumption in large-

- scale sensor networks through distributed data compression and hierarchical aggregation,” *IEEE Journal on Selected Areas in Communications*, Vol. 22, 2004, pp. 1130-1140.
20. Crossbow Technologies Inc., www.xbow.com, 2008.
 21. Intel Inc., <http://www.intel.com/research/exploratory/heterogeneous.htm>, 2008.
 22. QualNet Network Simulator, <http://www.scalable-networks.com/>, 2008.
 23. D. Liu, P. Ning, and W. Du, “Group-based key pre-distribution in wireless sensor networks,” in *Proceedings of the 4th ACM Workshop on Wireless Security*, 2005, pp. 11-20.
 24. A. Abbasi and M. Younis, “A survey on clustering algorithms for wireless sensor networks,” *Computer Communications*, Vol. 30, 2007, pp. 2826-2841.
 25. A. Silberstein, *et al.*, “Data-driven processing in sensor networks,” in *Proceedings of the Conference on Innovative Data Systems Research*, 2007, pp. 10-21.
 26. S. Capkun and J. P. Hubaux, “Secure positioning in wireless networks,” *IEEE Journal on Selected Areas in Communications*, Vol. 24, 2006, pp. 221-232.
 27. J. Blumenthal, F. Reichenbach, and D. Timmermann, “Minimal transmission power vs. signal strength as distance estimation for localization in wireless sensor networks,” in *Proceedings of the 3rd Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, Vol. 3, 2006, pp. 761-766.
 28. S. Ganeriwal, L. K. Balzano, and M. Srivastava, “Reputation based framework for high integrity sensor networks,” *ACM Transactions on Sensor Networks*, Vol. 4, 2008, pp. 1-37.



Suat Ozdemir has been with the Computer Engineering Department at Gazi University, Ankara, Turkey since March 2007. He received his M.Sc. degree in Computer Science from Syracuse University (August 2001) and Ph.D. degree in Computer Science from Arizona State University (December 2006). His main research interests include broad areas of wireless networks and network security. He serves on TPC for several conferences such as ICC, GLOBECOM, Sensor Networks, Information Security and Cryptography, *etc.* Dr. Ozdemir also serves as a reviewer for several journals, *e.g.*, Security and Communication Networks, IEEE Transactions on Mobile Computing, IEEE Transactions on Parallel and Distributed Systems, Computer Communications. Dr. Ozdemir is a member of IEEE.