

ENERGY-RATE BASED MAC PROTOCOL FOR  
WIRELESS SENSOR NETWORKS  
AND KEY PRE-DISTRIBUTION SCHEMES

A Thesis  
Submitted to the Graduate Faculty of the  
Louisiana State University and  
Agricultural and Mechanical College  
in partial fulfillment of the  
requirements for the degree of  
Master of Science in Systems Science  
  
in  
  
The Department of Computer Science

By  
Ramaraju Kalidindi  
B.Tech., Andhra University, India, May 2000  
August 2005

# Table of Contents

List of Figures . . . . .	iv
Abstract . . . . .	v
<b>1 Introduction . . . . .</b>	<b>1</b>
<b>2 Related Work . . . . .</b>	<b>4</b>
<b>3 Proposed MAC Protocol: Basic Scheme . . . . .</b>	<b>6</b>
3.1 Sensor Node Criticality . . . . .	6
3.2 Leader Election . . . . .	7
3.3 ER-MAC Protocol . . . . .	8
3.3.1 Protocol Packets and Data Structures at Each Node . . . . .	9
3.3.2 Protocol Description . . . . .	9
<b>4 Simulation Setup . . . . .</b>	<b>14</b>
4.1 Effect of Traffic Density . . . . .	15
4.1.1 Energy Savings . . . . .	16
<b>5 Security in Sensor Networks . . . . .</b>	<b>17</b>
<b>6 Proposed Key Pre-Distribution Scheme . . . . .</b>	<b>23</b>
6.1 Grid Key Vector Assignment . . . . .	24
6.1.1 Sub-Grid Key Vector Assignment . . . . .	24
6.1.2 Key Discovery Phase . . . . .	26
6.1.3 Path-Key Establishment Phase . . . . .	27
6.2 Security Analysis . . . . .	27
6.2.1 Link Capture . . . . .	28
<b>7 Simulation . . . . .</b>	<b>31</b>
7.1 Experimental Setup . . . . .	31

7.1.1	Impact of <i>Key Vector Size</i> on Connectivity and Path lengths . . . . .	32
7.1.2	Comparison to the Random Scheme . . . . .	33
7.2	Multiple Layer Pre-Deployment Strategies . . . . .	33
7.2.1	Key-Node Mapping Algorithm . . . . .	34
7.3	Connectivity . . . . .	35
7.4	Security Analysis . . . . .	37
7.5	Conclusions . . . . .	38
<b>8</b>	<b>Bibliography . . . . .</b>	<b>39</b>
<b>9</b>	<b>Vita . . . . .</b>	<b>41</b>

# List of Figures

1.	Difference in Average Energy of a Node Under ER-MAC versus Basic TDMA . . .	12
2.	Difference in Ranges of Energy of the Network Under ER-MAC versus Basic TDMA	12
3.	Average Number of Slots a Node is Awake and is Asleep . . . . .	13
4.	Difference in Energy of Minimum Energy Node under ER-MAC versus Basic TDMA	13
5.	Simple Grid Scheme Key Vector Assignment . . . . .	20
6.	Sub Grid Scheme Key Vector Assignment . . . . .	21
7.	Impact of Key Vector Size and Node Density on Connectivity and Path Length	30
8.	Performance Comparison between Random Scheme and Our Proposed Sub-Grid Scheme . . . . .	32
9.	Probability that two Nodes Share a Key with increasing value of $(k,M)$ where $k=M$	36
10.	Fraction of Total Nodes that are Connected with increasing value of $(k,M)$ , $k=M$	36
11.	Variance in number of Keys Disclosed per Node. . . . .	37
12.	Probability of Connectivity / Variance of Keys Disclosed . . . . .	37

## Abstract

Sensor networks are typically unattended because of their deployment in hazardous, hostile or remote environments. This makes the problem of conserving energy at individual sensor nodes challenging. S-MAC and PAMAS are two MAC protocols which periodically put nodes (selected at random) to sleep in order to achieve energy savings. Unlike these protocols, we propose an approach in which node duty cycles (i.e sleep and wake schedules) are based on their criticality. A distributed algorithm is used to find sets of winners and losers, who are then assigned appropriate slots in our TDMA based MAC protocol.

We introduce the concept of energy-criticality of a sensor node as a function of energies and traffic rates. Our protocol makes more critical nodes sleep longer, thereby balancing the energy consumption.

Security in sensor networks is more important than traditional networks as they are deployed in hostile environments and are more prone to capture. Trusted third party authentication schemes, public-key systems are not suitable owing to their high resource requirements. Key pre-distribution was introduced in to solve this problem. Our scheme achieves identical connectivity compared to the random key pre distribution using a less number of preloaded keys in each sensor node.

Our proposed key pre-distribution scheme is based on assigning keys to sensors by placing them on a grid. This approach has been further modified to use multiple mappings of keys to nodes. In each mapping every node gets distinct set of keys which it shares with different nodes. The key assignment is done such that, there will be keys in common between nodes in different sub-grids. After randomly being deployed, the nodes discover common keys, authenticate and communicate securely. The analysis and simulation results show that this scheme is able to achieve better security compared to the random schemes.

# Chapter 1

## Introduction

Wireless sensor networks (WSN) have become increasingly popular due to their wide range of applications in both military and civilian environments, ranging from battlefield surveillance to natural habitat monitoring. A typical WSN consists of a large number of autonomous sensor nodes that self-organize to form a multi-hop network [11]. Sensor nodes are battery operated, equipped with integrated sensors, and have embedded processing and short-range radio communication ability. Unlike standard wireless/ad-hoc networks, WSNs are severely resource constrained and energy conservation/efficiency is of paramount importance. The wireless radio-communication interface consumes a significant fraction of node energy. While substantial research has been done on the design of low-power electronics to reduce energy consumption at sensor nodes, due to fundamental hardware limitations further energy efficiency can only be achieved through the design of energy-aware communication protocols.

In this work we focus on the design of energy-efficient link layer protocols for sensor networks. Traditional MAC protocols focus on improving fairness, latency, bandwidth utilization and throughput (which are secondary for WSNs) and lack energy conserving mechanisms. Studies reveal that energy wastage in existing MAC protocols occurs mainly from collision, overhearing, control packet overhead and idle listening [14]. MAC protocols for sensor networks should try to avoid the above energy wastage while allocating shared wire-

less channels among sensor nodes as fairly as possible and prevent nodes from transmitting at the same time.

We present a MAC protocol specifically designed for wireless sensor networks. Our TDMA-based approach achieves significant energy savings by eliminating collisions, reducing idle listening and control packet overhead. Our protocol uses the periodic listen and sleep mechanism introduced in S-MAC [14]. Our work introduces a new notion: *energy-criticality of a node* which is a measure of the lifetime of the node. In our approach the entire network is divided into TDMA groups based on neighborhood information. We define the energy-criticality (henceforth called criticality) of a node as a function of the residual energies and traffic flow rates of its neighbors. We identify two parameters that define the criticality of a sensor node.

- The residual energy level of the sensor node.
- The packet flow rate through the node.

At certain times, a node may be more actively transmitting packets than the rest of the nodes. In such a case this node is assigned more number of slots to transmit its data packets. A node with lower energy level is also critical. Even if this node is not active it is assigned more transmission slots than its neighbors. During these slots, the node will be idle thereby reducing its energy costs due to listening. In our algorithm a set of leaders are elected based on their criticalities. Non-critical nodes are assigned fewer transmission slots. Since they are listening more frequently, future traffic will be predominantly routed through them, thereby balancing energy consumption across the link layer. Our adaptive slot assignment allows the energy management strategy to vary as the traffic and residual energy levels change.

Previous research on sensor networks ([14], [13], [18], [17]) does not consider the fact that critical nodes may deplete their energy faster than the remaining nodes. This may lead to the formation of holes in the network or even disconnect it, thereby reducing network lifetime substantially. Existing work, to the best of our knowledge, treats all nodes equally and tries

to minimize energy consumption at a single given node which will not necessarily extend the lifetime of the entire network. In the dynamic environment of wireless sensor networks, none of the previous schemes are optimal in terms of energy efficiency all the time.

Balancing energy consumption among nodes is the key solution to extending network lifetime. A centralized approach, though optimal will not be feasible in a distributed sensor network. A distributed mechanism which uses partial local information to achieve global benefit is the best approach to overall energy balancing. Under heavy and moderate traffic load existing MAC protocols designed for wireless sensor networks have negative effect on energy savings. This is due to the extra synchronization overhead and periodic exchange of sleeping schedules. An algorithm which balances the energy consumption among all the nodes is suitable for higher traffic loads.

# Chapter 2

## Related Work

Current MAC design for wireless sensor networks can be broadly classified into two categories: contention-based protocols and TDMA protocols. IEEE 802.11 [12], although widely used because of its simplicity and robustness against the hidden terminal problem, is not an energy-efficient protocol since it does not address the issue of avoiding overhearing and idle listening. PAMAS [13] tries to avoid overhearing but does not avoid collisions, which is a significant wastage of energy. Collisions can occur between probe messages or RTS/CTS messages. S-MAC [14], an improvement over PAMAS, reduces further wastage from idle listening by making idle nodes shut off their radios. It does not avoid collisions between two RTS or CTS messages, which is a significant wastage of energy. Also, the duration of sleep is the same for each node, which is unfair for the nodes with less energy. Making weaker nodes sleep more can increase efficiency. S-MAC assigns sleep schedules without taking into account the criticality of a node. This also has the same problem as PAMAS: two nodes simultaneously sending RTS packets can cause collisions. Another protocol proposed by Woo and Culler [16] uses an adaptive rate control mechanism based on carrier sense multiple access (CSMA). This protocol tries to achieve a fair bandwidth allocation to all nodes rather than saving energy at each node in a multi-hop network. Similar to S-MAC, Piconet [18] is another protocol which puts nodes into periodic sleep mode for energy conservation. For

synchronization, Piconet makes a node broadcast its address before it starts listening. The drawback of this scheme is that if a node wants to talk to its neighbor it has to wait until it gets the neighbor's address.

TDMA protocols have the natural advantage of having no collision or control-packet overhead from which the contention-based MAC protocols suffer. However, TDMA protocols are not as scalable as contention-based protocols. An example of TDMA protocol in wireless networks is the one proposed by Sohrabi and Pottie [15], where each node schedules different time slots to communicate with its known neighbors. The protocol uses FDMA or CDMA to avoid interference between adjacent links. The drawback of this protocol is low bandwidth utilization since a node can talk to only one neighbor during a time slot and collisions can occur when two neighboring nodes transmitting in the same slot are assigned the same frequency or code.

# Chapter 3

## Proposed MAC Protocol: Basic Scheme

In this work, we propose a TDMA based energy-efficient MAC protocol with good performance characteristics. Unlike several existing protocols, which treat all nodes equally with respect to energy conservation, our protocol is based on the crucial observation that over a period of time, there are several critical sensor nodes in the network, which must be treated differently (preferentially, in most cases) with respect to energy consumption. The criticality of a sensor node could be based purely on local state information, such as relative energy levels and traffic flows within the neighborhood group of sensors. Alternately, criticality is a function of a sensors location within dynamically changing query routing trees. The proposed MAC layer protocol is an improvement over [17] which uses a TDMA protocol with sensors sleeping when they have nothing to transmit. Our protocol initially assigns the same number of transmission slots to each node in a TDMA frame.

### 3.1 Sensor Node Criticality

Certain nodes may have more active participation in the sensing events or may be part of many routes in the event propagation trees. Hence, some nodes will deplete their energies faster than other nodes. Let  $E_i$  be the residual energy level of a sensor node. We label the flow rate of node as  $F_i$  which is obtained by counting the number of packets originating at

the sensor node and routed through the node periodically. We define the criticality  $C_i$  of a node to be

$$C_i = 1 - \frac{E_i}{\max\{E_j\}} + 1 - \frac{F_i}{\max\{F_j\}} \quad (3.1)$$

for all sensor nodes  $j$  in the TDMA-group(s) containing  $i$ . We assume that sensor nodes in a TDMA-group exchange their energy levels and flow rates periodically or whenever a new leader election phase is triggered.

## 3.2 Leader Election

Sensor nodes conduct a local election based on the criticalities of neighboring nodes, which are part of a TDMA group. The local election process is fully integrated with (i.e, part of) the regular TDMA communication schedule. Thus there is no extra throughput loss due to a separate local election phase. A sensor node  $i$  can independently decide to initiate an election if its current energy level  $E_i$  falls below a threshold value  $t_r E_w$  of the previous winner's then-energy level  $E_w$ . Once an election is initiated, each node transmits special 'energy-level' messages, which are appended, to its regularly scheduled transmission packet during its scheduled time slot. A property of our protocol is that all nodes listen to all transmitted packets i.e., there are no sleeping nodes when other nodes are transmitting. The motivation behind this constraint is to enable the integration of leader-election with regular TDMA communication and thereby save bandwidth/overhead. Since we enforce reception/listening by all nodes of all transmitted packets, there is no ambiguity about when an election is initiated. This approach is different from several standard MAC algorithms where a sensor nodes duty cycle consists of sleep and active periods and nodes can be sleeping while other nodes are active. Finally, the node with least energy in the group declares itself as the leader at the end of the election process. Also note that the entire election phase takes one (asynchronous) TDMA frame starting from the slot when the election is initiated. Once a leader is (or k-leaders are) elected at the end of this process, all the losers reduce their

number of slots by a constant factor (we choose two as the constant in our simulations) and the winners have slots twice that of the losers. The advantage behind this reallocation of slots is to reduce the idle listening time of critical nodes (those with lower energy) nodes. Thus nodes can power off/sleep when they they have nothing to transmit during their own slots. Since leaders have more allocated slots, their energy loss due to idle listening is less. Finally, note that the current leader also transmits its energy level once an election is initiated even though it may be a sleeping slot. This is to avoid election of an incorrect leader, which will lead to another unnecessary round of leader election.

### **3.3 ER-MAC Protocol**

ER-MAC, the distributed energy aware MAC protocol is based on TDMA and hence possesses the natural ability of avoiding extra energy wastage. The main advantages of a TDMA-protocol present in ER-MAC are the following.

- Packet loss due to collisions is absent because two nodes do not transmit in the same slot. Although packet loss may occur due to other reasons like interference, loss of signal strength etc.
- No contention mechanism is required for a node to start sensing its packets since the slots are pre-assigned to each node. No extra control overhead packets for contention are required.

ER-MAC uses the concept of periodic listen and sleep. A sensor node switches off its radio and goes into a sleep mode only when it is in its own time slot and does not have anything to transmit. It has to keep the radio awake in the slots assigned to its neighbors in order to receive packets from them even if the node with current slot has nothing to transmit. We describe the protocol in details in the next two subsections.

### 3.3.1 Protocol Packets and Data Structures at Each Node

The protocol has two types of packets, data packets and control packets.

- Data packets: These are normal data packets received from higher layer protocols, which are routed to the base station.
- Control packets: The normal packet contains two fields. The first field specifies the type of the packet and the second field specifies the value attributed to the type of the packet. There are two types of control packets.
  - a) Vote packet: This contains the decision of a node, which can be either positive vote or a negative vote. This packet is sent to nodes, which sent their energy values to this node.
  - b) Radio-power-mode packet: This packet contains the radio-power-mode of the sender, to indicate whether the sender is using one slot or two slots for transmitting its data packets.

Initially each sensor node is assigned two TDMA slots on which it can transmit packets. It also has a receiver table, a two-tuple  $\langle \text{source}, \text{slot} \rangle$ , which tells the sensor when to turn on its receiver to listen for a packet coming from its neighbors. It also has extra state variable Radio-power-mode, which tells the MAC to use two slots for transmission if it is set. It also maintains a local state variable Radio-mode[ $i$ ] for each of its neighbor indicating the Radio-power-mode of the neighbor  $i$ . This information about the neighbor is used to set its receiver to listen for packets from its neighbors.

### 3.3.2 Protocol Description

Initially each node is assigned two TDMA slots for transmission. The way these slots are assigned is not in the scope of our work. Each node knows which slots its neighbors will use to transmit packets. The main idea is to let the nodes exchange information about their

energy levels. Based on that energy level information, each node decides to use one or two of the slots for transmission. Initially the Radio-power-mode of all nodes is set to TRUE to allow nodes to transmit in two slots. Each sensor node can be in any of the two phases.

- Normal operation phase: The nodes operate normally, routing data packets to the base-station.
- Voting phase: Critical nodes enter the voting phase to do a local election to readjust their slots.

A node in the voting phase is integrated with the normal TDMA phase. So, control packets are sent along with normal data packets in the voting phase. The local voting phase is triggered by criticality of a node. A node is said to be critical if it falls below the previous election winner's energy value. When a node enters this critical phase a local voting phase is triggered. A node in the voting phase is a winner if all its neighboring nodes criticality values are greater than its own criticality value. Otherwise it is declared as a loser.

The sequence of steps followed by the sensor node  $i$  which triggers the voting phase is the following.

- The node  $i$  sends its current energy value to all of its neighbors and requests the criticality value of its neighbors.
- Node  $i$  calculates its own criticality value based on neighboring nodes energy and traffic information.
- If  $C_i > C_j$  for all  $j$  where  $j$  is set of neighbors of  $i$  Then it sets Radio-power-mode to TRUE and becomes the winner. Otherwise it sets Radio-power-mode to FALSE and declares itself a loser.
- At the end of the voting phase the node  $i$  sends its current value of Radio-Power-Mode to all its neighbors.

The sequence of steps followed by a receiver node  $j$  in the voting phase is the following.

- The receiver node  $j$  requests its neighbors energy levels and traffic rates and calculates the criticality value  $C_j$
- It then sends the criticality value  $C_j$  to  $i$ .
- If the Radio-power-mode value received from  $i$  is TRUE then it adjusts its TDMA frame to accommodate to slots for  $i$ .

Multiple nodes can initiate the voting phase at the same time. If more than one neighboring nodes initiate the voting phase at the same time, then the node/nodes with the minimum energy level becomes/become the winner. In normal operation mode, the activity of each node in a time slot is the following:

- If it owns the current slot then it sends its data in that slot. If it has nothing to transmit the radio is put to sleep.
- If it does not own the current slot, it checks by looking at its local state information whether any of the neighbors is transmitting in the current slot.
- If the current slot is being used the radio is put to receive mode. If current slot is not being used the radio is put to sleep.

A low energy node sleeps more than higher energy nodes, thus balancing the energy among the nodes and thus increasing the energy savings and thereby increasing the lifetime of the network. The performance of the TDMA protocol and the performance comparison with and without the Radio-power-mode is presented in the next section.

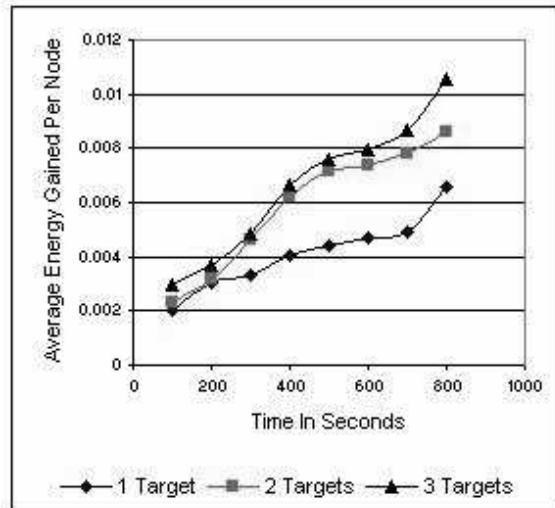


Figure 3.1: Difference in Average Energy of a Node Under ER-MAC versus Basic TDMA

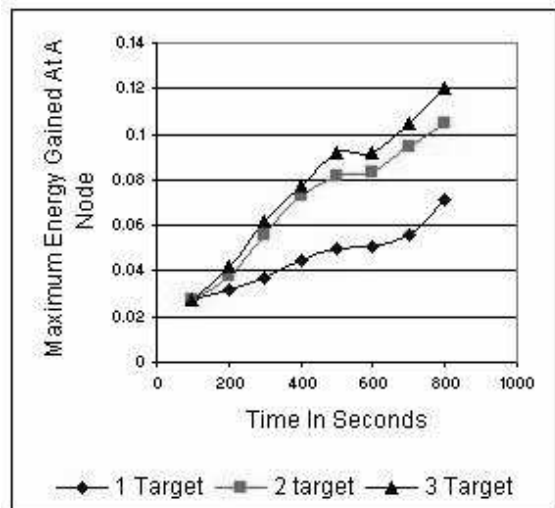


Figure 3.2: Difference in Ranges of Energy of the Network Under ER-MAC versus Basic TDMA

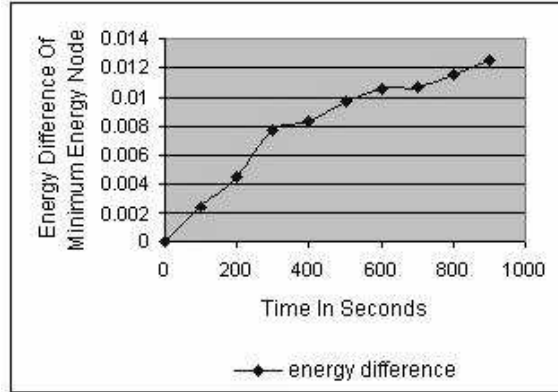


Figure 3.3: Average Number of Slots a Node is Awake and is Asleep

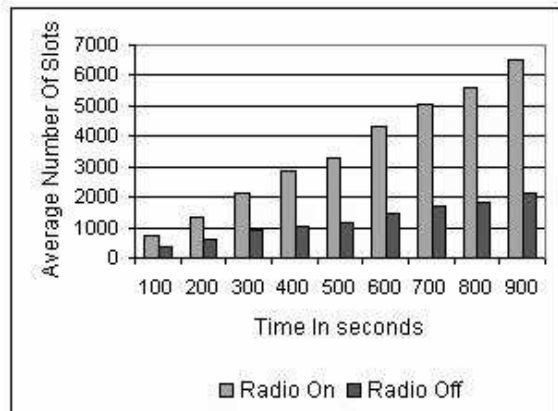


Figure 3.4: Difference in Energy of Minimum Energy Node under ER-MAC versus Basic TDMA

# Chapter 4

## Simulation Setup

ER-MAC was tested using the ns-2 simulator. In our experimental simulation we had 100 nodes distributed in a 1000X1000 meter area grid. We used the battery energy models for CPU, radio and sensor agent from [17]. In our simulation we have target nodes moving in the field, which transmit signals, and sensor nodes sense these signals. These signals are sent to the sensor application which sends the packets to be routed the user node. We run our simulation for a period of 1000 seconds. The targets move in the field between two points at constant speed of 10m/s repeatedly. We test our protocol with varying number targets moving across the grid. We change the packet traffic density by varying the number of targets moving across the region, which trigger more number of sensors to participate in the detection. We also, study the impact of increase in the radio range of networks to the lifetime of the network. We test the above setup with two radio ranges for each of the sensor node. Higher radio range will decrease the overall number of packet transmissions. But the high power consumption for longer transmissions might in effect cause more energy to be wasted. The purpose of our simulation is to test the energy savings at the nodes using our protocol. We compare our protocol with the basic TDMA protocol in [17] which uses a single slot for transmission for each of its node. We choose to use the following metrics for our simulation.

- Average energy remaining at the nodes with time.
- The minimum energy node in the network with time.
- Maximum gain in the energy at a node with time. The gain in the energy is difference in the energy remaining at the nodes under the two schemes. i.e. with and without energy balancing.
- The number of slots the radio is in sleep mode.

## 4.1 Effect of Traffic Density

We test our protocol under varying traffic densities. The number of targets moving in the region is slowly increased so that more sensor nodes will participate in target detection. Each of the targets starts at different points in the grid and move repeatedly across two points at the speed of 10m/s. The performance of our protocol is tested using the metrics mentioned above. We perform the test suite with single target, two targets and three targets moving in the region. Our simulation tests the performance of our MAC protocol under various traffic loads.

Figure 1 shows the performance of our protocol compared to the basic TDMA in [17]. We get a significant improvement in the energy savings, which is due to the balancing of the nodes in the energy consumption. Our protocol gives a higher gain in energy with slightly increasing the traffic load. In light traffic and lightly heavier traffic our protocol gives a significant savings in the energy. As traffic load increases the some of the nodes get depleted faster. Then our protocol saves energy at these nodes by reducing their idle listening time to half. This is the reason why our protocol is more effective in higher traffic loads. Other existing protocols, which do not balance the energy consumption among the sensor nodes, have fewer saving in energy at higher traffic loads. Even in extremely heavy traffic our protocol does some energy savings by dividing the energy load among all the possible nodes.

Our leader election phase and the slot assignment is integrated with normal TDMA packet transmission. Because of this integration our protocol does not require extra synchronization and extra control packets. This makes the protocol more beneficial in higher traffic densities.

#### 4.1.1 Energy Savings

Figure 2 shows the energy difference in the minimum energy nodes under the two methods (with and without energy balancing). The difference between the minimum energy nodes is always increasing. This shows that our approach maximizes the lifetime of the network by both maximizing the lifetime of a single sensor node as well as the entire network. This is due to the inherent energy balancing nature of the approach. Figure 3 shows the average number of slots a sensor node's radio is in power and the number of slots it is switched off. The TDMA frame has some unused slots in which neither any node nor any of its neighbors is transmitting. In this idle slots the sensor node's radio is also put to sleep. Figure 3 does not include these idle slots in calculating the number of slots radio is put to sleep. Figure 3 shows that our protocol puts the nodes to sleep 25 percent of the time; this is without even counting the idle slots. With the inclusion of idle slots the gain in energy is even higher, which comes natural with TDMA protocols.

Figure 4 shows that our protocol has higher minimum energy node in the network than the basic scheme without energy balancing.

# Chapter 5

## Security in Sensor Networks

Distributed Wireless Sensor Networks (DSN) consists of numerous tiny sensors deployed at high density in regions requiring surveillance and monitoring. These sensors are memory as well as energy constrained. A typical sensor node consists of one or more sensing elements (motion, temperature, pressure, etc.), battery, low power radio transmitter/receiver, micro-processor and limited memory. An important aspect of such networks is that the nodes are unattended, have un-replenishable energy and network topology is unknown. Sensor networks deployed in a hostile environment are prone to different types of malicious attacks like eavesdropping, masquerading, traffic-analysis, etc. To provide security communication should be encrypted and authenticated. Traditional secure communication schemes [8] [9] are not suited for sensor networks as they are more demanding in memory and computationally intensive.

The fundamental constraints under which the sensor networks operate prohibit them from using public key cryptosystems, third party authentication systems etc. These constraints are.

- Resource Constraints: A typical tiny sensor node has about 20-30 joules of initial energy. This imposes a strong restriction on the processing capabilities and available memory in a sensor node. For example, a Berkeley mote has a 8-bit 4 MHz processor

which supports a minimal RISC-like instruction set without support for multiplication and other complex operations. Perrig et al [4] showed that a simple RSA operation takes the order of tens of seconds on this processor. Moreover it has less than 4KB of memory after the node is loaded with the necessary operating system and applications. Hence the number of keys and the algorithms that can be stored on a node is very less.

- Short Radio Range: Sensor nodes are equipped with low power radio transmitters which have a very small transmission range of less than 20 meters. Thus a DSN uses multi-hop routing for sensor node to base station communication. Henceforth a base station cannot manage the key distribution for the nodes as it incurs high overhead.
- Hostile Environments: DSN's are deployed in hostile environments which make the nodes prone to capture. Using key distribution servers to establish shared keys is not feasible because a key distribution server if captured can disclose a large number of keys and thus is a single point of failure.

Identifying these limitations our work extends the seminal work done in this area by Eschenauer et al [3]. They introduced a random key pre distribution scheme in which each node is loaded with a set of keys randomly selected from a key pool. After the deployment phase neighboring nodes exchange information to establish common shared keys which are later used for secure communication. The basic idea behind this scheme is to have a large pool of keys, from which a set of keys is randomly chosen and stored in each of the sensor nodes. Any two nodes which are able to find common keys within their key subsets can use those shared keys for secure communication and authentication. The key idea of this method is to relegate the key establishment process from a key distribution server to the individual nodes thereby making it viable in hostile environments.

Key pre-distribution schemes [3] [5] [4] where key information is distributed to all sensor nodes prior to deployment is the most feasible method for secure communication between resource-constrained devices. A naive way of achieving complete connectivity for a network

of  $N$  nodes is to have  $N-1$  keys stored in each sensor node. But this method is infeasible due to memory constraints on sensor nodes. To tackle the memory constraints problem a single key can be used network-wide for encrypting data. Although this approach has the least storage cost it is most vulnerable to attack as compromising a single node will cause the entire network's security to be breached.

Chan et. al. [4] have extended the basic random scheme to enhance the security and resilience of the network using  $q$ -compositeness and multi-path key reinforcement. In the  $q$ -composite scheme instead of nodes sharing single key they are required to share at least  $q$  keys to establish communication. This method claims to achieve higher security under the assumption that network is more prone to small scale attacks and is unlikely to be subject to a large-scale attack. However a higher value of  $q$  makes the network less scalable and connectivity is reduced. In the multi-path key reinforcement scheme security is strengthened between any two nodes by exchanging information between the two nodes using multiple paths. In this method although an increase in the number of disjoint paths increases the security, communication overhead increases substantially.

The drawback of the above proposed schemes is that they are not suitable for large scale sensor networks as they require each node to be loaded with a large number of keys. Perrig, in his work in 2001 [5], showed that it is not feasible to implement public key cryptographic protocols in sensor nodes. These sensor nodes have less than 4KB of free memory after loading the necessary Operating System and other necessary applications. Implementation of key distribution schemes presented in [3] results in a requirement of memory for around 200 keys, thus occupying more than half the available memory. This aspect makes the previous proposed schemes impractical for large networks.

The motivation for our scheme is to reduce the number of keys to be loaded in each node. We propose a novel scheme, in which keys are assigned in a deterministic fashion so that any random deployment yields very high connectivity as well as high security among sensor nodes. The scheme is so modeled so as to maximize the connectivity with a small

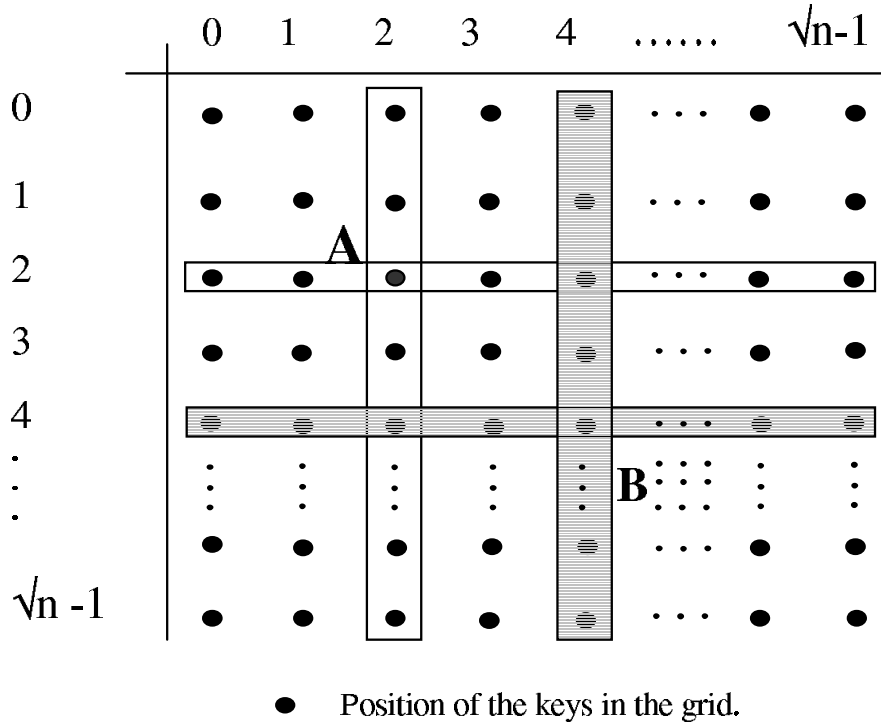


Figure 5.1: Simple Grid Scheme key vector assignment

number of keys loaded in each of the sensor nodes. Our scheme requires as few as 25 keys to be stored by each node thus minimizing required memory space in sensors and this enables implementation of our scheme more practical in these sensor nodes.

Any two nodes that share keys have atleast two keys in common. Thus our method is inherently 2-composite. For any two nodes to securely establish a common shared key they need to have exactly two common keys. This restriction makes the scheme more secure against node captures as described in the key discovery phase.

Our Contributions are summarized as below

- First we propose a deterministic key pre-distribution scheme in which every node shares at least two keys with every other node. Each node has  $2 \times \sqrt{N} - 1$  keys, where  $N$  is the number of nodes in the network. This scheme although guarantees complete connectivity is infeasible owing to its large number of keys. Then we extend this simple scheme to our novel sub-key vector assignment scheme. This scheme trades off connectivity for key space thereby making the network more scalable.

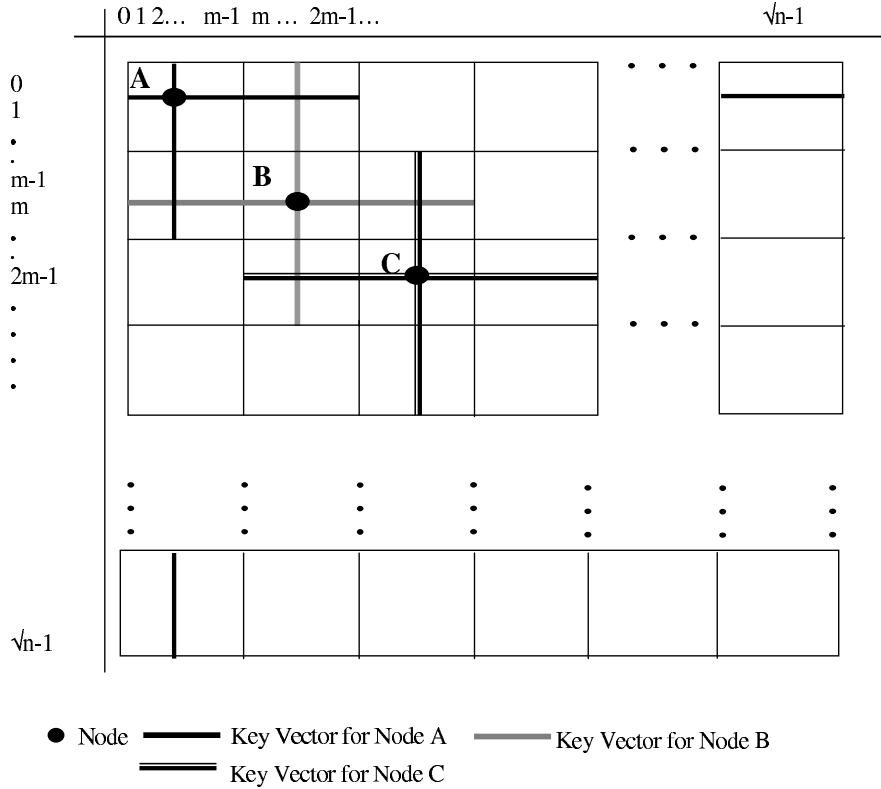


Figure 5.2: sub grid scheme key vector assignment

- Then we extend this simple scheme such that each node get a set of different keys from  $M$  mappings. In each mapping every node gets distinct set of keys. The final key set of the node is the combined set of keys from all the mappings.
- We use various metrics to show the tradeoffs in connectivity and memory and compare our proposed protocol with the random key pre-distribution scheme.
- We present analytical and simulation results to study the performance of our scheme in terms of security and connectivity.

Most of the previously proposed schemes tackle the issue of connectivity alone while assigning keys. In our proposed schemes we evaluate both connectivity and security. To analyze the performance of our schemes we use three metrics.

- Probability  $P$  that two nodes share a common key is a good indicator of the connectivity of the network. We use this metric to show the effectiveness of the proposed

protocols.

- The variance  $V$  in the number of keys revealed per node, when  $x$  nodes are captured, which indicates the security performance of the scheme.
- $\frac{P}{V}$  is the new metric proposed by us which combines the connectivity and security performance of the protocol.

## Chapter 6

# Proposed Key Pre-Distribution Scheme

This section describes the steps required to establish pair-wise secret keys among sensor nodes. The following three steps are essential in a key pre-distribution scheme.

- Key pre-distribution phase in which every node is loaded with a set of keys  $V_i$  (key vector), which are generated by the key server.
- Key discovery phase where every pair of neighboring nodes  $N_i, N_j$  finds a key path. The key path is a direct link if  $|V_i \cap V_j| = 2$  ( $N_i, N_j$  share exactly two keys). If neighboring nodes do not share exactly two keys they find a logical path  $\mathcal{P}_{ij}$  through a set of intermediate nodes  $N_1, N_2, \dots, N_l$  such that a subsequence of  $\mathcal{P}_{ij}$  ( $N_i, N_{m1}, N_{m2}, \dots, N_j$ ) exists where consecutive nodes in the subsequence share exactly two keys. This ensures that two neighboring nodes  $i$  and  $j$  can securely use this path to establish a shared key. Our scheme requires nodes to have two keys in common to ensure better security and making it more resilient to node capture.
- Key establishment phase in which neighboring nodes use the paths computed in the key discovery phase to establish new pair-wise shared keys which are used for secure communication.

We introduce a novel key pre-distribution method in which each sensor node is loaded with a set of keys chosen using our proposed sub-grid scheme, which is an extension of the simple

scheme described below.

## 6.1 Grid Key Vector Assignment

Figure 5 illustrates our simple pre-distribution scheme. First we construct a  $\sqrt{n} \times \sqrt{n}$  grid  $G$  with  $n$  keys such that exactly one key  $K_{ij}$  is at each position of the grid. A node  $N_{ij}$  gets the keys in row  $i$  and column  $j$ . Hence each node in this scheme gets a key vector of size  $2 \times \sqrt{n} - 1$ . Note that in this arrangement every pair of nodes share atleast two keys in common. This ensures that every neighboring node can establish a common shared pair-wise key after the nodes are deployed. Figure 1a shows that sensor node A shares two keys with node B. Although the basic scheme guarantees connectivity, it is not suitable for sensor networks because it requires a significant number of keys to be allocated to each node. The proposed sub-grid based scheme trades off direct connectivity for key space in the nodes by reducing the key vector size in each node.

### 6.1.1 Sub-Grid Key Vector Assignment

This scheme is an extension of the simple key vector assignment explained above. In this scheme the keys are placed in a grid  $G$  of size  $\sqrt{n} \times \sqrt{n}$  which is divided into  $k \times k$  cells each consisting of  $m \times m$  ( $m = \sqrt{n}/k$ ) keys as shown in the figure 6. A node in a particular cell is assigned the keys from the sub grid as explained below.

*Notations*

$$= \left\{ \begin{array}{ll} K_{ij} & \text{A unique key placed at position (ij)} \\ & \text{on the grid} \\ N_{ij} & \text{Node at position (ij) on the grid} \\ C_{xy} & \text{Represents a cell in the} \\ & \text{grid } G \\ SG_{xy} & \text{The grid formed by the cell} \\ & \text{ } C_{xy} \text{ and its 8 adjacent cells} \\ V_{ij} & \text{The key vector for a node } N_{ij} \\ & \text{in } SG_{xy} \end{array} \right.$$

We define a sub-grid  $SG_{xy}$  for a cell  $C_{xy}$ , which includes the cell itself and all its adjacent cells. For cells at the boundaries, adjacent cells also include cells at the respective opposite boundaries (wraparound). A node  $N_{ij}$  in cell  $C_{xy}$  is assigned keys from  $SG_{xy}$ . The key vector for the node is

$$V_{ij} = \cup K_{i,c} + \cup K_{r,j},$$

where,

$$(y - 1) \text{ mod } k \times m < c < (y + 1) \text{ mod } k \times m \text{ and}$$

$$(x - 1) \text{ mod } k \times m < r < (x + 1) \text{ mod } k \times m.$$

The size of the key vector  $V_{ij}$  is  $6 \times \sqrt{n}/k - 1$ . This is considerably smaller than the number of keys required in grid key vector scheme. The number of keys shared by two nodes

$$N_{i_1j_1} \text{ and } N_{i_2j_2} = \begin{cases} 3 \times \frac{\sqrt{n}}{k} & N_{i_1j_1}, N_{i_2j_2} \text{ are in the same cell} \\ & \text{and } i_1 = i_2 \text{ or } j_1 = j_2 \\ 2 \times \frac{\sqrt{n}}{k} & N_{i_1j_1}, N_{i_2j_2} \text{ are in cells which have} \\ & \text{common sides and } i_1 = i_2 \text{ or } j_1 = j_2 \\ 2 & \text{if } i_1 \neq i_2 \text{ and } j_1 \neq j_2 \text{ and } N_{i_1j_1} \text{ and} \\ & N_{i_2j_2} \text{ are adjacent} \\ 0 & \text{Otherwise} \end{cases}$$

The key vector size depends on the parameter  $k$  which determines the number of keys shared between nodes. A higher value of  $k$  reduces the key vector size at a node thereby decreasing the memory requirements for the keys. Also, a lower value of  $k$  decreases the security of the scheme as capture of a single node discloses a large number of keys. However, a very large value of  $k$  will produce lesser sharing of keys among nodes thereby decreasing the connectivity of the network. A suitable value of  $k$  should be chosen to maximize connectivity while satisfying stringent memory constraints of a sensor node.

Our scheme can be extended to use polynomials instead of keys as shown in our earlier work[10]. The extended polynomial scheme uses just one polynomial for each row and column in the grid. Each polynomial is divide into  $\sqrt{n}$  shares and placed at the  $\sqrt{n}$  positions in the column or row. In each mapping the node gets all the polynomial shares in the column and row of the subgrid. Using the polynomial shares the physically adjacent nodes can construct common shared keys.

### 6.1.2 Key Discovery Phase

Nodes loaded with their key vectors are randomly deployed in the area of interest. After the node deployment phase neighboring nodes exchange their node-id's to determine the number of keys they share. This can be done as the node-id can be used to determine the cell of the node that identifies the *key vector* of the node. Only neighboring nodes that share exactly

two keys are allowed to securely communicate with each other by establishing a common shared key to form a direct link. Nodes belonging to the same cell and in the same row or column share more number of keys. However these two nodes are not allowed to use the common keys because capturing of a single node in that row or column reveals those keys.

### 6.1.3 Path-Key Establishment Phase

On completion of the key discovery phase all the neighboring nodes may not have established common shared keys. In order that a node establishes keys with non-key-neighbors, it must go through the path-key establishment phase. In this phase, a node searches among its key-neighbors recursively to find a key-path to the non-key-neighbor. For example in figure 6 node A and node C are non-key-neighbors. In order for node A to communicate with node C it must find a intermediate node such that it shares keys with nodes A and C.

## 6.2 Security Analysis

Nodes deployed in hostile environments are prone to capture. Capture of a single node discloses all the information about the keys contained in them. An adversary can capture multiple nodes and use these keys to eavesdrop upon links. Hence the security of these keys is very essential for the overall security of the protocol.

We make the following assumptions about the adversary

- We assume that an adversary can capture only a fixed number of nodes in the network.
- Once a node is captured all the information is known to the adversary.
- The adversary can eavesdrop on any link and can decrypt the messages using the known key pairs.

### 6.2.1 Link Capture

We define *link capture* as the ability of the adversary to eavesdrop on the links between any two nodes. The adversary can decrypt the messages on a particular link, if the captured nodes disclose the keys shared between the nodes forming the link. This is a reasonable assumption because for  $x$  keys discovered it has to only check for  $x \times (x - 1)$  combinations.

We use the following metrics to analyze our protocol

- a)  $N_c$ : Number of nodes to be captured to compromise a single link across two specified nodes.
- b)  $P(x)$  probability of link compromise for  $x$  captured nodes
- c)  $N(x)$  Number of links compromised for  $x$  captured nodes

#### a) $N_c$

We calculate the expected number of nodes to be compromised for capturing a particular link. First we calculate the probability that  $x$  nodes are captured for single link compromise. Then we find the expectation over all values of  $x$ .

Let  $N_{capt}(x)$  be the number of ways of capturing  $x$  nodes such that the two keys are known.

$M(x)$  be the number of ways of capturing  $x$  nodes such that the two keys are not known.

$$N_{capt}(x) = \binom{n}{x} - M(x)$$

where

$$\begin{aligned} M(x) &= \text{None of the two keys captured} + \text{Only one key is captured} \\ &= \binom{n - 12m + 4}{x} + 2 \times \binom{n - 6m + 1}{x} - 2 \times \binom{n - 12m + 4}{x} \end{aligned}$$

$$N_c = \frac{\sum x \times N_{capt}(x)}{\sum N_{capt}(x)}, \text{ where } 2 \leq x \leq n$$

**b)  $P(x)$**

We calculate the probability of link compromise for  $x$  captured nodes.

$P(x) = 1 - P'(x)$  where  $P'(x)$  is probability of link not compromised for  $x$  captured nodes.

$$\begin{aligned} P'(x) &= \text{Probability that no key is known} + \text{Probability that only one key is known.} \\ &= \frac{\binom{n-12m+4}{x} + 2 \times \binom{n-6m+1}{x} - 2 \times \binom{n-12m+4}{x}}{\binom{n}{x}} \end{aligned}$$

**c)  $N(x)$**

Let  $Y$  be random variable which represents the number of compromised links, when  $x$  nodes are captured.

By using binomial distribution for probabilities we get

$$Prob(Y = r) = \binom{n}{r} P(x)^r (1 - P(x))^{n-r}$$

The expected value of the number of captured links,

$$N(x) = E(Y) = \sum r \cdot Prob(Y = r)$$

**Theorem:** The expected value of the number of links captured for  $x$  compromised nodes  $E(Y)$  is  $\mathcal{L}P(x)$ , where  $\mathcal{L}$  is the number of links between the deployed nodes.

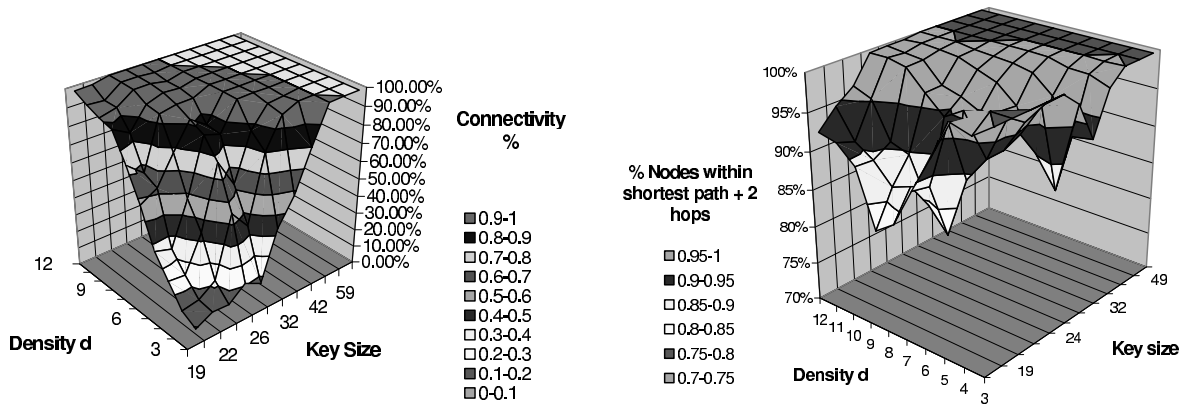
Proof: Let  $Q(x) = 1 - P(x)$

We prove the theorem by using the binomial expansion.

$$(P(x) + Q(x))^\mathcal{L} = \sum \binom{\mathcal{L}}{r} P(x)^r Q(x)^{\mathcal{L}-r}$$

Differentiating W.R.T  $P(x)$  on both sides we get the below equation.

$$0 = \sum r \binom{\mathcal{L}}{r} P(x)^{r-1} Q(x)^{\mathcal{L}-r} - \sum (\mathcal{L} - r) \binom{\mathcal{L}}{r} P(x)^r Q(x)^{\mathcal{L}-r-1}$$



(a) Connectivity versus Key Vector Size versus Node Density

(b) Percentage of Nodes Within  $sp+2$  versus Key Vector Size versus Node Density

Figure 6.1: Impact of Key Vector Size and Node Density On Connectivity and Path Length

Multiplying both sides by  $P(x)Q(x)$  the expression is simplified to.

$$0 = Q(x)E(Y) + P(x)E(Y) - \mathcal{L}P(x)$$

Hence,  $E(Y) = \mathcal{L}P(x)$

A smaller value of  $k$  here increases the connectivity of the network. However the security of the scheme is compromised by making the value of  $k$  too small. The value of  $k$  can be tuned to provide the desired balance between security and connectivity. Typically, higher connectivity of a network trades off different security issues. In the proposed sub grid key vector key pre-distribution scheme the tradeoff depends on the parameter  $k$ . The higher the value of  $k$  the smaller is the size of each cell. Consequently, the connectivity is reduced and the security is better.

# Chapter 7

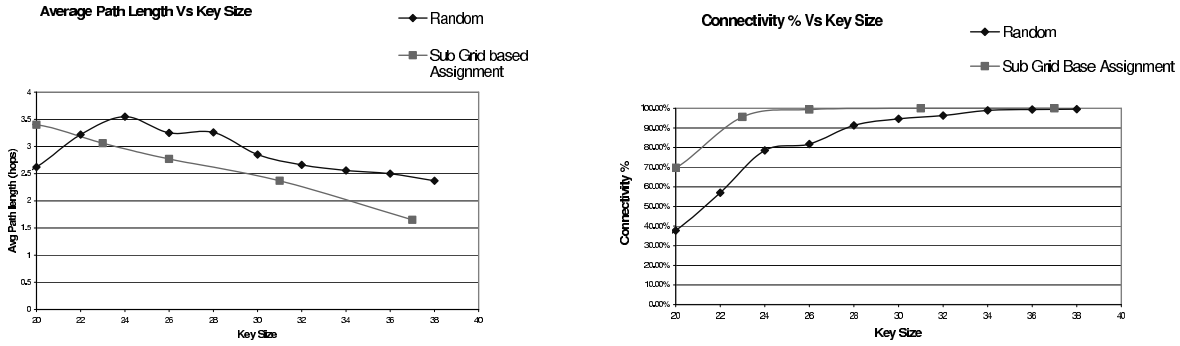
## Simulation

The effectiveness of the sub-grid key vector key pre-distribution scheme is tested through simulation. In the remainder of the section we show the impact of the value of  $k$  on connectivity under our subgrid scheme. Next we show the memory savings in our scheme compared to the random key pre-distribution scheme. Connectivity and average path length metrics were calculated for varying values of density, *key vector* size and network size.

### 7.1 Experimental Setup

The simulation assumes that nodes are deployed randomly in the target region. The deployment is done with varying densities  $d$ . We assumed a two hop neighborhood for our simulations. For our simulations a density  $d$  is equivalent to  $4\pi d$  nodes in a cluster. The simulations are done for different values of the sub-grid parameter  $k$  and density  $d$ .  $d$  determines the average number of nodes that lie in the neighborhood of a node. Each simulation was run 100 times with different seeds for the random number generator for deployment of nodes and the results presented are the average of 100 runs.

The different phases of the key pre-distribution scheme have been simulated. The logical key space is first defined in the form of a grid and the nodes are distributed keys depending on their position in the grid that determines their identity based on the sub-grid key vector



(a) Average Path Length versus Key Vector Size

(b) Network Connectivity versus Key Vector Size

Figure 7.1: Performance Comparison between Random Scheme and Our Proposed Sub-Grid Scheme

scheme described above. The nodes are then deployed at random. In the next phase a node finds out the nodes within its neighborhood it shares keys with. The cell to which a node belongs can be determined based on the identity of the node. The nodes share keys as specified by the key pre-distribution scheme earlier. Next we determine the connectivity of the network. With respect to a particular node, we determine if a path can be established between that node and every other node in the neighborhood, thus determining the connectivity of the network.

### 7.1.1 Impact of Key Vector Size on Connectivity and Path lengths

Figure 7a illustrates the relationship between *key vector* size, density and connectivity for a 2500 node network. The graph gives a clear picture of the key vector size that needs to be selected to achieve the desired connectivity for a given density of node distribution. We can observe that the key vector of 42 keys gives more than 98% connectivity for a very sparse neighborhood consisting of 37 nodes. The optimal *key vector* size and density for more than 98% connectivity is along the edge of the cliff in the 3D-graph (fig 7a). Note that as density increases the number of keys required to achieve desired 98% connectivity decreases. Figure 7b shows percentage of nodes within  $sp+2$  (shortest path distance + 2 according to the

actual deployment) in relation to *key vector* size and density. Larger key vector and higher densities increase percentage of nodes within  $sp+2$  increases.

### 7.1.2 Comparison to the Random Scheme

The performance of our protocol is compared with the random key pre-distribution scheme. [3]. The number of nodes  $n$  that are used in the predeployment was fixed as 1000 under both the protocols. However we consider a cluster of 60 nodes for our simulation of post deployment performance. We test our protocol with varying values of  $k$ . The key pool size for the random scheme is of size 10000 as in [3]. The key vector size under each protocol is same. Figure 8a shows a comparison of the average path length(for establishing shared keys between neighbors) under the two protocols using different key vector sizes. Figure 8b shows the relationship between connectivity and key vector size under the two protocols. Our protocol has better average path length than random scheme as key vector size increases. Although the random scheme is initially better it can be used because at such low key vector sizes its connectivity is very low as shown in figure 3b. Our protocol achieves the desired connectivity as low as 40% lesser memory compared to the random scheme.

In the next section we propose two deployment strategies in where each node gets its keys from multiple mappings of nodes to keys.

## 7.2 Multiple Layer Pre-Deployment Strategies

Consider  $M$  one-to-one mappings of  $N$  sensor nodes to random positions in the grid  $G$ . Our preliminary results indicate that a choice of a small constant for  $M$  yields good security-performance tradeoffs. In the given mapping  $i$ ,  $1 \leq i \leq M$ , each node is assigned a set of sub-row and sub-column keys from its subgrid and adjacent grids as shown in figure 2. For a given network deployment ,two physically adjacent sensors can encrypt messages using shared keys under one or more of these mappings.

- *Strategy 1* : First keys are placed at each position in the grid. Then for a mapping the nodes are placed at each position on the grid randomly. The keys are assigned to nodes as in the sub-grid scheme. We impose a restriction on the placement of nodes in each mapping. The mappings are such that a node does not get duplicate keys from different mappings. So, every node gets a unique set of keys in each mapping.
- *Strategy 2* : Unlike the previous strategy initially each grid position is occupied by a node. Subsequently in each mapping keys are placed randomly at grid positions. Similar to *strategy 1*, each node gets unique set of keys in each mapping

### 7.2.1 Key-Node Mapping Algorithm

Now we present the algorithm to assign the keys to nodes in each mapping  $i, 1 \leq i \leq M$ . Let  $pos[j]$  be the array of sets representing the available positions for nodes/keys at the beginning of each mapping. Initially  $pos[j]$  contains all the points in the grid every node/key. The following steps are done for each mapping  $i$ .

- Each node/key  $j, 1 \leq j \leq n$ , is mapped to a position in the grid such that it can be placed in grid positions available to  $j$  from  $pos[j]$ .
- For each node/key  $j$   $pos[j]$  is updated to the new set of available positions. This new set is obtained by removing the newly assigned keys/nodes to node/key  $j$  from the set  $pos[j]$ .

Note that the above approach is used for both the strategies presented above. Nodes and keys can be interchanged in the above algorithm to obtain mappings for the above two strategies. In the next section we evaluate the connectivity and security for the above two schemes.

## 7.3 Connectivity

After the nodes deployed randomly in the area of deployment nodes within communication range try to establish pair-wise keys if they have common shared keys. Hence the probability that a given pair of nodes share at least one key, is a good metric for evaluating the connectivity.

a) Probability of sharing a key  $p_1$  under *strategy 1* :

The placement of the nodes in each mapping is not completely independent of previous mapping. The following equation gives the probability  $p$  that two nodes share a key.

$p_1$  = Probability that two nodes share a key

$p_1 = 1 -$  Probability that two nodes do not share a key in any of the mappings

$$p_1 = 1 - \frac{(n-a)(n-b-a)(n-2b-a)\dots(n-Mb-a)}{n(n-b)(n-2b)\dots(n-Mb)},$$

$$\text{where } a = \frac{9n}{k^2} + \frac{4\sqrt{n}}{k}, b = \frac{6\sqrt{n}}{k} - 1$$

b) Probability of sharing a key  $p_2$  under *strategy 2* :

Under this strategy two nodes share keys in every mapping if they are in adjacent cells.

Otherwise they share a key if the same key is assigned to the nodes in different mappings.

$p_2$  = Probability that two nodes share a key

$p_2 =$  Probability that two nodes share a key in every mapping +

Probability that two nodes are not adjacent and share a key

$$p_2 = \frac{10\sqrt{n}}{n-1} + \left(1 - \frac{10\sqrt{n}}{n-1}\right) \left(\frac{(n-2b)(n-3b)\dots(n-Mb)}{(n-b)(n-2b)\dots(n-(M-1)b)}\right)$$

Figures 9 and 10 show the analytical and experimental connectivity results for the above

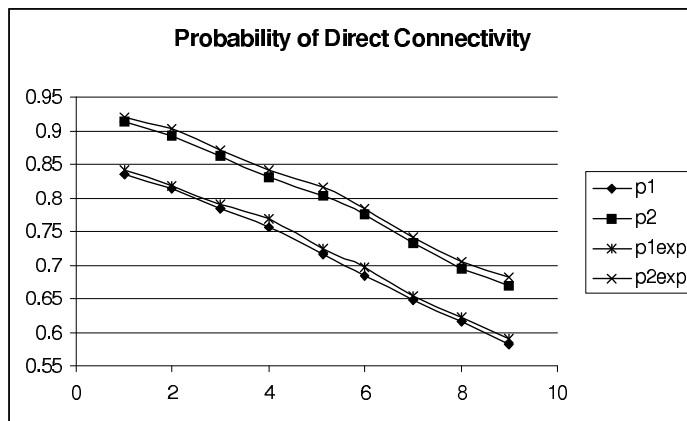


Figure 7.2: Probability That two Nodes Share a Key with increasing value of (k,M) where k=M

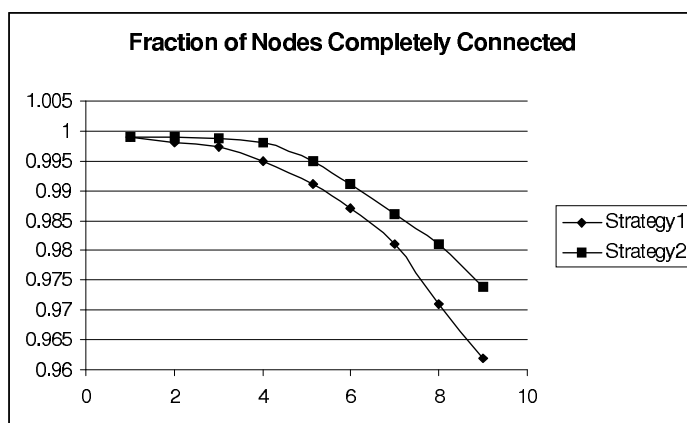


Figure 7.3: Fraction of Total Nodes that are Connected with increasing value of (k,M) where k=M

two proposed schemes. 10000 nodes were considered for this example. The values for  $k$  and  $M$  are chosen such that the number of keys each node gets remains the same. The values for  $p_1$ ,  $p_2$ ,  $p_1^{exp}$  and  $p_2^{exp}$  are calculated for different values of  $k$  and  $M$ . Also the overall connectivity of the network is obtained through simulations. This is the percent of the nodes that are connected after the completion of the key discovery and key path establishment phase.

The values  $p_1^{exp}$  and  $p_2^{exp}$  are values obtained through simulations. The table shows the closeness of the actual values to the analytical values obtained. Although direct connectivity comes down with increase in value of  $M$  and  $k$  overall total connectivity comes down at a much slower rate.

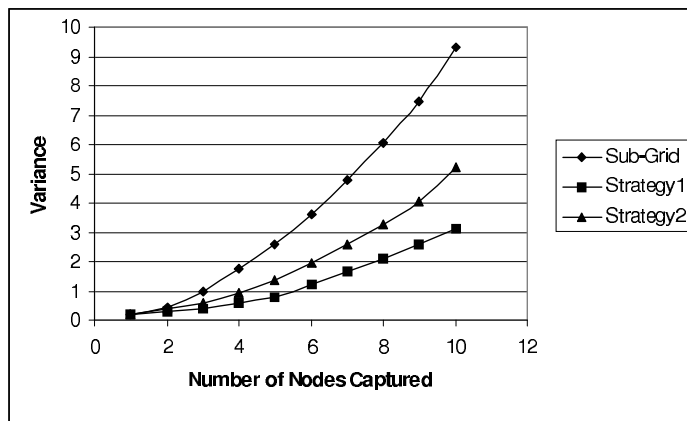


Figure 7.4: variance in number of keys disclosed per node.

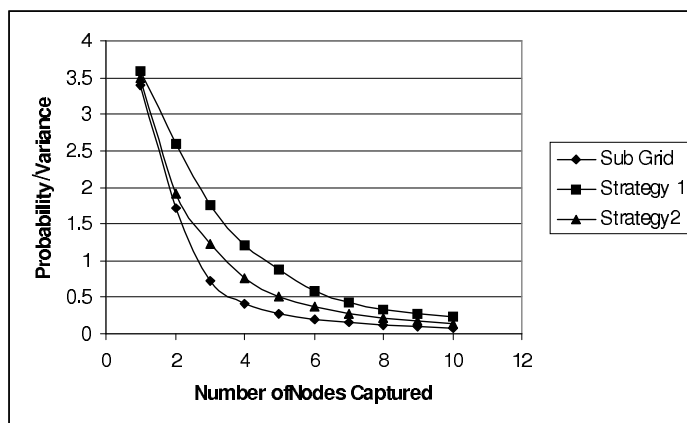


Figure 7.5: Probability of Connectivity / Variance of keys disclosed

## 7.4 Security Analysis

Nodes are grouped together under strategy2 and then keys are assigned to nodes. However in strategy1 keys are grouped together and then nodes are assigned keys. Therefore under strategy2 the capture of a single node discloses a number of keys which are shared by many nodes. But the capture of nodes under strategy1 does not disclose a significant portion of the keys of all the nodes. The captured keys are distributed evenly among the whole nodes in the network. However under strategy2 a large number of keys for some of the node's are disclosed. So, the whole set of disclosed keys have to be revoked and the number of available keys in some of the nodes is reduced significantly. But while using strategy1 individual nodes

still have a large number of available keys. Although strategy1 gives lower connectivity it is better suited owing to its better security performance.

Figures 11 and 12 show the security performance of the protocol. Obviously strategy1 has the lowest variance in the number of keys revealed per node. Under strategy2 and sub-grid schemes the position of the nodes is fixed the keys are placed on them. Hence capture a of a single node reveals keys of all its neighbors. Similarly, the connectivity security shows the better performance of strategy1.

## 7.5 Conclusions

We have proposed a novel approach for energy management at the MAC layer in a wireless sensor network. The protocol uses TDMA along with periodic listen and sleep to avoid energy wastage. The key feature of our protocol is the leader election method by which the most critical node is chosen to evade idle listening. Our simulation results show that ER-MAC achieves a significant gain in energy savings compared to other existing MAC layer protocols.

This work presents a new pre-distribution scheme for wireless networks. It has nearly 40% lesser memory requirements compared to the random scheme. Our detailed simulation shows the gain in memory savings and better connectivity under our scheme compared to the random scheme. This makes our scheme more scalable compared to the previous schemes.

# Bibliography

- [1] R.Kalidindi, V.Parachuri, S. Basavaraju, C. Mallanda, A. Kulshrestha, L. Ray, R. Kannan, and A. Durrezi, "Sub-Grid Based Key Vector Assignment: A Key Pre-Distribution Scheme For Distributed Sensor Networks" In ICWN 04.
- [2] Sanjay Shakkottai, R. Srikant, Ness Shroff, "Unreliable Sensor Grids: Coverage, Connectivity and Diameter". In Proc. of 22nd Annual Joint Conference of the IEEE Computer and Communications Societies (IEEE Infocom2003), San Francisco, April 2003.
- [3] Laurent Eschenauer and Virgil D. Gligor, "A key management scheme for distributed sensor networks". Proceedings of the 9th ACM Conference on Computer and Communication Security, pages 4147, November 2002.
- [4] H. Chan, A. Perrig, D. Song. "Random Key Predistribution Schemes for Sensor Networks". In Proc. of the IEEE Security and Privacy Symposim 2003, May 2003.
- [5] A. Perrig, R. Szewczyk, V. Wen, D. Culler, and J. Tygar, "SPINS: Security Protocols for Sensor Networks". In Proc. of Seventh Annual ACM International Conference on Mobile Computing and Networks(Mobicom 2001), Rome Italy, July 2001.
- [6] Donggang Liu, Peng Ning, "Establishing Pairwise Keys in Distributed Sensor Networks," In 10th ACM Conference on Computer and Communications Security (CCS '03), Washington D.C., October, 2003.
- [7] C. Karlof and D. Wagner, "Secure Routing in Sensor Networks: Attacks and Countermeasures". In Proc. of First IEEE Workshop on Sensor Network Protocols and Applications, May 2003.
- [8] R.L. Rivest, "The RC5 encryption algorithm", In Workshop on Fast Software Encryption (1995) pp. 86 96.

- [9] R.L. Rivest, A. Shamir and L.M. Adleman, "A method for obtaining digital signatures and public-key cryptosystems", *Communications of the ACM* 21(2) (1978) 120-126.
- [10] R.Kalidindi, R.Kannan, "Polynomial based Pre-key distribution schemes for sensor networks". In LSU technical report LSU-CSC-TR-04-003.
- [11] I.F. Akyldiz, W. Su, Y. Sankarasubramaniam and E. Cayirci, "Wireless Sensor Networks: A Survey," *Computer Networks*, Vol. 38, No. 4, pp. 393-422, March 2002.
- [12] LAN MAN Standards Committee of the IEEE Computer Society, Wireless LAN medium access control (MAC) and physical layer (PHY) specification, IEEE, New York, USA, IEEE Std 802.11 - 1997 edition, 1997.
- [13] S.Singh and C.S.Raghavendra, "PAMAS: Power Aware Multi-access Protocol With Signaling for Ad Hoc Networks," *ACM Computer Communication Review*, vol.28, no.3, pp.5-26, July 1998.
- [14] Wei Ye, John Heidemann and Deborah Estrin, "An Energy-Efficient MAC Protocol for Wireless Sensor Networks," *Proc. of 12th IEEE International Conference on Computer Networks*, INFOCOM 2002, New York, NY, USA, June 2002.
- [15] Katayoun Sohrabi and Gregory J. Pottie, "Performance of a Novel Self-organization Protocol for Wireless Ad Hoc Sensor Networks", *Proc. of the IEEE 50th Vehicular Technology Conference*, pp.1222-1226, 1999.
- [16] Alec Woo and David Culler, "A Transmission Control Scheme for Media Access in Sensor Networks," *Proc. of the ACM/IEEE International Conference on Mobile Computing and Networking*, Rome, Italy, July 2001.
- [17] S. Park, A. Savvides and M. B. Srivastava, "SensorSim: A Simulation Framework for Sensor Networks," *Proc, of MSWiM 2000*, Boston, MA, August 11, 2000.
- [18] Frazer Bennett, David Clarke, Joseph B. Evans, Andy Hopper, Alan Jones, and David Leask, "Piconet embedded mobile networking", *IEEE Personal Communications*, 4(5):8-15, October 1997.
- [19] T. S. Rappaport, "Wireless Communications", Prentice-Hall, 1996.
- [20] R. Steele, "Mobile Radio Communications", Pentech Press, London, 1992.

# Vita

Mr Ramaraju Kalidindi was born in the town of Bhimavaram in the state of Andhra Pradesh, in India. He did his bachelor of technology degree in computer sciences in Andhra University at Visakhapatnam in May 2000. He arrived at Louisiana State University for master's degree in systems science program in fall 2000. He successfully completed his degree of Master of Science in Systems Science in the department of Computer Science at Louisiana State University in August 2005.