A SECURED SECTOR BASED BI-PATH CLUSTERING AND ROUTING PROTOCOL FOR WIRELESS SENSOR NETWORK

By

YIHONG ZANG

Master of Engineering

Southeast University

Nanjing, China

1992

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
MASTER OF SCIENCE
May, 2007

A SECURED SECTOR BASED BI-PATH CLUSTERING

AND ROUTING PROTOCOL FOR WIRELESS

SENSOR NETWORK

Thesis Approved:

Dr. Johnson Thomas

Thesis Adviser

Dr. John P. Chandler

Dr. Debao Chen

Dr. A. Gordon Emslie

Dean of the Graduate Collage

ACKNOWLEDGEMENTS

I wish to express my sincere appreciation to Dr. Johnson Thomas, who directed me to finish this thesis. I would like to thank him for sharing his understanding and knowledge in this WSN area and for his continuous support and guidance for my research. I would also like to thank my committee members, Dr. John Chandler and Dr. Debao Chen, for their thoughtful comments and advices.

Heart-felt thanks go to my husband and my parents for their unending encouragement and emotional support throughout the years. A special thank-you goes to my two sons, Jerry and Kevin, for many happy and joyous moments they gave me.

Finally I would like to thank all my friends who stood beside me with their unfailing and indispensable support.

TABLE OF CONTENTS

LIST OF FINGURES

Figure Page

LIST OF TABLES

Table Page

LIST OF SYMBOLS

GPS	-	Global Positioning System
GTSNetS	-	Georgia Tech Sensor Network Simulator
MANET	-	Mobile ad hoc network
MANIACS	-	The Modeling & Analysis of Networks vIA Computer Simulation
MECN	-	Minimum Energy Communication Network
MEMS	-	Micro Electro Mechanical System
MRE	-	Minimum Residual Energy
NDV	-	Node Distribution Value
LEACH	-	Low-Energy Adaptive Clustering Hierarchy
PDF	-	Probability Density Function
PEGASIS	-	Power Efficient Gathering in Sensor Information Systems
RREQ	-	Route Request Message
RREP	-	Route Reply Message
SBCR	-	Sector Based Clustering and Routing
SPIN	-	Sensor Protocols for Information Negotiation
SSBBCR	-	Secured Sector Based Bi-path Clustering and Routing
WSN	-	Wireless Sensor Network

CHAPTER I

INTRODUCTION

With recent advancements and developments in wireless communication and Micro Electro Mechanical System (MEMS) technologies, low-cost and low-power consumption wireless micro sensors that possess sensing, signal processing and wireless communication capabilities have become available [1]. A wireless sensor network consists of hundreds or thousands of wireless sensor nodes which could either have a fixed location or are randomly deployed to monitor the environment. These sensor nodes are distributed in a region in uncontrolled and unorganized ways. The development of wireless sensor networks was originally motivated by military applications such as battlefield surveillance. However, wireless sensor networks now have many important applications in many civilian application areas, including environment and habitat monitoring, target tracking, security, healthcare applications, home automation, and traffic control, etc [2][3].

Sensor data obtained at sensor nodes are sent to a base station (also called sink) through wireless communications. A base station summarizes collected data and presents these to a user or sends these to a remote host. Since sensor nodes derive power from small disposable batteries, this puts significant constraints on the power available for communications, thus limiting the transmission range. A sensor in such a network can therefore communicate directly only with other sensors that are within a small distance.

To enable communication between sensors not within each other's communication range, the sensors form a multi-hop communication network (Fig. 1.1) [2]. Since the sensors are now communicating data over smaller distances in the clustered environment, the energy spent in the network will be much lower than the energy spent when every sensor communicates directly to the base station.

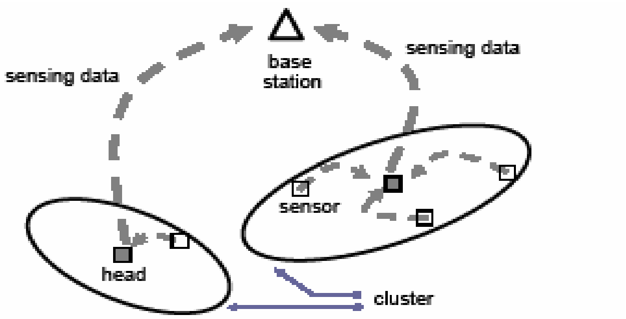


Figure 1.1 Cluster-based data gathering in sensor networks

Wireless sensor networks are similar to ad-hoc wireless networks in their clustering and multi-hop communication methods. However, they have many differences:

- More nodes are deployed in a sensor network, up to hundreds or thousands of nodes.
- Sensor nodes are more constrained in computational, energy and storage resources than nodes in an ad hoc network.
- Sensor nodes can be deployed in environments without the need for human intervention and can remain unattended for a long time after deployment.
- Neighboring sensor nodes often sense the same events from their environment, thus forwarding the same data to the base station resulting in redundant information.

- Aggregation and in network processing often require trust relationships between sensor nodes that are not typically assumed in ad hoc networks.

A wireless sensor node has very scarce resources in terms of memory, energy and computational capacities since it typically runs on battery power and has small size. Thus, sensing devices must operate under severe resource constraints and one of the foremost goals is to minimize energy consumption. Therefore, there is a need for an energy-efficient clustering and routing scheme to disseminate information to the sink. To date, little effort has been placed on protocols considering both security and energy efficiency. Sudheer et al [4] proposed a sector-based clustering and routing (SBCR) protocol for improving both security and energy efficiency. The sector based clustering provides better energy efficiency while multi-path routing algorithms enhance the security of data by using the overlapping cluster architecture to generate multiple paths for data transmission. Clustering also reduces energy consumption leading to increased survivability of the network. However, several drawbacks that exist in multi-path routing:

1. Multi-path routing introduces significant overhead in the Route-Discovery process due to broadcasting route request packets along multiple paths, resulting in reduced energy efficiency when clustering.
2. Data transmission along multiple paths causes performance degradation from a Quality of Service perspective.
3. The routing algorithm identifies and calculates as many paths as possible until all possible paths are discovered, consuming significant computational resources and energy and therefore decreasing the survivability of the network.

4. The security characteristic provided by multi-path routing is weak and impractical compared to a key pre-distribution scheme.

In this thesis, we propose a novel integrated protocol for further improving both the energy efficiency and security based on Sudheer's approach. The proposed Secured Sector Based Bi-path Clustering and Routing (SSBBCR) protocol increases the survivability of the network by reducing redundant computations involved in multiple route-discovery, thereby improving the energy efficiency and providing better security through the enforcement of the trust model proposed in chapter 3. In this trust model, a key pre-distribution scheme with deployment knowledge is established where keys are pre-distributed such that neighboring nodes that are in the route are more likely to share keys than neighboring nodes that are not in the route. We will evaluate the performance of the proposed scheme by comparing with Sudheer's approach. Our objective is to show that our scheme provides more energy efficiency and better security.

The remainder of this paper is structured as follows: In Chapter 2, we review related work. Chapter 3 explains the problem to be solved and describes the motivation for this work. Our proposed approach for Secured Sector Based Bi-path Clustering and Routing is given in chapter 4. Chapter 5 gives the implementation of simulation and, simulation results and observations are presented in Chapter 6. Chapter 7 concludes the thesis and point out several future research directions.

CHAPTER II

LITERATURE REVIEW

With the recent advances in wireless sensor networks, many new protocols have been designed specifically for sensor networks where energy awareness is an essential consideration. Most of the attention has been given to the routing protocols since they might differ depending on the application and network architecture. Routing in WSNs is very challenging due to several characteristics that distinguish them from contemporary communication and wireless ad-hoc networks. The differences are:

1. For the deployment of sheer number of sensor nodes, classical IP-based protocols cannot be applied to sensor networks due to lack of a global addressing scheme.
2. Almost all applications of WSNs require the flow of sensed data from multiple regions to a base station.
3. Routing protocols need to exploit the significant redundancy generated by the adjacent sensor nodes to improve energy and bandwidth utilization.
4. Resource management is typically required for WSNs because the sensor nodes are tightly constrained in terms of energy, transmission power, processing capacity and memory storage.

Besides these challenges, routing protocols in sensor networks still have to deal with issues such as scalability, robustness (maintain connectivity), load balancing in terms of energy used by nodes, low latency, minimum computation and memory usage, energy efficiency and security. Routing protocols are generally categorized as: data-centric, hierarchical and location-based [8].

2.1 Data Centric Protocols

In data-centric routing, the sink sends queries to certain regions and waits for data from the sensors located in the selected regions. Since data is being requested through queries, attribute based naming is necessary to specify the properties of data. Data – centric routing protocols have been proposed to be able to select a set of sensor nodes and utilize data aggregation during the relaying of data. SPIN [9] and Directed Diffusion [10] belong to this category.

The idea behind SPIN is to name the data using high-level descriptors or metadata. Before transmission metadata are exchanged between the sensors via a data advertisement mechanism, which is the key feature of SPIN. Directed diffusion aims at diffusing data through sensor nodes by using a naming scheme for the data. Directed Diffusion differs from SPIN in terms of the on demand data querying mechanism it has. One advantage of Directed diffusion over SPIN is that in it all communication is neighbor-to-neighbor with no need for a node addressing mechanism.

2.2 Hierarchical Protocols

The main aim of hierarchical routing is to efficiently maintain the energy consumption of sensor nodes by involving them in multi-hop communication within a particular cluster and by performing data aggregation and fusion in order to decrease the number of transmitted messages to the sink. Cluster formation is typically based on the energy reserve of sensors and sensor's proximity to the cluster head. LEACH [11], PEGASIS [12], Hierarchical-PEGASIS [13], TEEN and APTEEN [14] are fallen into this category.

LEACH uses one level cluster-based hierarchy model, in which nodes are grouped into one level of clusters and data is first aggregated to the cluster head and then sent directly to the base station. It consists of two phases: set-up phase and steady-phase. In the set-up phase, sensors may elect randomly among themselves a local cluster head with a certain probability. The optimal number of cluster heads is 5% of the total. After the cluster heads are selected, the heads advertise to all sensor nodes in the network that they are the new cluster heads. Once the nodes receive the advertisements, they decide which head they belong to. In the steady-phase, sensors sense and transmit data to the base through their cluster heads. After a certain period spent in the stead-state, the network goes into the set-up phase again and enters another round of selecting cluster heads. LEACH form clusters of the sensor nodes based on the received signal strength and use local cluster heads as routers to the sink therefore save energy since the transmissions will only be done by such cluster heads rather than all sensor nodes. However, LEACH assumes that all nodes have enough power to directly communicated with the base, such assumption is not applicable to networks deployed in large regions. Another drawback is that cluster heads communicate directly with the base, possibly causing channel overload at the base station.

2.3 Location Based Protocols

In most cases for sensor networks, location information is needed in order to calculate the distance between two particular nodes so that energy consumption can be estimated. Since, there is no addressing scheme for sensor networks like IP-addresses and they are spatially deployed, location information can be utilized in the routing data in an energy efficient way. MECN [15] is one of the energy-aware location based protocols for sensor networks. Minimum Energy Communication Network (MECN) sets up and maintains a minimum energy network for wireless networks by utilizing low power GPS. The main idea of MECN is to find a sub-network, which will have a small number of nodes and require little power for transmission between any two nodes. In this way global minimum paths are found without considering all of the nodes in the network.

2.4 Sector Based Clustering and Routing (SBCR) Protocol [4]

This protocol has been proposed for ad hoc networks aims to improve both energy efficiency and security. The Sector-based approach to clustering provides overlapping clusters while multi-path routing algorithms utilize this overlapping cluster architecture to generate multiple paths for data transmission. Clustering reduces the distance to which a node has to communicate and hence reduces energy consumption. It divides a node communication region into N equal sectors. The number of sectors N is the same for all nodes in the network and is defined at initialization. A node becomes a cluster head according to its Power Level (P), Node Distribution Value (NDV), and Security Level (L) (figure 2.1). A cluster is formed such that it contains no more than a maximum of M nodes and all the member nodes of the cluster are evenly distributed around the cluster head. Sector based clustering regulates the number of nodes present so that each cluster has more or less the same number of nodes and balances the load on the cluster head such that member nodes are evenly distributed. It also allows nodes to be members of multiple clusters (called gateway nodes) in a controlled way. This helps in even dissipation of energy among nodes, thereby increasing the survivability of the network.

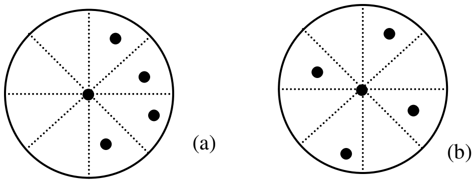


Figure 2.1 Sectors and Node Distribution Value (N=8)
Node b is more likely to be chosen as cluster head due to its higher NDV

The sector based routing approach generates multiple paths in a controlled way. The route generation starts at the destination instead of at the source node and the paths tend to be elliptical with source and destination at their foci. This increases the difficulty in predicting a node that belongs to a path used for data transmission and therefore provides better security.

The sector based clustering approach can be applied to WSNs due to its multi-hop communication mechanism. The characteristics of load balancing, cluster overlapping, and equality in cluster members provide more energy efficiency and increase the survivability of the network. However, the multi-path routing approach is not appropriate to WSN in that large amounts of communication are needed for data transmission along multiple paths and redundant computations are involved in calculating multiple paths, which consume a lot of energy. The security provided by multi-path routing is weak and not sufficient for the requirements of WSNs.

2.5 Security Models in WSNs

Security in wireless sensor networks is achieved by encrypting the messages sent between the sensors and the base. Keys for encryption purposes must be agreed upon by communicating nodes. Due to resource constraints, achieving such key agreement in wireless sensor networks is non-trivial. Pair-wise key establishment is a fundamental service provided in secure sensor networks. However, due to resource constraints, establishing pair wise keys is impractical because a large amount of memory is needed to store the keys. The latest and most advanced technique in achieving security approach is random key pre-distribution schemes and its improvements. Here we present three representative random key pre-distribution schemes.

(1) Random Key Pre-Distribution Scheme [5]

The Eschenauer-Gligor scheme proposed a large key pool, and keys randomly selected from the key pool are distributed to each node. Therefore, any two nodes can share one common key with a certain probability. In addition, a compromise of one node only causes the compromise of a limited part of the sensor network. The scheme consists of three phases: key pre-distribution phase, shared-key discovery phase, and path-key establishment phase. The drawback of this scheme is that a large number of keys have to be stored at each node to ensure high connectivity which consumes a lot of memory storage.

(2) Random Key pre-distribution scheme using deployment knowledge [6]

An improvement of the Eschenauer-Gligor scheme was proposed by Du et al [6]. It uses deployment knowledge for key pre-distribution. In the Du et al. scheme, the target deployment area is divided into N rectangles; different key pools are assigned to each of these rectangles in a logical manner: the nearer the rectangles, the greater the number of keys shared among key pools associated with the rectangles. As a result, two nodes deployed to neighboring rectangles can share keys with a high probability. This scheme presents better performance in terms of connectivity and memory usage and keeps the sensor networks secure. However, it can only be applied to the group-based deployment model.

(3) Pre-distribution scheme using probability density function of node deployment [7]

Ito et al improved Du et al's scheme by using Probability Density Function (PDF) of node deployment. In his scheme, different keys are logically mapped to two dimensional positions, and the keys that are distributed to a node are determined by positions estimated using a node PDF. This scheme is similar to Du's [6] scheme where the main difference lies in the granularity of the scheme. The granularity is finer, so that the scheme has the characteristic that the nearer the nodes, the higher the probability that they can share keys. The scheme can be applied to any deployment model provided the node PDF has already been determined. However, this scheme does not differentiate between neighboring nodes. In other words, it is important that communicating neighboring nodes (or neighboring nodes that are in the route) share keys whereas neighboring nodes that do not communicate (or not in the route) do not share keys.

CHAPTER III

PROBLEM DEFINITION

In wireless sensor networks, protocols have been considered separately for security and for energy efficient routing. There have been very few protocols, if any, which consider both of these important parameters. To date the most efficient schema that considers both energy efficiency and security has been the SBCR protocol [4]. In this approach, Sector Based Clustering reduces energy consumption leading to increased lifetime of the overall network by keeping the nodes alive for longer periods and regulating the number of nodes present in each cluster. The multi-path routing utilizes the overlapping cluster architecture to generate multiple paths for data transmission, which enhances the security of data. However, the SBCR protocol is proposed for mobile ad hoc networks (MANET) where the nodes are less energy and resource constrained. Comparing sensor nodes to ad-hoc wireless networks, wireless sensor networks share similarities with ad hoc wireless networks in the clustering and multi-hop communication methods. In our proposed SSBBCR protocol for WSN, we adopted the Sector based clustering algorithm proposed in [4] due to its improved minimal power consumption compared to the other clustering methods proposed in WSN. However, wireless sensor networks have many differences with MANET in that:

1. More nodes are deployed in a sensor network than in an ad-hoc network.

2. Sensor nodes are more constrained in computational, energy and storage resources than ad hoc,
3. Sensor nodes can be deployed in environments without the need of human intervention and can remain unattended for a long time after deployment.
4. Aggregation and in–network processing often require trust relationships between sensor nodes that are not typically assumed in ad hoc networks.

Due to the differences described above, the multi-path routing approach proposed in [4] is limited when applied to the wireless sensor network:

1. It introduces significant overhead in Route-Discovery due to broadcasting route request packets along multiple paths, resulting in reduced energy efficiency when clustering.
2. Data transmission along multiple paths introduces a large amount of communication between the nodes, and causes performance degradation from a Quality of Service perspective.
3. The routing algorithm identifies and calculates as many paths as possible until all paths are discovered, consuming significant computational resources and energy, therefore decreasing the survivability of the network.

The security characteristic provided by multi-path routing is weak and impractical in WSNs. Sensor networks are often used in mission critical environments such as in military and healthcare applications. These environments have demanding security requirements that must be addressed at the initial phase of design. A robust trust relationship between nodes is required to provide security for the network. To solve this problem, in our thesis, we propose a key pre-distribution scheme with pre-determined routing knowledge for sensor networks.

In this scheme keys are pre-distributed based on the knowledge of nodes that are likely to be the neighbors of each other. Keys are pre-distributed such that neighboring nodes in the route are more likely to share keys than neighboring nodes not in the route, therefore increasing the connectivity between neighboring nodes in a route while minimizing connectivity between neighboring nodes not in the route. The detailed process is discussed in Chapter 4.

CHAPTER VI

PROPOSED APPROACH

We assume that the sensor nodes in our wireless sensor network are homogenous, static and largely distributed. Homogenous means all the sensor nodes except the sink node have the same capabilities in terms of size, battery power, memory storage, and computational and communicational abilities.

4.1 General Description

SSBBCR (Secured Sector-based Bi-path Clustering and Routing) protocol includes 5 phases: network initialization, cluster formation, bi-path routing discovery, key picking and shared key discovery, and maintenance.

a. In the network initialization phase, the sensor nodes are randomly deployed into the network, and then keys and a key-position map are created. We assume all the nodes are stationary in our application.
b. In cluster formation phase, cluster heads are elected and cluster members are included into corresponding cluster heads.
c. In the bi-path routing discovery phase, the base station triggers the bi-path discovery mechanism to discovery a primary path and a back up path for data transmission.

d. In the shared key discovery phase, for all the nodes on the route, we pick the keys selected from the sector areas and delete the rest of keys pre-distributed on the nodes. Then the shared key discovery mechanism is applied to discover the shared keys between the neighboring nodes in the route.

e. The maintenance phase involves periodically checking the remaining power in the route nodes, re-organizing the network when the primary route path is abandoned in cases such as some of the nodes in the route die due to failure, compromised or running out of power, etc.

4.2 Network Initialization Phase

This phase consists of deployment of nodes in the network. In this phase keys generation and a key-position map are created. We assume that sensor nodes are deployed in a two dimensional area and the locations of nodes can be specified by coordinates in a two dimensional coordinate system. The communication range of each node is assumed to be equal and constant and is denoted as r.

1. Set security parameters *n*, *m*, *r,* and λ

   n is the size of the key pool, *m* is the number of keys stored by each node, *r* is the radius of the key picking area, and λ is the Angle of two opposite sectors of key picking area. These parameters are determined according to the capability of nodes and requirements of connectivity and security.

2. Divide the target deployment area into *n* rectangles of the same size.
3. Generate *n* random keys.

4. A key-position map is generated by assigning *n* keys to *n* subfields (rectangles) randomly in a one-to-one manner (see fig 4.1).
5. Pre-distribute *m* keys for all the sensor nodes

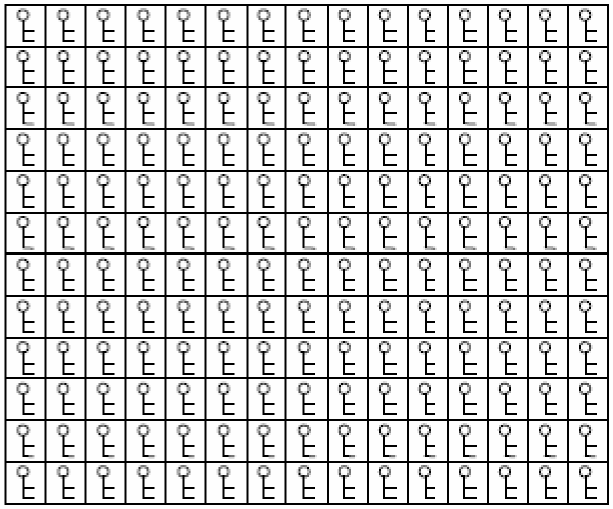


Figure 4.1: Key-position map

4.3 Cluster Formation Phase

Since our protocol is applied on the wireless sensor network, in contrast to SBCR, there exist many differences in SSBBCR clustering algorithm. We classify all the nodes in the network into cluster head nodes, cluster member nodes and gateway nodes. Every cluster member belongs to exactly one cluster head. A gateway node is a node that is in the communication range of two or more cluster-heads. A gateway node may be the member of one cluster or may not be the member of any cluster. The cluster head is one hop away from its cluster members and at least two hops away from its neighbor cluster head. Each node maintains a neighbor table. For each neighbor, the neighbor table of a node contains the state of the neighbor (cluster-head, gateway or member). A cluster-head keeps information about the members of its cluster and also maintains a cluster adjacency table that contains information about the neighboring clusters. For each neighbor cluster, the table has entry that contains the gateway through which the cluster can be reached and the cluster-head of the cluster.

In our approach, the Cluster Formation phase consists of two stages:

a. Cluster head Decision

b. Member Inclusion

4.3.1 Cluster Head Decision

Each node evaluates itself to decide whether it is eligible to become a cluster head according to its power level P and Node Distribution Value (NDV). The NDV is defined the same as in SBCR. The node with the highest NDV is most likely to become a cluster-head. The algorithm (Table 4.1) bellow is used to find the cluster head:

1. *Assign Pmin = the minimum acceptable power level a node is eligible to become a cluster head.*
2. *Assign Ns = the number of sectors a node have for all the nodes in the network.*
3. *For each node in the network, calculate its Node Distribution Value NDV.*
4. *If NDV > (Ns + 1) and P >= Pmin, this node is elected to be the cluster head.*
5. *Add CH_ID as the unique id of the cluster head into its routing table.*
6. *Add the node into the cluster head list CH_LIST*
7. *Repeat step 3 – 6*

Table 4.1: Algorithm for finding the cluster head

4.3.2 Member Inclusion

Since the sensor nodes have key pre-distribution scheme to ensure the security of the network, we do not need the parameter Security Level as evaluation values to decide the inclusion of the membership. Different from SBCR protocol, in which the nodes are moving, the sensor nodes are fixed in their location for most applications. We redefine the parameter Adjacency Value as the distance between two nodes, as the time and moving speed are not relevant here. To keep the balance of each cluster, the following conditions are satisfied when considering the inclusion of members:

1. The maximum number of *M* nodes can be accepted in each sector. For a cluster head, the maximum number of member nodes would be *M*Ns*.
2. If there are more than *M* nodes in a sector, the node with the least Adjacency Value is more likely to be included for membership. This ensures the cluster head includes the most adjacent members in its cluster.
3. If there exists other cluster heads that are eligible to be members of the current cluster head, then we compare the sum of NDV and P. The node with the greatest value of (NDV+P) is chosen as the cluster head and the others are subjected as its members. The cluster head with the greater value has the higher priority to choose its members. If one of its eligible members is a cluster head, it updates its status and becomes one of the member nodes of the cluster head. This way we ensure no other cluster head is falling within the communication range of another cluster head, making the clusters distribute evenly and less overlapping.

One node can be the member of exactly one cluster, meaning that if a node accepts the membership of one cluster head, it can not be the member of another cluster head. This ensures the data aggregation and fusion properly from the member nodes to their associated head node within one cluster.

If a node is not included as a member of any cluster by its neighbor cluster heads, the node may becomes a gateway node. The algorithm of Member Inclusion is given in Table 4.2:

```
8.  Sort the cluster head list CH_LIST by value (NDV+P) in descending order.
9.  For each head node in CH_LIST, do the following:
10. Broadcast bacons to all the neighbor nodes in its communication range, inviting them as its members
11. For I = 1 to Ns, in each sector do the following
12. For all nodes in sector I pick as many nodes as possible that are closest to the head node according to their Adjacency Values until the number reaches M.
13. For each node in the above picked nodes list, check if it is a cluster head. If true, then remove it from the CH_LIST. Move the rest of the head nodes in the list forwards and updating the CH_ID for the rest of the nodes after the removed nodes.
14. Add this node into the cluster list CT_LIST, assigning CT_ID = CH_ID, the id of the cluster head.
15. Repeat 13, 14.
16. Loop through all the sectors
17. Insert the head node as the first node of the CT_LIST. Thus the cluster corresponding to the cluster head is formed, with the first node as the cluster head and the rest of the nodes are members. They have the same CT_ID that is equal to the CH_ID of the cluster head.
18. Iterate all the nodes in CH_LIST, getting an array list of CT_LIST.
```

Table 4.2: Algorithm for cluster member inclusion

4.4 Bi-path Route Discovery Phase

We define the route nodes as the nodes that are on the discovered route. In this phase, two paths, a primary path and a backup path, are discovered from the base station to the source. The algorithm works as following:

When a source node S has to send data to base station D,

1) It sends a Route Request Message (RREQ) to its head. The head broadcasts the Route Request Message to all the cluster heads and gateway nodes surrounding it. Each sender will record its IP address in the RREQ.
2) On receiving the request a cluster-head first checks to see if it has received the same request before. If yes it discards the request. Then it checks to see if the destination D is in its cluster.
    - If yes, then it sends the request directly to the destination
    - Else, it sends it to all its adjacent cluster heads and gateway nodes, repeat step 2.
3) When the destination receives the request packet RREQ, it replies back the Route Reply Message (RREP) and initiates route discovery process to discover two paths between source S and base D. The RREP messages collect the route information
4) Two paths, a primary path and a back up path are discovered. Route nodes are selected based on the following rules:

- Shortest path mechanism: The next hop node try to be the node farthest away from the current head and closest to the source.
- Minimum residual energy requirement: Only when the residual energy of the node is great than the Minimum Residual Energy (MRE), can it be eligible to be selected as route node.
- Priority: Route nodes on the primary path are selected first. If a node is selected on one path, it could not be selected as a route node of another path, meaning that the two paths have no common nodes except the source and the base.

5) Each node in the network keeps a routing table. Routing ID (R_ID) is one entry of the table. Initially it is set to zero, meaning this node is not a route node. If this node is selected on the primary path, then R_ID is changed to 1. If it is on the back up path, R_ID is updated to 2. The primary path is the active route used to relay data from source to base, and the back up path is not active used as a candidate for the primary path. Once the primary path is broken, the back up path is upgraded to the primary path by setting its route nodes R_ID to 1.

6) When the source receives an RREP message it adds the paths to its list of routes. Then keys are picked and restored only for the route nodes on the primary path and data are transmitted from the source to the destination along the primary path.

4.5 Key Picking and Shared Key Discovery Phase

Since the routing is done after deployment, while keys are pre-distributed before deployment, the determined routing knowledge can guide the decision of key picking area.

1. For a sensor node, the deployment location is known. Suppose node *A*, *B*, and *C* are three adjacent route nodes. The key picking area from which the node *B* picks *m* keys is formed as follows (figure 4.2):
    a. Draw a disc of radius *Kr* and centered at location of node *B.*
    b. Four line segments start from node *B* divide the disc into four sectors where the two opposite sectors are chosen to be the key picking area and have the following properties:
        i. angle of sectors is λ.
        ii. Middle line of the sector is the route path from *B* to its next neighbor *A* and *C* (figure 4.2).
2. Pick the keys that belong to the sector areas and reserve them in the node's memory.
3. Delete all the rest of keys that pre-distributed on the node.
4. Repeat steps 1-3 for all route nodes that are on the route.
5. For the rest of nodes that are not on the route, delete all the keys they stored.

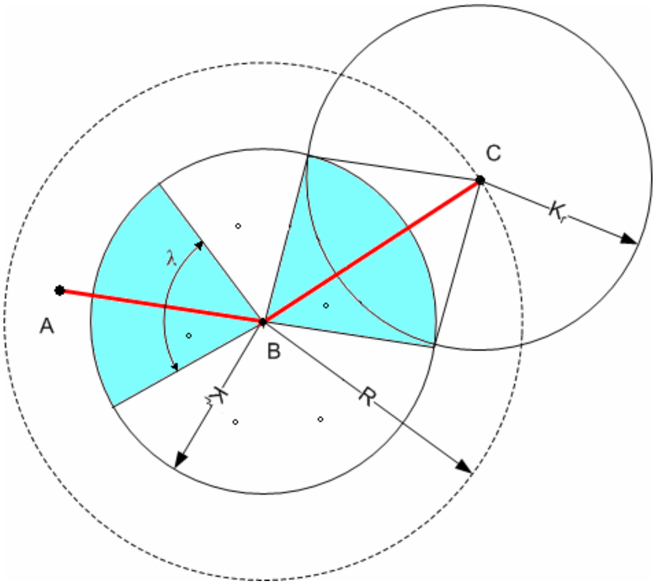


Figure 4.2 Key picking area

6. Node B broadcasts a challenge-response message to its neighboring node *A* and *C* (the node in the route) in one-hop.
7. For security reason, keys stored in node I are encrypted in function *Ek(i).* The decryption of *Ek(i)* by a recipient reveals the proper keys for establishing secured links between broadcaster and recipient. The recipient sends a reply to broadcaster to inform its node ID and challenge index, and stores the broadcaster's ID and respective keys in a table. The broadcaster does the same thing upon receiving the reply.
8. Repeat steps 6-7 for all nodes that are on the route.

Thus we establish a secured path between source and destination (base station).

Since the keys are picked and distributed only along the route nodes, so only the neighboring nodes that are on the route are likely to share keys, while the nodes that are not on the route have no keys stored on their memories, they could not share keys with the route nodes even if they are neighbors. The more overlapping of the two adjacent key picking areas, the more likely the two neighboring route nodes share common keys (Fig 4.3). In order to ensure the two adjacent sectors have overlapping areas, the radius *Kr* must satisfy: *Kr* > R/2. Here, R is the radius of the communication range of the sensor nodes.

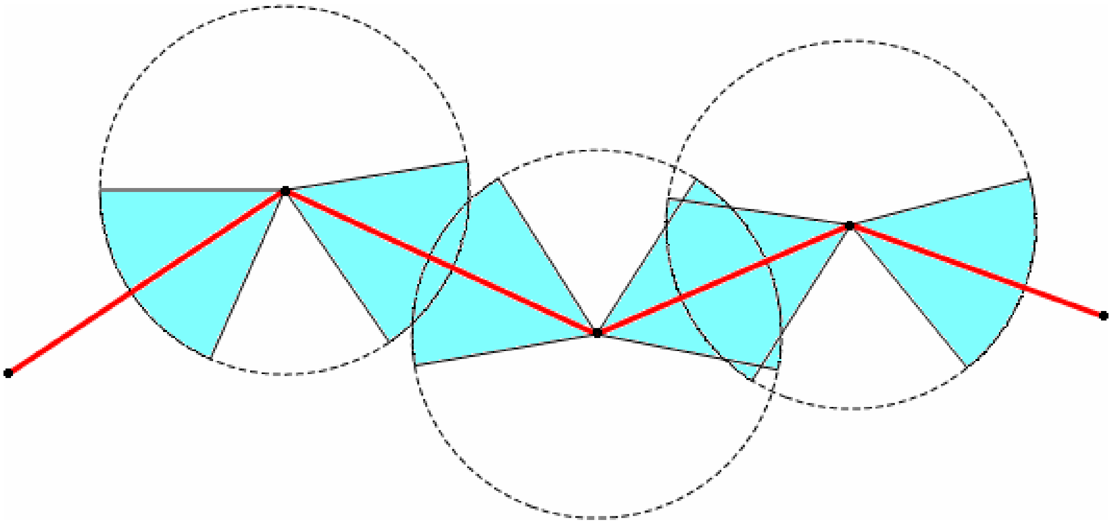


Figure 4.3 Proposed schemes for sharing keys

Given the communication radius R and key-picking radius *Kr*, the decision of sector angle λ has great affect on the probability of key sharing between adjacent route nodes. When λ is too small, we cannot pick enough keys for the nodes. However, when λ is too big, the proportion of overlapping area to the sector area is little, leading to less chance of keys being picked from the overlapping area. Therefore there exists an optimal value for angle λ. If two adjacent route nodes are at distance *R* far away, they have the least overlapping key picking area, which is the worst case for key sharing. Let us select the sector areas such that the overlapping areas are just covered (Figure 4.4). We can derive that:

$$\text{Cos}\ (\lambda / 2) = (R/2) / Kr$$

$$\lambda = 2^* \arccos (R / Kr/2)$$

If $R = 50$, $Kr = 30$, we have $\lambda = 67°$.

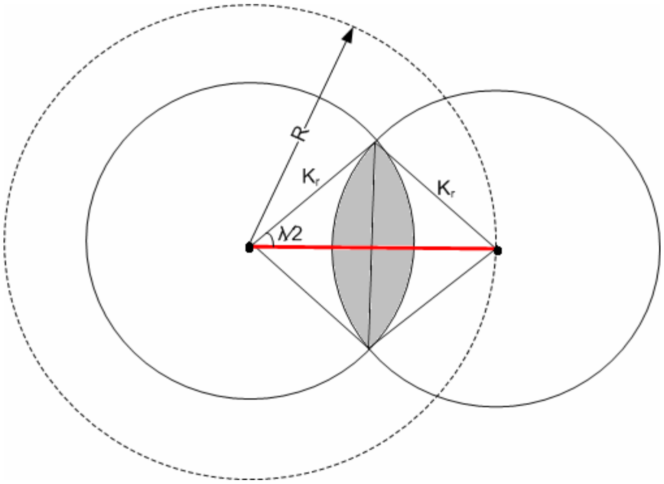
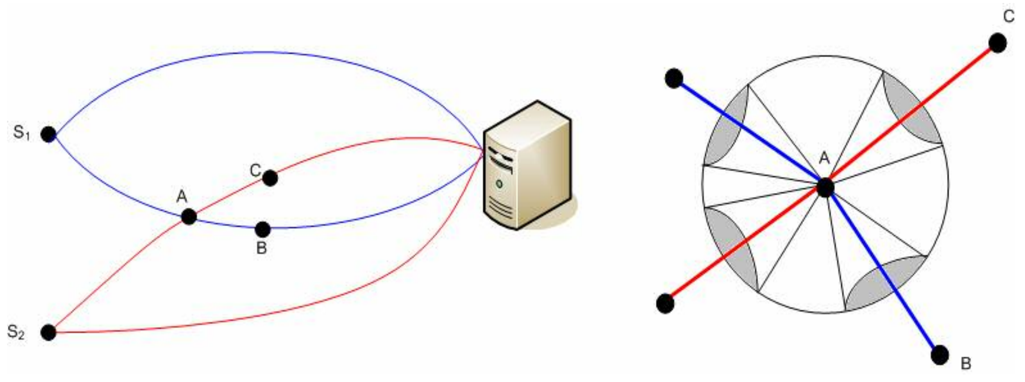


Figure 4.4 The relationship of sector angle λ, key picking radius *Kr*, and communication radius *R*

In a wireless sensor network, usually the base station collects data from multiple sources simultaneously. Multiple routes that are originated from different sources and end with the same destination may be formed in the same network at one time. Our proposed clustering and routing method can be applied in this scenario to discover two route paths for each source node. If any two paths intersect on some node, it will not have much affect on the key sharing of neighboring nodes on each of the two paths, since different path directions decide that they have different key picking area by not overlapping their sectors (Figure 4.5). The possibility of sharing common keys between nodes in one path and nodes in another path is therefore low.



(a) Two route paths with intersection on node A

(b) (A, C) and (A, B) share keys, (B, C) do not share keys.

Figure 4.5 The wireless sensor network with multiple sources and one base station

4.6 Maintenance Phase

The purpose of this phase is to reconfigure the network when the need for topology change arises. Example cases of such changes are discussed below.

Case 1: *Exhausted* Route Nodes

Since the route nodes have more functionalities than nodes not in the route, the energy stored in these nodes are exhausted quickly. When the energy level of a primary route node reaches the Minimum Residual Energy (MRE), it sends a report to the base. When the base gets the report, it broadcasts a message to the source along the route nodes to terminate the data transmission, abandon the current primary route and recall all the keys stored in the route nodes by releasing the allocated memories. Then it activates the backup route and uses it as the primary route. In the meanwhile, it starts from the cluster formation phase to reselect the cluster heads on those clusters whose cluster heads were not eligible to be heads because of low energy power. In path discovery phase, it only needs to discovery one back up path and add it into the route list. This is repeated until keys are picked and discovered on the new main route.

Case 2: *Dead* or *Damaged* Nodes

After a predefined interval of time, cluster head nodes require their member nodes to send their IDs to them. Member nodes that do not report to their heads are assumed to be dead or damaged and will be reported to the base. Similarly, if a member node did not receive any query from its head after the predefined interval of time (meaning the parent may be dead or damaged), it will turn to SLEEP mode and wait for further command from the base.

Case 3: *Re-positioned* Nodes

When a node's position changes (probably due to physical events, such as earthquakes, explosion, etc.), it will be considered as damaged by its head (case 2). After a node's position is changed, it will:

1. Automatically turns to SLEEP mode;
2. Broadcast a message indicating that its position needs to be updated.

Any node that has received the broadcast will forward the information to the base, which then updates the given node's position.

CHAPTER V

SIMULATION

5.1 Objective

The objective of this simulation is to compare the performance of Secured Sector Based Bi-path Clustering and Routing (SSBBCR) with Sector Based Clustering and Routing (SBCR) protocol. Another clustering-based protocol LEACH (Low-Energy Adaptive Clustering Hierarchy) is also selected for comparison. There are three types of metrics that are considered when comparing the performance of the three selected methods: total energy consumption, time before the first node dead, and the network lifetime. For SSBBCR we also evaluate the network connectivity under various node densities.

Th*e total energy consumption* is defined as the total energy that the network consumed for collecting and relaying to the base station certain type of sensed data.

The *time before first dead node* is the time when the first node of the network runs out of energy. This metric is a significant indication of the sensor network's 'well-being' or longevity.

The *network lifetime* is defined as the amount of time during which the network has been able to accomplish its tasks

5.2 Implementation Tools and Environment

The simulation program was implemented in C++ using Microsoft Visual studio.NET 2003 as development environment. The operating system is Windows XP Professional. In this program we adopted the model of sensor node and sink node defined in Georgia Tech Sensor Network Simulator (GTSNetS) [17]. Based on these models, we developed three application level classes for each of three protocols and assembled them in one simulation program.

The simulation model we have built to test the performance of the sensor network consists of five sub-models: a *sensor* model, a *sink* model, an *application* model*,* a *trace* model and a *simulation* model.

5.3 Simulation Architecture

A typical sensor node structure consists of 4 modules: Sensor, MCU, RF Radio and a battery. MCU (or Micro Controller Unit) is the module that is responsible for controlling all activities of node and executing communication protocols. The sensor module includes sensors attaching to the node and the Radio module is responsible for wireless communication. A battery provides energy for the three modules. The Modeling & Analysis of Networks vIA Computer Simulation (MANIACS) research group at Georgia Institute of Technology has designed a generic sensor node model. It is composed of three main functional units: a sensing unit, a communication unit and a computing unit. [17] (Fig 5.1). Based on this generic design, we built our simulation model as illustrated in Figure 5.2

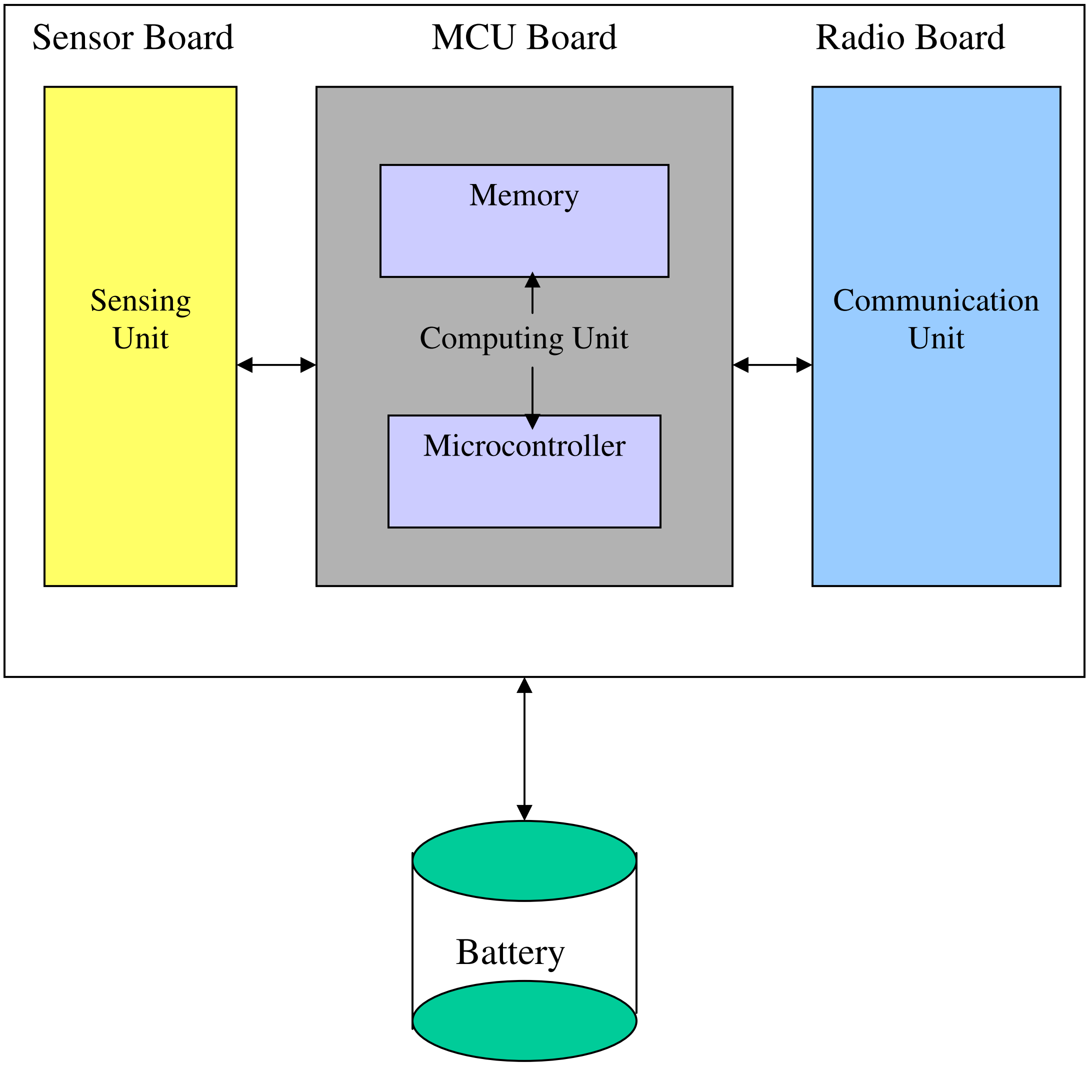


Figure 5.1: A simulated sensor node structure

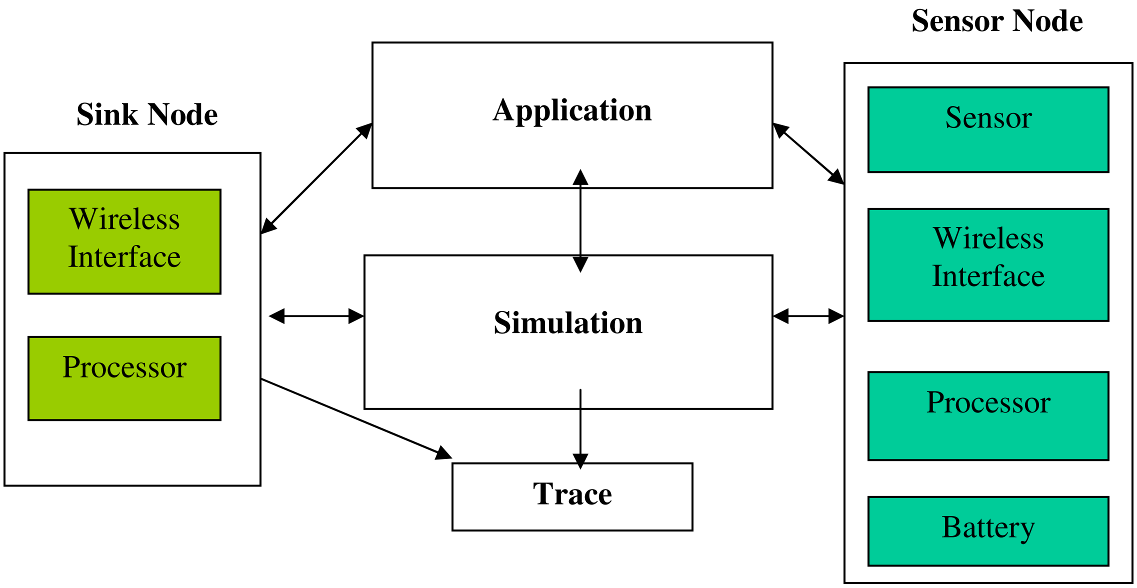


Figure 5.2: Architecture of simulation model

- The *Application* module represents the application layer (layer 5) of the simulation model. It is used for implementing protocol algorithms for sensor networks, including processing sensed data; forwarding requests or messages from node to node; selecting cluster head; cluster member including and route-discovery, etc.

- The *Simulation* module is an interface to connect all modules together. It corresponds to the C++ main program to coordinate all the modules and run the simulator. It first initiates an instance of a simulator object and other various network elements to describe a simulated topology, then creates an application object and uses the protocols defined in it to transmit the sensed data to the destination (e.g., sink node). It also creates a trace object to collect the statistics into a given file.

- The *Trace* module is used for packet tracing. The module creates a text file and all the traced data are written in this file. The state of each packet can be traced as it is forwarded through the network. Each time a message is sent or received by a sensor node, tracing occurs. Packet tracing can be enabled or disabled by different layers. A wide set of tracing options can be chosen to trace the different types of energies at node level as well as network-wide, node and network lifetime, etc.
- The *Sensor Node* module describes the properties and behaviors of wireless sensor node. It consists of four objects: the sensor object, the processor object, the wireless interface of sensor network object and the battery object. The *sensor* object corresponds to the sensing Unit in a sensor node. Two accuracy models can be chosen to collect the sensed data depending on the type of the sensor and the sensing environment. If the parameter *isActive* is set TRUE, the sensor is constantly collecting data at a fixed sensing rate and consumes energy. If FALSE, the sensor turns to SLEEP. The simulator keeps a record of the cumulative amount of energy consumed by the sensing unit. Users can define their own energy models to represent the sensing energy consumption for their specific applications. The *wireless interface* object represents the communication unit in a sensor node. It is in charge of relaying sensed data to the sink node and other sensor nodes. When it is on Active mode, the module consumes energy. The energy consumption is determined by energy consumption per bit for transmission and reception as well as energy consumption per bit for amplification purpose. Section 5.3 gives the detailed method for energy consumption calculation. The *Processor* object performs the computing functionality (e.g., MCU) of a sensor

node. It controls the sensor object, performs the signal processing and executes the communication protocol. The energy consumption of the MCU depends greatly on the required level of performance and the specific sensor network application. In our application, we assume that we have constant energy dissipation per bit processed (Ec), and so *E*comp=*E*c*b, where b is the number of bits in the processed stream. The *Battery* object represents the battery in a sensor node. It is modeled as a reservoir initially with a fixed amount of Joules. It is reduced every time an activity that consumes energy occurs. Every time the remaining energy is updated, the node lifetime is also updated by adding the time since last update.

- The *Sink Node* module represents the base station. The sink node has all the components of a regular sensor node except the sensing unit and battery. It is in charge of gathering the sensed data from sensor nodes. It can interact with Trace object to keep track of the network lifetime.
- Input parameters for the simulation:
    - NO_NODE: The number of nodes present in the simulation environment. This is the number of nodes placed in random locations at the start of the simulation.
    - XSIZE: The maximum size of X dimensional boundary
    - YSIZE: The maximum size of Y dimensional boundary. The simulation will randomly deploy all the number of nodes within these two dimensional boundary.

- initEnergy: The initial energy level of sensor nodes. At the beginning of the simulation, all the sensor nodes are at the same initial energy level. A node is dead when it is depleted of its power resources
- perBitSenE: Energy consumption for modulating or demodulating one bit of the circuits.
- Radius: A node's radius of communication range. This is the maximum distance a node will be able to communicate in any particular direction. This is also the longest one-hop distance in the route.

5.4 Energy Consumption Calculation

The respective ways of calculating energy dissipation for each of the selected simulation protocols are discussed in this section. As shown in Figure 5.1, there are three units contributing to the energy consumption: the *communication unit*, *Sensing* unit, and *Computing* unit.

5.4.1 Theoretical background

- *Communication unit*: The communication unit is responsible for wirelessly communication among nodes. A typical radio module for communication used in wireless devices is shown in Figure 5.3 (redrawn from [11]). The *Transmit Electronics* represents electronics circuit performing signal modulation. *Tx Amplifier* is used to amplify the modulated signal and output it to the antenna. The *Receive Electronics* is used to decode the modulated signal. E_{elec} is the energy needed for modulating or demodulating one bit of the circuits. ϵ_{amp} is the energy for the amplifier circuit to transmit one bit to an area of radius d = 1 meter (i.e., πd^2). In a real device, the transmit module (*Transmit Electronics* and *Tx Amplifier*) normally stays in sleep mode. It only wakes up when there is any bit that needs to be sent. The receiver module (*Receive Electronics*) performs the reverse function. It needs to be ON when waiting to receive messages.

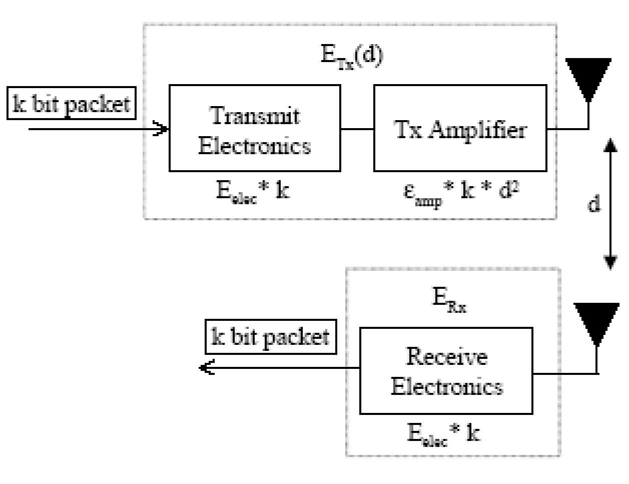


Figure 5.3: Radio model for a wireless device

The formulas for sending a k-bit message to a distance *d* are shown in Figures 5.4 and Figure 5.5 [11].

$$E_{Tx}(k, d) = E_{Tx\text{-}elec}(k) + E_{Tx\text{-}amp}(k, d)$$

$$E_{Tx}(k, d) = E_{elec} * k + \varepsilon_{amp} * k * d^2$$

Figure 5.4: Energy consumption formula for sending a k-bit message to a distance *d*

$$E_{Rx}(k) = E_{Rx\text{-}elec}(k)$$

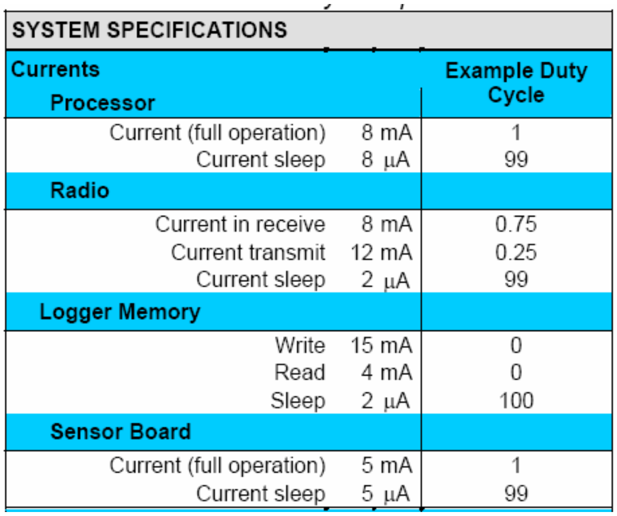
$$E_{Rx}(k) = E_{elec} * k$$

Figure 5.5: Energy consumption formula for receiving a k-bit message

$E_{Tx}(k, d)$ represents the energy needed to spread k bits to an area of radius d, while $E_{Rx}(k)$ is the energy needed to de-modulate k bits. The second formula in the two figures is a re-writing of the respective first formula per Figure 5.3.

- *Sensor board, MCU (CPU board, Memory board), and Radio board of a sensor network*:

  These boards work in two modes: full action and sleep. In the sleep mode, the energy dissipation is almost zero. The full action mode consumes energy as shown in Fig 5.6 [16]. In which, mA mean milli-ampere, µA is micro-ampere.



SYSTEM SPECIFICATIONS		
Currents		**Example Duty Cycle**
Processor		
Current (full operation)	8 mA	1
Current sleep	8 µA	99
Radio		
Current in receive	8 mA	0.75
Current transmit	12 mA	0.25
Current sleep	2 µA	99
Logger Memory		
Write	15 mA	0
Read	4 mA	0
Sleep	2 µA	100
Sensor Board		
Current (full operation)	5 mA	1
Current sleep	5 µA	99

Figure 5.6: Current of boards in sensor node MICA2DOT (MPR 500)

From Figure 5.6, we deduce that the current of the MCU board in full operation is equal to that of the radio board in the receiving mode. And, the current of the sensor board in full operation is around 2/3 of the current of the radio board in receiving mode. The current of the memory board to write data is around 2 times of the current of the MCU board in full operation, and the current of the memory board to read data is ½ of the current of the MCU board in full operation.

5.4.2 Calculation

We use the following assumptions in [11] as the basis when calculating the energy dissipation for our simulations:

- Energy comsumption for modulating or demodulating one bit:

  Eelec = 50nJ/bit

- Energy consumption for spreading one bit to an area of radius r = 1 meter (i.e., πm^2):

  $\epsilon amp = 100pJ/bit/\,m^2 = 0.1nJ/bit/\,m^2$

- Data rate = *2000bits/s*

- Data package size = *2000*-bit

- Signal package size = *64*-bit

From Figure 5.6, we deduce that the current of the MCU board in full operation is equal to that of the radio board in the receiving mode. And, the current of the sensor board in full operation is around 2/3 of the current of the radio board in receiving mode. Derivations of the remaining parameters for calculating energy consumption are shown below:

- For each received data message, the radio board consumes:

  $E_{Rx\_data} = E_{elec} * k\text{-}bit/message$

  $= 50nJ/bit * 2000\ bits/message = 100\ \mu J/message$

- For each received signal message, the radio board consumes:

  $E_{Rx\_signal} = E_{elec} * k\text{-}bit/message$

$= 50nJ/bit * 64\ bits/message\ = 3.2\ \mu J/message$

$\sim= 3\ \mu J/message$

- For transmitting a data message to a distance d, the radio board consumes:

$E_{Tx\_data} = E_{elec} * k\text{-}bit/message + \epsilon_{amp} * k * d^2$

$= 50\ nJ/bit * 2000\ bits/message + 0.1\ nJ/bit*2000\ bits/message * d^2$

$= (100 + 0.2* d^2)\ \mu J\ /message$

- For transmitting a signal message to a distance d, the radio board consumes:

$E_{Tx\_signal} = E_{elec} * k\text{-}bit/message + \epsilon_{amp}*k* d^2$

$= 50\ nJ/bit * 64\ bits/message + 0.1\ nJ/bit*64\ bits/message* d^2$

$= (3.2 + .0064* d^2)\ \mu J\ /message$

We assume that the optimized communication radius of nodes is 50m. Then we have:

$E_{Tx\_data} = 100 + 0.2*50*50 = 600\ \mu J/message$

Meaning: The radio board consumes 600 µJ for transmitting a data message to distance d <=50m.

$E_{Tx\_signal} = 3.2 + 0.0064*50*50 = 19.2\ \mu J/message\ \sim= 19\ \mu J/message$

Meaning: The radio board consumes 19 µJ for transmitting a signal message to distance d <=50m.

$E_{Radio} = Eelec * data\_rate = 50nJ/bit* 2000\ bits/s\ = 100\ \mu J/s$

Meaning: If the radio board is in receiving mode, it consumes 100 µJ at each second.

$E_{Sensor} = E_{Radio} * 2/3\ = 66\ \mu J/s$

Meaning: if the sensor board is in full operation mode, it consumes 66 µJ at each second

$$E_{MCU\_data} = 2000(bit/message)*50(nJ/bit) = 100\ \mu J/message$$

Meaning: The MCU board consumes 100 µJ for processing a data message.

$$\boldsymbol{E}_{MCU\_signal} = 64(\mathrm{bit}/message)*50(\mathrm{nJ}/bit) = 3.2 \sim= 3\ \mu J/message$$

Meaning: The MCU board consumes 3 µJ for creating a signal message.

In SSBBCR protocol, route nodes are required to store some number of keys in the memory. Assume one key takes 64 bits of memory space, so we have:

$$E_{Mem\_Wr} = 2 * 50(nJ/bit) * 64\ (bit) = 6.4\ \mu J$$

Meaning: The Memory board consumes 6.4 µJ for storing one key in the memory.

$$E_{Mem\_Rd} = 1 / 2 * 50(\mathrm{nJ}/bit) * 64\ (\mathrm{bit}) = 1.6\ \mu J$$

Meaning: The Memory board consumes 1.6 µJ for reading one key from the memory.

CHAPTER VI

SIMULATION RESULTS

6.1 Build Scenarios

The simulation environment is built in a square region with the sink node positioned on upper right corner of the region. Nodes' locations are generated by random number generator in the given area and therefore are positioned randomly. We performed our simulation in two different area regions: a large region A with an area 250x250m and a small region B with an area 100x100m. We define the network density as 1 for having an average of one node per 25x25 square meter. For both regions, we increase the density of nodes from 1 to 10. Therefore, in region A, the number of nodes in the network are 100, 200, 300, 400, 500, 600, 700, 800, 900, and 1000, with 2000mJ of initial energy for each node; In region B, the number of nodes are 16, 32, 48, 64, 80, 96, 112, 128, 144, and 160 respectively, each node has same initial energy as above. We set the communication radius of a node to 50m and the key picking area radius to 30m. For SSBBCR and SBCR node, the number of sectors is set to 6 and the nodes per sector are 2. The simulations were run for many times for each of the three protocols. Under each setting, each simulation was run 10 times. An average for these 10 runs is used as the final result.

6.2 Simulation Results

1) Angle of the key picking area λ

Before we conduct simulations on the metrics mentioned before, we need to find the optimal value on the sector angle λ of the key picking area. We define On-route Key Sharing Probability as the probability that any two adjacent nodes on a route have at least one common key. Figure 6.1 is the simulation of On-route Key Sharing Probability based on different λ values. We calculate the probabilities using the number of times the key is shared between two adjacent nodes that are in the route divided by the total number of simulations done. From Figure 6.1, we can see that the optimal value for λ is around 75º, at which the probability reaches highest.

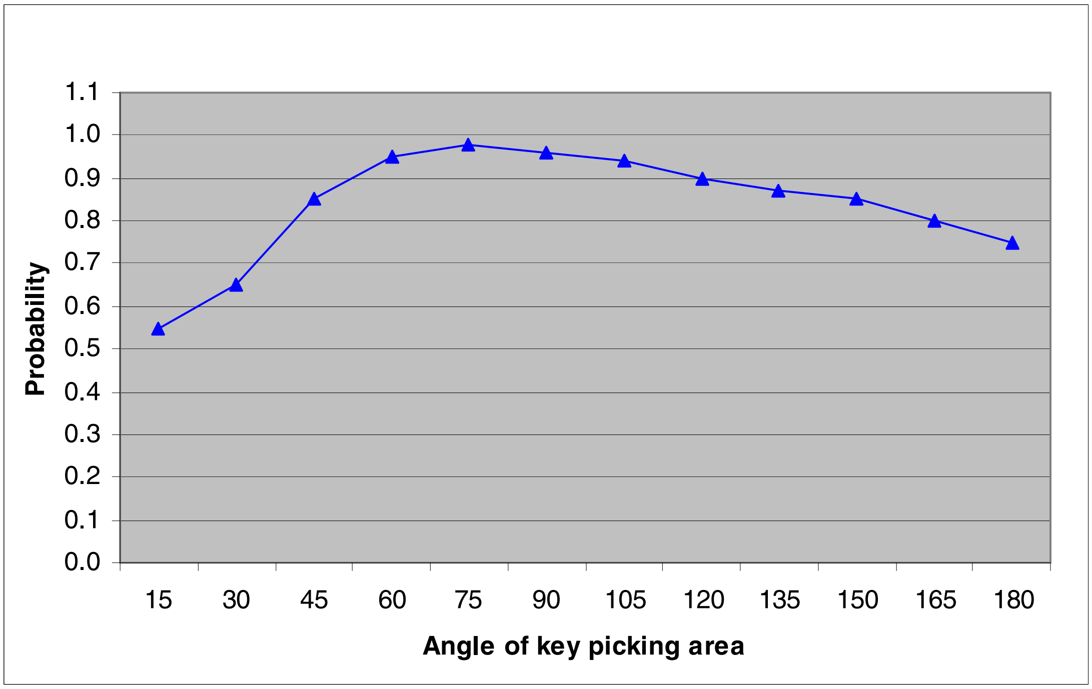


Figure 6.1 On-route key sharing probability vs. Angle of key picking area

2) Network Connectivity

We evaluate the Network Connectivity for SSBBCR only by comparing two probabilities: On-route Key Sharing Probability and Non-route Key Sharing Probability.

Non-route Key Sharing Probability is defined as the probability that any two nodes, one node is on the route and another node is not on the route but within the communication range of the previous node, have at least one common key. We calculate the probabilities using the number of times the key is shared between the two nodes divided by the total number of simulations done. For each simulation, the angle of key picking area is set to 75°. The results are shown on Figure 6.2. As we expected, the Non-route Key Sharing Probabilities are zero under each of the node densities simulated. The reason is obvious; since all the nodes that are not in the route have no keys stored on the memory, therefore there are no any possibilities for them to share keys with nodes on the route. For the on-route connectivity, the figure shows that any two neighboring nodes on the routes have high probabilities sharing common keys regardless the density of the sensor nodes. This result confirms that the key pre-distribution scheme with pre-determined routing knowledge ensures high on-route connectivity and low non-route connectivity.

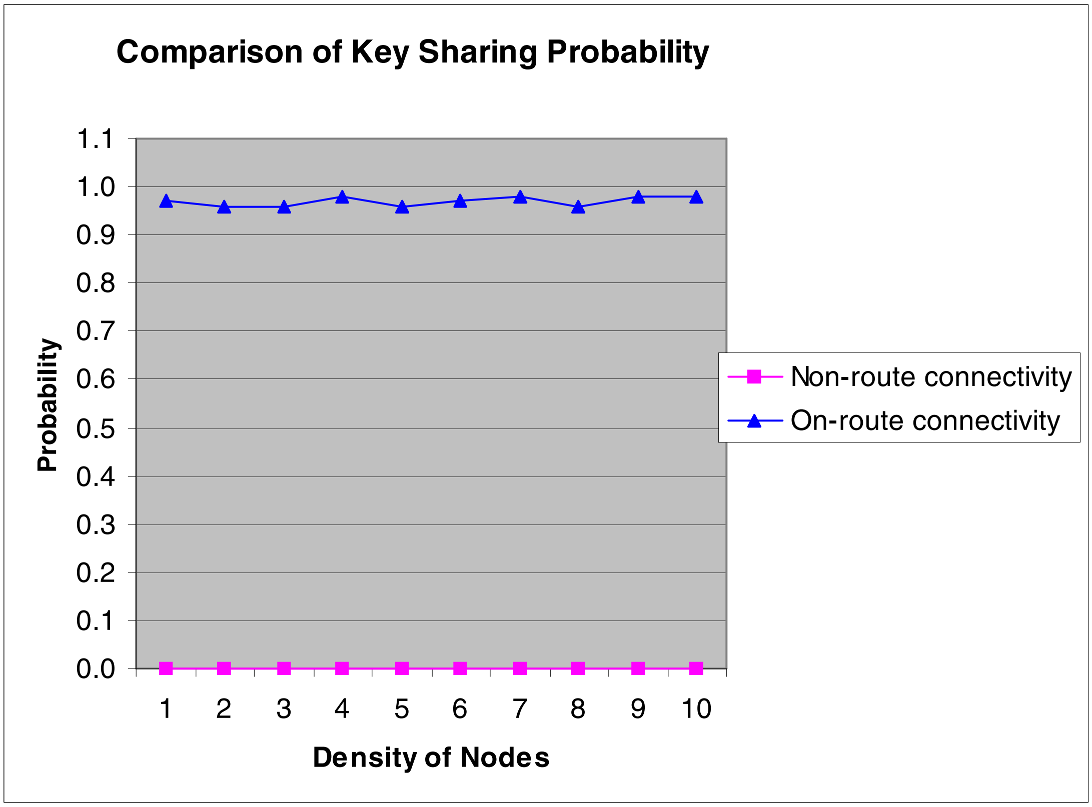


Figure 6.2 On-route and Non-route Connectivity

3) Network Energy Consumption

The second metric, energy consumptions are measured where nodes are deployed in both two areas. Here in region A the simulations are run for time units 300, 600 and 900 (Figure 6.3, 6.4, and 6.5), in region B simulations are run for time units 20, 50, and 100 (Figure 6.6, 6.7, and 6.8).

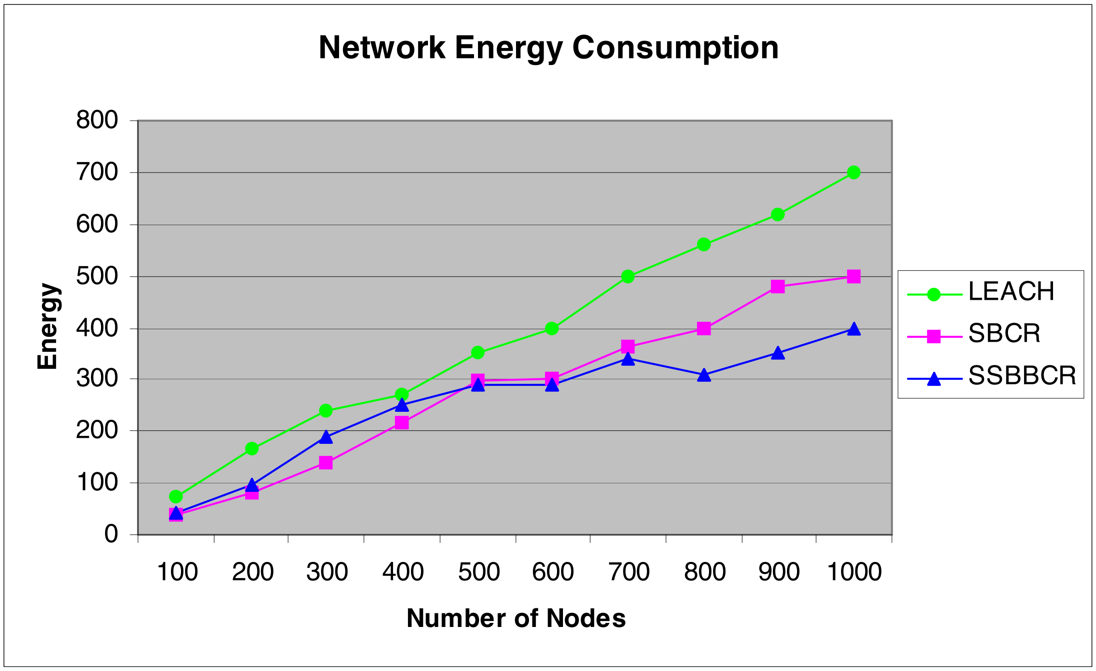


Figure 6.3 Network Energy Consumption with simulation time 300 (Region A)

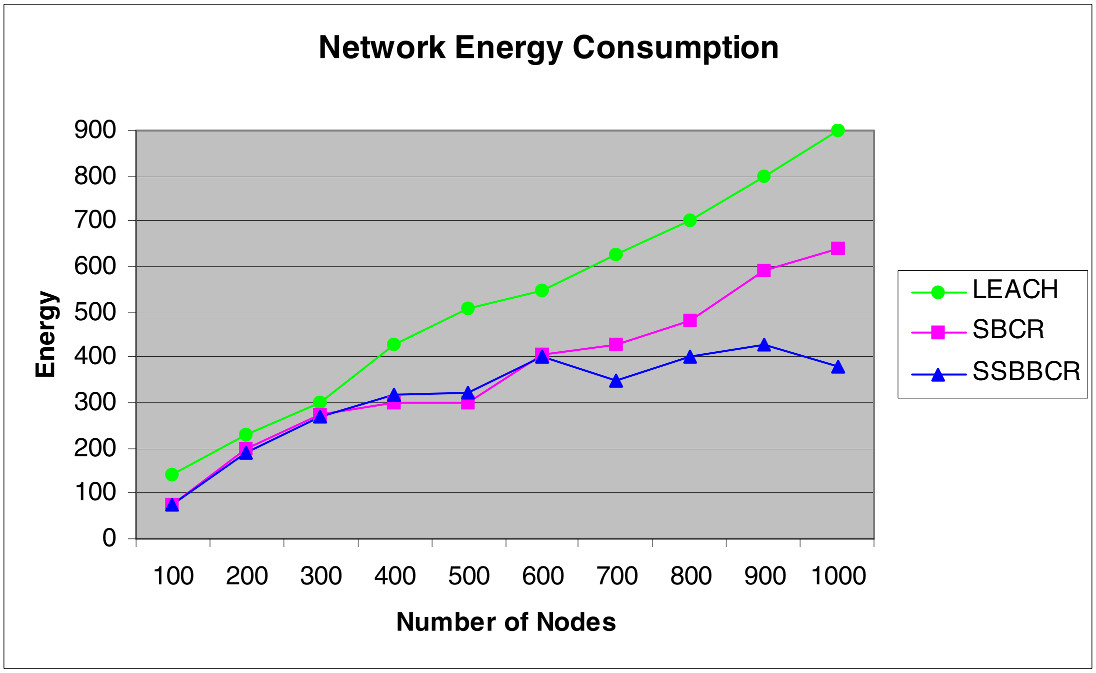


Figure 6.4 Network Energy Consumption with simulation time 600 (Region A)

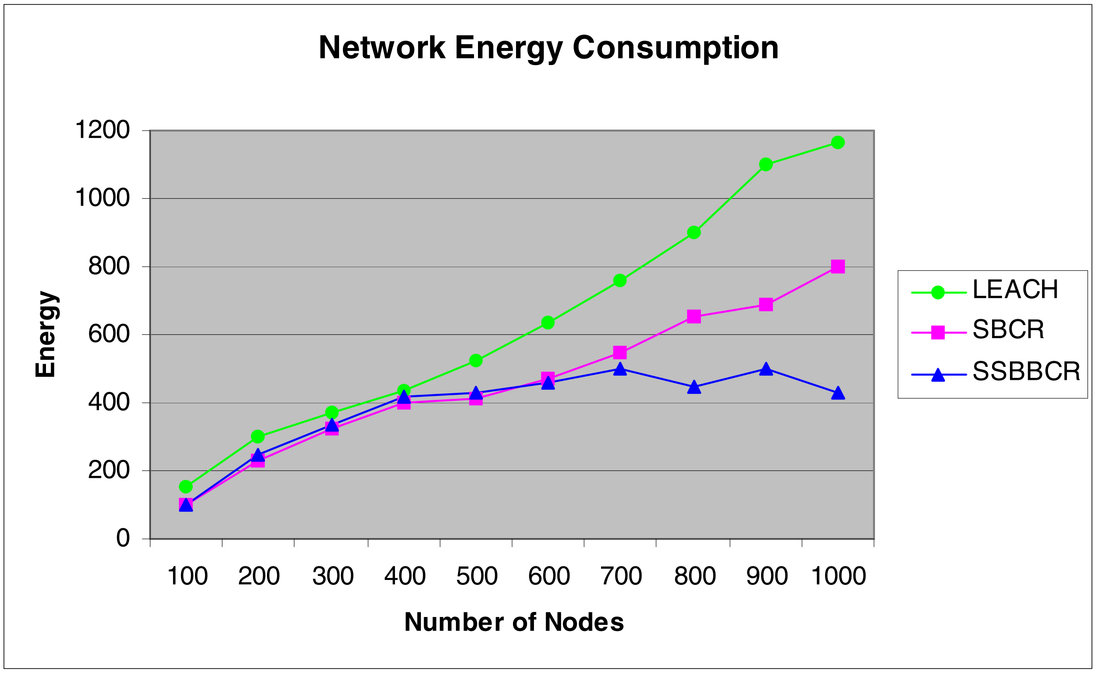


Figure 6.5 Network Energy Consumption with simulation time 900 (Region A)

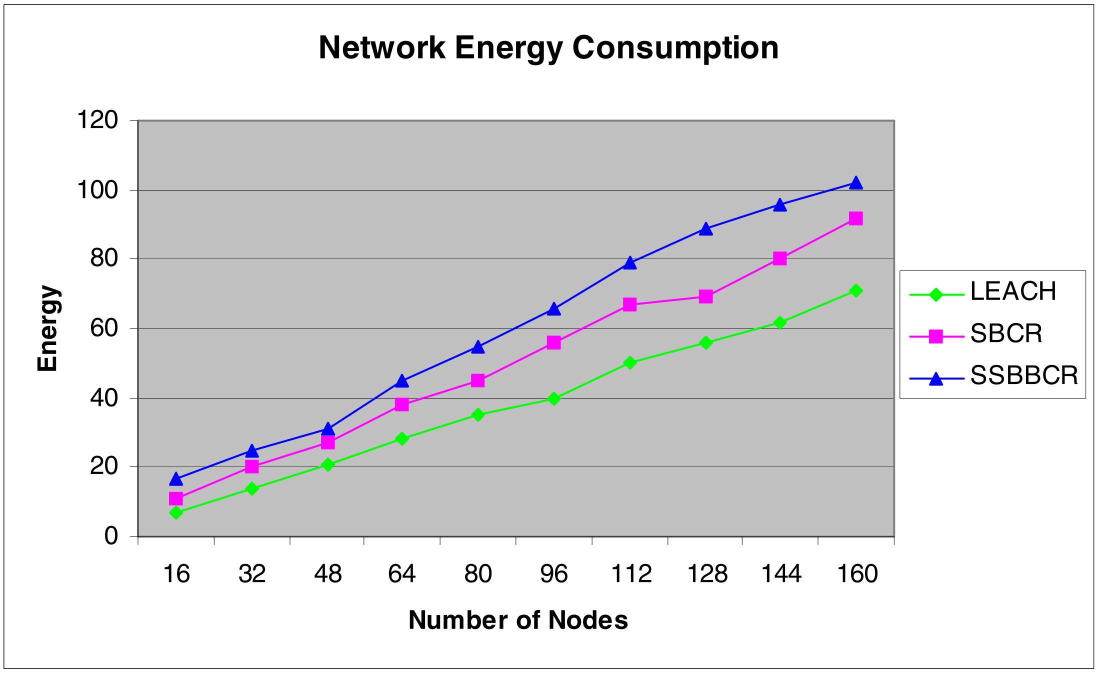


Figure 6.6 Network Energy Consumption with simulation time 20 (Region B)

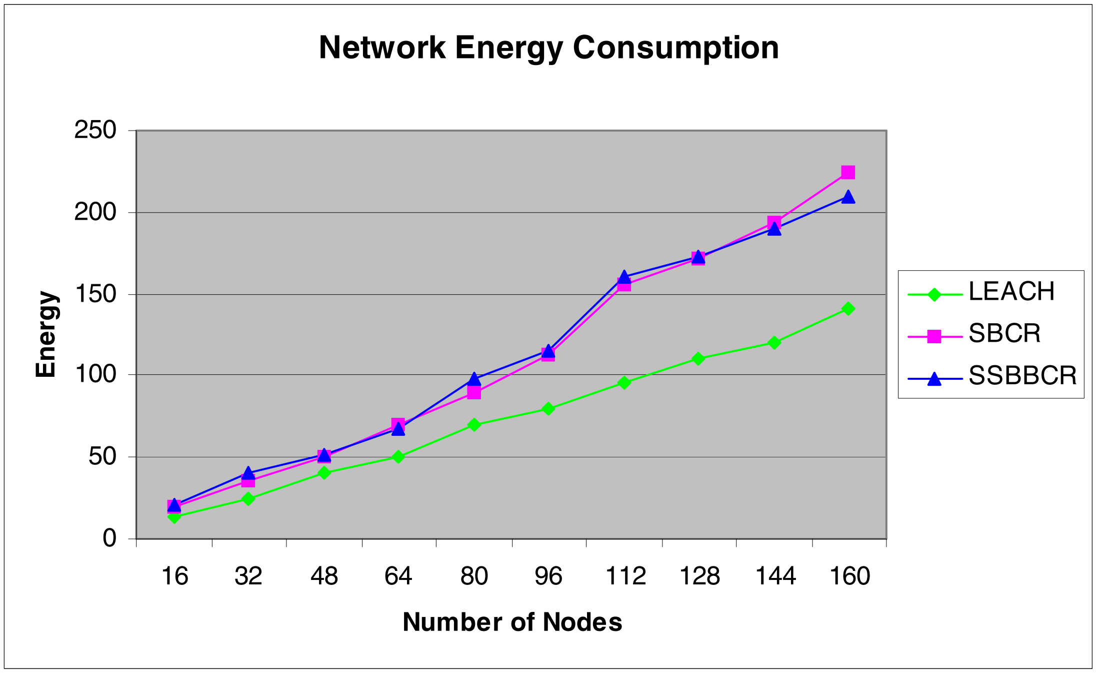


Figure 6.7 Network Energy Consumption with simulation time 50 (Region B)

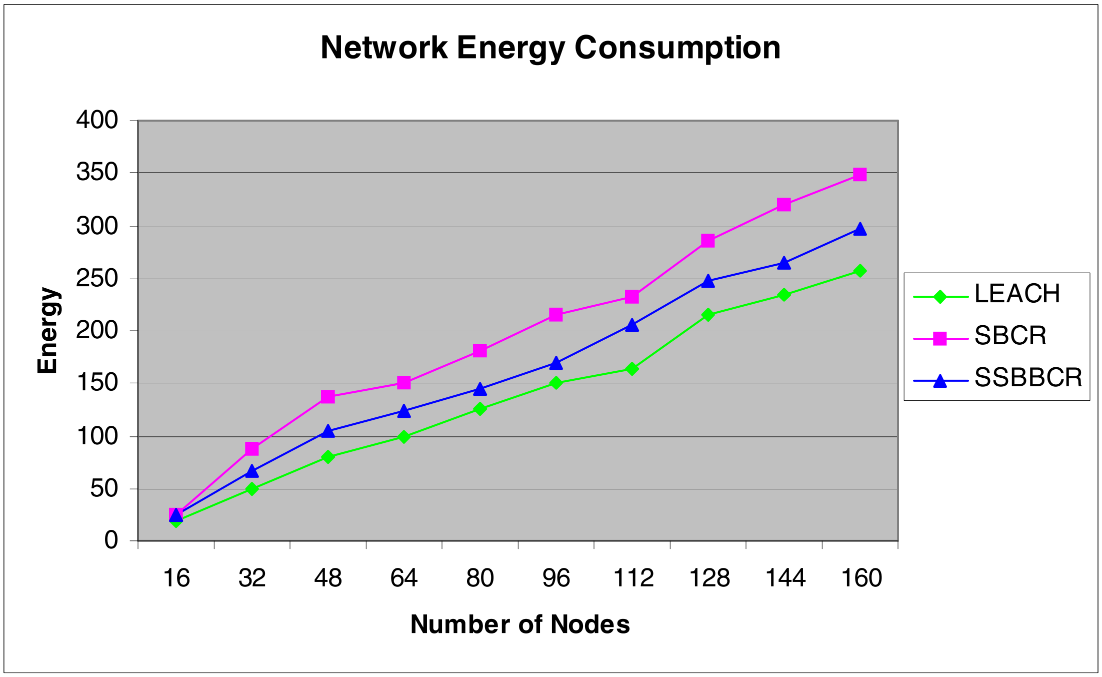


Figure 6.8 Network Energy Consumption with simulation time 100 (Region B)

From Figures 6.3~6.8 we can see that with the network size growth, the total energy consumption increases for all the three protocols in each of the three simulation times. This result is expected, since larger networks result in more message overheads.

In large Region A (Figure 6.3 ~ 6.5), when the node density is less, SSBBCR consumes a bit more energy than SBCR. However, when the number of nodes increases (greater than 600), the energy dissipation in SSBBCR increases slowly, delivering much better results than the other protocols. The reason is that when the number of nodes is small, fewer nodes are available for routing discovery in SBCR and there is less overhead involved, while SSBBCR has to spend extra power for key picking and discovery. As the node density becomes high, the packets overhead in SSBBCR does not increase much while SBCR uses more and more nodes as route nodes in the multiple route-discovery phase which introduces more and more overhead. As we expected, LEACH has the most energy consumption than the other two protocols. This is due to the fact that a lot of head nodes are located far away from the base station resulting in energy being depleted quickly for transmitting data from them to the base station.

In small region B, Figure 6.6~6.8 shows that LEACH is the most energy efficient as compared to SBCR and SSBBCR. The reason is that in small area all the nodes have enough power to communicate with the base. LEACH uses cluster-based hierarchical model to group the sensor nodes. Data are aggregated by the cluster heads and then directly sent to the base, this reduces the overheads involved in the discovery of routing path therefore less energy are consumed.

2) Time before the first dead node

Time before the first dead node, as illustrated in Figure 6.9, shows that SSBBCR lasts longer than SBCR and LEACH. Our explanation is that SSBBCR protocol dissipates energy evenly in sensor nodes. Sector based clustering is one way to balance the energy, another way is to exclude it from being a route node or a cluster head when its energy is below some level and therefore extend the lifetime of the node.

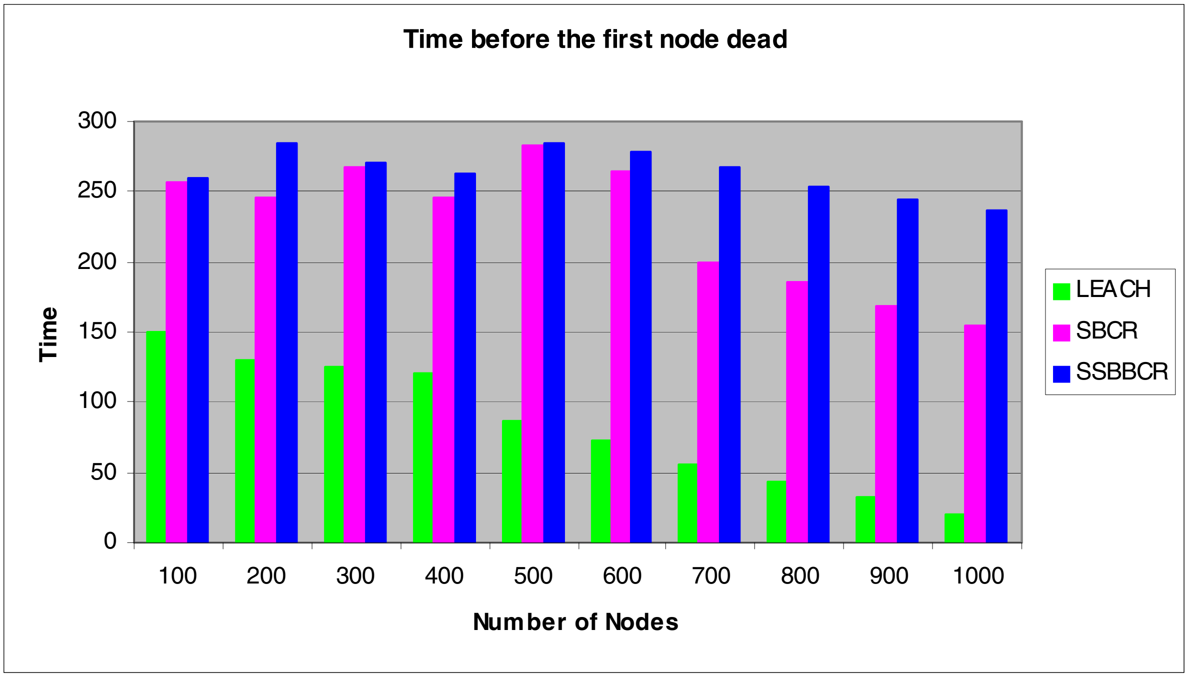


Figure 6.9 Time before the first dead node (Region A)

3) Network lifetime

In this simulation, we measure the network survival time during three phases: The time when 5% of nodes dead, 25% of nodes dead and until the simulation terminates. Figures 6.10~6.13 show that when the network gets larger, the network survival time under 5% of nodes dead, under 25% of nodes dead, as well as the lifetime decrease. This is because larger networks result in more overhead messages and data transmission. As expected, LEACH has the lowest lifetime when the network region is large and the longest lifetime when the region is small. In large region, most of the head nodes are far away from the sink. They exhaust energy quickly and die soon. This in turn lowers the lifetime of the overall network. In small region, most of the head nodes can communicate with the sink in one hop and the rotation of the cluster heads make these nodes energy dissipation minimized therefore increase the lifetime of the nodes and the overall network. SSBBCR outperforms SBCR and LEACH when nodes are deployed in large region. In the small region, though SSBBCR does not last as long as LEACH, the performance degradation is not much.

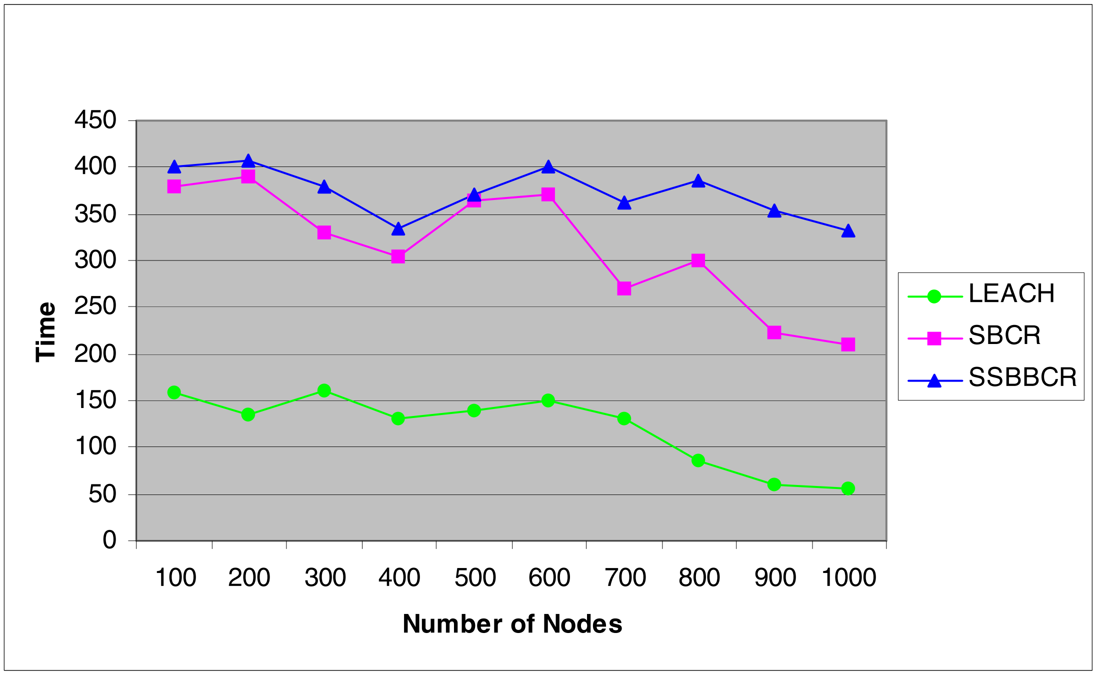


Figure 6.10 Network survival time when 5% of node dead (Region A)

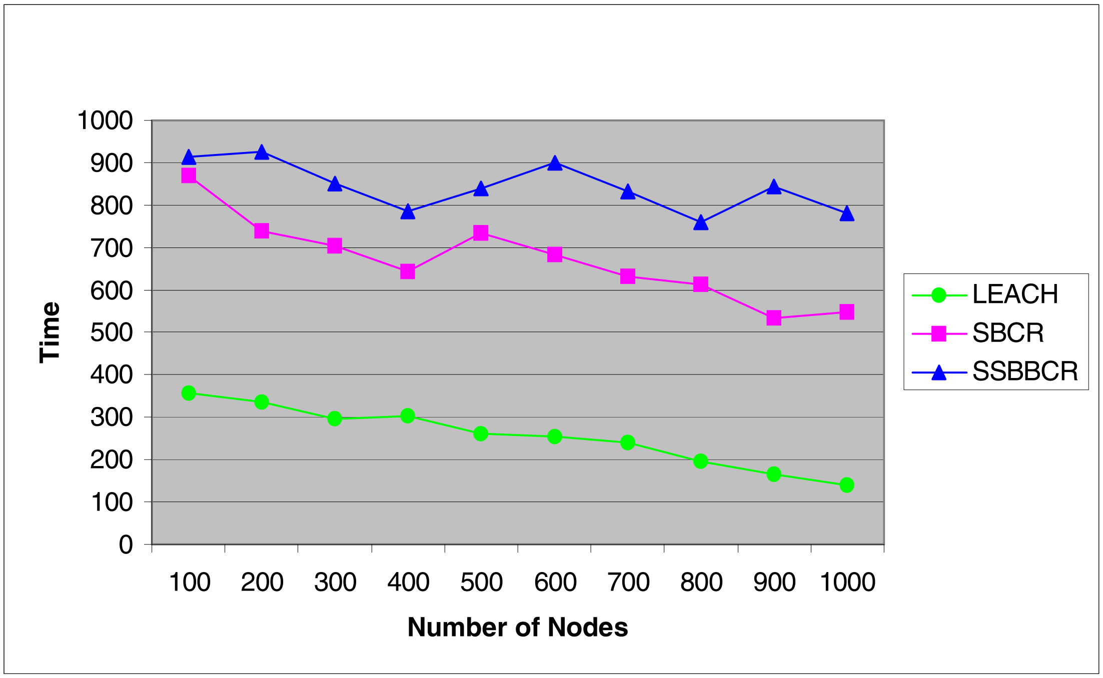


Figure 6.11 Network survival time when 25% of nodes dead (Region A)

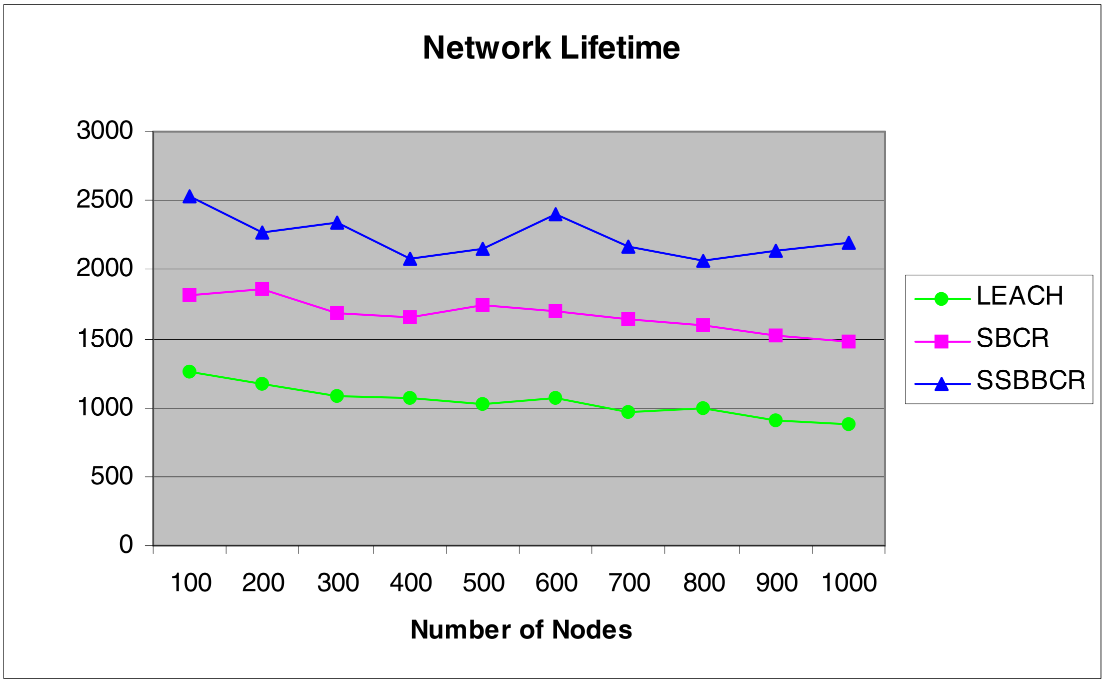


Figure 6.12 Network Lifetimes (Region A)

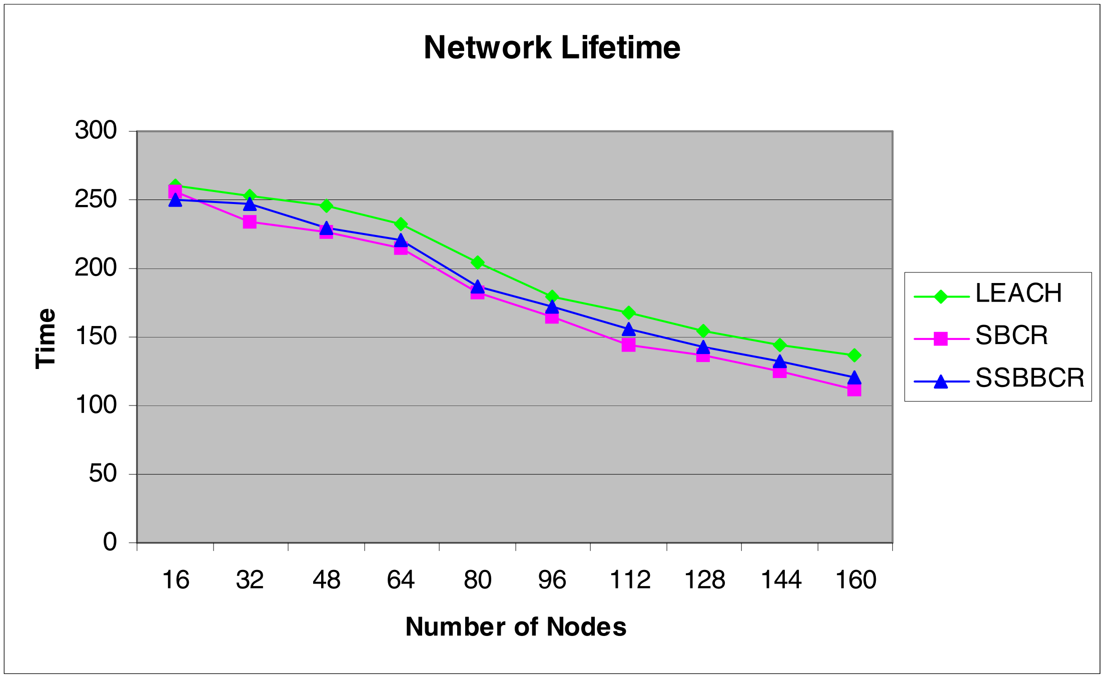


Figure 6.13 Network Lifetimes (Region B)

CHAPTER VII

CONCLUSION AND FUTURE WORK

We have proposed a protocol, Secured Sector Based Bi-path Clustering and Routing (SBBCR), for security and energy efficient in wireless sensor networks, and have evaluated its performance in various simulation scenarios against two other protocols (SBCR and LEACH). Based on the simulation results, we confirm that our proposed protocol controls the energy consumption, provides higher-level security for the network and effectively extends the network lifetime without performance degradation. These features are clearly presented when the network is deployed in large area and the network density is high (that is when the network size is moderate or high scale). Our proposed protocol provides the sensor network an easy and practical way to implement the security scheme by allowing only neighboring nodes in a main route to share common keys. The main strengths of SSBBCR lies in its energy efficiency and ability to keep the network secure, which mean that, SSBBCR is able to extend the lifetime of the network.

For our future work, we will incorporate a hierarchical clustering model into our sector-based approach for clustering and routing. Moreover, some quantitative parameters that can be used to evaluate the performance of the routing protocol such as latency and effective data dissemination will be investigated.

REFERENCES

[1] I. F. Akyildiz, W.Su, Y. Sankarasubramaniam, and E.Cyirci, Wireless sensor networks: a survey, Computer Networks, pp. 393–422, March 2002.

[2] Bandyopadhyay Seema and Coyle Edward J. An Energy Efficient Hierarchical Clustering Algorithm for Wireless Sensor Networks, Proceedings IEEE INFOCOM, pp. 186-191, 2003.

[3] Wireless Sensor Network, http://en.wikipedia.org/wiki/Sensor_network, [last accessed Sep. 2006]

[4] Sudheer. K. Chimbli, Johnson P. Thomas, Venkatesh Sarangan, and Ruiyi Zhang, Sector-based Routing for Secured Energy Efficient Communications. IEEE Wireless Communications and Networking Conference 2006, Las Vegas, April 2006.

[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, pp. 41-47, November 2002.

[6] W. Du, J. Deng, Y. S. Han, S. Chen, and P. K. Varshney. A key management scheme for wireless sensor networks using deployment knowledge. In Proceedings of the IEEE INFOCOM 2004, pp. 586-597, March 2004.

[7] Takashi Ito, Hidenori Ohta, Nori Matsuda, and Takeshi Yoneda, A Key Pre-Distribution Scheme for Secure Sensor Networks Using Probability Density Function of

Node Deployment, In Proceedings of 2003 ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN '05)

[8] Akkaya Kemal and Younis Mohammed, A Survey on Routing Protocol for Wireless Sensor Networks, Elsevier Ad Hoc Network Journal, Vol. 3/3 pp. 325-349, 2005.

[9] W. Heinzelman, J. Kulik, and H. Balakrishnan, Adaptive protocols for information dissemination in wireless sensor networks, in the Proceedings of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'99), Seattle, WA, pp. 148-157, August 1999.

[10] C. Intanagonwiwat, R. Govindan and D. Estrin, Directed diffusion: A scalable and robust communication paradigm for sensor networks, in the Proceedings of the 6th Annual CM/IEEE International Conference on Mobile Computing and Networking, Boston, MA, pp. 235-249, August 2000.

[11] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, Energy-efficient communication protocol for wireless sensor networks, in the Proceeding of the Hawaii International Conference SystemSciences, Hawaii, pp. 126-134, January 2000.

[12] S. Lindsey, C. S. Raghavendra and K. Sivalingam, Data Gathering in Sensor Networks using the Energy*Delay Metric, in the Proceedings of the IPDPS Workshop on Issues in Wireless Networks and Mobile Computing, San Francisco, CA, pp. 243-257, April 2001.

[13] S. Lindsey and C. S. Raghavendra, PEGASIS: Power Efficient Gathering in Sensor Information Systems, in the Proceedings of the IEEE Aerospace Conference, Big Sky, Montana, pp. 312-319, March 2002.

[14] A. Manjeshwar and D. P. Agrawal, TEEN: A Protocol for Enhanced Efficiency in Wireless Sensor Networks, in the Proceedings of the 1st International Workshop on Parallel and Distributed Computing Issues in Wireless Networks and Mobile Computing, San Francisco, CA, pp. 864-870, April 2001.

[15] V. Rodoplu and T.H. Ming, Minimum energy mobile wireless networks, IEEE Journal of Selected Areas in Communication*s*, Vol. 17, No. 8, pp. 1333-1344, 1999.

[16] Rev B (2005). MPR/ MIB User's Manual. DOCUMENT 7430-0021-06. http://www.xbow.com/Support/Support_pdf_files/MPR-MIB_Series_Users_Manual.pdf, [last accessed Nov. 2006]

[17] E. Ould-Ahmed-Vall, G. F. Riley, B. S. Heck, and D. Reddy, Simulation of large-scale sensor networks using gtsnets, in International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS). Atlanta, GA, USA: IEEE, pp. 211-218, Sep. 2005.

VITA

Yihong Zang

Candidate for the Degree of

Master of Science

Thesis: A SECURED SECTOR BASED BI-PATH CLUSTERING AND ROUTING PROTOCOL FOR WIRELESS SENSOR NETWORK

Major Field: Computer Science

Biographical:

Personal Data: Born in Dafeng, Jiangsu, CHINA, on November 18, 1968, the daughter of Mr. Qi Zang and Ms. Jifeng Wang.

Education: Received the degree of Bachelor of Engineering in Civil Engineering from Hohai University, Nanjing, China, in July 1988; Received Master of Engineering degree in Civil Engineering from Southeast University, Nanjing, China, in January 1992; completed the requirements for the Master of Science degree at the Computer Science Department at Oklahoma State University, Tulsa, Oklahoma, in May 2007.

Experience: Employed as Structural Engineer in Nanjing Cement Industry Design & Research Institute, Nanjing, China from 1992 to 2000; employed as Research/Teaching Assistant in the Department of Computer Science, Oklahoma State University from 2003 to 2006.

Name: Yihong Zang Date of Degree: May, 2007

Institution: Oklahoma State University Location: Tulsa, Oklahoma

Title of Study: A SECURED SECTOR BASED BI-PATH CLUSTERING AND ROUTING PROTOCOL FOR WIRELESS SENSOR NETWORK

Pages in Study: 56 Candidate for the Degree of Master of Science

Major Field: Computer Science

Scope and Method of Study: Security and efficient routing are two important parameters in sensor networks. There are few protocols that consider both routing and security. Existing methods suffer from computational overheads. In this thesis we propose SSBBCR (Secured Sector Based Bi-path Clustering and Routing) protocol that takes both these two issues into consideration. It uses bi-path routing algorithm to decrease the computational overhead as well as energy dissipation in the nodes. A key pre-distribution scheme is proposed such that nodes in the route share keys while those that are not in the route do not share keys with nodes that are in the route.

Findings and Conclusion: Simulations have been done for small and large area, low and high density, on-route and non-route connectivity scenarios. Simulation results show that the proposed protocol maximizes the sensor network lifetime and achieves a high probability of connectivity. The proposed approach achieves the best performance when the network is large and the sensor nodes are deployed over a large area.

Advisor's Approval: ____________Dr. Johnson Thomas__________________________