Armor-LEACH for Wireless Sensor Network

M.A. Abuhelaleh T. M. Mismar A. A. Abuzneid
School of Engineering, University of Bridgeport
Bridgeport, CT 06604
{mabuhela, tmismar, abuzneid} @bridgeport.edu

Abstract

The use of sensor networks is increasing day by day; which offer more research topics to be discuss and modified; one of these topics is the power consumption that has to be reduced as possible, where the resources are limited; another topic is the security level that should be offer by such kind of networks. Clustered networks have been proposed in many papers to reduce the power consumption in sensor networks. LEACH is one of the most interested techniques that offer an efficient way to minimize the power consumption in sensor networks. TCCA provides LEACH with higher performance, by applying some modification to the way LEACH works. In this paper we combine two of the most powerful proposed techniques that can be applied on LEACH to reduce the power consumption and to increase the level of security.

1. Introduction

Special applications with high limited and constrained resources need special kind of networks that can handle their needs. Wireless sensor networks (WSNs) most of the time are the perfect solutions for these kinds of applications, where sensors are distributing around the base stations (BSs) and sensing reports are transmitting from these sensors to these BSs. Many researches in recent years were focused on this kind of networks and they mainly proposed different ways to provide this kind of networks with high throughputs, low delays, and less power consumption.

Few papers focused on the security of WSNs even that the nature of this kind of networks may leads to low level of security which make these networks easy targets for intruders. Low Energy Adaptive Clustering Hierarchy (LEACH) [1] provides an efficient communication protocol for WSNs to reduce power consumption and at the same time to provide some level of security. Secure LEACH (Sec-LEACH) [2] provides high level of security to LEACH without affecting its performance; in Sec-LEACH the random Key Distribution (KD) technique has been applied dynamically on LEACH communication. Time Controlled Clustering Algorithm (TCCA) [3] presented to increase the efficiency of energy saving for LEACH, where they apply the time factor to control the distribution and the communication of the WSNs with LEACH protocol.

LEACH organize the communication between nodes in WSNs in clever way, where the network is divided into clusters with cluster head (CH) for each cluster; the CHs mainly responsible about communicate the other nodes (sensors) with the BS to reduce the total energy consumption of the network. The CHs are to be elected based on the desired percentage of CHs and the round number; CHs are to be changed dynamically after each round. Sec-LEACH presents some modifications on the way that nodes, CHs and the BS communicate with each other, where random KD schema has to be applied for each transaction of the network communication to prevent many kinds of possible attacks. TCCA modified the way CHs have to be elected by adding another condition for electing CH; this condition is the availability of the energy comparing to the maximum energy of the node, which is controlled by time stamp. This algorithm reduced the power consumption of LEACH by good percentage.

In this work, we combine both solutions provided by Sec-LEACH and TCCA in one complete solution, to offer a high level of security on WSN with low power consumption. In the second section we describe LEACH protocol and we briefly discuss some of existing works on this protocol. In the third

section we describe the Sec-LEACH protocol and we highlight the main improvement that offered by this protocol to the LEACH. In the fourth section we describe TCCA with its modifications to the original LEACH; in the fifth section we describe how to combine TCCA with Sec-LEACH in one protocol to provide one complete solution, Armor-LEACH, to WSNs that offers high level of security and high power of power saving. In the sixth section we analyze our solution comparing to the original protocols in order to maintain the complete performance of this hybrid solution.

2. LEACH

Low Energy Adaptive Clustering Hierarchy has been presented by [1] to balance the draining of the energy during communication between nodes in sensor networks. The BS assumed to be directly reachable by all nodes by transmitting with high enough power. Nodes send their sensor reports to their CHs, which then combine the reports in one aggregated report and send it to the BS. To avoid the energy draining of limited sets of CHs, LEACH rotates CHs randomly among all sensors in the network in order to distribute the energy consumption among all sensors. It works in rounds; in each round, LEACH elects CHs using a distributed algorithm and then dynamically clusters the remaining sensors around the CHs. Sensor-BS communication then uses this clustering result for the rest of the round. (see Figure1)

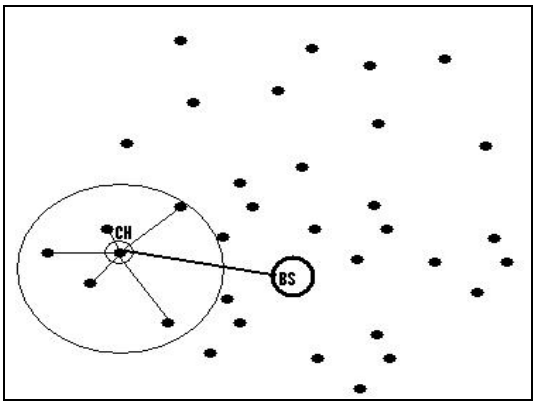

Figure1. Cluster organization for sensor networks

2.1 LEACH Protocol

Routing in LEACH works in rounds and each round divides into two phases, the Setup phase and the Steady State; each sensor know when each round starts using synchronized clock.

Initially, each sensor decides if it will be a CH or not based desired percentage of the CHs for the network and the number of times the sensor has been a CH (to control the energy consumption), this decision is made by the sensor s choosing a random number between Zero and One. Then it calculates the threshold for s T(s), then compare the random number with result T(s); if the number is less than T(s), s becomes CH for the current round. T(s) for x round with desired percentage of cluster heads P is calculated by:

$$T(s) = \begin{cases} \dfrac{P}{1 - P * (x \bmod \frac{1}{P})} & \text{if } n \in G \\ 0 & \text{otherwise} \end{cases}$$

G is a set of nodes that have not been CHs in the last 1/p round.

Setup phase includes three steps. Step1 is the advertisement step, where each sensor decides its probability to become a CH, based on the desired percentage of CHs and its remaining energy, for the current round; Sensors who decides to be CHs broadcast an advertising message to other nodes that they are ready to become CHs. Carrier sense multiple access protocol is used to avoid the collision. Clustering joining step is the second step, where the remaining sensors pick a cluster to join according to the highest signal received; then they send request messages to desired CHs. Step three starts after CHs receive all requests from other sensors, where CHs broadcast confirmation messages to their cluster members; these messages include the time slot schedule to be used during the steady state phase.

The Steady State phase (the actual communication) then starts and it consists of two steps; in the first step each nodes start to send its sensor report to its CH based on the time provided by the time slot schedule. When CH receives all reports, it aggregates them in one report and it sends this report to the BS (step 2). Next we show the details of each step by providing the content of each one, for this purpose we combine the two phases in one phase with five steps.

In step one, CH broadcasts to the rest of sensors, its ID and the Advertising message, then, in step two, each sensor sends its ID, CH ID, and the Join Request message to its desired CH. When CH received all requests, it will broadcast its ID, and the time slot schedule for sensors that includes each member with its time slot (step three). Each sensor then sends its ID, CH ID, and the sensing report to its CH (step four). Finally, each CH sends its ID, BS ID, and the aggregate report of all it members to the BS.

The transmission of information between sensors and between sensors and BSs are perform using CSMA MAC protocol; In other hand, all are communicate using CDMA codes to reduce the interference that may occur from communication of nearby nodes.

2.2 Energy saving in LEACH

LEACH provides many techniques to save the total energy during network communication; in this section we focus on the main concepts that have been applied by LEACH to save the energy.

LEACH is self-organization, adaptive protocol and it uses randomization to evenly distribute the energy load among the sensors in the network; in addition to the random way that CHs rotate around the various sensors is reducing the possible draining of the battery for each sensor.

Performing a local data compression to compress the amount of data being sent from clusters to BS reduces the energy consumption and it also enhances the system lifetime.

The time schedule that being performed by CHs to their members, gives break time for sensors that are not reach their time yet to be in sleeping mode which helps them to save their energy for their scheduled time.

Finally, the nature of the way that LEACH change CHs each round and the way that each CH can be elected provide high energy saving for the whole network.

2.3 Security in LEACH

LEACH is more powerful against attacks than most other routing protocols [4]. CHs in LEACH that are directly communicate with BS can be anywhere in the network and they are changing from round to round, which make it harder for intruders to identify the critical nodes in the network.

On other hand, LEACH vulnerable to a number of security attacks [4], including spoofing, jamming, and replay attacks. Since LEACH is a cluster based protocol, it relies mainly on the CHs for routing and data aggregation, which makes the attacks involving CHs, the most damaging attacks.

Some kinds of attacks, such as sinkhole and selective forwarding, may occur if an intruder manages to become a CH, which result in disrupting the working of the network.

3. Sec-LEACH

It proposes a new modification for LEACH to increase the level of security and to protect the network from many kinds of attacks, specially the sinkhole and selective forwarding attacks [2].

Sec-LEACH proposes to generate a large pool of keys and their IDs at the time the network is deployed. Each sensor then is assigned a ring of keys taken from key pool pseudo randomly [5]. First it generates a unique ID for each sensor using pseudo randomly function (PRF), then a large enough number of keys is assigned to each sensor from the key pool; also assign each node by a pair-wise key shared with the BS.

3.1 Sec-LEACH Protocol

When elected CH broadcast its advertising message, it includes the ID of the keys in its key ring, the other sensors cluster around the nearest CH with whom they share a key. The details of Sec-LEACH protocol as follows [2].

CHs are elected as in LEACH, and then these CHs broadcast their IDs and a nonce (step 1). In step 2, other sensors computes the set of CHs key IDs and choose the nearest CH with whom they share a key; these sensors then send the Join Request messages, protected by MAC that produced by the share key, and the nonce that broadcasts by CH, to prevent reply attack; the ID of the key chosen to protect the link is also sends with the message in order for CH to know which key to use to verify the MAC. To complete the setup phase, CHs send the time schedule to sensors that choose to become their members (step3).

Step4 is the first step in Steady State phase, where sensor-to-CH communications are protected using the same key used to protect the Join Request message. To prevent replay, a value computed from the nonce and the reporting cycle is also included. CH then decrypts sensors reports and performs a data aggregation then it sends it to the BS protected by the symmetric key shared with the BS. A counter is included in the MAC value also, to provide freshness.

3.2 Energy saving in Sec-LEACH

Sec-LEACH works as original LEACH, but here some extra bits have to be added to the total transactions that occur during the communication in the network. As discussed in [2], these overloads will not affect the efficiency of the original LEACH if suitable size of the key pool and suitable number of keys assigned to each sensor are chosen.

3.3 Security in Sec-LEACH

Sec-LEACH provides more protection to the network than it is in LEACH, where it protects against spoofing, jamming, and replay attacks. In addition, it prevents sinkhole and selective forwarding attacks.

4. TCCA

Time-Controlled Clustering Algorithm (TCCA) allows multi-hop clusters using message time-to-live (TTL) and timestamp to control the way the clusters form. Residual energy is also considered before volunteering of a sensor to become a CH, and a numerical model is provided to quantify its efficiency on energy usage.

4.1 TCCA Protocol

Similar to LEACH, TCCA operation is divided into rounds with two phases conclude in each round (Setup phase and the Steady State phase). CHs are elected and the clusters are formed in Setup phase; then the complete cycle of data collection, aggregation and transfer to the BS is occur in the Steady State phase.

To determine the eligibility of sensor to be CH, TCCA adds some modifications to LEACH technique. A sensor residual energy is considered and a random number between 0 and 1 (Tmin) is to be generating by each sensor to determine its eligibility to become CH. If this number is less than the variable threshold, the sensor becomes a CH for the current round. The threshold for sensor s in round r, with desired CH percentage *p*, residential energy RE and maximum energy MaxE is calculated by:

$$T(s) = \left(\frac{p}{1 - p(r \bmod \frac{1}{p})} \times \frac{RE}{MaxE}, T\min\right) \forall s \in G$$

$$T(s) = 0 \qquad \forall s \notin G$$

G is a set of nodes that have not been CHs in the last 1/p round

When CH is elected, it advertises to other sensors to become its members; this advertisement message contains CH ID, initial TTL, timestamp and its residual energy. Sensors received the message will forwarded to their neighbors based on TTL value which may be based on the current energy level of CH; at the same time they join this CH with the rest of sensors who received the message. Once a sensor decides to join the cluster, it informs the corresponding CH by sending a join request message carries sensor ID, CH ID, the original timestamp from advertising message and the remaining TTL value.CH uses the timestamp to approximate the relative distance of its neighbors and to learn the best setup phase time for future rounds.

The time schedule that is to be advertised by CH is based on the total number of its members and their relative distance to avoid the collision.

Timestamp and TTL are used in TCCA to give the CH the ability to produce multi-hops clusters in efficient way that has the same performance of the one-hop clusters.

4.2 Energy saving in TCCA

TCCA applies new condition to electing CHs by considering the remaining energy of the sensors. At the same time, it guarantees that every sensor will become a CH at least one time per 1/P rounds, where P is the desired percentage of CHs. These modifications provide the network with high energy balance by distributed the energy among all sensors.

TCCA provides optimum cluster size (K) for K-hops in order to produce high performance to the network that is similar to the performance in one-hop. Also it reduces the complexity of transmission schedule generation to O (1).

TCCA uses timestamp and Time to live (TTL) tags to control the cluster formation; this leads to gain more energy balance.

4.3 Security in TCCA

TCCA follows the main steps provided by LEACH with some modifications that do not affect the level of security that is provided by LEACH; this means that TCCA does not have enough protection against Spoofing, Jamming, Replay and some other kind of attacks.

5. Armor-LEACH

The operation of Armor-LEACH combines the operations of TCCA and SecLEACH. TCCA provides LEACH with more energy control results in less power consumption. SecLEACH provides LEACH with high level of security against many kinds of attacks.

Armor-LEACH proposes a complete powerful solution for sensor networks communications, where it offers high level of security with high performance.

5.1 Armor-LEACH Protocol

The operation of Armor-LEACH is occurs in rounds, and each round pass through five steps divided into two phases, Setup phase and Steady State phase.

Prior to network deployment, each sensor assigned by group of keys randomly from big key pool provided by the BS. A pseudorandom function is used to produce keys IDs which then map to their corresponding values in the key pool. Also each sensor assigned by a pair-wise key share with the BS for secure direct communication.

At the beginning of each round, CHs elect themselves. In order to determine the eligibility of sensor to be a CH, each sensor (S) generates a random number between 0 and 1; then this number compared by sensor variable threshold value T(S). if the value of threshold is greater than the random number, the sensor become CH for the current round (R). The threshold value can be calculated as follows:

$$T(S) = \begin{cases} \max\left(\dfrac{P}{1 - P\left(R \bmod \frac{1}{P}\right)} \times \dfrac{RemainingEnergy}{MaximumEnergy}, T\min\right) & if\ S \in G \\ 0 & otherwise \end{cases}$$

Where P is the desired percentage of CHs, Tmin is a minimum threshold (to avoid the possibility of remaining energy shortage), and G is the set of sensors that have not became CHs in 1/P round.

Setup phase is then begins with the first step (step1). Each elected CH broadcasts an advertising message to its neighbors announcing that it is a CH for current round; this message consists of CH ID, nonce, initial Time to live (TTL), remaining energy, timestamp and the advertising message. When sensor receives the message, it choose a key that is share with CH to use in the current round, and at the same time it will forward the message to its next neighbors based on TTL value which is calculated depends on the current remaining energy for CH. For step2 in Setup phase, each sensor, receives an advertising message, sends its request to the CH to join its cluster. The request message consists of sensor ID, CH ID, sharing key ID, join request message, original advertising message timestamp, the remaining TTL value, and the encryption of sensor ID, CH ID, sharing key ID, and the nonce sent by CH; the encryption is produced using message authentication code, which is produced using the sharing key. Timestamp helps CH to approximate the relative distance of its members and to learn the best Setup phase time to be used in future rounds. TTL with time stamp helps CH to form a multi-hops view of its clusters, in order to create a collision-free transmission schedule.

In step3, CH broadcasts its ID and the time slot schedule for each sensor in its cluster to organize the communication between nodes in the cluster.

The real communication begins in Steady State phase. In step1, each sensor sends its report to its CH; the report message consists of sensor ID, CH ID, sensor report, and the encryption of sensor ID, CH ID, sensor report, and the nonce with its reporting cycle within the current round; the encryption produce using the same MAC that is produced in step2 in Setup phase. In step2, CH sends the aggregation of sensors reports to the BS; this message consists of CH ID, BS ID, the aggregation of sensors reports, and the encryption of the aggregation report, encrypted using MAC that is produced using the sharing key between CH and the BS.

Using random key distribution (RDK) to create sharing keys leads to mach most of sensors with available CHs. Small number of sensors may not have sharing keys; these sensors may follow another technique. [3] Suggests two different techniques to deal with these kinds of sensors (orphans). The first solution is to make these orphans communicate directly with the BS in the current round. The second solution is to have these orphans sleep for the current round. In both scenarios, the number of orphans in each round depends on the size of the key pool, the size of the group of keys of each sensor, and the number of CHs.

5.2 Security in Armor-LEACH

In next section, we presents the main security issues that covered by our solution in order to know what improvements have been applied to LEACH; then we discuss the main factors that may affect the performance of security level in our solution.

5.2.1 Security issues

Armor-LEACH provides authenticity, confidentiality, integrity, and freshness to sensor-to-sensor communications. The message in the second step of the Setup phase, is encrypted with a key from the key pool; CH can conclude that the message came from legitimate sensor in the network by successfully decrypt the message; Also it can conclude that the message is not a stale message being replay form the nonce that is encrypted in the message. Nonce value is incremented every cycle which guarantee a freshness of all subsequence sensor reports from the sensors to their BS. By the counter value shared between the CH and the BS in step2 in Steady State phase, the freshness is guaranteed.

The biggest security issue in Armor-LEACH is the resiliency against sensor captures, where the link keys used for sensor-to-CH communications are not pairwise in Armor-LEACH.

5.2.2 Factors: Affect performance

In our solution we generate a key pool that contains big enough numbers of keys to be used during network communications. For WSNs, the size of the key group that is to be assigned to each sensor is almost fixed. Once the key group size (KGS) is set, the choice of the key pool size (KPS) will affect the system performance in two ways:

1. The sharing key probability between sensors:

The probability (Pr) that two sensors, with specific KGS and KPS, will share a key is calculated by:

$$\Pr = 1 - \frac{[(KPS - KGS)!]^2}{KPS!(KPS - 2KGS)!}$$

When KGS is fixed, the size of KPS will affect this probability; the larger the KPS, the smaller the Pr [2]. 2. The security level

The probability that a link that is choosen randomly is not compromise when a sensor is not either end of the link is compromise is based on KPS and KGS and it determine the security level by the following formula:

Security Level= $1-\frac{KGS}{KPS}$

Given fixed KGS, KPS will determine the level security; the larger the KPS, the higher the security level [2]. (See Figure2)

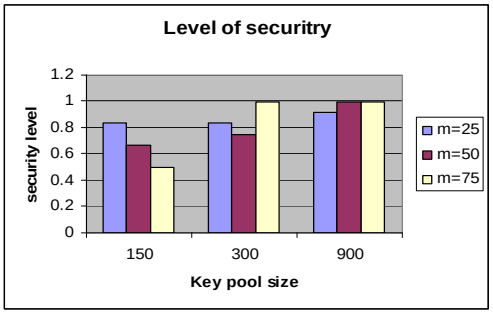

Figure2. Security level affected by KGS and KPS

6. Analysis

In this section we compare our solution with LEACH, Sec-LEACH, and TCCA solutions. We start with energy saving, then we discuss the expected dead nodes (network performance) for each solution after specific number of rounds, then we reevaluate the security level that can be applied by each solution.

6.1 Energy saving

LEACH provides many techniques to save the total energy during network communication where it is a self-organization, adaptive protocol and it uses randomization to evenly distribute the energy load among the sensors in the network; in addition to the random way that CHs rotate around the various sensors is reducing the possible draining of the battery for each sensor. Also, performing a local data compression to compress the amount of data being sent from clusters to BS reduces the energy consumption and it also enhances the system lifetime. These factors, in addition to way that CHs changing every cycle, provide LEACH with High energy saving. Armor-LEACH applies the same factors to Sensor networks which provide it with similar energy saving to the LEACH at this point.

TCCA adds additional factors to save energy; it uses timestamp and Time to live (TTL) tags to control the cluster formation, which leads to gain more energy balance; also it adds new condition to elect CHs each round results in more energy control. [3] Shows that TCCA works almost three times better than LEACH according to energy saving. Armor-LEACH applied TCCA factors, which means that it works three times better than LEACH according to energy saving. Sec-LEACH applied the same factors that have been applied by LEACH, which mean that Armor-LEACH is also works three times better than Sec-LEACH, according to energy saving.

6.2 Performance

Here, we analyze the performance based on the expected Dead Nodes that may results in each solution after same number of rounds.

According to energy saving analysis, we can figure that number of Dead Nodes that may appear in LEACH and Sec-LEACH may reach triple value of what may appear in TCCA and Armor-LEACH, where number of Dead Nodes depends on the energy consumption by the network.

6.3 Security

Sec-LEACH provides flexible level of security that is affected by the key pool size and the group key size. And in the worst case, Sec-LEACH provides more security to sensor networks than what is provided by TCCA and LEACH. Armor-LEACH applies the techniques that are applied by Sec-LEACH, which provide it with same level of security that has been applied by Sec-LEACH.

7. Conclusions

Armor-LEACH provides sensor networks with high energy saving, and high level of performance, three times better than LEACH and Sec-LEACh. At the same time it produces higher level of security than it produced by LEACH and TCCA. These results produced a very efficient solution for sensor networks communications.

Mohammed Abuhelaleh is a full-time Ph.D. student of Computer Science and Engineering at the University of Bridgeport. He worked as a lecturer for some computer science courses in addition to college courses like Data structure, computer skills 1 & 2 in Alhusein Bin Talal University / Jordan for three years. He has master degree computer science from University of Bridgeport, and graduated with a GPA of 3.48. Mohammed now is in second semester of PHD program, and he is working as a graduate assistant for prof. Elleithy at Engineering and Computer Science department at the University of Bridgeport.

Thabet Mismar is a full-time M.Sc. student of Electrical Engineering at the University of Bridgeport. He has B.Sc. degree of Electrical Engineering from the University of Jordan. Thabet is now in the second semester of the M.Sc. program, and he is working as a graduate assistant for Prof. Elleithy at the dean of engineering and technolgy office at the University of Bridgeport.

Abdelshakour Abuzneid has received his BS degree in Computer Engineering and Control from Yarmouk University and MS degree in Computer Engineering from the University of Bridgeport in May 2007. Currently he is pursuing his PhD in Computer Science & Engineering from the University of Bridgeport. His research interest is in Data / computer / wireless / mobile communications. He has published few journal and conference papers. Abelshakour has 12 years of experience in this field.

REFERENCES

[1] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. In IEEE Hawaii Int. Conf. on System Sciences, pages 4–7, january 2000.

[2] Leonardo B. Oliveira, Hao C. Wong, M. Bern, Ricardo Dahab, A. A. F. Loureiro. SecLEACH - A Random Key Distribution Solution for Securing Clustered Sensor Networks. Fifth IEEE International Symposium on Network Computing and Applications (NCA'06)

[3] S. Selvakennedy, and S. Sinnappan. A Configurable Time-Controlled Clustering Algorithm for Wireless Networks. 2005 11th International Conference on Parallel and Distributed Systems (ICPADS'05).

[4] Chris Karlof, Naveen Sastry, and David Wagner. TinySec: A Link Layer Security Architecture for Wireless Sensor Networks. 2004 Conference on Embedded Networked Sensor Systems Proceedings of the 2nd international conference on Embedded networked sensor systems.

[5] Sencun Zhu, Shouhuai Xu, Sanjeev Setia, and Sushil Jajodia. Establishing Pairwise Keys for Secure Communication in Ad Hoc Networks: A Probabilistic Approach. Proceedings of the 11th IEEE International Conference on Network Prot1ocols (ICNP'03)