朝陽科技大學

資訊工程系

碩士論文

一種動態的金鑰管理及備用雙層資料匯集網格無線感測網路之研究

The Study of a Dynamic Session Key Management and Spare Two-Tier Data Aggregation for Grid-Based Wireless Sensor Networks

指 導 教 授：陳金鈴 博士

研 究 生：林宜賢

中華民國 99 年 1 月

朝陽科技大學資訊工程系

Department of Computer Science and Information Engineering

Chaoyang University of Technology

碩士論文

Thesis for the Degree of Master

一種動態的金鑰管理及備用雙層資料匯集網格無線感測網路之研究

The Study of a Dynamic Session Key Management and Spare Two-Tier Data Aggregation for Grid-Based Wireless Sensor Networks

指 導 教 授： 陳金鈴 博士 (Chin-Ling Chen)

研 究 生： 林宜賢 (I-Hsien Lin)

中華民國 99 年 1 月

January 2010

摘要

在軍事和其他敵對的環境中，安全性是無線感測網路的關鍵的議題。例如，當在無線感測網路裡的無線感測節點被散佈在不安全的地區時，一把秘密的金鑰必須用來保護被傳送的訊息。我們提出了一種動態的金鑰管理及備用雙層資料匯集網格無線感測網路，能改進金鑰的安全性並且使能源消耗和網路故障容錯減到最小。另外，我們的協定包含動態更新金鑰的方式，能降低被猜測金鑰的可能性，及防止目前已知的攻擊方式。透過利用本地訊息，我們的方法為了路由降低選擇通路過程中消耗的能量，因此限制廣播泛濫區域(Flooding Region)。在每個網格裡會有一個無線感測節點被選擇為叢集頭(Header)。叢集頭的責任是偵測發生的事件，並對其他叢集頭宣佈事件，且對移動的匯集點(Mobile Sink)作出回應。一個移動的匯集點從它所在的網格中的叢集頭獲得發生的事件訊息。每個網格有兩個叢集頭，全部叢集頭會被隨機分類成為兩個群組 group-0 和 group-1，能達到容錯，二條路徑(path-0 為 group-0 和 path-1 為 group-1)被建立。本文提出在根節點(Root)和移動的匯集點之間的二條路徑，使能源消耗和網路故障容許減到最小。最後，我們經由一些模擬顯示我們的方法優於 CODE 和 TTDD，結果顯示我們的方法較現有的方法更有效降低能量效耗。

關鍵詞：攻擊、以網格為基礎、金鑰管理、安全、無線感測器網路

Abstract

Security is a critical issue for sensor networks used in military and other hostile environments. For example, when wireless sensor nodes in a wireless sensor network are distributed in an insecure area, a secret key must be used to protect the transmitted messages. We propose dynamic session-key management and a spare two-tier data aggregation for grid-based wireless sensor networks which can improve the security of the key and minimize energy consumption and network fault tolerance. In addition, our protocol includes a key that is dynamically updated, which can lower the probability of the key being guessed correctly and thus defend against currently known attacks. By utilizing the local information, the proposed scheme also can limit the flooding region in order to reduce the energy consumed in discovering routing paths. A sensor node is selected to be a header, in each grid. The responsibilities of header are to detect generated event, announce to all other headers, and respond to a mobile sink. A mobile sink obtains the occurred event information from its grid header. Each grid has two headers. All header nodes are randomly classified into two types, group-0 and group-1. To achieve the fault tolerance, two paths (path-0 for group-0 and path-1 for group-1) were constructed. This work provides two paths between roots and mobile sinks. This work can minimize energy consumption and network fault tolerance. Finally, we conducted some simulations to show that the thesis outperforms the CODE and TTDD. Results show the approaches are more energy-efficiency than the existing approaches.

Keywords: attack; grid-based; key management; security; wireless sensor network

致謝

在朝陽受到師長的栽培及鼓勵，讓我在做研究及做人處事上收穫良多，使我能有足夠的知識背景來完成碩士畢業論文，並且由於學校給予我們完善的設備及整潔的環境，讓我在做學問的過程中得以全心致力於研究的探討中，在此致上我由衷的感謝。

本篇論文得以順利完成，最主要感謝指導教授陳金鈴老師的指導，無論是在研究方面或是學習態度上的指導，每當研究上有所疑惑時，老師都給予我正確的方向，並分享寶貴的經驗，使我獲益良多。其次感謝口試委員呂芳懌老師、張庭毅老師以及施再繁老師在論文口試時給予本篇論文的建議及指導，使本篇論文更加完善。

在研究期間，我要感謝我的家人，在我求學及創業的過程中給予我最大的支持與鼓勵，讓我在創業及學業沒有其他顧慮下完成論文與碩士學位。

Contents

List of Tables

List of Figures

Chapter 1

Introduction

In recent years, there have been major advances in the development of wireless sensors. Due to recent wireless and IC process technological advances, wireless sensor networks (WSNs) have been replacing traditional network technologies [1-4]. There are a number of advantages that WSNs have over wired networks, such as ease of deployment, extended transmission range, and self-organization.

Currently, WSNs are used in various applications. Figure 1 shows a schematic of applications for WSNs, including military, agriculture, transportation, industry, and smart homes. However, there are a few inherent limitations in WSNs, such as low communication bandwidth, small storage capacity, limited computation resources and limited device energy. In terms of energy, many researchers assume that all nodes in a sensor network are battery-driven [5-6]. Because of energy is a very scarce resource for such sensor networks. Therefore, energy efficiency is an important design issue for WSNs.

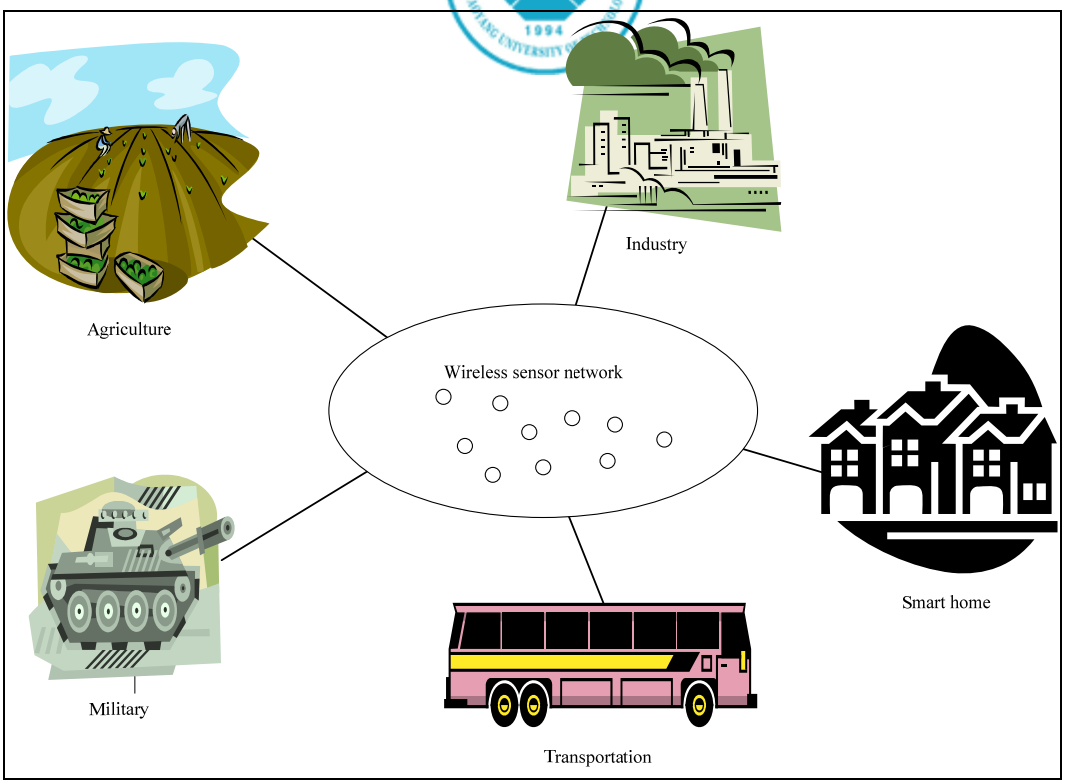


Figure 1 .Wireless sensor network applications

Generally speaking, a WSN consists of hundreds or thousands of densely populated sensor nodes that sense the environment and collaboratively work to process and route the sensor data. These sensor nodes relay data streams to sinks either periodically or based on events. The sink can be stationary or mobile such that it can move around the sensors and collect data. In these densely populated sensor networks, the area detected by the sensors may overlap and the data sensed by the sensors may be similar. Many wireless collisions occur in this type of network.

The general characteristics of wireless sensor networks include the following: ability for multiple deployments, low cost, small size, and limited battery power supply. Also, route

transmissions of a wireless sensor network include the following two types:

(1) **Cluster:** the cluster structure is the most representative of the routing protocols. The general scheme is to group a large number of sensors into several clusters. In each of the clusters, a node is chosen as the cluster header, which collects information from other sensor nodes and transmits the information to the base station.

(2) **Chaining:** the chaining structure differs from the cluster structure. Each detector node in the network is linked as a chain. In each round, a node in the chaining structure is chosen as the chaining header. Both ends of the chain then start transmitting data to adjacent nodes in the direction of the chaining header, and each receiving node gathers the information. Finally, the chaining header transmits the information to the base station.

In WSNs, one of several basic network topologies may be used. Each of the sensor nodes has simple computing power and can transmit and receive messages over wireless communication links. The basic network topologies are shown in the Figure 2. These include the star, tree, ring, fully connected, bus, and mesh topologies. Fully connected networks suffer from problems of NP-complexity. If nodes are added, the number of links increases exponentially. Mesh networks are regularly distributed networks that generally allow transmission only to a node's nearest neighbors. In the star topology, all nodes are connected to a single hub node.

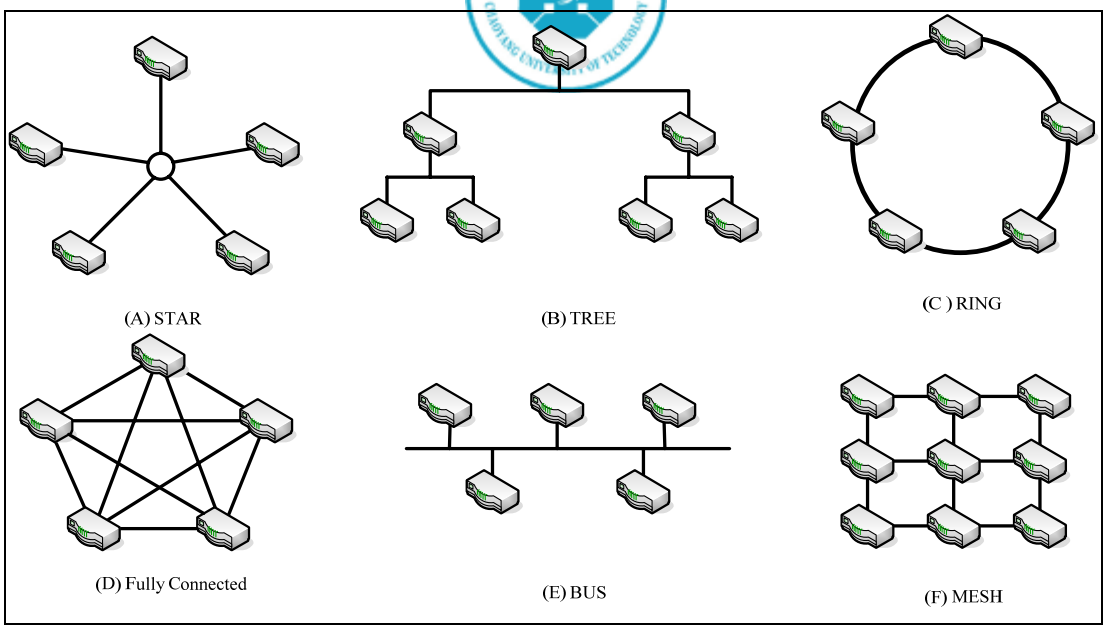


Figure 2. Network topology

Topology affects many network characteristics, such as robustness, energy consumption, and latency. The complexity of data routing and processing also depends on the topology.

Therefore, it is very important to choose a proper encryption system to protect any transmitted messages. Due to the limited computation resource of a wireless sensor node, public key algorithms (such as Diffie-Hellman key management [7] or the RSA mechanism [8]) are not feasible for solving security problems. We therefore propose a low cost dynamic session-key management for grid-based wireless sensor networks.

The rest of this thesis is organized as follows. Section II presents related work. Section III presents a dynamic session key management and spare two-tier data aggregation for grid-based wireless sensor networks Section IV presents simulation results. Section V presents conclusions.

Chapter 2

Related Work

2.1 Basic Assumptions

In this section, we present the basic model for the sensor networks. The network model has the following basic assumptions:

(1). After deployment, sensor nodes remain stationary at their initial locations.

(2). Each sensor node is assumed to be aware of its own geographic location. Sensors and mobile sinks can determine their own locations using Global Positioning System (GPS) [24-26] or other location approaches [19-20, 27-30]

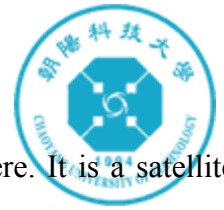
(3). Sensor nodes communicate with sinks by delivering data across multiple hops [31, 32]. That is to say, sources and sinks are typically much further apart than a single radio radius.

(4). The sensor nodes are homogeneous, and wireless channels are bidirectional. Each sensor node has limited battery energy.

(5). The sensor nodes are assumed to know a network's location which is in the interest region.

2.2 Global Positioning System (GPS)

GPS is the system which is able to give the exact location of an object on the earth at



anytime, in any weather, and anywhere. It is a satellite-based, radio navigation system. The satellite is continuously monitored by ground stations located worldwide. The satellites transmit signals that can be detected by anyone with a GPS receiver. Using the receiver, one can determine the location of an object with great precision [23-26].

GPS has three parts: a space segment, a user segment, and a control segment. The space segment consists of 24 satellites, each in its own orbit 11,000 nautical miles above the Earth. The user segment consists of receivers which can be held in a hand or mounted in a car. The control segment consists of ground stations (five ground stations located around the world) that make sure the satellites are working properly. GPS receivers typically work well outdoors, which can provide positioning accuracy within a 15 meters range.

2.3 Three Types of Key Management Protocols

In this section, we will review the existing key management protocols for wireless sensor networks. We have classified these protocols into three types: the Random Key Pre-distribution Protocol, the Group-based Key Pre-distribution Protocol, the Hierarchical Key Pre-distribution Protocol.

2.3.1 Random Key Pre-distribution Protocol

This protocol includes three steps: key pre-distribution, key sharing, and key establishment. Before the deployment of any sensor node, *m* sets of keys are selected from a

large key pool. The *m* keys form a key chain which is sent to each sensor node. One key is selected from among the nodes and is used to transmit data within the group. However, each sensor node must store *m* keys.

Blom's method [9] used a global matrix pool to replace the global key pool. In the key pre-distribution phase, each node randomly selects several matrices from the global matrices pool, and then loads a row of elements from each determined matrix into the node. In this case, any two adjacent nodes have a row of elements from the same matrix that can establish a pair wise key. In 2003, Pietro et al. [10] proposed a random key transmission protocol. The random keys are transmitted between the sensor nodes such that any two nodes can establish a secure communication channel. The shortcoming of this method is that each sensor node must store more than three sets of keys. In order to enhance security, the number of keys must be increased. However, the addition of the number of keys increases the loading of the sensor nodes. Furthermore, power consumption is also increased.

2.3.2 Group-based Key Pre-distribution Protocol

The Group-based Key Pre-distribution Protocol is used to divide the area where the nodes into several sections. A helicopter airdrops the nodes into a pre-defined area such that the sensor nodes have a higher probability of good communication.

Liu and Ning [11] proposed a paired key protocol, which used a polynomial key pool and pre-distribution of a grid key. This protocol has higher elasticity on catch and attack, and

superior sensor node communication. However, a key management algorithm is relatively complicated. More time is required to generate a key.

2.3.3 Hierarchical Key Pre-distribution Protocol

The hierarchical key pre-distribution protocol's elements include a base station, a cluster header, and sensor nodes. Before deployment, each cluster header stores keys. After deployment, the nodes will exchange identification codes. At the same time, the cluster header will be informed of the identification codes of the sensor nodes. The whole network can then communicate with each other. However, if one of the nodes is caught, the information transmitted between the cluster header and the sensor nodes could be easily observed by an enemy. Therefore, the cluster header must increase the number of keys to improve security. However, the resources of the sensor nodes are limited, making this impracticable.

Cheng and Agrawal [12] proposed an improved key distribution mechanism (IKDM) which makes use of bivariate polynomials to develop a secure wireless sensor network. In this scheme, each gateway does not directly store gateway keys of the nodes, but each gateway stores bivariate polynomial functions. After deployment, a node sends its identification code and the gateway numbers to the nearest gateway. Then the gateway receiving the data asks other gateways to obtain sub-keys. The gateway can then compute the gateway keys of its neighboring nodes from these sub-keys. Other similar schemes, such as the one by Jolly et al. [13], are also based on the Identity-Based symmetric keying scheme. This paper discusses

using additional sensors.

2.4 Routing Protocols

We also classified routing protocols into two types: the Direct Communication Protocol, and the Grid-based Routing Protocol.

2.4.1 Direct Communication Protocol

For direct communication in WSNs [14-15], all sensor nodes gather data as well as transmit the collected data directly to the sink. This is extremely energy inefficient, since path loss in wireless systems is proportional to R^n, where R is the distance between a sensor node and the sink, and n typically ranges between 2 and 4. Long distance transmission in direct communication consumes more power. In addition, in a network which consists of a large amount of sensor nodes, direct communication may not be feasible because of the large number of collisions. Multiple access schemes can be used to reduce the number of collisions, but the radio bandwidth for each sensor node would be reduced. Alternatively, direct communication can only be used in a small area with just a few sensor nodes within the area.

2.4.2 Grid-based Routing Protocol

In recent years, many researchers [15-20] have studied providing efficient grid-based data dissemination protocols for sinks. In the following section, we will introduce the relevant works.

2.4.2.1 Two-Tier Data Dissemination

A Two-Tier Data Dissemination (TTDD) approach [15-16] provides scalable and efficient data delivery to multiple mobile sinks. The mobile sinks are in constant motion because they are building a two-tier structure in sensor networks. TTDD exploits local flooding within a local cell of a grid, which the sources build proactively. Each source transmits data along the nodes on the grid line to the sink. However, TTDD does not optimize the path from the source to the sinks. Also, TTDD frequently resumes establishing the entire path to the sinks when the path is down. Data forwarding in TTDD is shown in Figure 3.

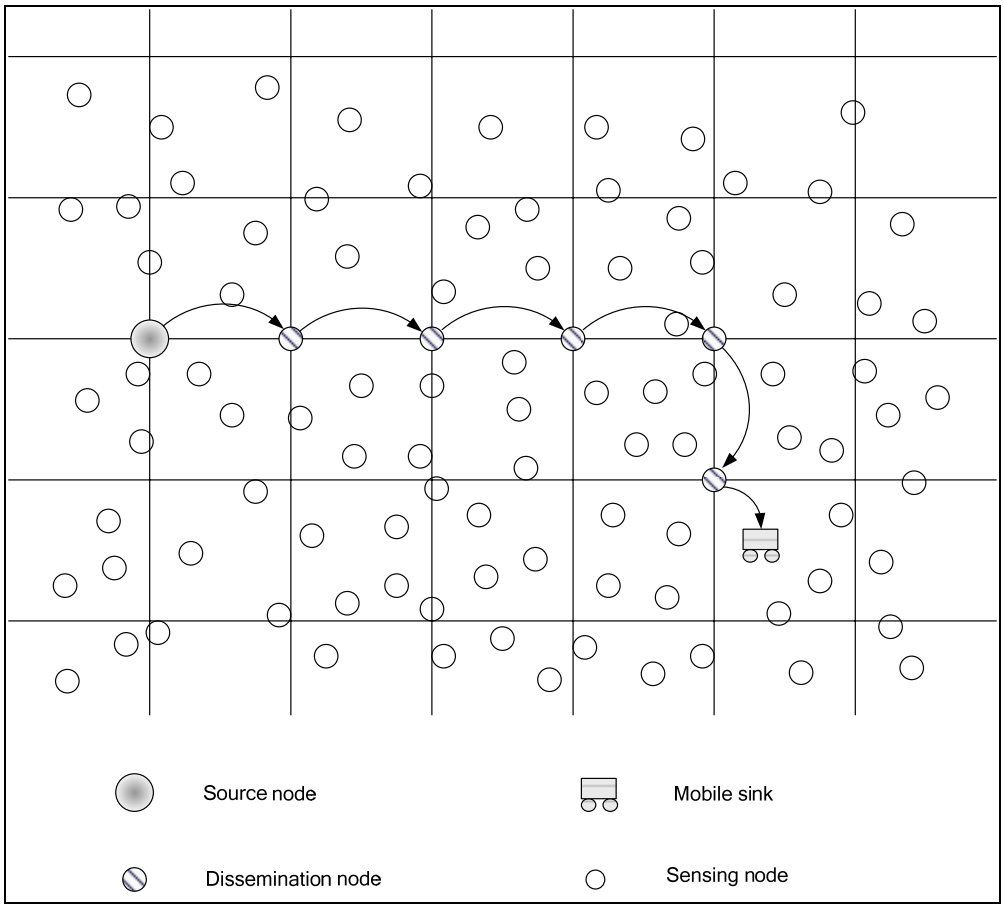


Figure 3. The TTDD scheme for the source node forwarding data to the mobile sink

2.4.2.2 Coordination-based Data Dissemination Protocol for Wireless Sensor Networks

Mobile sinks send queries to the nearest grid points using flooding. Queries are routed along the grid and data is traced along the path back to the sinks. As a consequence, the control overhead introduced by sink mobility is limited to the grid cell where a sink is located. A coordination-based data dissemination protocol for wireless sensor networks (CODE) [17] considers energy efficiency and network lifetime, especially for sensor networks with high node density. CODE uses a grid structure to establish an efficient data dissemination path between the sources and the mobile sink. For example, in Figure 4, a sensor node is selected to be a coordinator in each grid. The mobile sink sends a query to the source node. The source node transmits data to the mobile sink when it receives a query.

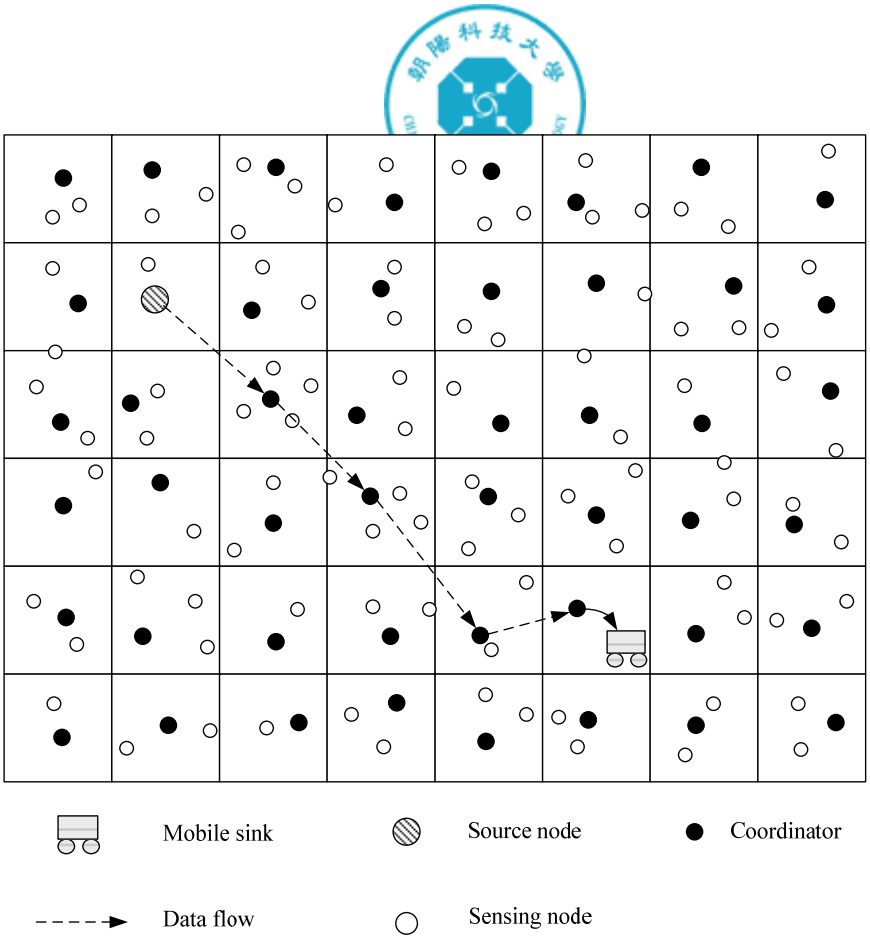


Figure 4. The CODE scheme for multi-hop routing through coordinators

2.4.2.3 Data Aggregation for Range Queries

Chen et al. [20] presented an efficient data aggregation algorithm for range queries (DARQ). DARQ is based on a grid structure. For example, in Figure 5, sensor nodes can determine their own locations using GPS. In Chen's scheme, a mobile sink makes requests for the source to aggregate regular-sharp data. When a source receives a query packet, it constructs an aggregate data tree. This scheme is able to aggregate data in the sensor field with void regions. Void regions are regions in which there are no nodes in a grid because some obstacles exist in the grid, no node is deployed, or a node has already died. Chen's

scheme utilizes the proposed face routing [21] to discover void regions and detour void regions. When a node cannot deliver packets by greedy-forwarding, it uses face routing to detour void regions.

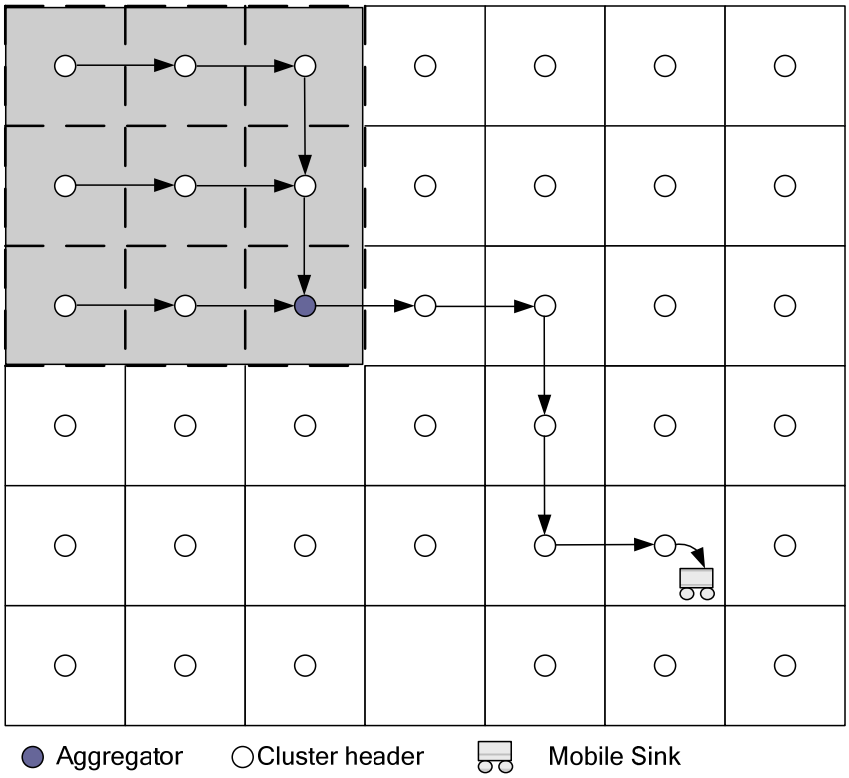


Figure 5. The DARQ scheme for data aggregation with regular-shape ranges

Our proposed scheme is based on the DARQ scheme, one-way hash function, "two-way" mutual authentication, and symmetric encryption mechanism to generate dynamic key management. A new key will be generated from the previous two old keys when the sensor node transmits data in each transaction. The new key will be used for encryption to protect the gathered data. When the sensor node transmits data to a header node, the header

node will request the decryption key of the sensor node from the base station. Since the base station has recorded two primary keys for all sensor nodes, it will transmit the required keys of that sensor node to the header node. After receiving the key list, the header node can decrypt the protected keys. When the number of sets of the received data is larger than a threshold value *t*, the data will be encrypted and transmitted to the base station. In order to ensure information security, the method for generating the keys for the sensor nodes is the same as the method for the header node. In addition, one of the keys between the base station and the header node and another between the base station and the sensor nodes will be updated dynamically in order to improve network security.

Although Liao et al. [22] solved the flooding problem [15], they still did not solve the security problem. This paper investigates the security problem in a grid-based routing protocol by exploiting local flooding within a local cell of a grid which sources build proactively.

Chapter 3

A Dynamic Session Key Management and Spare Two-tier Data Aggregation for Grid-based Wireless Sensor Networks

In this section, we propose a novel scheme for grid-based generation of a dynamic key to improve upon the security of previous methods. Our protocol is based on grid-based sensor networks. A sensor node, called a header node, is selected from the sensors in a grid to make announcements and for routing. Each sink can obtain information on an event from a grid header where it is located. If the sink is interested in an event, it queries the source via the grid header. In the interesting region, the sink designates the range for data aggregation. The proposed scheme can defend various attacks and reduce energy consumption. Figure 6 shows the grid structure.

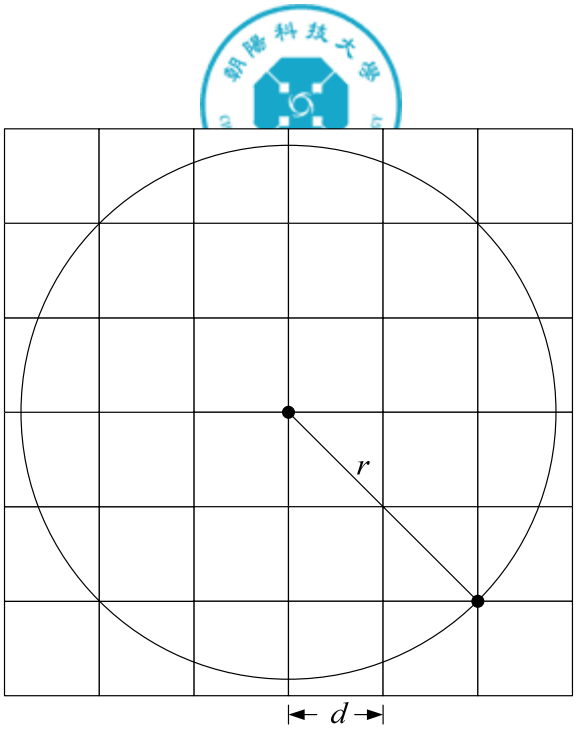


Figure 6. Grid structure

3.1 Eliminating the Broadcast Storm Effect

To eliminate any broadcast storm effect, each node will first broadcast their information to its neighboring nodes. We will use the locations of the source and the interesting region (see section 3.3) to limit the forwarding region. Thus, the broadcast storm problem can be reduced to a certain degree. However, if the forwarding zone is large, there will be a lot of redundancies, contentions, and collisions in the zone. In our protocol, the parameter range in a query packet is used to limit the forwarding zone.

Let S and X be the source and destination of the header node. Figure 7 shows the limits of the forwarding zone fan (θ, r), which is an area in the shape of a fan from the grid S to the grid X with angle θ and radius r [22].

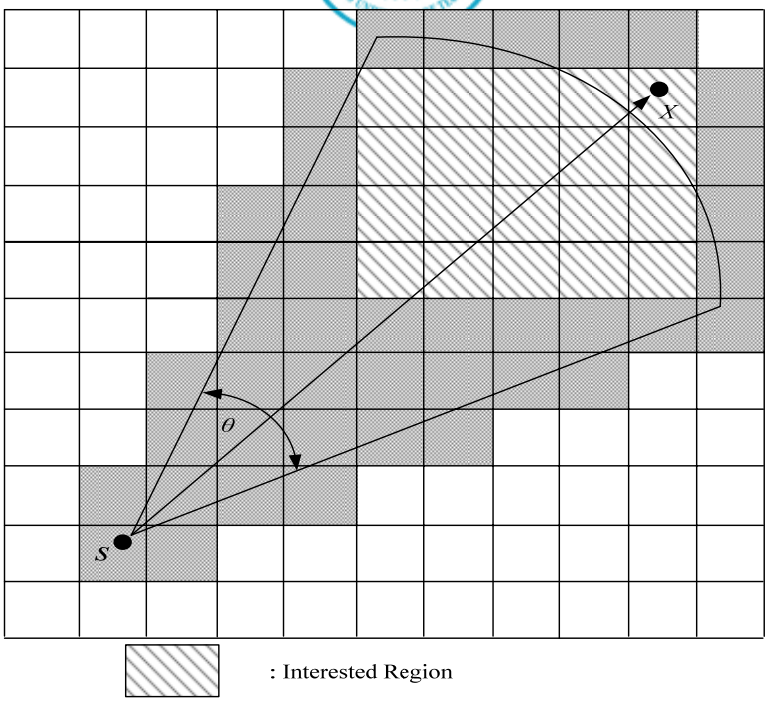


Figure 7. The flooding region

3.2 Grid Formation

The entire area of a wireless sensor network is partitioned into a 2D logical grid (a 4×4 grid, as illustrated in Figure 8).

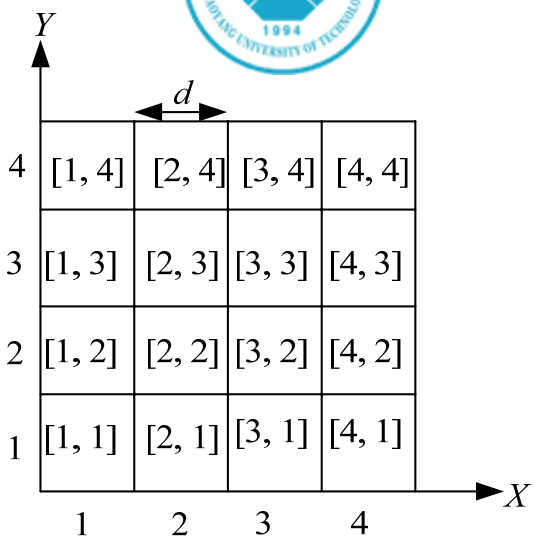


Figure 8. Grid index

Each grid is a square of size $d \times d$. Grids are identified (x, y) using the conventional xy-coordinate system. To be aware of its location, each node is equipped with a positioning device, such as a GPS receiver, from which it can read its current location. For any given location, there is a predefined mapping of a location to its grid coordinate. Each grid ID, which is given by [C_X, C_Y], is assigned as follows: in the first row, from left to right, the grid IDs are [1, 1], [2, 1], [3, 1], and [4, 1]. In the second row, the grid IDs are [1, 2], [2, 2], [3, 2], and [4, 2], and so on. Based on the coordinate (x, y), each node computes its C_X and C_Y as follows:

$$C_X = \left\lfloor \frac{x}{d} \right\rfloor, \; C_Y = \left\lfloor \frac{y}{d} \right\rfloor \qquad (1)$$

d is the grid size, where C_X and C_Y are the largest integers not greater than $\frac{x}{d}$ and $\frac{y}{d}$, respectively.

From equation (1), each node determines which grid it belongs to. Moreover, each node will maintain a neighbor table. The neighbor table is generated by using the periodic HELLO protocol [25] at the beginning of a network life. The HELLO packet is small. The HELLO overhead from the periodic HELLO protocol is very small.

The length of the sides of the grids will affect the performance of our protocol. Let r be the transmission distance of a radio signal. We use the maximum value of d is limited to $r/\sqrt{5}$ [20]. The maximum value d is a header node located at a grid and is capable of talking to any of the header nodes of its 8 neighboring grids. However, a smaller d also means more header nodes in the network, which in turn implies a higher overhead for delivering a packet and more broadcast storm. Thus, there exists some tradeoff in choosing a moderate d value.

In each grid, a sensor node is selected to be the header of that grid. We call a node is header when it has more remaining energy than other nodes in that grid. Figure 9 shows a physical area partitioned into logical grids.

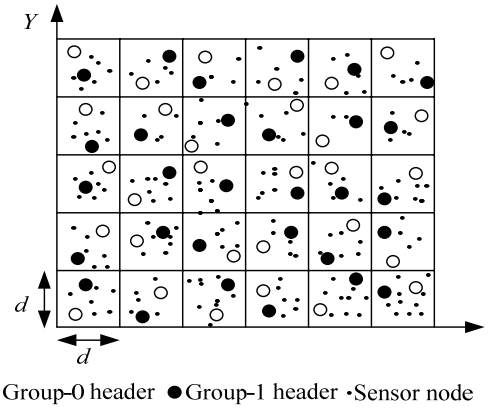


Figure 9. A physical area partitioned into logical grids

3.3 Neighboring Table

A node recorded the information of other nodes that were within the transmission range when it could hear the hello packet. The field of the table was <*node_ID*, *distance*>.

3.4 Routing Table

This table contained the path that was used for the transmission of data. The format of the path table is <*source_ID*, *destination_ID*, *seq_number*, *route_group*, *next_hop*>. The *source_ID* and *destination_ID* fields contained the unique addresses of the source and the destination node, respectively. The *seq_number* field contained the sequence number of the source node (guaranteeing the loop-freedom of all routes to the destination node). The *route_group* field records the group of route for group-0 or group-1. The *next_hop* field contained the address of the neighboring node to which data packets had to be forwarded.

In each grid, two sensor nodes of group is selected to be headers of that grid. We call a node is header when it has more remaining energy than other nodes in that grid. Figure 9 shows a physical area partitioned into logical grids.

3.5 Selection of Interest Region

When an interesting event happens, a sensor node will be conscious of this event in the selected region. After the event, a sensor node will broadcast a packet to find one-hop

neighboring nodes. If a neighboring node is conscious of the event, it will forward this packet and store this message in an events table. We describe the event process below.

Step 1: When an event of interest happens, a sensor node will be conscious of this event in the interest region. The sensor node will broadcast a packet to its one-hop neighboring nodes.

Step 2: If receiving the packet, the neighboring nodes will be conscious of the event then go to the next step; otherwise, the neighboring nodes will drop this packet.

Step 3: The neighboring nodes will forward the packet and store this message in their events table.

For example, in Figure 10, node E is conscious of an event occurring in its region. Node E will broadcast a packet to its one-hop neighboring nodes A, B, C, D, F, G, H and I. Because Nodes A, B, C, D, F, G, H and I are now conscious of the event in his region, they will forward this packet and store this message in their events table. Nodes J, K, and L will drop this packet since they are not conscious of the event.

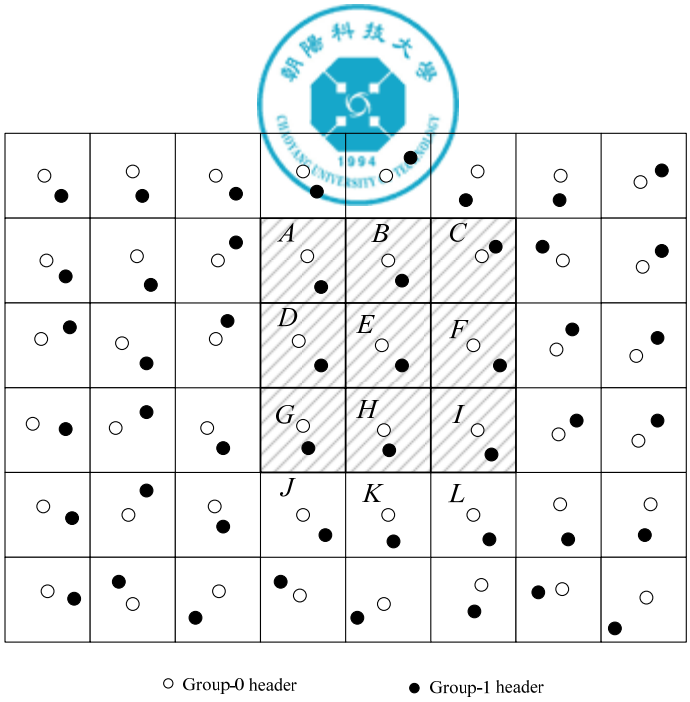


Figure 10. Selected interest region

3.6 Root Nodes of Both Groups Election

To maintain the header node so that it is alive in each grid, an efficient method for header node selection is necessary. In each grid, when header node residual energy is less than the assumed threshold (for example: defined as 10), one node will be selected as the header node of that grid. To maintain the quality of routes, we let the header node of a grid node with the largest residual energy in that grid.

After a sensor node detects an interesting region. The steps for electing header nodes of both groups are described as follows:

In the interesting region, each header of both groups sends a root node election request packet <*Node_id*, *Grid_id*, *Residual_Energy*, *Group*, *Timeout*> to the other nodes, where

Grid_id is an identification code for the grid and *Residual_Energy* is the residual energy of a node. *Group* field recorded the group of route for group-0 or group-1. If the time it takes for a header node to receive the selection request packet is greater than the *Timeout* value, then the packet is discarded.

When a header node receives the root node election request packet, the header node judges whether its grid location is the most approach of the mobile sink and it has the largest residual energy.

If it is true, the node becomes a root node; otherwise, the request packet is discarded. The header node will be selected periodically to keep the gateway from running out of energy.

The main feature of our scheme is as follows. We assume that a header node has the most residual energy of both groups. When a header node detects an interesting event, it will broadcast a packet to all the header nodes. Thus, all the header nodes will know whether an event has occurred. When a sink wants to know whether there has been an interesting event, it sends a request packet to ask the header node in its grid. When the header node receives the request packet, it sends a reply packet to the sink.

3.7 Spare Two-Tier Path Establish in Interesting Region

After selecting root of both groups, the steps of establishing spare two-tier path are

listed below.

1. Root node of both groups broadcast a path construct packet <*R_Grid*(*x,y*), *Residual_Energy*, *Group*, *Timeout*> to its one-hop neighboring nodes, where *R_Grid*(*x*, *y*) is the root's grid identity. *Residual_Energy* is the residual energy of node. For example, in Figure 11. Node F is a root of group-0, node K is a root of group-1. They broadcast a path construct packet to one-hop neighboring nodes. Root-A{D, J, L}, Root-B{C, E, I, O, Q}

2. When each one-hop neighboring node receives the path construct packet from the root node, the node judges whether it is leaf node of a tree or not. If no, the node will sends a path construct request packet < *N_Grid(x, y)*, *R_Grid*(*x*, *y*) *Residual_Energy*, *Group* , *Timeout*> to the root node, where *N_Grid*(*x, y*) is node's grid ID. For example, in Figure 11. Nodes {D, J, L} receives the path construct packet from Root-A, Nodes {C, E, I, O, Q} receives the path construct packet from Root-B. They judges whether it is leaf node of any tree or not. If no, the node will sends a path construct request packet to the root node.

3. Root of both groups will compare with same group residual energy of the path construct request packets which come from his neighboring nodes. Then it will choose the node with the most residual energy of both groups and to become its leaves node. If the leaves node with the residual energy less than the threshold (for

example: defined as 10), then drop it.

In addition, nodes receive two path construct packets which are from different root. It will send path construct request packet to the root which have minimum value of node number.

For example, in Figure 11, root of both groups (node F and node K) were selected. Node F will broadcast a path construct packet to nodes D, J and L. At the same time, node K will broadcast a path construct packet to nodes C, E, I, O and Q.

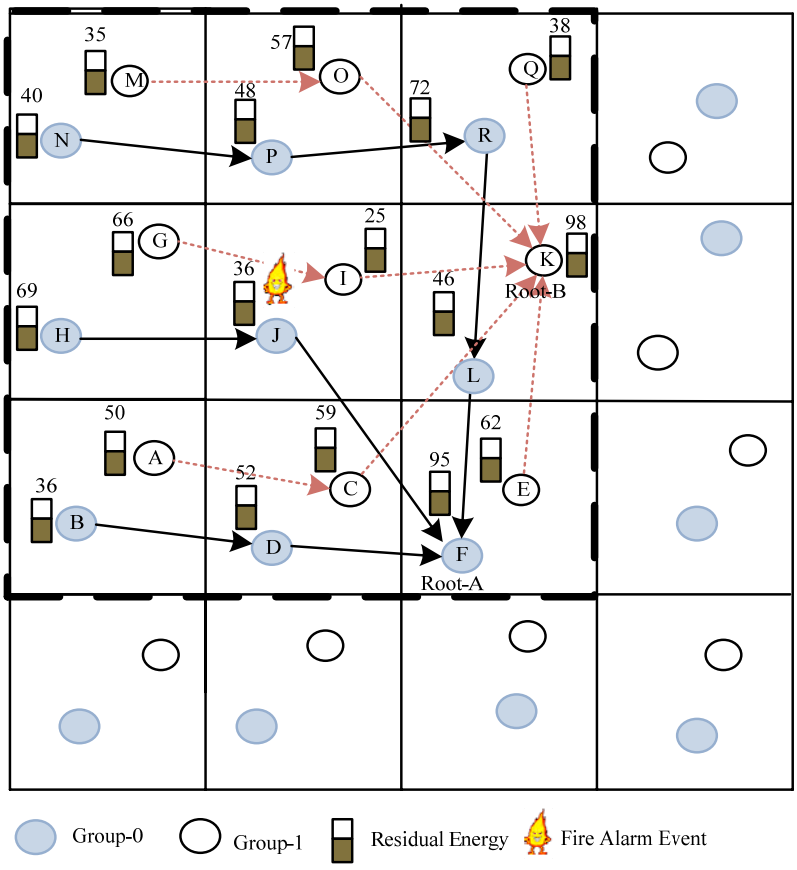


Figure 11. The path construction of Interest Region

3.8 Spare Two-Tier Query Forwarding

If a mobile sink is interested in the occurred event, it sends a query packet to find route and acquire the interesting data. When a mobile sink wants to acquire the interesting data, it will send a request packet to ask the header node in the same grid. The header node receives the request packet, and it sends reply packet to the mobile sink. Then the header node which the mobile sink belongs will send a query message to the root node of both groups. When the root node receives the query message, it will reply the message to the mobile sink with reverse route. The mobile sink has been decided which path is active path, and which path is sleep path. We consider different weight levels. We assume that mean $\bar{x}$ of weight is 0.6; standard deviation σ of weight is 0.4 and header node's threshold of residual energy greater than 10. Which path have bigger value of $\bar{x}$, which one be selected as active path. If they are same mean of the values, we use standard deviation to calculate which one is active path.

If the random variable X takes on N values (which are real numbers) with equal probability, where X is a residual energy of header, then its standard deviation σ can be calculated as follows:

1. By equation (2), find the mean $\bar{x}$.
2. For each value X_i calculate its deviation $\left(x_i - \bar{x}\right)$ from the mean.
3. Calculate the squares of these deviations.
4. Find the mean of the squared deviations. This quantity is the variance σ^2.

5. Take the square root of the variance.

The arithmetic mean of the values x_i, defined as:

$$\overline{X} = \frac{x_1 + x_2 + 6 \ + x_n}{N} \qquad (2)$$

This calculation of standard deviation σ is described as follows:

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(x_i - \bar{x}\right)^2} \qquad (3)$$

From equation (3), a low standard deviation indicates that the data points tend to be very close to the mean, whereas high standard deviation indicates that the data are spread out over a large range of values. The mobile sink decides which path is low standard deviation. The low standard deviation path will be *active path*.

For example: in Figure 12, path of group-0 is {79, 92, 43, 53, 36, 56} and path of group-1 is {80, 75, 65, 35, 66, 38}. From equation (2), the mean $\overline{X}$ of the Group-0 is 59.83 and the mean $\overline{X}$ of the group-1 is also 59.83, they are same mean of the values and same weight value. From equation (3), the standard deviation of the group-0 is 19.6419336 and the standard deviation of the group-1 is 17.29563208, the group-1 have a low standard deviation indicates that the data points tend to be very close to the mean, the group-1 standard deviation σ of weight is 0.4 but group-0 standard deviation σ of weight is 0, so the low standard deviation group-1 will be *active path*.

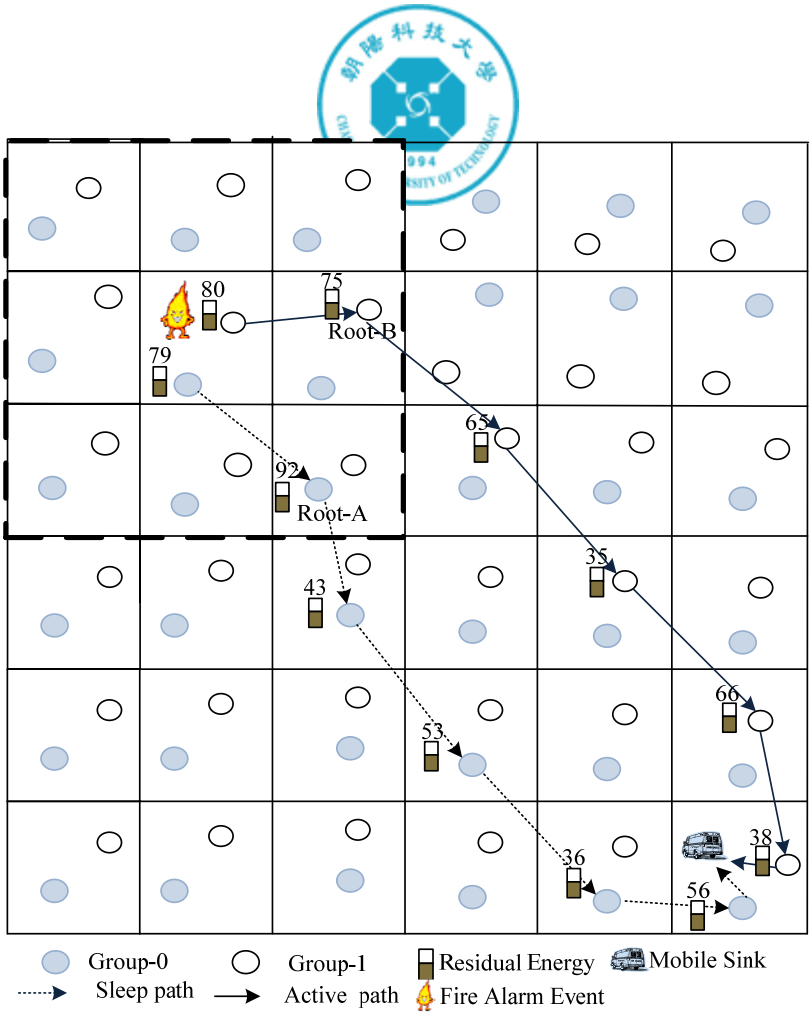


Figure 12. Data dissemination construction

3.9 Communication protocol

In our proposed protocol, we use a dynamic key management mechanism. Two keys are preset in each sensor node. A new key for the next round is generated by these two keys. The two keys will also be preset in the header node. The generation of the session key will be the same as the generation of the key in the sensor node. Using key management, we can ensure the security of the data transmission. We show the transmission paths of the sensor network in Figure 13.

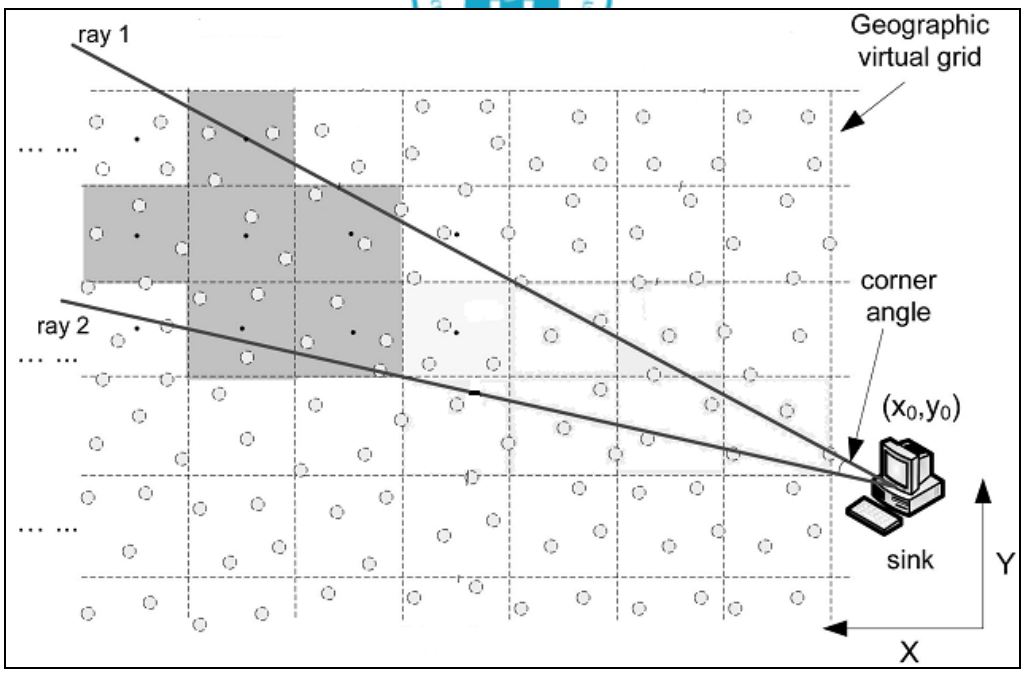


Figure 13. Transmission paths for the sensor network

We divide our protocol into the following steps, as shown in Figure 14.

3.10 Notation

The notations used in our thesis and in this paper are given below, along with their meaning.

$h( )$	the one-way hash function, used for key generation.
a_j , a_{j-1}	two parameters, used for generating a key pre-deployed in the j^{th} sensor node.
b_j , b_{j-1}	two parameters, used for generating a key pre-deployed in the j^{th} header node.
N_1 , N_2 , N_3	three nonces.
K_{si}	the i^{th} key of the sensor node.

K_{ci}	the i^{th} key of the header node.
K_{msg}	the key used for encrypting or decrypting the updated-key message msg_{finish}.
Seed	the seed for updating the key pre-deployed in each of the sensor nodes.
ID_{si}	the identity of the i^{th} sensor node.
ID_{ci}	the identity of the i^{th} header node.
ID_{Bi}	the identity of the i^{th} base station.
C_{si}	the encrypted information generated by the i^{th} sensor node.
C_{ci}	the encrypted information generated by the i^{th} header node.
C_b	the encrypted information generated by the base station.
ID_{list}	the identity set list of the t sensor nodes received from the header nodes, such as $ID_{list} = (ID_{s1}, ID_{s2}, ..., ID_{st})$.
K_{list}	the key of the sensor nodes generated by the header node, such as $K_{list} = (K_{s1}, K_{s2}, ..., K_{st})$.
M_i	the plaintext information is generated by the i^{th} sensor node.
M_f	the latest information received by the base station.
E(M,K)	the symmetric encryption infrastructure makes use of key *K* to encrypt *M*.
D(M,K)	the symmetric decryption infrastructure makes use of key *K* to decrypt *M*.

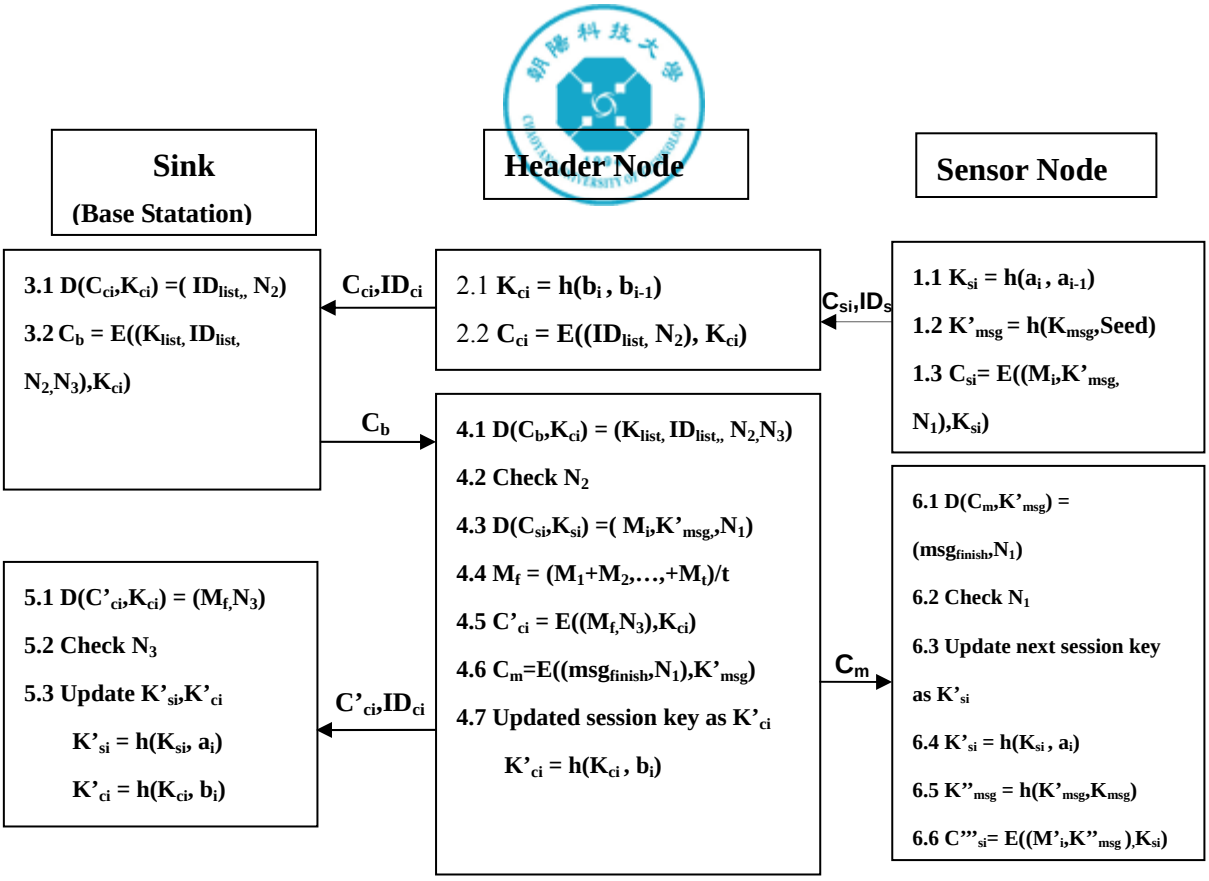


Figure 14. The communication protocol

Step 1: When the deployed sensor node *i* returns the collected information, the sensor node will make use of the preset parameters a_j and a_{j-1} to generate a key, K_{si}, where

$$K_{si} = h(a_j, a_{j-1}) \qquad (2)$$

Furthermore, the two parameters K_{msg} and the Seed preset in each of the nodes will use the hash function to generate a new message key, K'_{msg}, where:

$$K'_{msg} = h(K_{msg}, Seed) \qquad (3)$$

At that moment, the sensor node generates N_1 and makes use of K_{si} to encrypt the detected data M_i, the preset K'_{msg} , and N_1 into packet C_{si} as follows:

$$C_{si} = E((M_i, K'_{msg}, N_1), K_{si}) \qquad (4)$$

The (C_{si}, ID_{sj}) is then transmitted to the header node.

Step 2: When the header node receives more than t packets, or when the period is longer than a specified time, the header node will record and transmit the identity, ID_{sj}, of the sensor node. It will also arrange a list, ID_{list}

$$ID_{list} = (ID_{s1}, ID_{s2}, ..., ID_{st}) \quad (5)$$

The header node will make use of the two preset parameters b_j and b_{j-1} to generate a key, K_{ci}, where

$$K_{ci} = h(b_j, b_{j-1}) \quad (6)$$

At that moment, the header node will generate N_2 and make use of K_{ci} to encrypt ID_{list} and N_2

$$C_{ci} = E((ID_{list}, N_2), K_{ci}) \quad (7)$$

Together with the code ID_{cj} of the header node, the C_{ci} will be transmitted to the sink.

Step 3: When the base station (sink) receives the packet from the header node, it will verify the code ID_{cj} and seek the key K_{ci} to decrypt C_{ci}; otherwise, this packet is discarded.

$$D(C_{ci}, K_{ci}) = (ID_{list}, N_2) \quad (8)$$

After receiving the ID_{list} sent from the header node, the sink will search for the key of the corresponding sensor node from the database and arrange them into the key list K_{list}, where

$$K_{list} = (K_{s1}, K_{s2}, ..., K_{st}) \quad (9)$$

At that moment, the sink will generate N_3 and make use of K_{ci} to encrypt $(K_{list}, ID_{list}, N_2, N_3)$. The encrypted data, C_b, will be sent to the header node, where

$$C_b = E((K_{list}, ID_{list}, N_2, N_3), K_{ci}) \qquad (10)$$

Step 4: When the header node receives the response data from the sink, it will make use of the key K_{ci}, which is generated by itself, to decrypt C_b

$$D(C_b, K_{ci}) = (K_{list}, ID_{list}, N_2, N_3) \qquad (11)$$

The header node will then check whether N_2 is equal to the N_2 in step 2.

The header node can only use K_{si} of K_{list} to decrypt C_{si}; otherwise, this packet is discarded.

$$D(C_{si}, K_{si}) = (M_i, K'_{msg}, N_1) \qquad (12)$$

Afterward, the header node will calculate the average value of each set of data and obtain M_f as follows:

$$M_f = (M_1 + M_2 +,...,+M_t)/t \qquad (13)$$

This ensures that the data is accurate when it is transmitted to the backend. This header node will make use of K_{ci} to encrypt M_f and N_3 into C'_{ci},

$$C'_{ci} = E((M_f, N_3), K_{ci}) \qquad (14)$$

Together with the header node identity, ID_{cj}, C'_{ci} are transmitted to the sink. At that

moment, the header node will update the session key into K'_{ci} for the next round.

$$K'_{ci} = h(K_{ci}, b_j) \tag{15}$$

Moreover, the header node will make use of the key K'_{msg}, transmitted from the sensor node, to encrypt the transmitted update message msg_{finish} as follows:

$$C_m = E((msg_{finish}, N_1), K'_{msg}) \tag{16}$$

The encrypted packet C_m will then be broadcasted to the sensor nodes to inform the sensor nodes that message transmission is completed.

Step 5: When the sink receives the packet from the header node, it will confirm the identity, ID_{cj}, of the header node first. Also, it will search for the key K_{ci} to decrypt C'_{ci}.

$$D(C'_{ci}, K_{ci}) = (M_f, N_3) \tag{17}$$

The sink will then check whether N_3 is equal to the N_3 in step 3.

Simultaneously, the sink will update the key of the header node and sensor node, which will be updated into K'_{si} and K'_{ci}, as follows:

$$K'_{si} = h(K_{si}, a_j) \tag{18}$$

$$K'_{ci} = h(K_{ci}, b_j) \tag{19}$$

Step 6: After receiving the message C_m, the sensor node will make use of K'_{msg} for decryption, and obtain the message msg_{finish} as follows:

$$D(C_m, K'_{msg}) = (msg_{finish}, N_1) \tag{20}$$

The sensor node will then check whether N_1 is equal to the N_1 in step 1.

The previously generated keys K_{si} and a_j are used to generate a new key, K'_{si}, where:

$$K'_{si} = h(K_{si}, a_j) \tag{21}$$

K'_{si} will be used to encrypt the transmitted data for the next transmission. When the sensor node transmits the data in the third round, the original message key K'_{msg} will be updated to K''_{msg}, where

$$K''_{msg} = h(K'_{msg}, K_{msg}) \tag{22}$$

The message key, K''_{msg} and the detected message M'_i will make use of K'_{si} to encrypt them to C'_{si}, where

$$C'_{si} = E((M'_i, K''_{msg}), K_{si}) \tag{23}$$

When the sensor node transmits data for the fourth time, the message key must be updated to K'''_{msg}, where

$$K'''_{msg} = h(K''_{msg}, K'_{msg}) \tag{24}$$

The updated message key K'''_{msg} and the detected message M''_i will make use K'''_{si} to encrypt them into C''_{si},

$$C''_{si} = E((M''_i, K'''_{msg}), K_{si}) \tag{25}$$

The session keys K_{si}, K'_{si} and K''_{si}, etc. are used for encrypting messages between the header node and the sensor node. In addition, the updated K''_{msg} and K'''_{msg} are the message keys which the header node uses to transmit complete messages msg_{finsh} to the sensor node during communication.

Chapter 4

Security Analysis and Performance Analysis

4.1 Security Analysis

4.1.1. Against Malicious Guessing Attacks

When a sensor network has been deployed for a certain period, the key database of the base station will be updated after a transaction such that the attacker cannot obtain the correct key for the next transmission. Furthermore, each node includes the records of not more than three keys, which include two old keys and one newly generated key. When the new key is generated, the oldest key will be updated. This can improve the security of the network and reduce the memory load of the nodes.

4.1.2. Against Replay Attacks

In each communication session, including sessions from the sensor node to the header node or sessions from the header node to the base station, “two-way” mutual authentication is used to prevent the replay attack. We use the nonces N_1, N_2 and N_3 to confirm each communication message. Any illegal communication can be found by checking the correctness of the *nonces*. Our scheme is able to prevent replay attacks.

4.1.3. Against Falsification Attacks

For secure transmission, we use the keys K_{si} and K_{ci} for the encryption of data transmitted between the header node and the sensor node or between the header node and the base station, respectively. When the sensor node returns the data to the header node, $C'_{si} = E((M'_i, K''_{msg}), K_{si})$ is used for encryption. When the communication between the header node and the base station is finished, K_{list} is obtained. The base station returns K_{si} to the header node. If the received key cannot decrypt the received packet, it will be regarded as an illegal packet and will be abandoned. This practice can ensure the integrity of the data transmission, and guarantee that the data is sent from the sensor node administered by the header node.

4.1.4. Against Man-in-the-Middle-Attacks and Guarantee Data Privacy

When the sensor node communicates with the header node, the encryption mechanism is used to prevent man-in-the-middle attacks and ensure data privacy. The transmitted message is encrypted into $C_{si} = E((M_i, K'_{msg}), K_{si})$. The header node and the base station also use a similar method to prevent similar attacks and ensure data privacy.

The attacker cannot obtain the protected data. Furthermore, the header node makes use of K_{msg} to encrypt the complete message, and the message key will be updated for each transaction. Therefore, the attacker cannot imitate the header node to transmit a message. The

man-in-the-middle-attack can thus be prevented.

4.1.5 Against Node Capture Attacks

To transmit data between the header node and the sensor node or between the header node and the sink node, we use the keys K_{si} and K_{ci} for encryption. We make use of the one-way hash function to generate the key because the one-way hash function can prevent an attacker from inverting the key. (1) *h(x)* is relatively easy to compute for any given x, making both hardware and software implementations practical. (2) For any given value y, it is computationally infeasible to find x such that $h(x) = y$. This is sometimes referred to in the literature as the one-way property. (3) For any given block x, it is computationally infeasible to find z not equal to x with $h(z) = h(x)$. This is sometimes referred to as weak collision resistance.

Table 1. The security and characteristic comparison of the grid-based thesis

Grid-based / Attacks	TTDD [15, 16]	CODE [17]	DARQ [20, 21]	Our proposal
Against malicious guessing attacks	NA	NA	NA	Yes
Against replay attacks	NA	NA	NA	Yes
Against falsification attacks	NA	NA	NA	Yes
Against man-in-the-middle-attacks and guarantee of data privacy	NA	NA	NA	Yes
Against node capture attacks	NA	NA	NA	Yes
Grid-based protocol	Yes	Yes	Yes	Yes
Event-driven data dissemination	Yes	Yes	Yes	Yes
Limit interest region to prevent flooding storm	No	No	Yes	Yes
Routing problem with obstacles in sensor field	Yes	No	Yes	Yes
Routing problem with voids in sensor field	Yes	No	Yes	Yes

In Table 1, we analyze the security and characteristic comparison of the grid-based thesis. The following observations are made:

(1). Because TTDD, CODE and DARQ do not support dynamic session-key management, they may be susceptible to various attacks.

(2). In CODE and TTDD, the sinks have to reissue a query to request data or use local flooding to request data when they move out of the original grid. This will increase energy consumption and the number of collisions. DARQ and our thesis will limit the interest region to prevent flooding storm.

(3). If the sink moves out of the original grid, it reconstructs a new routing path. But CODE does not solve the routing problem when there are obstacles or voids in a sensor field.

4.2 Performance Analysis

In Table 2, we analyze the performance of the proposed protocol.

Table 2. Performance analysis of the proposed protocol

Relationship between the nodes	Rounds	*Time complexity*
Sensor node and header node	2	$2T_E + 1T_M$
Header node and base station	$3N$	$N(3T_E + 3T_M)$

Notes:
T_E: the time complexity of using symmetric encryption algorithm.
T_M: the time complexity needed for plaintext (e.g., $ID_{sj,}$ $ID_{cj,}$) transmission.
N: the number of hop count.

We developed a simulation based on NS2 (Network Simulation 2), shown in Table 3. We use the energy model used in NS2 that requires about 0.7 Watt, 0.35 Watt, and 0.035 Watt for transmitting, receiving and idling, respectively. The initial battery status is equal to 10 Joule. The mobility model is according to random waypoint model. The sensor nodes are deployed uniformly in a 500m×500m field. The simulation lasted for 100 seconds. Each

simulation runs 50 times.

Table 3. Parameters used in the simulation environment

Parameter	Values
Simulation tool	NS2
Simulation area	500 m × 500 m
Number of nodes	200 nodes
Sink mobility model	5 m / sec
Radio transmission range	Random waypoint model
Data packet size	512 bytes
Data transmission rate	100 m

In the following section, we compare the energy consumed in our scheme versus the energy consumed in other scheme.

A comparison of the total energy consumption of the related works for various different numbers of nodes is shown in Figure 15. The total energy consumption of the proposed thesis and the DARQ, CODE and TTDD thesis increased when the number of grids increased, but the total energy consumption in our thesis is lower than the CODE and TTDD thesis but higher than DARQ. Since our thesis uses a grid-based mechanism to restrict the possibility of packet flooding, such a result meets our expectations. However, although our thesis is based on DARQ, more energy is consumed in encryption computation.

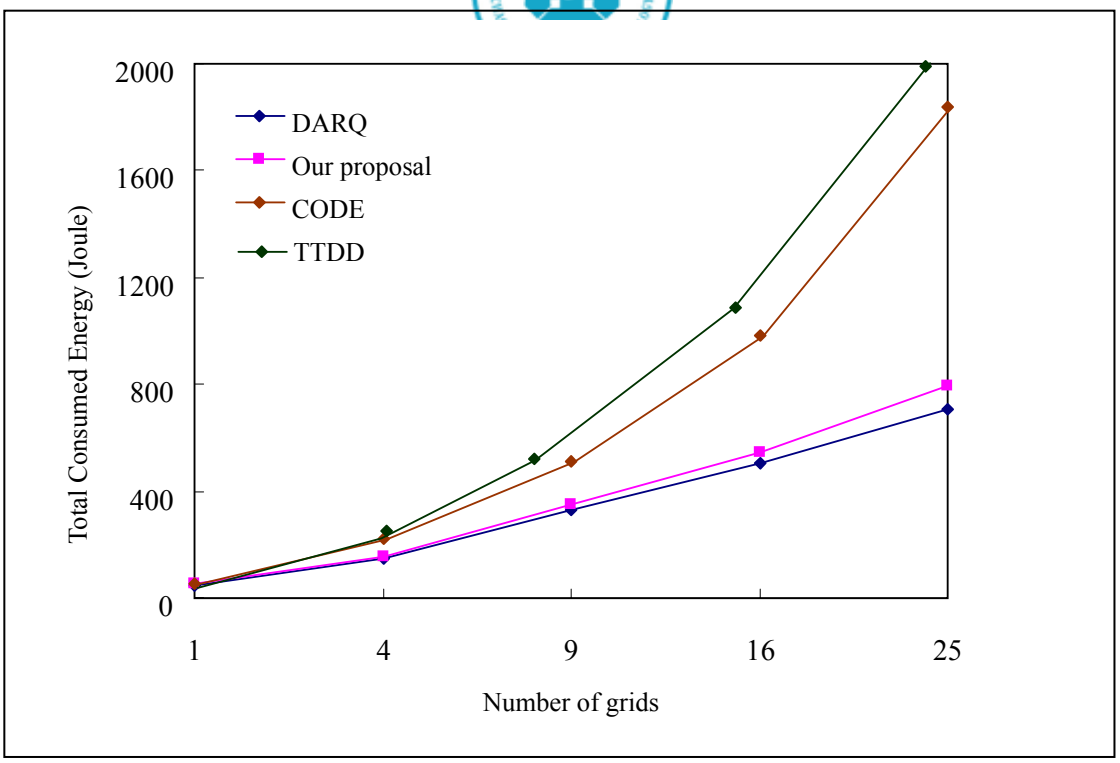


Figure 15. Comparison of energy consumption versus number of grids for different schemes

The performance metrics used were as follows:

1. Total energy consumed: The total energy consumed of our system. It includes three energy dissipations: Communication Energy Dissipation, Computation Energy Dissipation and Sensing Energy Dissipation.
2. Average delay: The delay is defined as the average time between the time sensing nodes transmit a packet and the time a sink receives the packet.

4.2.1 Total Energy Consumed

A comparison of the total consumed energy of the whole network is shown in Figure 16 shows the performance of the total consumed energy under various different numbers of

sink, which ranged from 1 sink to 5 sinks. We assumed that the mobility speed was 15 m/sec. The interesting region is 3x3 grids. The 200 sensor nodes are deployed uniformly in a field. The total energy consumption of proposed scheme, CODE, and TTDD both increased when the number of sinks increased, but the total energy consumption of our scheme is lower than CODE and TTDD.

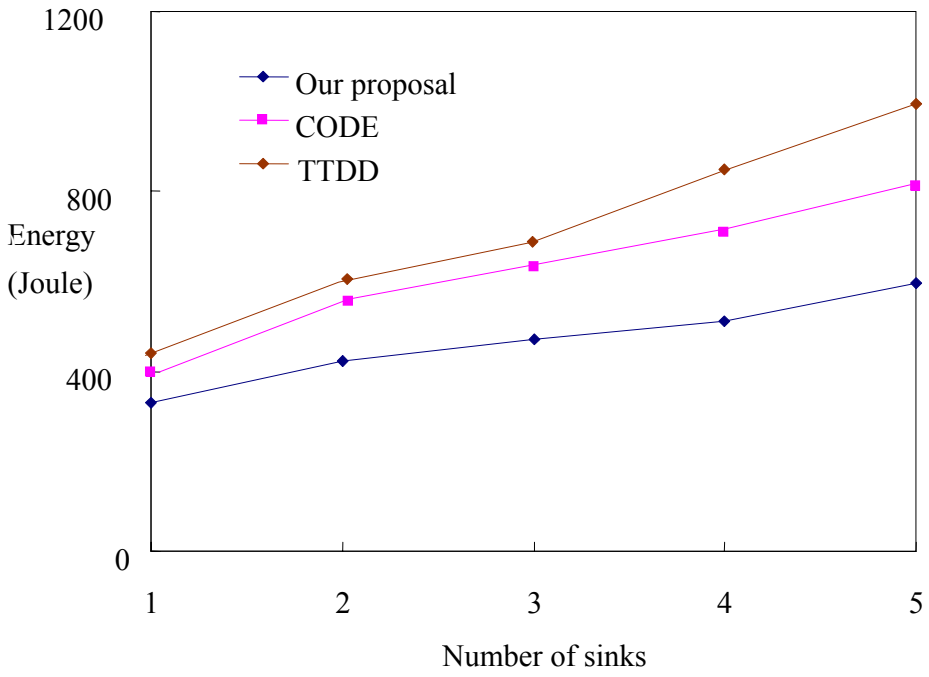


Figure 16. The energy consumed of different number of sinks

Figure 17 shows the performance of the total consumed energy under various different numbers of grids, which ranged from 1 grid to 25 grids that is region of interest. We assumed that the mobility speed was 15 m/sec. Only one sink is in this network. The total energy consumption of proposed scheme, CODE, and TTDD both increased when the

number of grids increased, but the total energy consumption of our scheme is lower than CODE and TTDD when the number of grids increased. The energy of transmission will increase.

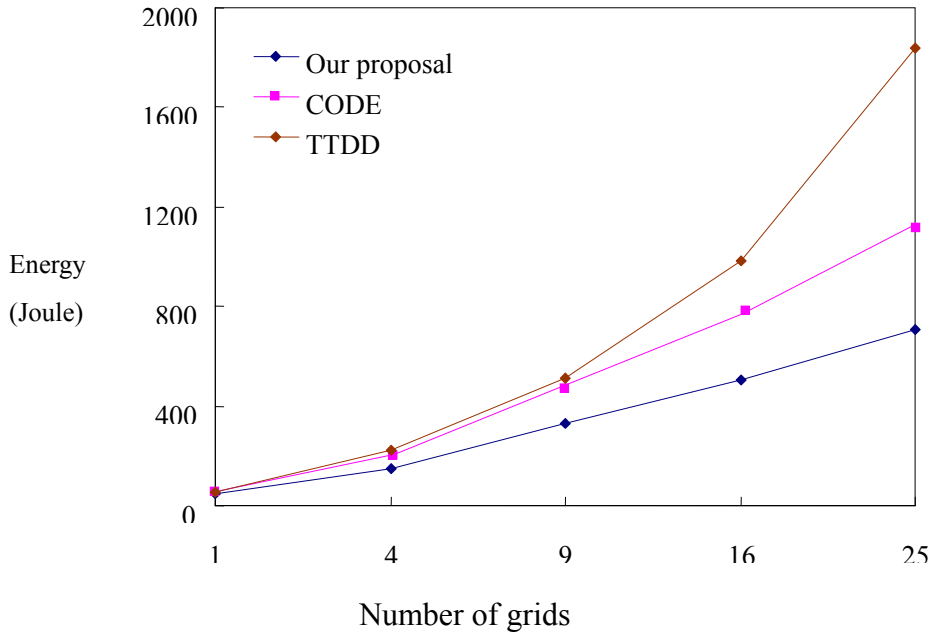


Figure 17. The energy consumed of different number of grids.

Figure 18 shows the performance of the total consumed energy under various different maximum speeds of sinks, which ranged from 0 m/s to 20 m/s that is region of interest. We assumed that the number of sink was one. The total energy consumption of proposed scheme, CODE and TTDD increased when the maximum speed of sinks increased, but the total energy consumption of our scheme is lower than CODE and TTDD.

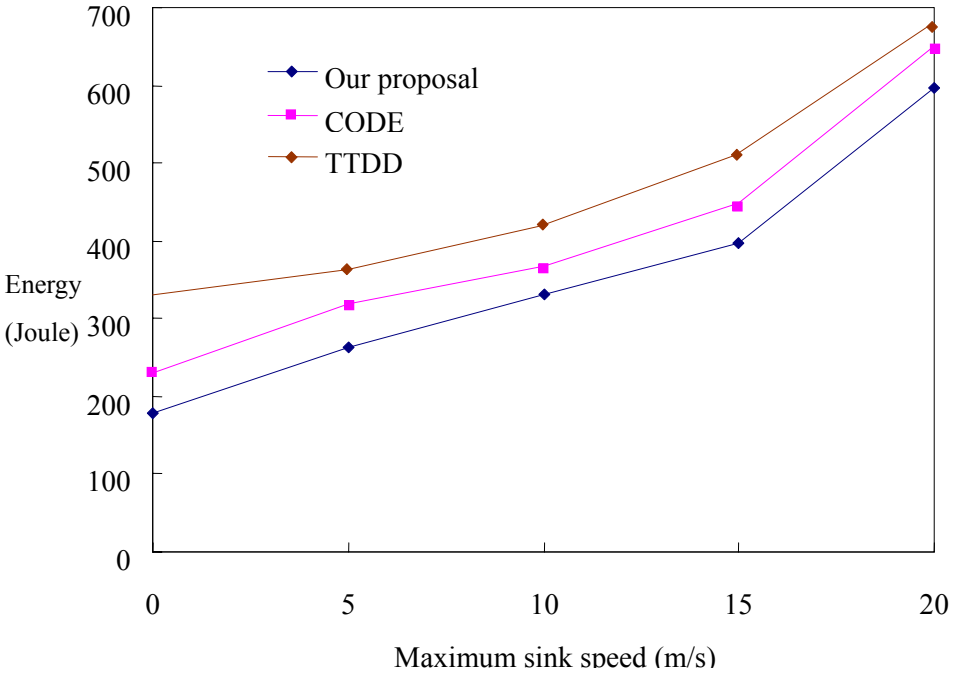


Figure 18. The energy consumed of different maximum speed of sinks.

.

Chapter 5

Conclusions

We proposed an efficient management mechanism for WSNs that includes the following benefits:

(1). The proposed method can significantly conserve the memory of a sensor node.

(2). Dynamic key management for each transmission will be used only once. This method reduces the probability of an attacker guessing a key correctly and improves security.

(3). The total energy consumption in our thesis is lower than in the other schemes except the DARQ scheme.

(4). The proposed thesis is a grid-based approach. We also limited the flooding region to decrease the overhead for routing discovery to reduce the probability of a flooding storm.

(5). In future research, we will propose a solution to find a routing detour for void regions, since void regions exist in a network because some grids do not deploy sensors. . In addition, the design of multiple interest regions will be taken into consideration to provide data aggregation for WSNs.

In this thesis, we proposed a dynamic session key management and spare two-tier data

aggregation for grid-based wireless sensor networks wireless sensor networks scheme. The goals are to establish spare path that has network fault tolerance. In terms of energy consumption, the proposed scheme establishes a spare two-tier data aggregation in the interesting region. In the interesting region, two paths can enhance fault tolerance. Each root node can be a spare path.

In terms of the energy consumption, the presented approach also established spare path in the interesting region. In the interesting region, the node which has the highest residual energy is chosen as the roots. Each other nodes choose its child node among its neighbors based on the information of the residual energy distance to the root. Therefore, the roots which have more residual energy can connect to more neighboring gateways and this can reduce energy consumption to extend the system lifetime. Simulation results show that our proposed scheme outperforms CODE and TTDD.

References

[1] C. Intanagonwiwat, R. Govindan, D. Estrin, J. Heidemann, and F. Silva, “Directed Diffusion For Wireless Sensor Networking,” Proceedings of the IEEE Transactions on Networking, pp.2-16, February 2003.

[2] D. Estrin, R. Govindan, J. Heidemann, and S. Kumar, “Next century challenges: Scalable coordination in sensor networks,” Proceedings of the 5th IEEE/ACM Annual Conference on Mobile Computing and Networks, pp. 263-270, August 1999.

[3] J. M. Rabaey, M. J. Ammer, J. L. Silva, D. Patel, and S. Roundy, “PicoRadio Supports Ad Hoc Ultra Low Power Wireless Networking,” IEEE Computer Magazine, pp. 42-48, July 2000.

[4] J. Hill, R. Szewczyk, A. Woo, S. Hollar, D. Culler, and K. Pister, “System Architecture Directions for Networked Sensors,” Proceedings of the International Conference on Architectural Support for Programming Languages and Operating Systems, pp. 93-104, November 2000.

[5] I. F. Akyildiz, W. Su, Y. Sankarasubramanian, and E. Cayirci, “A Survey on Sensor Networks,” IEEE Communications Magazine, Vol. 40, Issue 8, pp. 102-114, August 2002.

[6] Y. J. Zhao, R. Govindan, and D. Estrin, "Residual Energy Scan for Monitoring Sensor Networks" Proceedings of the IEEE Wireless Communications and Networking Conference, pp. 356-362, March, 2002.

[7] W. Diffie, and M. E. Hellman, "New directions in cryptography," IEEE Transactions on Information Theory, Vol. 22, pp.644–654, November 1976.

[8] R. Rivest, L., A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," Communications of the ACM, Vol. 21, pp.120–126, February 1978.

[9] R. Blom, "An optimal class of symmetric key generation systems," Proceedings of EUROCRYPT'84, Vol. 209, pp. 335–338, 1985.

[10] R. Pietro, L. Mancini, and A. Mei, "Random key-Assignment for Secure Wireless Sensor Networks", ACM workshop on Security of ad hoc and sensor networks, pp. 62-71, October 2003.

[11] D. Liu and P. Ning, "Establishing Pairwise Keys in Distributed Sensor Networks," ACM conference on Computer and communications security, Vol. 8, pp. 41-77, February 2005.

[12] Y. Cheng and D. P. Agrawal, "An improved key distribution mechanism for large-scale hierarchical wireless sensor networks," Ad Hoc Networks, Vol.5, No. 1, pp.35-48, January 2007.

[13] G. Jolly, M.C. Kuscu, P. Kokate, M. Younis, "A Low-Energy Key Management Protocol for Wireless Sensor Networks,"Proceedings of Eighth IEEE International Symposium on Computers and Communication (ISCC 2003), Vol.1, pp.335-340, 2003.

[14] C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks," Proceedings of the Seventh Annual ACM/IEEE International Conference on Mobile Computing and Networking, pp. 56-67, August 2000.

[15] H.-L. Xuan, D.-H. Seo, S. Lee, and Y.-K. Lee, "Minimum-Energy Data Dissemination in Coordination-Based Sensor Networks," Proceedings of the 11th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications, pp. 381-386, August 2005.

[16] F. Ye, L. Haiyun, C. Jerry, L. Songwu, and Lixia Zhang, "Sensor Networks: A Two-Tier Data Dissemination Model for Large-Scale Wireless Sensor Networks," Proceedings of the Eighth Annual ACM/IEEE International Conference on Mobile Computing and Networks, pp. 148-159, September 2002.

[17] H. Xuan and S. Lee, "A Coordination-Based Data Dissemination Protocol for Wireless Sensor Networks," Proceedings of the Sensor Networks and Information Processing Conference, pp. 13-18, December 2004.

[18] H. Kim, T. Abdelzaher, and W. Kwon, "Minimum-Energy Asynchronous Dissemination to Mobile Sinks in Wireless Sensor Networks," Proceedings of the 1st International Conference on Embedded Networked Sensor Systems, pp. 193-204, November 2003.

[19] R. Want, A. Hopper, V. Falcao, and J. Gibbons, "The Active Badge Location System," Proceedings of the ACM Transactions on Information Systems, pp. 91-102, January 1992.

[20] T.-S. Chen, Y.-S. Chang, H.-W. Tsai, and C.-P. Chu, "Data Aggregation for Range Query in Wireless Sensor Networks" Proceedings of the IEEE Wireless Communications and Networking Conference, Hong Kong, March 2007.

[21] W. Chen, L. Chen, Z. Chen, and S. Tu, "WITS: A Wireless Sensor Network for Intelligent Transportation System," Proceedings of the First International Multi-Symposiums on Computer and Computational Sciences, pp. 635-641, April 2006.

[22] W.-H. Liao, Y.-C. Tseng, and J.-P. Sheu, "GRID: A Fully Location-Aware Routing Protocol for Mobile Ad Hoc Networks," Telecommunication Systems, Vol. 18, No. 1, pp. 37-60, 2001.

[23] B. H. Wellenhoff, H. Lichtenegger, and J. Collins, "Global Positions System: Theory and Practice," Fourth Edition, Springer Verlag, 1997.

[24] D. Niculescu and B. Nath, "Ad Hoc Positioning System (APS) Using AoA," Proceedings of the IEEE 22nd Annual Joint Conference of the IEEE Computer and Communications Societies, San Francisco, pp. 1734-1743, April 2003.

[25] F. Kuhn, R. Wattenhofer, Y. Zhang, and A. Zollinger, "Geometric ad-hoc routing: of theory and practice," Proceedings of the 22nd ACM Symposium on the Principles of Distributed Computing, Boston, Massachusetts, pp. 63-72, July 2003.

[26] N. Bulusu, J. Heidemann, and D. Estrin, "GPS-Less Low Cost Outdoor Localization for Very Small Devices," IEEE Personal Communications Magazine, pp. 28-34, October 2000.

[27] A. Savvides, C. C. Han, and M. Srivastava, "Dynamic Fine-Grained Localization in Ad-Hoc Networks of Sensors," Proceedings of the 7th IEEE/ACM Annual Conference on Mobile Computing and Networks, Rome, Italy, pp. 166-179, July 2001.

[28] A. Savvides, H. Park, and M. B. Srivastava, "The Bits and Flops of The n-hop Multilateration Primitive for Node Localization Problems," Proceedings of the first ACM international workshop on Wireless Sensor Networks and Applications, Atlanta, GA, pp. 112-121, September 2002.

[29] A. Nasipuri and K. Li, "A Directionality Based Location Discovery Scheme for Wireless Sensor Networks", Proceedings of ACM Workshop on Wireless Sensor Networks and Applications, Atlanta, Georgia, pp. 105-111, September 2002.

[30] C. Savarese, J. Rabaey, and K. Langendoen, “Robust Positioning Algorithms for Distributed Ad-Hoc Wireless Sensor Networks,” Proceedings of the USENIX Technical Annual Conference, Monterey, CA, pp. 317-327, June 2002.

[31] R.-G. Lee, K.-C. Chen, S.-S. Chiang, C.-C. Lai, H.-S. Liu, and M.-S. Wei, “A Backup Routing with Wireless Sensor Network for Bridge Monitoring System,” Proceedings of the 4th Annual Communication Networks and Services Research Conference, pp. 157-161, May 2006.

[32] S. Aeron, V. Saligrama, D.A. Castaon, Efficient Sensor Management Policies for Distributed Target Tracking in Multihop Sensor Networks “, IEEE transactions on signal processing, Vol. 56, No6, pp. 2562-2574, 2008.