Scheduling of Relay Nodes in a Heterogeneous Wireless Sensor Network

Submitted by

Mohammad Saber

(Student ID : 100505052P)

A thesis submitted to the Department of Computer Science and Engineering of Bangladesh University of Engineering and Technology for partial fulfillment of the degree of Master of Science in Computer Science and Engineering

October 2009

Department of Computer Science and Engineering

Bangladesh University of Engineering and Technology

The thesis titled **"Scheduling of Relay Nodes in a Heterogeneous Wireless Sensor Network"** submitted by **Mohammad Saber**, Roll No. 100505052P, Session October 2005, to the Department of Computer Science and Engineering, Bangladesh University of Engineering and Technology, has been accepted as satisfactory in partial fulfillment of the requirement for the degree of Master of Science in Computer Science and Engineering. The examination was held on October 26, 2009.

Board of Examiners

1. ________________

Dr. Mahmuda Naznin — Chairman (Supervisor)
Assistant Professor
Department of Computer Science and Engineering
Bangladesh University of Engineering and Technology, Dhaka – 1000

2. ________________

Dr. Md. Monirul Islam — Member (Ex-officio)
Professor & Head
Department of Computer Science and Engineering
Bangladesh University of Engineering Technology, Dhaka – 1000

3. ________________

Dr. Reaz Ahmed — Member
Assistant Professor
Department of Computer Science and Engineering
Bangladesh University of Engineering Technology, Dhaka – 1000

4. ________________

Dr. M. Sohel Rahman — Member
Assistant Professor
Department of Computer Science and Engineering
Bangladesh University of Engineering Technology, Dhaka – 1000

5. ________________

Dr. Mohammad Nurul Huda — Member (External)
Associate Professor
Department of Computer Science and Engineering
United International University, Dhaka.

Declaration

This is to declare that the work presented in this thesis has been done by **Mohammad Saber**, Student No. 100505052P, under the supervision of **Dr. Mahmuda Naznin**, Assistant Professor, Department of Computer Science and Engineering, Bangladesh University of Engineering and Technology, is the result of extensive study and experiments. It is further declared that neither this thesis nor any part of it has been submitted elsewhere for the award of any degree or diploma.

Mohammad Saber	**Dr. Mahmuda Naznin**
MSc. Student, BUET	Supervisor of the Thesis

To

My Beloved Parents

Without whom any achievement in life would be meaningless

Acknowledgements

At first, I would like to thank my supervisor Dr. Mahmuda Naznin, Assistant Professor, Department of Computer Science and Engineering (CSE), Bangladesh University of Engineering and Technology (BUET), for his continuous support, advice and care. Her keen interest in research, constructive criticism, continual encouragement, deep sense of responsibility towards his students and dynamic supervision influenced me to carry out and finish this difficult work.

Secondly, I would like to express intense gratitude to Dr. A. S. M. Latiful Haque, Associate Professor, Dr. Reaz Ahmed, Assistant Professor and Dr. M. Sohel Rahman, Assistant Professor, Department of Computer Science and Engineering, BUET, for their cordial cooperation and suggestions.

Thirdly, I would like to acknowledge with thanks to the Head of the Department and other faculty members, officers and staffs of the Department of Computer Science and Engineering, BUET for their friendly cooperation.

I would like to convey my sincere thanks to the Head of the Department and other faculty members of the Department of Computer Science and Engineering of Ahsanullah University of Science and Technology for their support and cooperation.

Finally, I would like to express my deep respect and gratitude to my father Md. Nurul Quadir, my mother Mrs. Shahrya Quadir and my brother Md. Shoaib, without whose sacrifice, support and encouragement, this work would have been impossible.

Abstract

Wireless sensor network is composed of large number of sensor nodes with limited battery power which cannot support long range communication. Sensor nodes' battery cannot be replaced due to adverse environment which causes connectivity failure in sensor network. Besides, sensors have short communication ranges and relay nodes are usually more powerful than sensor nodes. Relay nodes have longer communication range than sensor nodes. Therefore, to assist sensor nodes if relay nodes are added, communication will be more reliable. Usually relay nodes communicate with sensor nodes, or other relay nodes, or the base stations. It is desirable to place the minimum number of relay nodes in sensor network and keep the minimum number of relay nodes working to save network energy consumption. Most of the existing work on placing relay nodes in wireless sensor networks assume homogeneous network where sensor nodes and relay nodes have identical characteristics which is not realistic. This thesis work addresses the problem to find the optimal load balancing strategy where the minimum number of relay nodes be working to provide network connectivity considering heterogeneous properties of the network nodes. We study heterogeneous property of relay nodes and show that, existing algorithms do not support heterogeneous environment. Then, we give models, considering heterogeneous properties where we schedule wake up and sleep times for relay nodes to optimize the connectivity with the goal of saving network energy expenditure. We give two different guaranteed optimal models which keep the minimum number of relay nodes awake resulting to the optimal load balancing.

Table of Contents

Table of Contents (continued)

List of Tables

List of Tables (continued)

List of Figures

List of Figures (continued)

Chapter 1

Introduction

1.1 Overview

A transducer that converts a physical phenomenon such as heat, light, sound or motion into electrical or other signals that may be further manipulated by other apparatus is called a *sensor* [11]. Basic unit of a sensor node consists of processor, memory wireless modem, power supply and on-board sensor. Recent advancement in wireless communication and electronics have enabled the development of low cost, low power, multifunctional sensor nodes which are small in sizes and compatible to communicate within short distances. These tiny sensor nodes, which consist of sensing, data processing and communicating components, leverage the idea of sensor networks [15]. When a large number of sensor nodes create a network to communicate between them is called *sensor network*. Sensor network is a new family of wireless networks which is different from traditional wireless network like cellular network or Mobile Ad-hoc Network (MANET). In sensor network, sensors are densely deployed either inside the phenomenon or very close to it. The position of sensor nodes does not need to be engineered or predetermined. This allows random deployment of sensors in inaccessible terrains or disaster relief operations. This means that, sensor networks' protocols and algorithms must possess self-organizing capabilities. Another unique feature of sensor network is the cooperative effort of sensor nodes. Instead of sensing raw data to the nodes responsible for the fusion, they use their processing abilities to locally carry out simple computations and transmit only the required and partially processed data.

1.2 Preliminaries

We can divide sensor network based different attributes of sensor nodes. If all of the sensor nodes in a wireless sensor network have similar transmission, sensing range and battery power then, we can term this type of network as *homogeneous wireless sensor network* i.e. all sensors posses identical property in homogeneous wireless sensor network. But, in real life it's very difficult to get homogeneous wireless sensor network. Suppose we have sensor nodes with identical property in a given wireless sensor network. But some sensor node may loose energy quickly than other nodes due to heavy traffic in some area. In this case, the battery power might vary from node to node after some operation. Thus we can say, absolute homogeneous wireless sensor network is very difficult to be found in real world. If two or more sensors in a sensor network have different battery power and transmission/sensing range, then we can term this type of network as *heterogeneous wireless sensor network* (*H-WSN*).

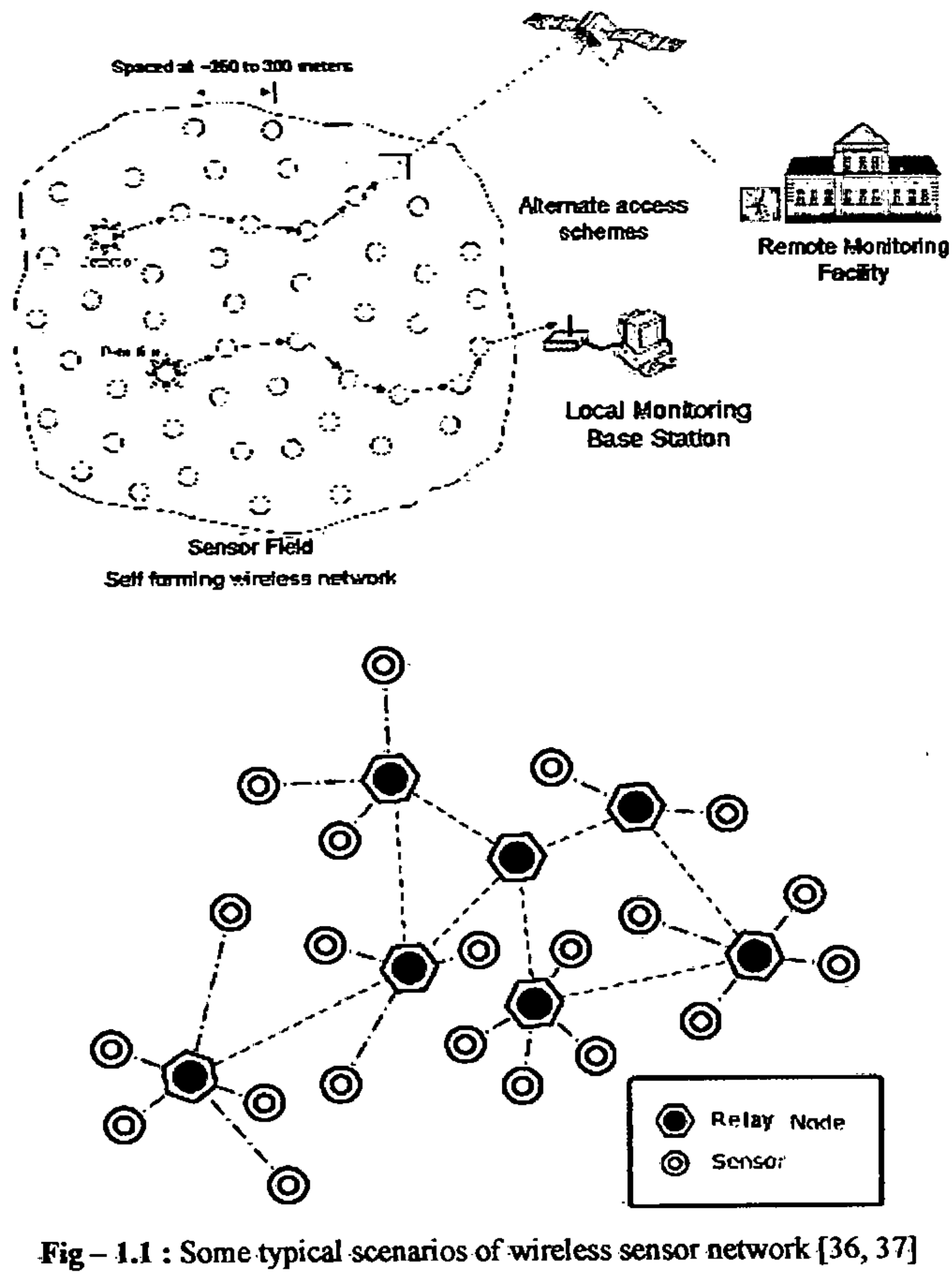

Fig – 1.1 : Some typical scenarios of wireless sensor network [36, 37]

Wireless sensor networks (WSNs) have received significant attention in recent years due to their potential applications in military sensing, wildlife tracking, traffic surveillance, health care, environment monitoring, building structures monitoring etc. We can deploy sensors under the sea also to do research about under water life and environment monitoring. Sensor can be put into bodies of human being for medical research. In this way, we can monitor human body functions very closely which may lead to new medicine invention for deadly diseases. In hill tracks area surveillance operation is very difficult due to lack of accessibility. We can put sensors in those areas for environment monitoring for wild life surveillance and for border protection. By environment monitoring we can make early prediction of some natural calamities like flood situation, earthquake, tsunami etc. which may save many innocent lives.

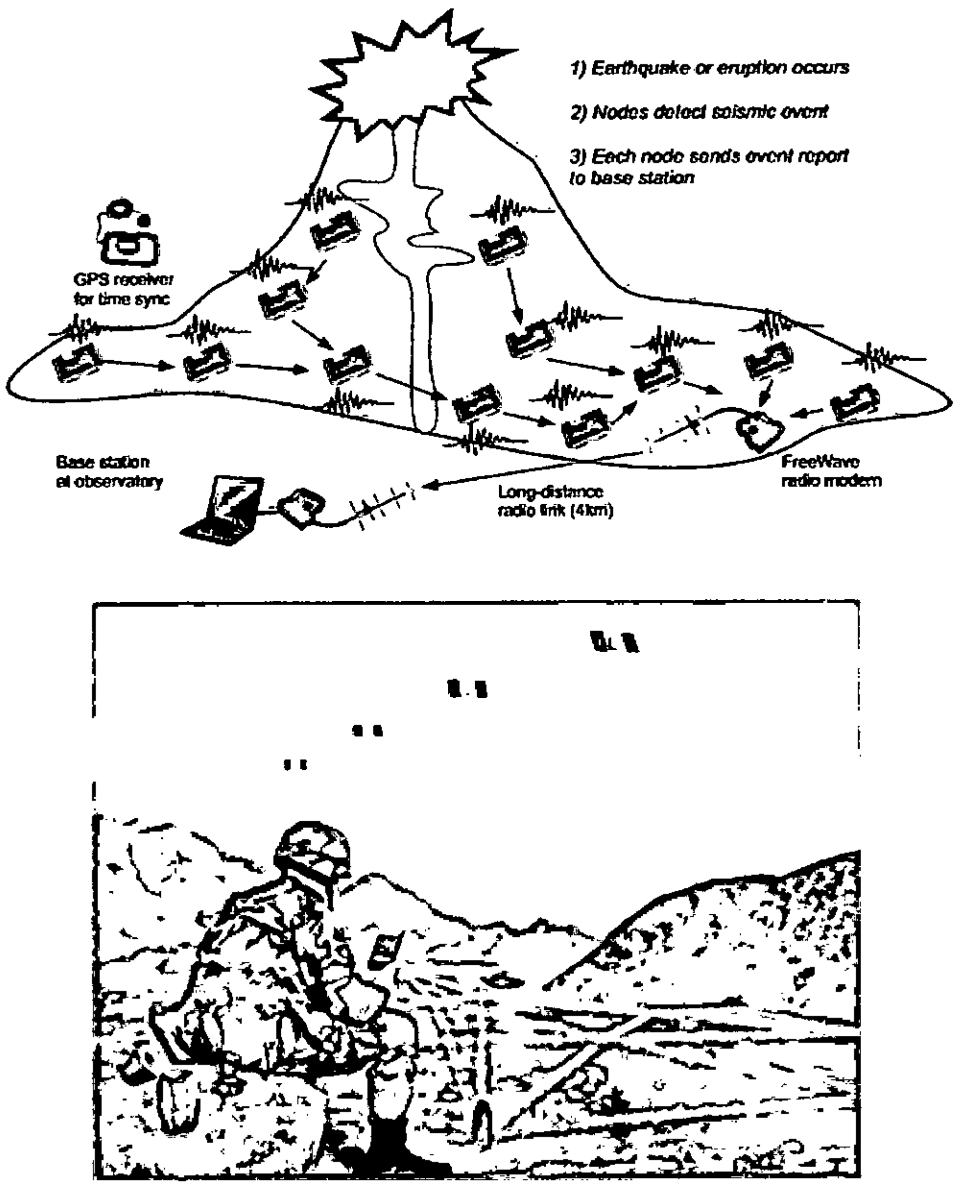

Fig – 1.2 : Applications of wireless sensor network [38, 39]

Sensor networks have very limited energy supply and their batteries cannot usually be replaced due to adverse environment. Since sensors' energy cannot support long range communication to reach a sink which is generally far away from the data source, multi-hop wireless connectivity is required to forward data to the remote sink. It is a key problem regarding how to gather data packets from sensor nodes to the sink in applications. From this concept, researchers started research adding some additional nodes which are called *relay nodes* [14]. To prolong network lifetime while preserving network connectivity we can deploy small number of costly but more powerful relay nodes whose main task is to communicate with the other sensor or relay nodes. If we can minimize the number of total relay nodes in a WSN, then overall energy consumption by WSN will be minimized. Therefore, many algorithms were proposed on efficient relay node placement in WSN [2, 6, 9, 13, 14, 17, 23, 28, 31, 34]. Most of the existing works on placing additional relay nodes in wireless sensor networks assumes homogeneous wireless sensor nodes and relay nodes with an identical transmission radius and similar battery power. But in real life, one sensor network has different types of sensors also. Thus, heterogeneous properties of sensor nodes should be considered in every work related to sensor network to obtain better result.

1.3 Objective of the Thesis

The objectives of this thesis work are as follows:

- Study a heterogeneous sensor network and finding the challenges.
- Study the role of relay nodes and applicability in heterogeneous sensor network environment.
- Propose a model considering heterogeneous property of sensor network.
- Finding load balancing strategy in sensor network.
- Performance measurement by simulation.

1.4 Thesis Organization

In this section we give the outline of our thesis organization. The rest of the thesis is organized as follows:

In Chapter 2, a brief discussion about previous research on this topic has been given. In Chapter 3, we discuss the problem domain. We study the heterogeneous property of sensor network and then we discuss all possible nine cases to prove the existence of multiple paths between two sensors after incorporating heterogeneous property within an existing relay node placement algorithm [9]. We provide two solution models for energy saving strategy by scheduling relay nodes in heterogeneous environment which are guaranteed optimal solution.

In Chapter 4, we show the performance measurement of our models through some empirical results. We have performed simulation on extensive sets of data and the average of the results achieves from multiple data sets are presented in tables and graphs with description. Raw data of simulation results are available in Appendix-A with standard deviation and average which are used to draw graphs in Chapter 4.

In Chapter 5, we summarize our work and give some direction for future scope of research on this topic.

Chapter 2
Related Work

2.1 Overview

Relay node placement in wireless sensor network is a very interesting problem, which prolongs network lifetime by placing some number of relay nodes in sensor network to ensure connectivity between sensors. In this chapter, we briefly discuss some articles on this area.

2.2 Related Research

In 2006, H. Liu et al. [14] propose three approximation algorithms for placing relay node in homogeneous sensor network which they have called the minimum relay node placement problem (MRP). They discuss the MRP problem for two cases. In case 1, the network is connected and in case 2, the network is bi-connected. Liu propose a $(6 + \epsilon)$–approximation algorithm for case one, where $\epsilon > 0$ can be any given constant. They use the minimum disk cover algorithm of Hochbaum [8] to place a set of relay nodes in WSN. But the networks of these relay nodes may be disconnected, if the distance between two relay nodes are larger than the range of relay node. They place a set of additional relay nodes to make WSN connected through Steiner Tree Problem with Minimum number of Steiner Points (STP-MSP) algorithm [7] using previously found relay nodes as input. Then, they also propose a $(24 + \epsilon)$–approximation algorithm and a $(6/T + 12 + \epsilon)$–approximation algorithm for case two, respectively, for any $\epsilon > 0$, where T is the ratio of the number of relay nodes placed and the number of sensors in case one. They add three backup nodes within the communication range of each relay node to make the network bi–connected. The three backup relay nodes are placed on the vertices of an equilateral triangle with length d respectively, where d is the communication range of each relay node and the center of the equilateral triangle is on the previously placed relay node. They improve this 2–connected relay node placement algorithm by deploying only one relay node as backup node. They put relay node at the mid point of every steiner tree branch which improves the algorithm from (24 + ϵ)–approximation to (6/T + 12 + ϵ)–approximation algorithm. They further extend the result where communication radius of sensor nodes and relay nodes are different from each other. They also provide mathematical proof to establish their ideas. But work of this paper consider homogeneous sensor environment which is not practical.

In 2007, W. Zhang et. al [32] study both single-tired and two-tired relay node placement problems that ensure 2–connectivity, under the mild condition $R \geq r$, where R and r stand for transmission range of relay nodes and sensor nodes respectively. At first they introduce *single tired fault tolerant relay node placement* (1tFTP) problem and give a 14–approximation algorithm to solve it. They construct a Steinerized graph from given sensor nodes and then compute a spanning sub graph from the Steinerized graph using the algorithm of [27] to ensure fault tolerance through 2–connectivity for solving 1tFTP. Then they extend this problem to include base stations which is called *single-tired relay node placement with base stations* (1tFTPB) and propose a 16–approximation algorithm to solve it. They also propose a (10 + ϵ)–approximation algorithm for *two-tired relay node placement* (2tFTP) problem where they use the algorithm of [9] for relay node placement and then put extra relay nodes to make the network 2–connected. They conclude their paper with a (20 + ϵ)–approximation algorithm for *two-tired relay node placement with base stations* (2tFTPB) and some experimental results. In their paper, they do not consider heterogeneous property of relay nodes as well as sensor nodes. They also assume all nodes' transmission ranges are the same which might not happen in practical scenario.

In 2007, E. L. Lloyd et al. [9] study two versions of relay node placement problem. In first version they want to deploy the minimum number of relay nodes so that between each pair of sensor nodes, there is a connecting path consisting of relay and/or sensor nodes. In the second version, they want to deploy the minimum number of relay nodes so that between each pair of sensor nodes, there is a connecting path consisting solely of relay nodes. They propose a polynomial time 7–approximation algorithm for the first problem and a polynomial time (5 + ϵ)–approximation algorithm for the second problem, where $\epsilon > 0$, can be any given constant. They study relay node placement in two kinds of WSNs. In single tired WSN, both sensor nodes and relay nodes can receive and forward data packets from neighbor node. In two tired WSN, relay nodes can receive and forward data packets, while sensor nodes do not forward data packets; they only receive and transmit sensed information to the relay nodes. For solution of problem one, they create a spanning tree from given sensor nodes. Then they put relay nodes on every edge of that spanning tree after every R distance to make the network connected, where R is the communication range of relay nodes. For the solution of problem two, they have applied the Disk Cover algorithm of [8] to find out a set of relay nodes which can cover whole WSN. Then they have applied the *Steiner Minimum tree with minimum number of Steiner points and bounded* *edge length* algorithm (SMT-MSPBEL) [6], on relay nodes found in step one from Disk Cover algorithm to get the final output of required relay nodes. Lloyd et al. do not incorporate heterogeneous property in their proposed algorithm which claims only one connecting path between two sensor nodes. We find and prove the existence of multiple paths between two sensor nodes in Lloyd's algorithm, if we consider heterogeneous property of the relay node.

In 2007, X. Chen et al. [33] study the relay node placement problem and propose two approximation algorithms with detailed performance analysis. This problem is modeled by a problem named *Steiner minimum tree with minimum number of Steiner points and bounded edge length* (SMT-MSP) which is proved as a NP-hard network optimization problem in [12]. The first algorithm proposed by Chen has a performance ratio of 3 and the second has a performance ratio of 2.5. They do not consider heterogeneous property of relay nodes in their proposed algorithms.

In 2007, Han et al. [34] address the problem of deploying relay nodes to provide fault-tolerance with higher network connectivity in heterogeneous wireless sensor networks. Depending on level of fault-tolerance, they categorize such problems as : 1) full fault-tolerant relay node placement, which aims to deploy a minimum number of relay nodes to establish k ($k \geq 1$) vertex disjoint paths between every pair of sensor and/or relay nodes; 2) partial fault-tolerant relay node placement, which aims to deploy a minimum number of relay nodes to establish k ($k \geq 1$) vertex disjoint paths only between every pair of sensor nodes. Due to different transmission radii of sensor nodes these problems are further extended to two-way paths, along which wireless communication exist in both directions; and one-way paths, along which wireless communication exist in only one direction. X. Han analyzes the problem for different level of fault-tolerance (full or partial) coupled with the different types of path (one-way and two-way). X. Han propose a $O(\sigma k^2)$ approximation algorithm for both one-way and two-way partial fault tolerance relay node placement, as well as $O(\sigma k^3)$ approximation algorithm for both one way and two way full fault tolerance relay node placement (σ is the best performance ratio of existing approximation algorithms for finding a minimum k-vertex connected spanning graph) for each of these NP-hard problems. They also show heuristic implementation of these algorithms through simulation results. They only consider different sensing ranges for sensor nodes but relay nodes' transmission range is homogeneous in their proposed algorithm which contradicts with their used title 'Heterogeneous Wireless Sensor Network'. Relay nodes might have different battery power supply or different transmission range also in practical scenario which should be considered.

Chapter 3

Saving Energy by Optimal Load Balancing in H-WSN

3.1 Overview

In this chapter, we discuss about the problem domain in details and we give our solution method. It is very challenging to balance the load to prolong the network life time with the minimum cost where different powered nodes are present in the same network. Besides we know, heterogeneous sensor and relay nodes have different radio ranges. Therefore, ensuring connectivity is also important. Unfortunately, most of the works in this area consider only homogeneous relay nodes [9, 14, 32, 33]. In [34], different range of sensor nodes has been considered but the properties like, range and power of relay nodes are not considered in their proposed algorithm. Most of the algorithms mentioned above only considered geographic position of relay nodes to enhance network life time for providing service. But if the power and different range of the relay nodes are not considered, network connectivity will be lost and some nodes will be over loaded which ultimately cause network failure. Therefore, it is very important to incorporate heterogeneous property of relay nodes and sensor nodes.

We show that, existing algorithms do not load balance for a complete heterogeneous environment. For example, the algorithm in [9], claims having only one connecting path between two sensor nodes in homogeneous environment. After incorporating heterogeneous property in [9] we show that, there exist multiple connecting paths between two sensor nodes, and we can save energy by keeping only the required number of connectivity by load balancing. We propose two different load balancing models by scheduling of relay nodes which ultimately maximize network life time with minimizing the energy consumption.

Scheduling is a very old interesting problem. It is the process of deciding how to assign resources between various tasks. Lot of work has been done on scheduling algorithms. First-come First-serve (FCFS), Shortest Job First (SJF), Round Robin (RR) and Priority based scheduling are well known CPU scheduling algorithms for allocating the CPU cycles to process/threads to achieve good performance[1, 5]. Input/Output devices are scheduled using above technique by the Operating Systems to allow multitasking. Job Shop Scheduling [16] algorithm is another familiar scheduling algorithm. We have some other scheduling algorithms called earlier deadline first (EDF) [10] and least slack time (LST) [26] scheduling for real time scheduling. But in this thesis, scheduling has been done using linear programming models which are totally different approach than other scheduling techniques.

3.2 Preliminaries and the Problem Domain

In this section, we define some terms which are necessary to describe our problem domain. We are given four positive real constants $R_{max} > R_{min} > r_{max} > r_{min} > 0$,

where, r_{max} = the maximum sensing range of a sensor node,

r_{min} = the minimum sensing range of a sensor node,

R_{max} = the maximum transmission range of a relay node, and

R_{min} = the minimum transmission range of a relay node.

In addition, we are given a set $S = \{S_1, S_2, \ldots, S_n\}$ of $n > 0$ of sensor nodes on the Euclidean plane and the goal is to deploy the minimum number of relay nodes and schedule them in such a way that we can save maximum amount of energy used by the wireless sensor network.

Sensor nodes can use location services such as those in [3, 18, 23, 35], to estimate their locations without a global positioning receiver. We model scheduling duty times by dividing operation time into multiple time slots. We assume that, the sensor/relay nodes are static and each sensor/relay node has a specific predefined identifier.

We study the relay node placement in heterogeneous environment and also apply scheduling strategy to reduce the energy consumption of relay nodes. Lloyd et al. [9] proposed a 7–approximation algorithm for homogeneous single-tired relay node placement. We apply this algorithm in heterogeneous environment. The transmission range of all sensors in Heterogeneous Wireless Sensor Network (H-WSN) is different. Thus, we replace relay node's transmission range, r by δ in [9] to adjust Lloyd's algorithm to work with heterogeneous scenario,

where, $\delta = \min\{T(u), T(v)\}$,

$T(u)$ = transmission range of sensor node u, and

$T(v)$ = transmission range of sensor node v.

Suppose, we have two sensor nodes u and v respectively then, δ represents the minimum transmission range between u and v. We also replace relay node transmission range R by R_{min}, which is the minimum transmission range among all relay nodes. After incorporating heterogeneous property in [9], we get the following algorithm for relay node placement :

Suppose, we have two sensor nodes S_i and S_j respectively then, if $\delta < ||S_i S_j|| \leq 2\delta$, then we need only one relay node to communicate between sensor S_i and S_j. Again, if $2\delta < ||S_i S_j||$, then we have to place $\left\lceil \frac{||S_iS_j|| - T(S_i) - T(S_j)}{R_{\min}} - 1 \right\rceil$ relay nodes between sensor S_i and S_j to ensure connectivity. We need not to place any relay node between sensor S_i and S_j if $||S_i S_j|| < \delta$.

where,

$||S_i S_j||$ = distance between sensor S_i and S_j,

$T(S_i)$ = transmission range of sensor node S_i,

$T(S_j)$ = transmission range of sensor node S_j, and

R_{min} = the minimum transmission range among all relay nodes in wireless sensor network.

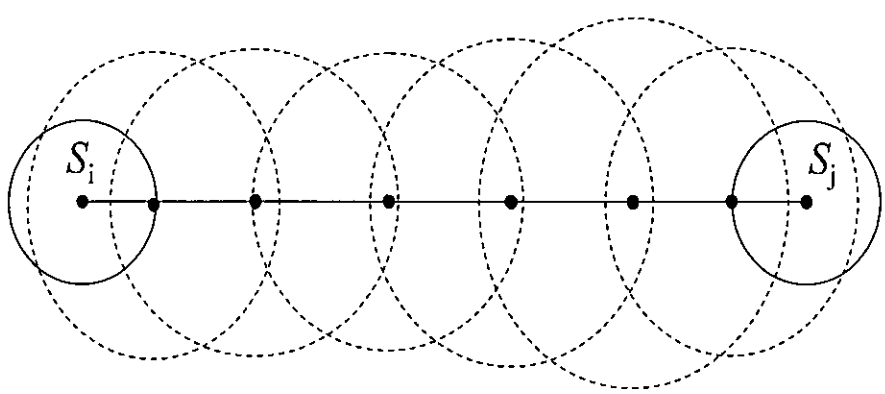

Fig- 3.1: Relay node placement between sensors S_i and S_j [10]

According to the algorithm in [9], there exists only one connecting path between each pair of sensor nodes for homogeneous wireless sensor networks. After applying this algorithm on heterogeneous wireless sensor networks (H-WSN), we observe that multiple communication paths may exist between two sensor nodes. We give a proof to support our observation.

3.3 Multiple Paths in Heterogeneous Wireless Sensor Network

***Lemma* 1.1.** There is a unique path from the root to every node of the spanning tree [29].

Here we give *Theorem* 1 and its proof by construction as follows.

***Theorem* 1.** For sensor nodes S_1 and S_2 of a given heterogeneous sensor network N of n nodes where $n>2$, and the relay node set $R = \{\alpha_1, \alpha_2, \ldots, \alpha_n\}$ where R_i is the transmission range of the relay node α_i, $R_{min} = min(R_1, R_2, \ldots, R_n)$ and any $R_i \geq R_{min}$ and the relay node α_i is placed at Steiner points between S_1 and S_2, then there are multiple transmission paths from S_1 to S_2.

***Proof*:** Suppose, S_1, S_2 and S_3 are given sensor nodes in a sensor network. At first, we construct a spanning tree T of the given sensor network, and then we put relay node on the Steiner points. We know that, a spanning tree contains exactly one path between any two nodes. Thus, we have only one path from S_1 to S_2 and S_1 to S_3 of T.

Now, we observe the following cases.

__Case 1.__ $||S_1 S_2|| > 2\delta$ and $\delta > ||S_1 S_3|| < 2\delta$, where $\delta = min(r_1, r_2)$ and r_i is transmission range of S_i sensor where, $i = 1, 2, \ldots, n$. In this case, we need more than two relay nodes to communicate between S_1 and S_2 (for simplicity we use three relay nodes in Fig - 3.2). In Fig - 3.2, we have another sensor node S_3 where, $\delta < ||S_1 S_3|| < 2\delta$. Again S_1 and S_3 can communicate between themselves using relay node α_4 only according to the algorithm [9]. Here, position of S_1, S_2, S_3, α_1, α_2, α_3 and α_4 are denoted by p, r, q, a, b, c and d respectively in Fig - 3.2.

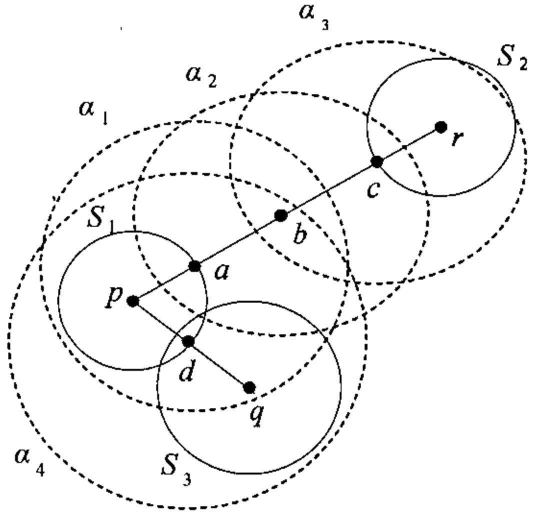

Fig - 3.2: *Case* 1 : $||S_1 S_2|| > 2\delta$ and $\delta > ||S_1 S_3|| < 2\delta$

If all relay nodes' transmission ranges are equal to R_{min}, then there exists only one path between S_1 to S_2 and S_1 to S_3 according to the algorithm of [9]. But in case of heterogeneous network, multiple paths may exist between two sensors. In Fig - 3.2, α_1, α_2, α_3 and α_4 are relay nodes deployed according to algorithm [9] where, R_{min} = minimum transmission range of relay nodes and $R_4 >> R_{min}$.

Here, $\|p\ b\| = \delta + R_{min}$

$$< R_{min} + R_{min};\ [\ \delta < R_{min}\ ]$$

$$< 2R_{min}$$

and $\|p\ d\| < \delta < R_{min}$; $[\|p\ q\| < 2\delta$ and 'd' is the midpoint of $\|p\ q\|$]

In $\Delta\, pbd$, if $\angle bpd \leq 90^\circ$ then,

$$\|d\ b\| \leq \sqrt{\|p\ b\|^2 + \|p\ d\|^2}$$

$$< \sqrt{4R_{min}^{\ 2} + R_{min}^{\ 2}}$$

$$\approx 2.2R_{min}$$

If $\angle bpd > 90^\circ$ then, S_2 and S_3 are not in same direction from S_1. In worse case $\angle bpd = 180^\circ$ then,

$$\|d\ b\| = \delta + \delta + R_{min}$$

$$< R_{min} + R_{min} + R_{min}$$

$$= 3R_{min}$$

Thus, it is possible for relay node, α_4 to cover the point 'b' in Fig – 3.1, if $R_4 >> R_{min}$. Hence, there is another path [path : *p-d-b-c-r*] from S_1 to S_2 using relay node, α_4.

<u>Case 2.</u> $\|S_1\ S_2\| > 2\delta$ and $\|S_1\ S_3\| > 2\delta$, we need more than two relay nodes to communicate between S_1 and S_2. Similarly at least three relay nodes require for communicating between S_1 and S_3 (Fig - 3.3). In this case, if $R_4 >> R_{min}$ then we can prove in similar way of *Case*1 that, there is another path [path : *p-d-b-c-r*] from S_1 to S_2 using relay node α_4.

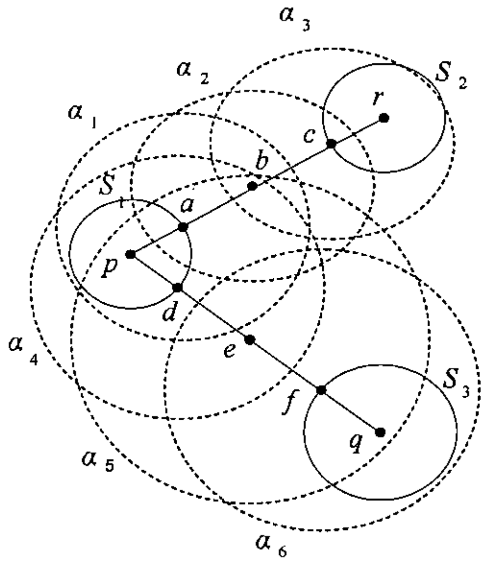

Fig - 3.3 : *Case* 2 : $||S_1 S_2|| > 2\delta$ and $||S_1 S_3|| > 2\delta$

Case 3. $||S_1\ S_2|| > 2\delta$ and $||S_1\ S_3|| < \delta$, we need more than two relay nodes to communicate between S_1 and S_2 . But no relay node is required to communicate between S_1 and S_3. In this case, S_1 sensor will communicate with S_2 via S_3, if S_2 and S_3 are in same direction from S_1.

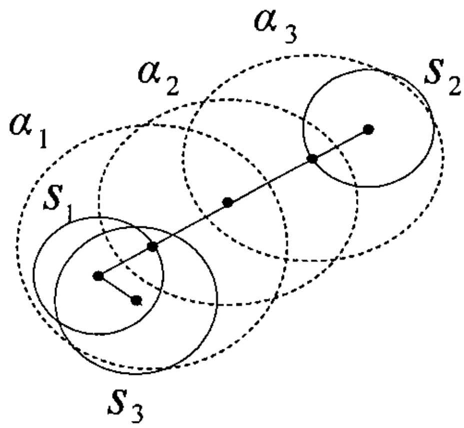

Fig - 3.4 : *Case* 3 : $||S_1 S_2|| > 2\delta$ and $||S_1 S_3|| < \delta$

Case 4. $\delta < ||S_1\ S_2|| < 2\delta$ and $||S_1\ S_3|| > 2\delta$, this case is similar to *Case* 1, if we exchange the position of S_3 and S_3. Thus, we can proof similarly there are multiple path between S_1 and S_2.

Case 5. $\delta < ||S_1 S_2|| < 2\delta$ and $\delta < ||S_1 S_3|| < 2\delta$, then we need one relay node α_1 to communicate between S_1 and S_2. Similarly S_1 and S_3 can communicate between them using one relay node α_4 (Fig - 3.5). In this case, if $R_4 >> R_{min}$ then we can prove in similar way of *Case*1 that, there exists multiple paths (*p-d-a-r* & *p-a-r*) from S_1 to S_2.

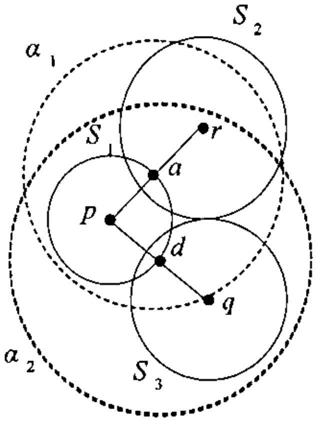

Fig - 3.5 : *Case* 5 : $\delta < ||S_1 S_2|| < 2\delta$ and $\delta < ||S_1 S_3|| < 2\delta$

Case 6. $\delta < ||S_1 S_2|| < 2\delta$ and $||S_1 S_3|| < \delta$, this case is almost similar to *Case* 3.

Case 7. $||S_1 S_2|| < \delta$ and $||S_1 S_3|| > 2\delta$, this case is similar to *Case* 3.

Case 8. $||S_1 S_2|| < \delta$ and $\delta < ||S_1 S_3|| < 2\delta$, this case is similar to *Case* 6.

Case 9. $||S_1 S_2|| < \delta$ and $||S_1 S_3|| < \delta$, in this case no relay node required for communication because all sensor nodes are within δ distance to communicate with each other (Fig - 3.6). In this case, there are multiple paths (*p-r* & *p-q-r*) available to communicate between S_1 and S_2.

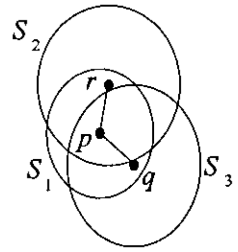

Fig - 3.6 : *Case* 9 : $||S_1 S_2|| < \delta$ and $||S_1 S_3|| < \delta$

Thus, from all possible observations, we find multiple paths exist between a source and a sink sensor node while connecting through relay nodes according to algorithm in [9], if we consider heterogeneous property of the wireless sensor network.

3.4 Linear Programming Method and Preliminaries

Efficient relay node management requires that the active and sleeping periods to be scheduled in such a way that the minimum numbers of relay nodes are in working state ensuring network connectivity to prolong network lifetime. The main problem is to assign time slots to individual relay nodes in such a way that the minimum number of relay nodes became awake and actively transmit data. The relay node is in "sleeping" state and conserving energy when it is non-active. In the simplistic case, where M relay nodes are charged with transmitting a single point over a time period of length T, an equitable load sharing assignment would have each relay node active for time (T/M) and sleeping for time $(T-T/M)$. But it is not efficient, as different relay nodes have different battery life and transmission range. The relay node assignment problem can then be formulated as the Minimum Cost Network Flow problem. We can also map this problem into the Minimax model to maximize network life time with minimizing the energy consumption. The formulation seeks an assignment of time slots to relay nodes fulfilling the required differentiated transmission that optimizes load balancing among the relay nodes, and thus promotes energy conservation.

Linear Programming is the branch of applied mathematics which deals with the problem of minimizing or maximizing linear function over a convex polyhedron specified by linear and non-negativity constrains. Informally, linear programming determines the way to achieve the best outcome (such as maximum profit or lowest cost) in given mathematical model given some list of requirements represented as linear equations [19, 20, 24].

In general, if c_1, c_2, ... c_n are real numbers, then the function f of real variables x_1, x_2, ... , x_n defined by [19, 20, 24] :

$$f(x_1,x_2, \dots , x_n) = c_1x_1 + c_2x_2 + \dots + c_nx_n = \sum_{j=1}^{n} c_j x_j \quad (3.1)$$

is called a *linear function.* If f is a linear function and if b is a real number, then the equation

$$f(x_1,x_2, \dots , x_n) = b \quad (3.2)$$

is called a *linear equation.*

$$f(x_1, x_2, \ldots, x_n) \leq b \tag{3.3}$$

$$f(x_1, x_2, \ldots, x_n) \geq b \tag{3.4}$$

is called *linear inequalities*. *Linear equations* and *linear inequalities* are both referred to as *linear constraints*. Finally, *linear programming* problem is the problem of maximizing (or minimizing) a *linear function* subject to a finite number of *linear constrains*.

3.4.1 The Minimax Model

In this section, we define one of our methodologies to obtain the load balancing in heterogeneous wireless sensor network. Two apparently non-linear objective functions can be converted into Linear Programming forms by tricks. The first is where we wish to :

$$\textit{minimize} \quad max(t_1, t_2, t_3, \ldots, t_n) \tag{3.5}$$

i.e. minimize the maximum of a set of decision variables. We can model this by introducing a new decision variable, say s and then

$$\textit{minimize} \quad s \tag{3.6}$$

$$\textit{subject to} \quad s \geq t_1 \tag{3.7}$$

$$s \geq t_2 \tag{3.8}$$

$$s \geq t_3 \tag{3.9}$$

.

.

.

$$s \geq t_n \tag{3.10}$$

We can see that s has to be no less than each of the t's and the minimization objective will force it down to take the value of the largest t. Unfortunately, the same trick cannot be applied where we want to *minimize* $min(t_1, t_2, t_3, \ldots, t_n)$ or *maximize* $max(t_1, t_2, t_3, \ldots, t_n)$. The above model of solving linear programming is called the *Minimax* model [4].

3.4.2 Scheduling using the Minimax Method

We propose the Minimax model to formulate the problem of maximizing network life time with minimizing the energy consumption for scheduling excess relay nodes. The formulation seeks an assignment of time slots to relay nodes that optimizes load balancing among relay nodes, and thus promotes energy conservation. The integer linear programming (ILP) formulation is given below :

The notation used in specifying the model is as follows:

- A set R of M static relay nodes; $R = \{\alpha_1, \alpha_2, \alpha_3, \dots, \alpha_M\}$.
- Transmission ranges for the individual relay nodes are $R_1, R_2, R_3, \dots, R_M$ respectively.
- There is a set S of P sensors which are transmitting; $S = \{S_1, S_2, S_3, \dots, S_P\}$.
- A set of N transmitting time periods during which the sensors are transmitting through relay nodes; $T = \{t_1, t_2, t_3, \dots, t_N\}$.
- Parameter , $r_{ik} = 1$, if sensor k is within transmission range of relay node i, otherwise, it is 0, where $i = 1, 2, \dots, M$ and $k = 1, 2, \dots, P$.
- Decision variable x_{ij}, where $x_{ij} = 1$, if relay node α_i is assigned to be active in time slot t_j; otherwise, $x_{ij} = 0$, where i = 1, 2, … , M and j = 1, 2, … , N.

We assume, one relay node is enough to adequately transmit information to the individual sensor during the time periods T.

$$\sum_{i=1}^{M} r_{ik} x_{ij} \geq 1 \tag{3.11}$$

for $k = 1, 2, \dots, P$; $j = 1, 2, \dots, N$.

Equation set (3.11) is modified to capture the excess transmission through sensor k during time period j in a variable called v_{kj}.

$$\sum_{i=1}^{M} r_{ik} x_{ij} - 1 - v_{kj} = 0 \tag{3.12}$$

for $k = 1, 2, \dots, P$; $j = 1, 2, \dots, N$.

The excess transmission of all relay nodes over all time periods is captured in a variable, z.

$$z - v_{kj} \geq 0 \tag{3.13}$$

for $k = 1, 2, \ldots, P$; $j = 1, 2, \ldots, N$.

The objective function is to minimize the variable z, subject to constraint sets (3.12) and (3.13). The optimal value of z will be less than the minimum of the v_{kj} by virtue of optimality of the solution [21]. Thus, the value of z will be exactly equal to the Minimax solution, which is the minimum of maximum excess transmission of all points over all time periods, which achieves the load balancing effect.

We describe the Minimax model by the example in Fig - 3.7. In this example, the number of relay nodes $M = 5$, the number of sensors is $P = 10$ and the number of time slots is $N = 3$.

Relay nodes = $\{\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5\}$, Time slots = $\{T_1, T_2, T_3\}$

Sensors = $\{S_1, S_2, S_3, S_4, S_5, S_6, S_7, S_8, S_9, S_{10}\}$

Parameter, r_{ik} for i = 1, 2, ..., 5. and k = 1, 2, ..., 10

	S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_8	S_9	S_{10}
α_1	1	1	0	0	0	0	1	1	0	1
α_2	0	0	1	1	0	0	0	0	1	0
α_3	0	0	0	1	1	1	0	0	0	1
α_4	1	1	1	0	0	0	1	1	1	0
α_5	1	1	0	0	1	1	0	0	1	1

Fig - 3.7 : An example with 10 sensor, 5 relay nodes and 3 time slots for the Minimax model

$$
\begin{aligned}
\textit{minimize} \quad & z \\
\textit{subject to} \quad & z \geq x_{11} + x_{41} + x_{51} - 1 \\
& z \geq x_{21} + x_{41} - 1 \\
& z \geq x_{21} + x_{31} - 1 \\
& z \geq x_{31} + x_{51} - 1 \\
& z \geq x_{11} + x_{41} - 1 \\
& z \geq x_{21} + x_{41} + x_{51} - 1 \\
& z \geq x_{11} + x_{31} + x_{51} - 1 \\
& z \geq x_{12} + x_{42} + x_{52} - 1 \\
& z \geq x_{22} + x_{42} - 1 \\
& z \geq x_{22} + x_{32} - 1 \\
& z \geq x_{32} + x_{52} - 1 \\
& z \geq x_{12} + x_{42} - 1 \\
& z \geq x_{22} + x_{42} + x_{52} - 1 \\
& z \geq x_{12} + x_{32} + x_{52} - 1 \\
& z \geq x_{13} + x_{43} + x_{53} - 1 \\
& z \geq x_{23} + x_{43} - 1 \\
& z \geq x_{23} + x_{33} - 1 \\
& z \geq x_{33} + x_{53} - 1 \\
& z \geq x_{13} + x_{43} - 1 \\
& z \geq x_{23} + x_{43} + x_{53} - 1 \\
& z \geq x_{13} + x_{33} + x_{53} - 1
\end{aligned}
$$

Fig - 3.8 : The Minimax model for the example in Fig - 3.7

3.4.3 The Minimum Cost Network Flow Model

In this section, we describe another linear programming model which can be used for determining optimal load balancing of our problem domain. Let $\boldsymbol{G} = (\boldsymbol{V}, \boldsymbol{E})$ represents a directed network defined by as set $\boldsymbol{V}$ of n nodes and a set $\boldsymbol{E}$ of m directed arcs. Each arc (i, j) has an associated cost c_{ij} that denotes the cost per unit flow on the arc. The flow cost varies linearly with the amount of flow. Each arc has an upper bound u_{ij} on the arc capacity that denotes the maximum flow on the arc and lower bound for each arc is l_{ij}. Each node i is a member of $\boldsymbol{V}$ and has a label $b(i)$ represents its supply or demand. The decision variable in the Minimum Cost Network Flow problem are the arc flows on each arc (i, j) and are represented by x_{ij} [22, 25, 30]. The Minimum Cost Network Flow model for the relay node scheduling can be formulated as follows :

Minimize $z = \sum c_{ij} x_{ij}$ for $i = 1, 2, \ldots, n$ and $j = 1, 2, \ldots n$ (3.15)

s.t. $\sum x_{ij} - \sum x_{ji} = b_i$, where as $i \varepsilon V$ and $j \varepsilon V$ (3.16)

and

$$l_{ij} \leq x_{ij} \leq u_{ij} \quad (3.17)$$

We formally define the above two constraints as follows:

Constraint (3.16) is called *flow balance constraint*. It states that the difference between the flow into a node j and the flow out of node j must equal its supply or demand of flow [22].

Constraint (3.17) is called *capacity constraint*. It states that the flow x_{ij} satisfies both these constraints. Flow x_{ij} is greater than or equal to the lower bound l_{ij} and less than or equal to the upper bound u_{ij} for arc (i, j) [22].

3.4.4 Scheduling using the Minimum Cost Network Flow Method

In this section, we describe the Minimum Cost network Flow Model for finding the minimum energy expenditure of relay nodes. We first define the different attributes and notations for our model. The notation used in specifying the model is as follows:

- There is a set of M static relay nodes, $R = \{ R_1, R_2, \ldots, R_M\}$
- A set S of k static sensors, $S = \{ S_1, S_2, \ldots, S_k \}$
- The battery life time of the individual relay nodes are $T_{R1}, T_{R2}, \ldots, T_{RM}$
- The time slot available for transmitting are $T_1, T_2, \ldots, T_N$

We assume that, if a relay node is active the sensors which are within its transmission range can transmit data though it. A given relay node R_M has sufficient battery power to be awake and transmitting for a known number of time periods, which can be scheduled at any of the N time periods. Thus, relay nodes may be turned on and off at time period boundaries. When a relay node is awake (actively transmitting), individual relay nodes contribute to transmit data to all sensors or relay nodes which are within its transmission range. Each sensor S_i has a known number of relay nodes c_i that can transmit data to S_i during a given time period T_i. More than one path of relay nodes between two sensors may occur due to heterogeneous property of relay nodes.

The objective is to schedule time period for each relay node to be awake in such a way that the total number of wake time periods summed over all sensors is minimized. This schedule minimizes the total energy required to meet all of the transmission requirements. In Fig - 3.7, the Minimum Cost Network Flow model for the problem is illustrated below for time period T_1 to T_N.

We assume all time slots are of equal length. Now, we explain the model as follows:

- The leftmost node R has an external flow of P. The rightmost node t is the target node which has flow of $-P$. Here, P is a large number which is equal to the sum of all relay nodes battery power in wireless sensor network. The value of positive sign represents the supply and the negative value represents the demand of the nodes. Intermediate node demands are zero, and they function as transshipment nodes.
- All arc costs in the model are 1 except the arc from leftmost node R to rightmost node t, which represents excess transmission arc. The cost of the arc from R to t is a large number which is greater than all other arc's cost. Minimizing total flow on leftmost arcs minimizes the number of wake time periods assigned to the relay nodes.

- The parameters such as (0,15) on the second leftmost arcs give lower and upper bounds on the arc flow. The upper bound models battery capacity in terms of the maximum number of time periods for which the relay nodes can be awake.

- The nodes labeled R_1, R_2, ... , R_M represent the individual relay nodes.

- The nodes in the tier with levels (R_i, T_j) represent individual relay node R_i at time period T_j.

- The upper bound of the arc leading form R_i is a large number to allow sufficient amount of flow through the arc to meet the demand of target node.

- The arc leading from the node (R_i, T_j) represents the available time periods in which the node can be awake. The number of arcs in this tier that support flow is limited by the capacity of the arc leading into the node.

- The nodes labeled (S_i, T_j) which represent sensor S_i at time period T_j receive the flow from their incoming arcs.

- The arc that leave (S_i, T_j) nodes have the required number of relay nodes to transmit data to sensor S_i at the time period T_j. At least one relay node is required to adequately transmit information to the individual sensors during the time periods, thus the arc have lower bound of one. Over transmission is allowed. From a computational view we replace the unlimited arc capacity with a large upper bound M. Fig - 3.8 shows the flow of the network when R_1 relay node is 'on' at T_1 time period.

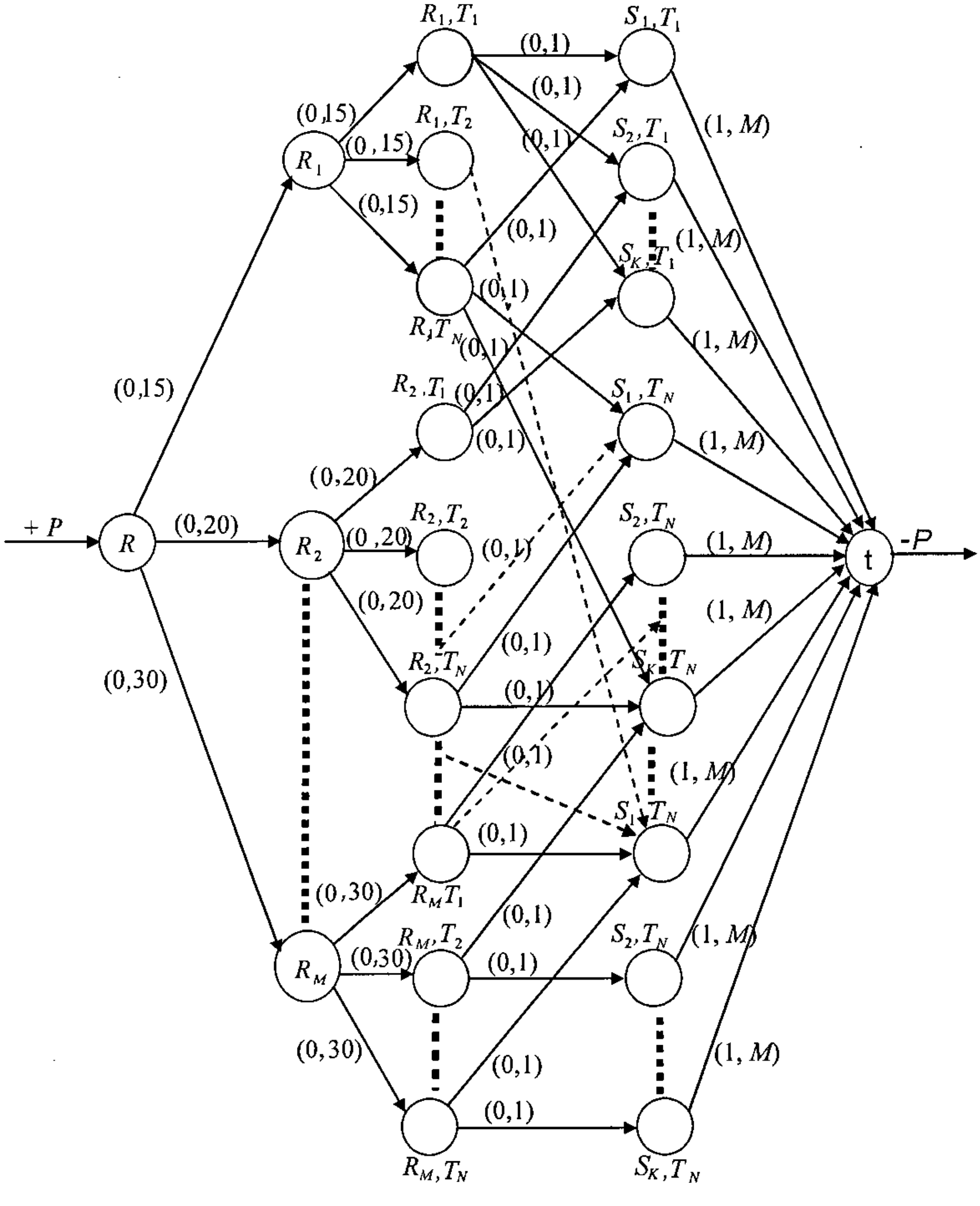

Fig - 3.9 : The Minimum Cost Network Flow model for scheduling relay nodes in a heterogeneous wireless sensor network field.

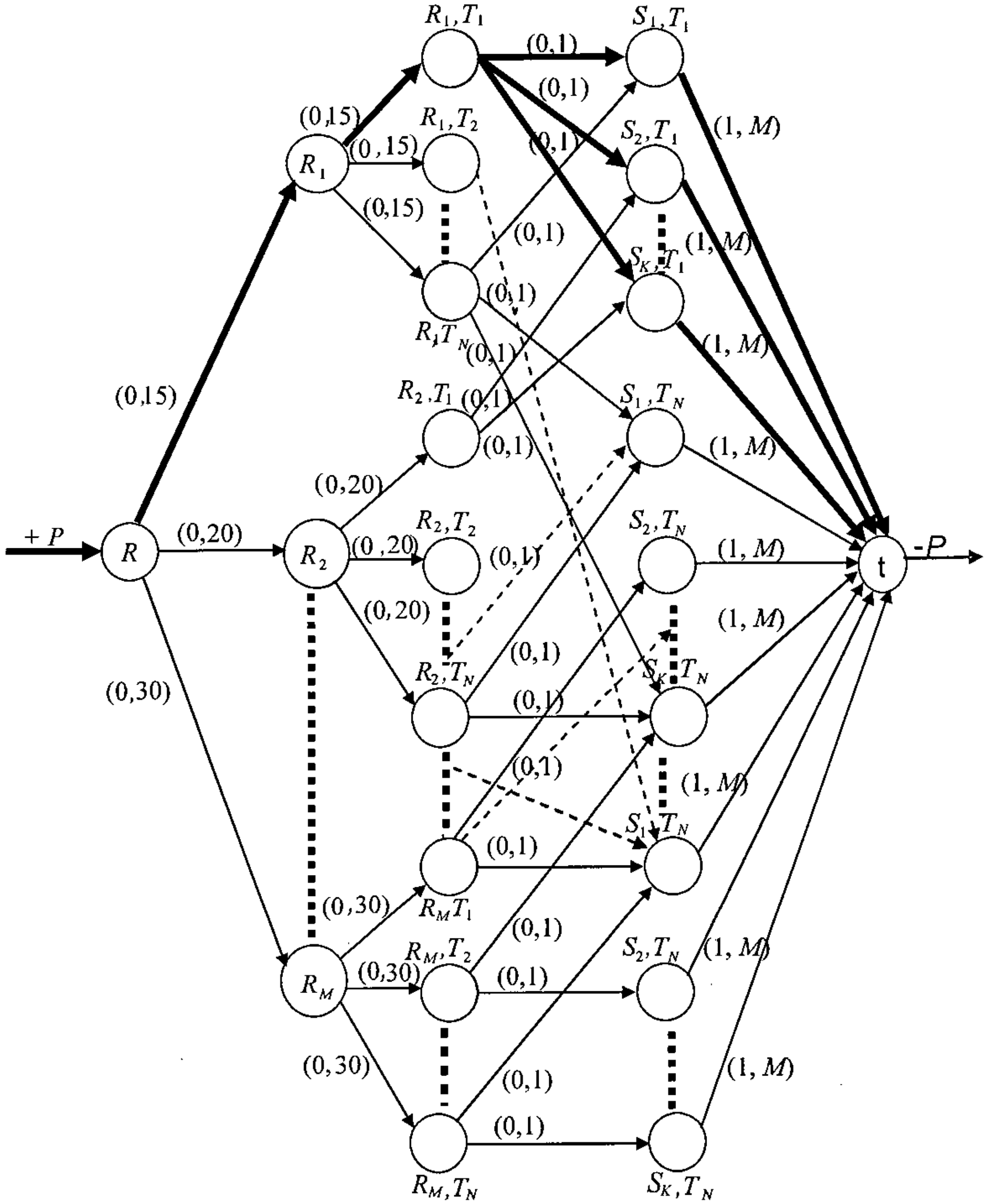

Fig - 3.10 : Data flows in the Minimum Cost Network Flow model when relay node R_1 is in active state at time period T_1

We describe the Minimum Cost Network Flow model by the following example in Fig - 3.11. In this example, the number of relay nodes M = 2, the number of sensors k = 4 and the number of time slots N = 2.

Relay nodes = $\{R_1, R_2\}$, Sensors = $\{S_1, S_2, S_3, S_4\}$, Time slots = $\{T_1, T_2\}$
Range[R_1] = $\{S_1, S_2, S_3\}$, Range[R_2] = $\{S_1, S_3, S_4\}$
Battery life time for relay nodes = $\{R_1$- 15 unit, R_2 - 20 unit$\}$

Fig - 3.11: An example with 2 relay nodes, 4 sensors and

2 time slots for the Minimum Cost Network Flow model

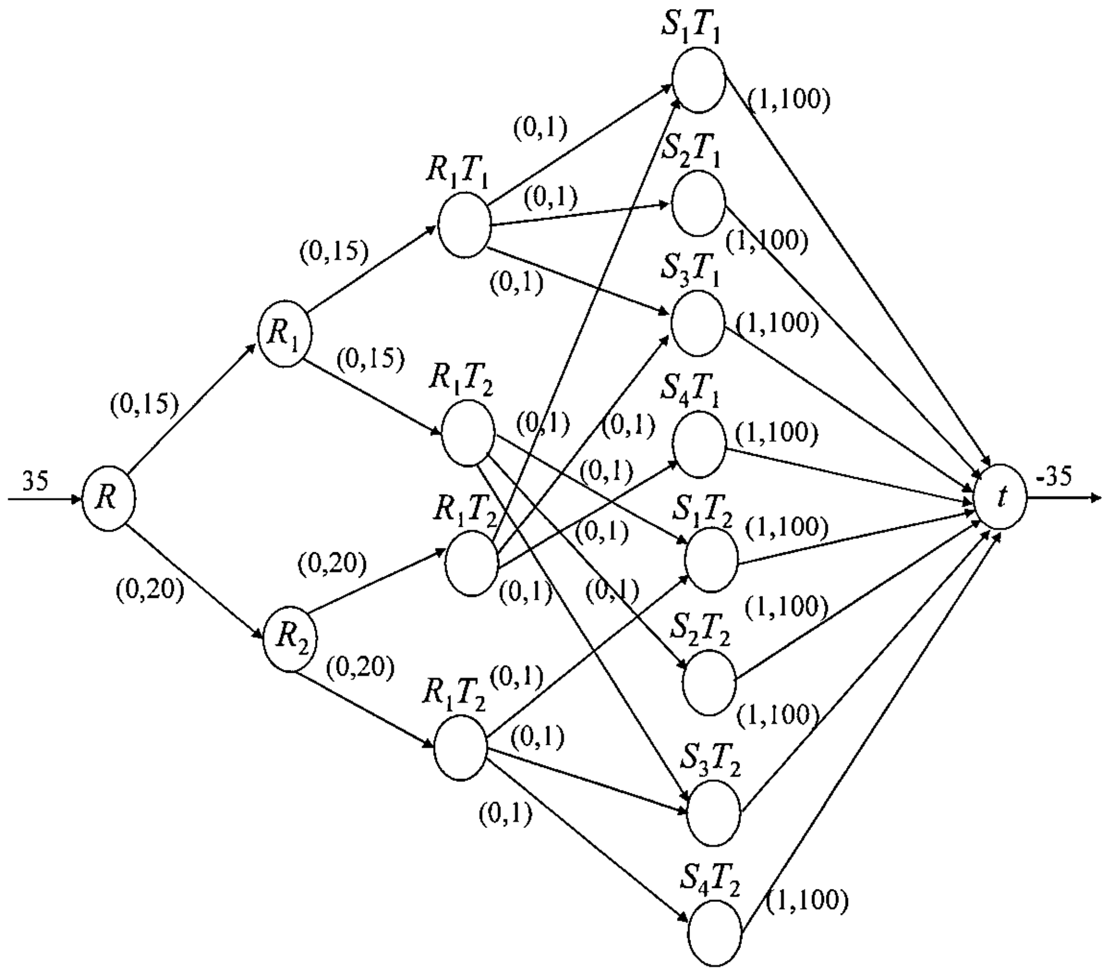

Fig - 3.12: The Minimum Cost Network Flow model for the example in Fig - 3.11

Chapter 4

Experimental Results

4.1 Overview

In order to study the performance of our proposed Minimax and Minimum Cost Network Flow models, we use *lpsolve* and *cplex* solver respectively. Data files considering different scenario of the problem domain are generated using C/C++ program. Model and data files for the Minimax and the Minimum Cost Network Flow models were executed in AMPL (Advanced Mathematical Programming Language) to generate output files. Then C/C++ codes have been used to analyze all output files generated by different data files and summary of output files have been created. If there is lack of relay nodes in sensor field to communicate with all sensors, then our program can identify and also display the sensor position which is out of transmission range through relay nodes. In this way we can put extra relay nodes into the exact position to connect whole sensor network to perform data transformation.

We have performed experiment on extensive sets of data. Both Minimax and Minimum Cost Network Flow models have been implemented using AMPL on a Pentium-IV 1.83 GHz with 1GB of RAM running Windows XP. Data files were generated using C/C++ language where relay node placement was made using a random function. Relay nodes transmission range and battery power were also determined by random function. We consider the sensing and transmission area as circular for simplicity. We map the total network area in small rectangular grid. We assumed that sensors are randomly placed on integer grid point. Relay nodes can be at any position in the grid. Our method allows any other grid shape.

4.2 Model and Data Files

We have performed experiments on an extensive set of randomly generated data files. We use 30 units spacing between two grid points. Grid size is different for every data set in our experiment. We assume that, the minimum transmission range of each relay node is 40 units and the maximum transmission range is 80 units. We also assume that, the minimum battery power for every relay node is 7 units and the maximum battery power is 17 units. The average of the results achieved from multiple data sets are presented in tables and graphs.

In data files, sensor fields were created in such a way that every grid point represents a sensor node. Time slots for scheduling relay nodes were varied from 2 to 4 in our extensive data sets. Parameters used in our experiments are given below:

Table – 4.1 : Parameters for creating data files for the Minimax model

Parameters	Value
Number of Sensor nodes	35 – 70
Number of Relay nodes	5 – 35
Transmission Range of Sensor nodes	20 – 30 units
Transmission Range of Relay nodes	40 – 80 units
Time Slots	2 – 4

Table - 4.2 : Parameters for creating data files for the Minimum Cost Network Flow model

Parameters	Value
Number of Sensor nodes	10 – 18
Number of Relay nodes	3 – 9
Transmission Range of Sensor nodes	20 – 30 unit
Transmission Range of Relay nodes	40 – 80 unit
Time Slots	2 – 4
Battery Power of Relay nodes	7 – 17 unit

In the Minimax model, three sets are declared to indicate relay nodes, sensor nodes and time slots respectively. Parameter 'cover' is used to indicate whether a sensor node is within the transmission range of a relay node or not. If the value of 'cover' parameter is 1 for a particular pair of sensor and relay node then, the sensor node is within the transmission range of that relay node. Otherwise, the value of the parameter 'cover' is 0. Decision variable *x* is 1 when a relay node is assigned for particular time slot. Otherwise the value of *x* is 0. We can calculate the amount of energy saving from analyzing the output of the decision variable x. If the decision variable is 0 then, relay node is in sleeping state and conserving energy at that time. Objective of this model to find out the minimum of maximum excess transmission of all relay nodes to save total energy consumption.

Model file written in AMPL to solve the Minimax model is given below:

```
## Minmax algorithm - Model for Relay node scheduling
set Relay;           # Set of m relay nodes
set Sensor;          # Set of k sensors
set TimeSlot; # Set of n time slots
param cover{i in Relay, j in Sensor} binary;        # = 1 if sensor j is within
                                                    # range of relay node i
var x{i in Relay, j in TimeSlot} binary;      # = 1 if relay node i is assigned
                                              # to be active in time period j,
                                              # 0 otherwise;
var u{k in Sensor, j in TimeSlot};
var z;
# Minimum of Maximum excess transmission range of all point
minimize Z: z;
subject to maxcover{k in Sensor, j in TimeSlot}:
      sum{i in Relay} cover[i,k]*x[i,j] - u[k,j] = 1;
subject to mincover{k in Sensor, j in TimeSlot}:
      z - u[k,j] >= 0 ;
subject to positive {k in Sensor, j in TimeSlot}:
      u[k,j] >= 0 ;
```

Fig - 4.1 : AMPL code for the Minimax model

A typical data file for the Minimax model with 10 sensor nodes, 5 relay nodes and 4 time slots are given below :

```
## Minimax algorithm - Data for relay node scheduling
set Relay := R0 R1 R2 R3 R4 ;
set Sensor := s0 s1 s2 s3 s4 s5 s6 s7 s8 s9 ;
set TimeSlot := t0 t1 t2 t3 ;

param cover: s0    s1    s2    s3    s4    s5    s6    s7    s8    s9:=
      R0     1     1     0     0     0     1     0     0     0     0
      R1     0     0     0     0     0     1     0     0     0     0
      R2     1     1     1     0     0     1     1     1     1     1
      R3     1     1     1     1     1     1     1     1     0     0
      R4     0     0     0     0     0     1     1     0     0     0;
```

Fig – 4.2 : Sample AMPL Minimax data file

In the Minimum Cost Network Flow model NODES and ARCS are two sets which contain the node and arc list of the model respectively. Parameter *b* which indicates supply/demand is 0 for every internal node or transshipment node in this model. Parameter *b* is positive for supply and negative for demand. Other parameters *c*, *l* and *u* contain cost, lower bound and upper bound of every arc respectively. According to our model all arcs cost is 1 except source to destination arc. The cost of source to destination arc is a big number. Decision variable x indicates the amount of flow through each arc. Objective of this model is to minimize the total cost (battery power) of the model (sensor network). Model file written in AMPL for the Minimum Cost Network Flow is given below :

```
# AMPL model for the Minimum Cost Network Flow Problem
# By default, this model assumes that b[i] = 0, c[i,j] = 0,
# l[i,j] = 0 and u[i,j] = Infinity.
# Parameters not specified in the data file will get their default values.
set NODES; # nodes in the network
set ARCS within {NODES, NODES}; # arcs in the network
param b {NODES} default 0; # supply/demand for node i
param c {ARCS} default 0; # cost of one of flow on arc(i,j)
param l {ARCS} default 0; # lower bound on flow on arc(i,j)
param u {ARCS} default Infinity; # upper bound on flow on arc(i,j)
var x {ARCS}; # flow on arc (i,j)
minimize cost: sum{(i,j) in ARCS} c[i,j] * x[i,j];
# Flow Out(i) - Flow In(i) = b(i)
subject to flow_balance {i in NODES}:
sum{j in NODES: (i,j) in ARCS} x[i,j] - sum{j in NODES: (j,i) in ARCS} x[j,i] = b[i];
subject to capacity {(i,j) in ARCS}: l[i,j] <= x[i,j] <= u[i,j];
```

Fig – 4.3 : AMPL code for the Minimum Cost Network Flow model file

A typical data file for the Minimum Cost Network Flow model with 10 sensor nodes, 4 relay nodes and 2 time slots is given below :

```
## Minimum Cost Network Flow algorithm - Data for relay node scheduling
set NODES := R R0 R1 R2 R0T0 R0T1 R1T0 R1T1 R2T0 R2T1 s0T0 s0T1 s1T0 s1T1 s2T0 s2T1
s3T0 s3T1 s4T0 s4T1 s5T0 s5T1 s6T0 s6T1 s7T0 s7T1 s8T0 s8T1 s9T0 s9T1 t;
set ARCS := (R,R0) (R0,R0T0) (R0,R0T1) (R,R1) (R1,R1T0) (R1,R1T1) (R,R2) (R2,R2T0)
(R2,R2T1) (R0T0,s0T0) (R0T1,s0T1) (R0T0,s1T0) (R0T1,s1T1) (R0T0,s2T0) (R0T1,s2T1)
(R0T0,s3T0) (R0T1,s3T1) (R0T0,s4T0) (R0T1,s4T1) (R0T0,s5T0) (R0T1,s5T1) (R0T0,s6T0)
(R0T1,s6T1) (R0T0,s7T0) (R0T1,s7T1) (R0T0,s8T0) (R0T1,s8T1) (R0T0,s9T0) (R0T1,s9T1)
(R1T0,s0T0) (R1T1,s0T1) (R1T0,s1T0) (R1T1,s1T1) (R1T0,s2T0) (R1T1,s2T1) (R1T0,s3T0)
(R1T1,s3T1) (R1T0,s4T0) (R1T1,s4T1) (R1T0,s5T0) (R1T1,s5T1) (s0T0,t) (s0T1,t) (s1T0,t)
(s1T1,t) (s2T0,t) (s2T1,t) (s3T0,t) (s3T1,t) (s4T0,t) (s4T1,t) (s5T0,t) (s5T1,t)
(s6T0,t) (s6T1,t) (s7T0,t) (s7T1,t) (s8T0,t) (s8T1,t) (s9T0,t) (s9T1,t) (R,t);
param b:=
R 29
t -29;
#           cost   lower  upper bound
param:      c      l      u :=
R R0        1      0      12
R0 R0T0     1      0      1000
R0 R0T1     1      0      1000
```

Fig – 4.4 : Sample AMPL Minimum Cost Network Flow data file

```
R R1          1     0     8
R1 R1T0       1     0     1000
R1 R1T1       1     0     1000
R R2          1     0     9
R2 R2T0       1     0     1000
R2 R2T1       1     0     1000
R0T0 s0T0     1     0     1
R0T1 s0T1     1     0     1
R0T0 s1T0     1     0     1
R0T1 s1T1     1     0     1
R0T0 s2T0     1     0     1
R0T1 s2T1     1     0     1
R0T0 s3T0     1     0     1
R0T1 s3T1     1     0     1
R0T0 s4T0     1     0     1
R0T1 s4T1     1     0     1
R0T0 s5T0     1     0     1
R0T1 s5T1     1     0     1
R0T0 s6T0     1     0     1
R0T1 s6T1     1     0     1
R0T0 s7T0     1     0     1
R0T1 s7T1     1     0     1
R0T0 s8T0     1     0     1
R0T1 s8T1     1     0     1
R0T0 s9T0     1     0     1
R0T1 s9T1     1     0     1
R1T0 s0T0     1     0     1
R1T1 s0T1     1     0     1
R1T0 s1T0     1     0     1
R1T1 s1T1     1     0     1
R1T0 s2T0     1     0     1
R1T1 s2T1     1     0     1
R1T0 s3T0     1     0     1
R1T1 s3T1     1     0     1
R1T0 s4T0     1     0     1
R1T1 s4T1     1     0     1
R1T0 s5T0     1     0     1
R1T1 s5T1     1     0     1
s0T0 t        1     1     1000
s0T1 t        1     1     1000
s1T0 t        1     1     1000
s1T1 t        1     1     1000
s2T0 t        1     1     1000
s2T1 t        1     1     1000
s3T0 t        1     1     1000
s3T1 t        1     1     1000
s4T0 t        1     1     1000
s4T1 t        1     1     1000
s5T0 t        1     1     1000
s5T1 t        1     1     1000
s6T0 t        1     1     1000
s6T1 t        1     1     1000
s7T0 t        1     1     1000
s7T1 t        1     1     1000
s8T0 t        1     1     1000
s8T1 t        1     1     1000
s9T0 t        1     1     1000
s9T1 t        1     1     1000
R t           2     0     1000;
```

Fig – 4.4 : Sample AMPL Minimum Cost Network Flow data file (Continued)

4.3 Simulation Results

It is observed that, in the Minimax model relay nodes are in sleeping state for on an average $\frac{32+31.1+32.1}{3}$ = 31.7% of total time. It means that, sensor network can save 31.7% of their total energy according to this model. We incorporate battery power in the Minimum Cost Network Flow model. Our test result shows on an average $\frac{26.7+14.9+7.5}{3}$ = 16.4% of the total battery power can be saved in the Minimum Cost Network Flow model. We present the experimental results in tables and graphs.

Tables 4.3, 4.4 and 4.5 show the percentage of relay nodes are in sleeping state for the Minimax model with time slots 2, 3 and 4 respectively in different sensor fields. Tables 4.6, 4.7 and 4.8 show the percentage of relay nodes are in sleeping state for the Minimum Cost Network Flow model with time slots 2, 3 and 4 respectively in different sensor fields.

The graphs from Fig - 4.5, 4.6 and 4.7 shows the percentage of relay nodes are conserving energy in the Minimax model for time slots 2, 3 and 4 respectively in different sensor fields. Fig - 4.8, 4.9 and 4.10 shows the percentage of time relay nodes are conserving energy in the Minimum Cost Network Flow model for time slots 2, 3 and 4 respectively in different sensor fields. Fig - 4.11 shows the comparison between average percentages of time relay nodes in sleeping state for different time slots in the Minimum Cost Network Flow model. Data sets of the Minimum Cost Network Flow model is smaller compared to the Minimax model, because small sensor field in the Minimum Cost Network Flow model creates large amount of arcs and nodes which is difficult to solve. We have used student version of AMPL for simulation which has restriction of 300 variables. Even medium size data sets in the Minimax model takes a lot of time and more than 1 billion iterations to find out the optimal solution for single run. Thus we could not take large data sets for calculating optimal solution of both models due to time and restriction of the student version of AMPL. The percentage of relay nodes in sleeping state is calculated using the following formula.

$$\text{Percentage of relay nodes in sleeping state} = \frac{(\text{Total number of relay nodes} - \text{Number of active relay nodes})}{\text{Total number of relay nodes}} \times 100$$

Scheduling technique has been applied on total relay nodes to find out active relay nodes in particular time period. All the plotted data in the above mentioned graphs are the average of 5 data sets.

In Table 4.3, for example, the ratio of relay and sensor nodes 20% means there exist 20 relay nodes for every 100 sensor nodes in that particular sensor field. When some relay nodes are in sleeping state in particular time, they can conserve energy for the whole network. The percentage of relay nodes are in sleeping state has been calculated for the number of sensor nodes 35, 40, 45, 50, 55, 60, 65 and 70 respectively. The average percentage of relay nodes are in sleeping state for a particular time period is shown at the right column of the Table 4.3. Two equal time slots are used in Table 4.3 to schedule all relay nodes which result on an average 32% of relay nodes are in sleeping state i.e. conserving energy. The average percentage of relay nodes are in sleeping sate is zero (0) when the ratio of relay and sensor nodes is less than or equal to 10% i.e. the number of deployed relay nodes is not sufficient to make the whole network connected. It is observed that, the relay nodes can be scheduled when the ratio of relay and sensor nodes are more than or equal to 15%. The average 4.7% of relay nodes are in sleeping state when relay and sensor nodes ratio is 15%. The average of the average percentage of sleeping relay nodes for the Minimax model using two time slots is shown in the bottom row of the table 4.3.

Table - 4.3 : Percentage (%) of relay nodes in sleeping state for different sensor network in the Minimax model (Time Slot = 2)

Ratio of Relay and Sensor nodes (%)	Total Number of Sensor Nodes in Sensor Network (% of relay nodes are in sleeping state) **Time Slot = 2**								Average % of Sleeping Relay nodes
	35 nodes	40 nodes	45 nodes	50 nodes	55 nodes	60 nodes	65 nodes	70 nodes	
0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0
15	3.2	4.1	4.7	5	6	4.8	5.2	4.2	4.7
20	13.6	17	20	22.1	21.9	17.7	22	24.9	19.9
25	24.6	30	33.6	36.6	37.3	37.3	40	43.8	35.4
30	36.1	42.1	44	45.9	47	47.6	50.1	50.9	45.5
35	45.7	50.7	51.8	53.9	57.9	57	60.4	56.8	54.3
40	52.4	54	59	57.8	63.7	61.9	63.5	64.2	59.6
45	58.1	60.7	61.2	61.4	65.2	65.9	67.4	68.5	63.6
50	61.9	63.5	66	67.8	69.4	72	73.5	74.6	68.6
The average percentage of relay nodes are in sleeping state =									32

In Fig - 4.5, we observe that, when relay node's ratio is greater than 10% to sensor nodes, we have some extra relay nodes to schedule. When the ratio of relay nodes and sensor nodes is 15%, then we notice that approximately 5% relay nodes are in sleeping state to conserve energy for the whole network. When we increase the number of deploying relay nodes in a sensor field, we observe more relay nodes are available to communicate between sensors. Thus we can schedule more relay nodes and at the same time we get more relay nodes are in sleeping state for particular sensor field. For this reason, we observe in fig - 4.5, the relay node's sleeping time is increasing with the increment of the ratio of relay nodes in a sensor field. The highest percentage of the relay nodes are in sleeping state is 75%, when we put 70 sensor nodes and 35 relay nodes in a sensor field. Again, if we deploy less than 10% relay nodes, then the graph shows infeasible solution i.e. the number of relay nodes is insufficient to transmit data to all sensor nodes and we need more relay nodes to communicate with all sensor nodes.

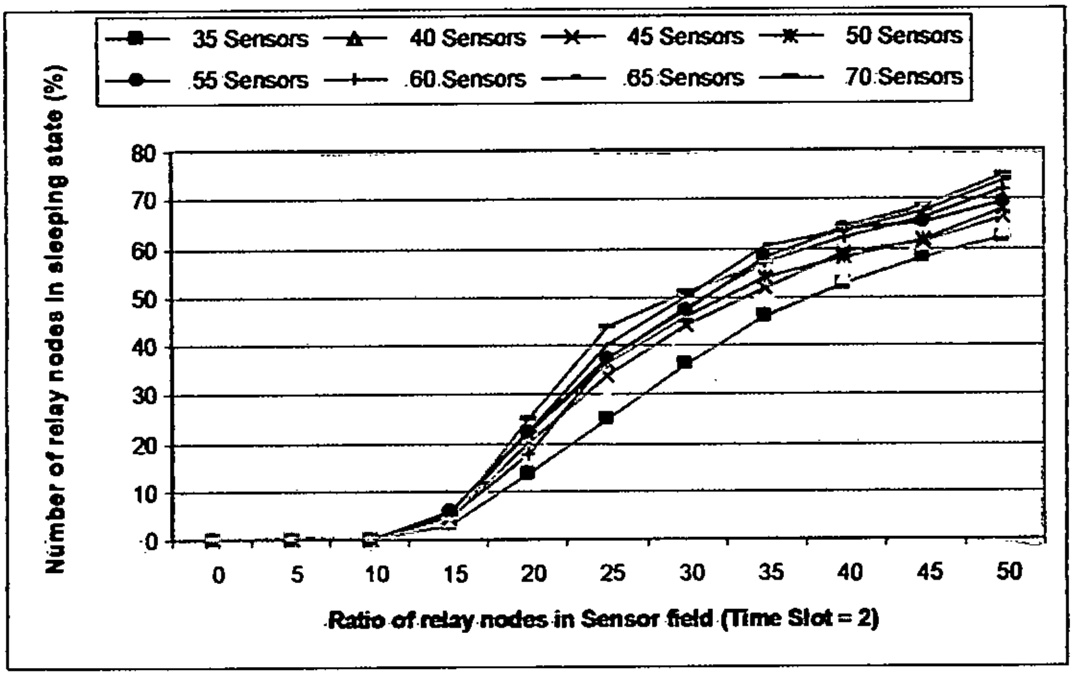

Fig - 4.5 : Percentage (%) of relay nodes in sleeping state for different sensor network in the Minimax model (Time Slot = 2)

In Table 4.4, we use three time slots to calculate the average percentage of relay nodes in sleeping state for different sensor field. The table shows the percentage of sleeping relay nodes are increasing with the increment of sensor field size. The percentage of relay nodes are in sleeping state has been calculated for the number of sensor nodes 35, 40, 45, 50, 55, 60, 65 and 70 respectively. We observe the highest 72% of relay nodes are in sleeping stage when we put 70 sensor nodes along with 35 relay nodes to transmit data between all sensors. The average sleeping time is 31.1% for the Mini-max model when we consider three time slots for scheduling all relay nodes. According to the table 4.4, the average percentage of relay nodes are in sleeping sate is zero (0) when the ratio of relay and sensor nodes is less than or equal to 10%. When the relay nodes are in sleeping state, they are conserving energy to prolong network life time. It is observed that, the relay nodes can be scheduled when the ratio of relay and sensor nodes are more than or equal to 15%. The average 4.5% of relay nodes are in sleeping state when relay and sensor nodes ratio is 15%. The average of the average percentage of sleeping relay nodes for the Minimax model using three time slots is shown in the bottom row of the table 4.4.

Table - 4.4 : Percentage (%) of relay nodes in sleeping state for different sensor network in the Minimax model (Time Slot = 3)

Ratio of Relay and Sensor nodes (%)	Total Number of Sensor Nodes in Sensor Network (% of relay nodes are in sleeping state) **Time Slot = 3**								Average % of Sleeping Relay nodes
	35 nodes	40 nodes	45 nodes	50 nodes	55 nodes	60 nodes	65 nodes	70 nodes	
0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0
15	3.1	4.3	3.2	4.5	6.5	4.7	5.1	4.1	4.5
20	11.1	15.6	18	20	23.3	21.9	23.4	24.7	19.8
25	23.5	28	33.9	35.2	34.7	35.3	40.9	43.6	34.4
30	35.2	40.1	44	47.3	43.6	44.3	48.1	50.1	44.1
35	45.5	50.3	49.7	55.4	51.3	51.1	54.7	57.5	52
40	53.3	57	52	59.2	57.1	56.1	61.2	63.5	57.5
45	59.4	59.7	59.7	61.6	60.9	64.9	65.4	68.9	62.6
50	62.6	62	65.2	66	67.2	69.3	70.3	72	66.9
The average percentage of relay nodes are in sleeping state =									31.1

Fig - 4.6 shows the percentage of relay nodes in sleeping state for the Minimax model in different sizes of sensor fields. In our experiment, the number of deployed sensors in sensor field ranges from 35 to 70. If we deploy less than or equal to 10% relay nodes comparing to sensor nodes, then the graph shows 0% percent of relay nodes in sleeping state i.e. we need more relay node to make the whole network connected through relay nodes for data transmission. When the ratio of relay nodes and sensor nodes is 15%, then we notice that approximately 5% relay nodes are in sleeping state to conserve energy for the whole network. For the similar reason of fig - 4.5, the graph of fig - 4.6 shows the relay node's sleeping time is increasing with the increment of the ratio of relay nodes in a sensor field. The highest percentage of the relay nodes are in sleeping state is 72%, when we put 70 sensor nodes and 35 relay nodes in a sensor field.

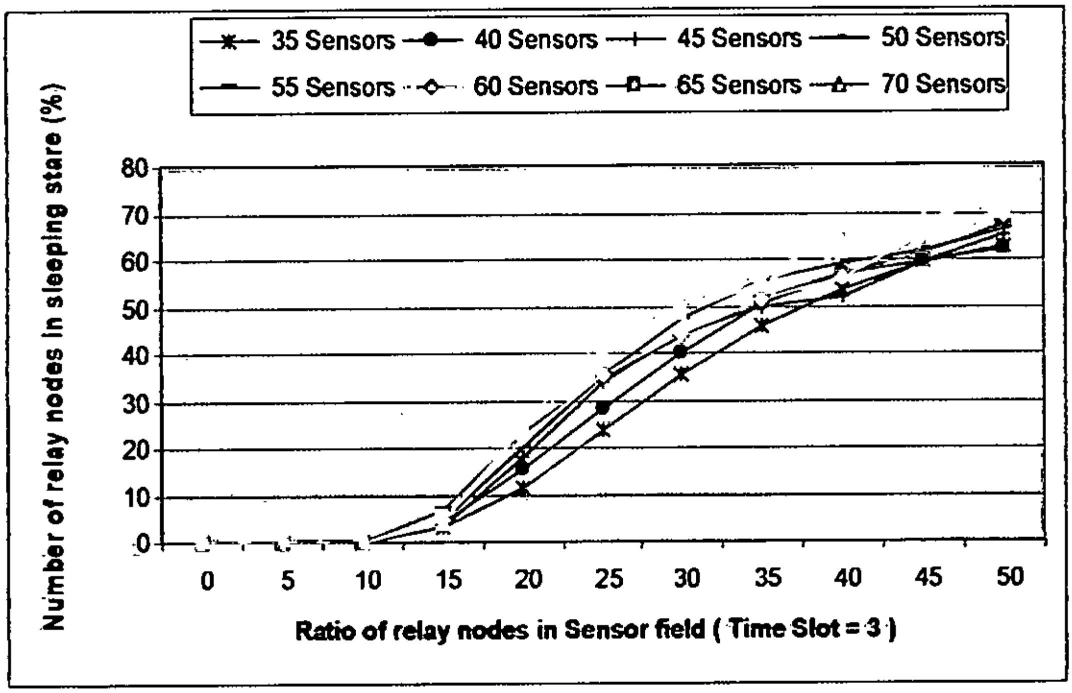

Fig - 4.6 : Percentage (%) of relay nodes in sleeping state for different sensor network in the Minimax model (Time Slot = 3)

In Table 4.4, we use four time slots to calculate the average percentage of relay nodes in sleeping state for different sensor field. The table shows the percentage of sleeping relay nodes are increasing with the increment of sensor field size. It is observed from the table, the maximum 74.3% of relay nodes in sleeping stage when we put 70 sensor nodes along with 35 relay nodes to transmit data between all sensors. The percentage of relay nodes are in sleeping state has been calculated for the number of sensor nodes 35, 40, 45, 50, 55, 60, 65 and 70 respectively. According to the table 4.5, the average sleeping time is 32.1% for the Mini-max model when we consider four time slots for scheduling all relay nodes. The average percentage of relay nodes are in sleeping sate is zero (0) when the ratio of relay and sensor nodes is less than or equal to 10%. When the relay nodes are in sleeping state, they are conserving energy to prolong network life time. It is observed that, the relay nodes can be scheduled when the ratio of relay and sensor nodes are more than or equal to 15%. The average 4.7% of relay nodes are in sleeping state when relay and sensor nodes ratio is 15%. The average of the average percentage of sleeping relay nodes for the Minimax model using four time slots is shown in the bottom row of the table 4.5.

Table – 4.5 : Percentage (%) of relay nodes in sleeping state for different sensor network in the Minimax model (Time Slot = 4)

Ratio of Relay and Sensor nodes (%)	Total Number of Sensor Nodes in Sensor Network (% of relay nodes are in sleeping state) **Time Slot = 4**								Average % of Sleeping Relay nodes
	35 nodes	40 nodes	45 nodes	50 nodes	55 nodes	60 nodes	65 nodes	70 nodes	
0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0
15	3.1	4.6	4.1	5.1	7.1	4.5	5	4	4.7
20	10.3	16.6	20.1	21	22.5	18.2	21	22.8	19.1
25	21.8	29	33.5	33.8	36.3	40.6	41.4	45.6	35.3
30	33.1	40.4	44.9	45	48.2	50.2	53	55.1	46.3
35	43.4	49.3	54	53.4	56.5	57.8	58.9	59.6	54.2
40	51.7	56.4	58.7	58	60.3	61.5	62.9	64.5	59.3
45	58.3	59.8	62.2	63.9	65.1	67.6	69.4	70.8	64.7
50	62.3	63	67.1	69	70.7	72.3	72.6	74.3	69
The average percentage of relay nodes are in sleeping state =									32.1

Fig - 4.7 shows the percentage of relay nodes in sleeping state for the Minimax model in different sizes of sensor fields. In our experiment, the number of deployed sensors in sensor field ranges from 35 to 70. If we deploy less than or equal to 10% relay nodes comparing to sensor nodes, then the graph shows 0% percent of relay nodes in sleeping state. When the ratio of relay nodes and sensor nodes is 15%, then we notice that approximately 5% relay nodes are in sleeping state to conserve energy for the whole network. For the similar reason of fig - 4.5, the graph of fig - 4.7 shows the relay node's sleeping time is increasing with the increment of the ratio of relay nodes in a sensor field. The highest percentage of the relay nodes are in sleeping state is approximately 75%, when we put 70 sensor nodes and 35 relay nodes in a sensor field.

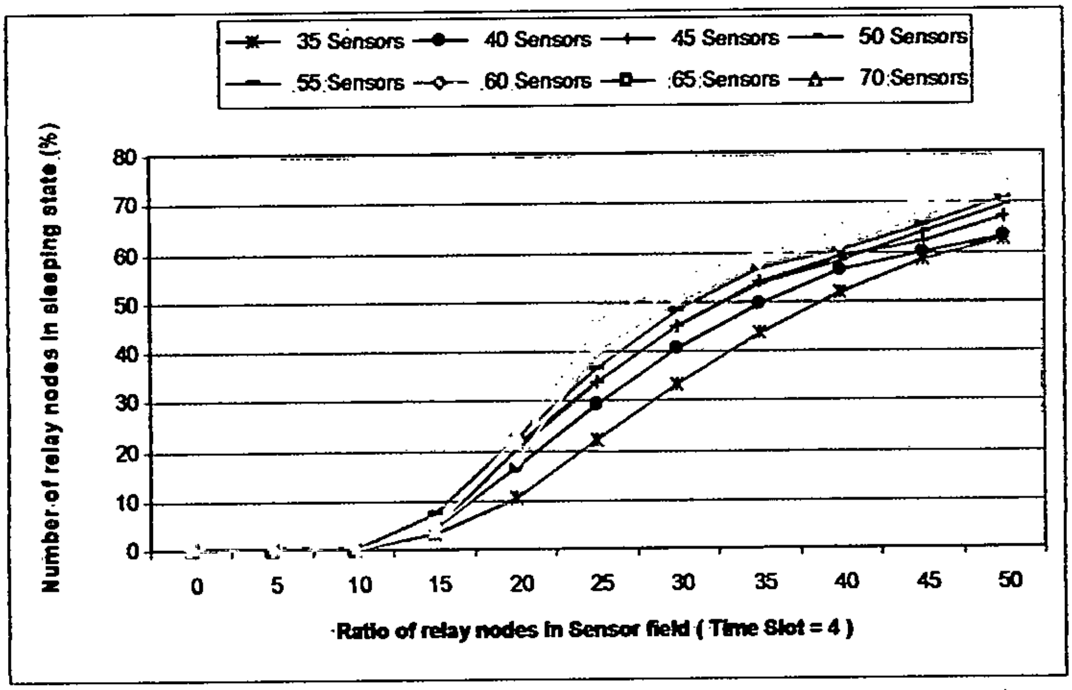

Fig - 4.7 : Percentage (%) of relay nodes in sleeping state for different sensor network in the Minimax model (Time Slot = 4)

Table 4.6 shows the result of the Minimum Cost Network Flow model for two equal time slots. It presents the percentage of relay nodes in sleeping state for different size of sensor fields. Right column of the table shows the average number of relay nodes in sleeping state according to our given model. When relay nodes are in sleeping state, they are saving battery power which prolongs the network life time. Thus we can refer the amount of relay nodes are in sleeping state as energy saving. The percentage of relay nodes are in sleeping state has been calculated for the number of sensor nodes 10, 12, 14, 16 and 18 respectively. From the table 4.6, we can observe that, the highest amount of battery power is saved when we put 8 relay nodes in sensor field size of 16 sensors i.e. the ratio of relay and sensor node is 50%. In this case, the highest amount of battery saving is recorded as 67.3%. If we consider two time slots, the average battery power saving observed in Minimum Cost Network Flow model is 26.7%. The average percentage of relay nodes are in sleeping sate is zero (0) when the ratio of relay and sensor nodes is less than or equal to 10%. When the relay nodes are in sleeping state, they are conserving energy to prolong network life time. It is observed that, the relay nodes can be scheduled when the ratio of relay and sensor nodes are more than or equal to 15%. The average 2.7% of relay nodes are in sleeping state when relay and sensor nodes ratio is 15%. The table shows the percentage of sleeping relay nodes are increasing with the increment of sensor field size and the ratio of relay and sensor nodes. The average of the average percentage of sleeping relay nodes for the Minimum Cost Network Flow model using two time slots is shown in the bottom row of the table 4.6.

Table - 4.6 : Percentage (%) of relay nodes in sleeping state for different sensor network in the Minimum Cost Network Flow model (Time Slot = 2)

Ratio of Relay and Sensor nodes (%)	Total Number of Sensor Nodes in Sensor Network (% of relay nodes are in sleeping state) **Time Slot = 2**					Average % of Sleeping Relay nodes
	10 nodes	12 nodes	14 nodes	16 nodes	18 nodes	
0	0	0	0	0	0	0
5	0	0	0	0	0	0
10	0	0	0	0	0	0
15	5	3	1.5	1.6	2	2.7
20	13.2	13.1	11.5	5.2	7	10
25	26.4	25.5	18.4	13.7	16.6	20.2
30	38.3	31.4	27.9	33.9	39.4	34.2
5	46.2	39.3	44.1	49.2	51.4	46.1
40	46.1	49.6	53.9	58	59.3	53.4
45	54	60.4	63.2	63.9	63.5	61
50	61.7	65.3	67.3	67.3	64.5	65.3
The average percentage of relay nodes are in sleeping state =						26.7

Fig - 4.8 shows the percentage of relay nodes in sleeping state for the Minimum Cost Network Flow model in different sizes of sensor fields. In our experiment, the number of deployed sensors in sensor field ranges from 10 to 18. We observe that, when relay node's ratio is grater than 10% to sensor nodes, we have some extra relay nodes for scheduling. When we have 15% relay nodes, we observe approximately 3% of time relay nodes are in sleeping state to conserve energy for the whole network. For the similar reason of fig - 4.5, the graph of fig - 4.8 shows the relay node's sleeping time is increasing with the increment of the ratio of relay nodes in a sensor field. If we deploy 10 or less relay nodes in sensor field size of 100 sensor nodes, then the graph shows infeasible solution in relay node placement in sensor field i.e. relay node's number is insufficient to transmit data to all sensor nodes and we need more relay nodes to communicate between all sensor nodes. The highest percentage of the relay nodes are in sleeping state is approximately 68%, when we put 16 sensor nodes and 8 relay nodes in a sensor field.

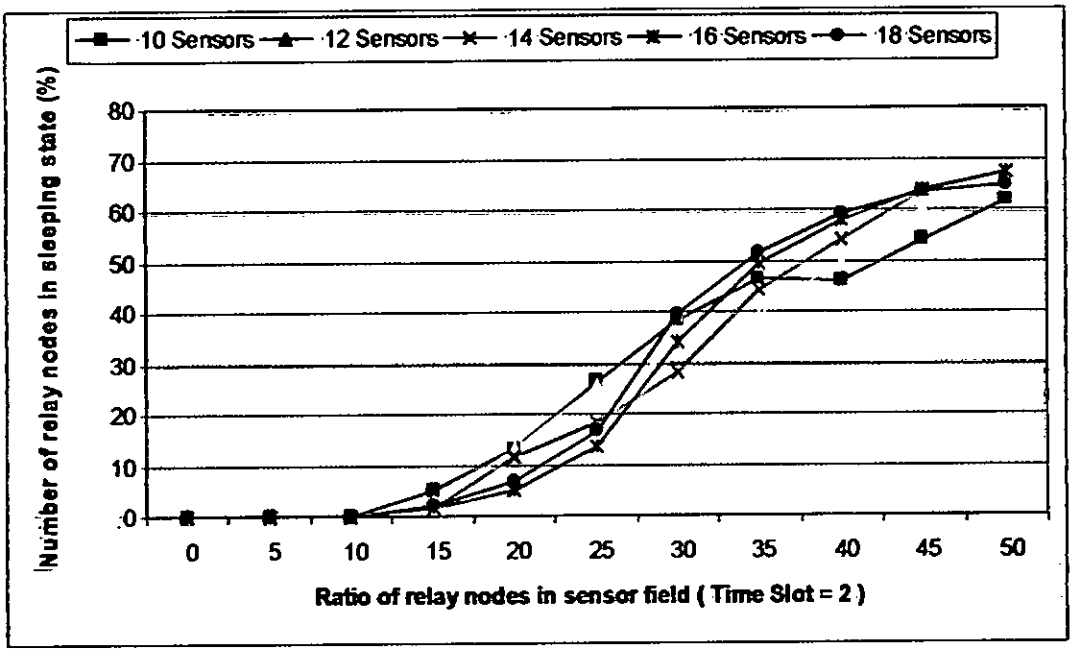

Fig - 4.8 : Percentage (%) of relay nodes in sleeping state for different sensor network in the Minimum Cost Network Flow model (Time Slot = 2)

Table 4.7 demonstrates the result of the Minimum Cost Network Flow model for three equal time slots. It represents the percentage of relay nodes in sleeping state for different number of sensors in a sensor field. The percentage of relay nodes are in sleeping state has been calculated for the number of sensor nodes 10, 12, 14, 16 and 18 respectively. The highest amount of battery power is saved when we put 7 relay nodes in a sensor field of 14 sensors i.e. the ratio of relay and sensor node is 50%. In this case, the highest amount of battery saving recorded as 52%. The average battery saving observed in Minimum Cost Network Flow model for three time slots is 14.9%. The average percentage of relay nodes are in sleeping sate is zero (0) when the ratio of relay and sensor nodes is less than or equal to 20%. On an average 1.8% of relay nodes are in sleeping state when relay and sensor nodes ratio is 25%. The table shows the percentage of sleeping relay nodes are increasing with the increment of sensor field size and the ratio of relay and sensor nodes. The average of the average percentage of sleeping relay nodes for the Minimum Cost Network Flow model using three time slots is shown in the bottom row of the table 4.7.

Table - 4.7 : Percentage (%) of relay nodes in sleeping state for different sensor network in the Minimum Cost Network Flow model (Time Slot = 3)

Ratio of Relay and Sensor nodes (%)	Total Number of Sensor Nodes in Sensor Network (% of relay nodes are in sleeping state) **Time Slot = 3**					Average % of Sleeping Relay nodes
	10 nodes	12 nodes	14 nodes	16 nodes	18 nodes	
0	0	0	0	0	0	0
5	0	0	0	0	0	0
10	0	0	0	0	0	0
15	0	0	0	0	0	0
20	0	0	0	0	0	0
25	4.6	1.1	0.9	1	1.1	1.8
30	11.1	12.7	7.1	9.3	12.1	10.5
35	19.6	26	21.2	26.4	29.7	24.6
40	24.5	31.2	33.4	40.1	40.4	34
45	35.1	41.1	46	46.7	46.5	43.1
50	45.2	49.6	52	50.9	46.7	48.9
The average percentage of relay nodes are in sleeping state =						14.9

Fig - 4.9 shows the percentage of relay nodes in sleeping state for the Minimum Cost Network Flow model in different sizes of sensor fields. In our experiment, the number of deployed sensors in sensor field ranges from 10 to 18. We observe that, when relay node's ratio is grater than 20% to sensor nodes, we have some extra relay nodes for scheduling. When we have 25% relay nodes, we observe approximately 2% of time relay nodes are in sleeping state to conserve energy for the whole network. For the similar reason of fig - 4.5, the graph of fig - 4.9 shows the relay node's sleeping time is increasing with the increment of the ratio of relay nodes in a sensor field. If we deploy 20 or less relay nodes in sensor field size of 100 sensor nodes, then the graph shows infeasible solution in relay node placement in sensor field The highest percentage of the relay nodes are in sleeping state is approximately 52%, when we put 14 sensor nodes and 7 relay nodes in a sensor field.

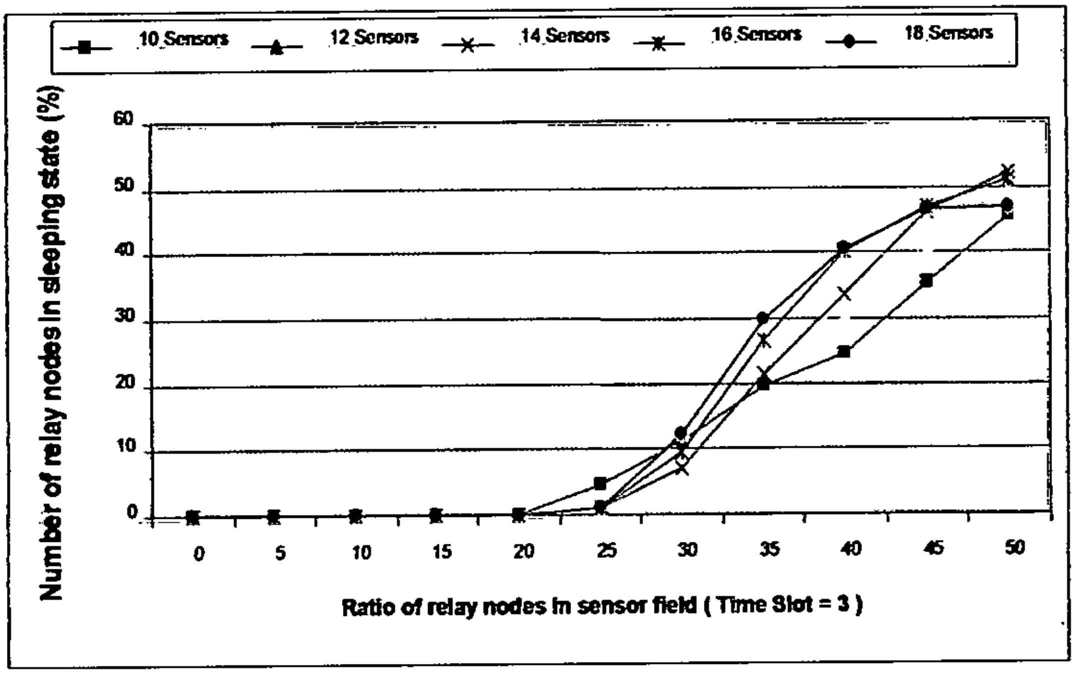

Fig - 4.9 : Percentage (%) of relay nodes in sleeping state for different sensor network in the Minimum Cost Network Flow model (Time Slot = 3)

Table 4.8 displays the result of the Minimum Cost Network Flow model for four equal time slots. It represents the percentage of time relay nodes in sleeping state for different number of sensors. The percentage of relay nodes are in sleeping state has been calculated for the number of sensor nodes 10, 12, 14, 16 and 18 respectively. The maximum amount of battery power is saved when we put 7 relay nodes in sensor field size of 14 sensors or 6 relay nodes in sensor field size of 12 i.e. ratio of relay and sensor node is 50%. In this cases, the maximum amount of battery saving recorded as 37.4%. If we consider four time slots, the average battery saving is noticed as 7.5% for Minimum Cost Network Flow model. The average percentage of relay nodes are in sleeping sate is zero (0) when the ratio of relay and sensor nodes is less than or equal to 30%. On an average 6% of relay nodes are in sleeping state when relay and sensor nodes ratio is 35%. The table shows the percentage of sleeping relay nodes are increasing with the increment of sensor field size and the ratio of relay and sensor nodes. The average of the average percentage of sleeping relay nodes for the Minimum Cost Network Flow model using four time slots is shown in the bottom row of the table 4.8.

Table - 4.8 : Percentage (%) of relay nodes in sleeping state for different sensor network in the Minimum Cost Network Flow model (Time Slot = 4)

Ratio of Relay and Sensor nodes (%)	Total Number of Sensor Nodes in Sensor Network (% of relay nodes are in sleeping state) **Time Slot = 4**					Average % of Sleeping Relay nodes
	10 nodes	12 nodes	14 nodes	16 nodes	18 nodes	
0	0	0	0	0	0	0
5	0	0	0	0	0	0
10	0	0	0	0	0	0
15	0	0	0	0	0	0
20	0	0	0	0	0	0
25	0	0	0	0	0	0
30	0	0	0	0	0	0
35	3.7	3.1	6	7.1	9.7	6
40	6	11.6	15.7	19.6	21.8	15
45	17	24.4	29	30.6	30.1	26.3
50	30.3	35.2	37.4	37.4	31.7	34.4
The average percentage of relay nodes are in sleeping state =						7.5

In Fig - 4.10, we observe that, when the relay node's ratio is greater than 30% to sensor nodes, we have some extra relay nodes for scheduling. When we have 35% relay nodes, we notice approximately 6% of relay nodes in sleeping state to conserve energy for the whole network. In our experiment, the number of deployed sensors in sensor field ranges from 10 to 18. According to the fig - 4.10, the relay node's sleeping time is increasing with the increment of the number of sensors in the sensor field. If we deploy 30 or less relay nodes in sensor field of 100 sensor nodes, then the graph shows infeasible solution i.e. relay node's number is insufficient to transmit data to all sensor nodes and we need more relay nodes to communicate between all sensor nodes. According to the fig - 4.10, the highest percentage of the relay nodes are in sleeping state is approximately 38%, when we put 14 sensor nodes and 7 relay nodes in a sensor field.

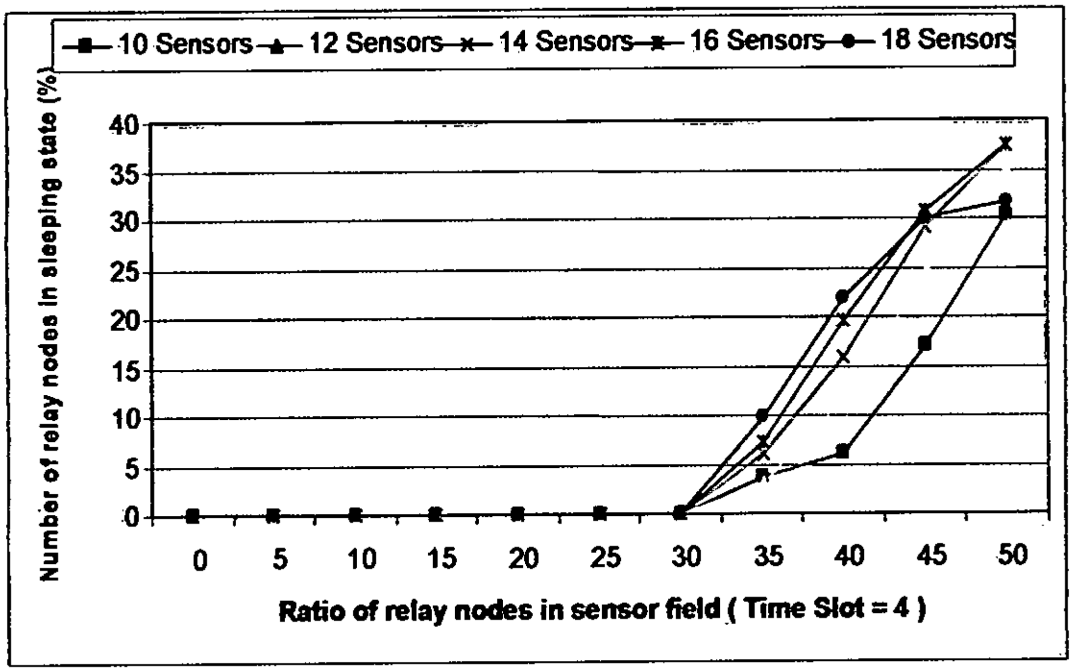

Fig - 4.10 : Percentage (%) of relay nodes in sleeping state for different sensor network in the Minimum Cost Network Flow model (Time Slot = 4)

Fig - 4.11 shows the comparison of the number of relay nodes in sleeping state for time slots from 2 to 4 in the Minimum Cost Network Flow model. The highest amount of relay nodes in sleeping state when we consider two time slots and the lowest amount of time relay nodes in sleeping state when four time slots are considered. When a relay node is in sleeping state, it preserves energy for the whole wireless sensor network to prolong network life time. Suppose the total battery power for every relay nodes is fixed for time slot 2, 3 and 4 respectively. When we use two time slots then every relay nodes have two options to be active i.e. a relay node can be active in time slot 1 or time slot 2. Similarly when we use four time slots then every relay nodes have four options to be active in four different time slots. Thus relay nodes activation time is more in four time slots compared to two time slot which means maximum number of relay nodes are in sleeping state when consider two time slots compared to four time slots. Thus we can conclude that, the number of sleeping relay nodes are decreasing with the increment of time slots. Fig – 4.11 shows that, if we consider more time slots in the Minimum Cost Network Flow model, then we can save less battery power from relay node scheduling in wireless sensor network and vice-versa.

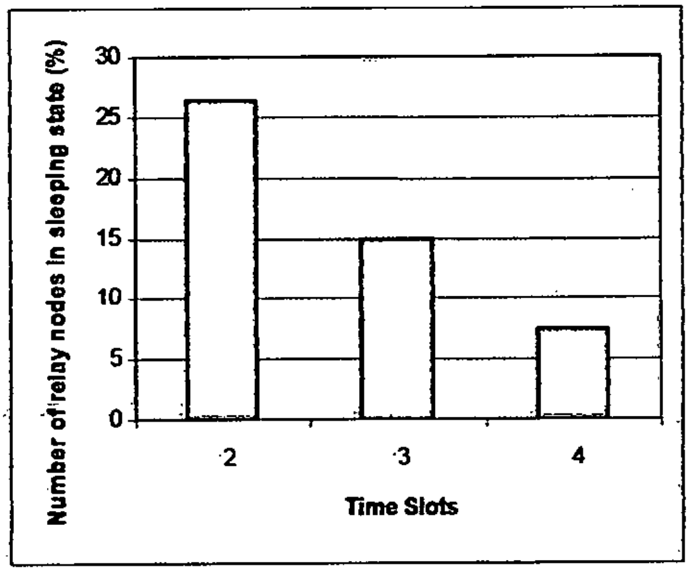

Fig - 4.11 : Comparison of the percentage of relay nodes in sleeping state for different time slots in the Minimum Cost Network Flow model

Chapter 5

Conclusions and Future Work

5.1 Conclusions

In this chapter, we summarize our work and give direction to further research. We have studied the heterogeneous property of wireless sensor network. The most of the work on relay node placement problems consider homogeneous environment. Recently some work have been done on heterogeneous environment, but so far to the best of our knowledge the most of the work only considers relay nodes' geographic position. It does not consider relay nodes' properties such as battery power supply and their different transmission ranges. We prove that existing algorithms do not work in heterogeneous environment, and we provide models considering heterogeneous property.

We prove that, there are multiple transmission paths between two sensors in heterogeneous environment, if we apply existing minimum number of relay node placement algorithm. We can save energy, if we keep the minimum number of transmitting nodes active in a particular time period. When some nodes are participating in transmitting data, then the other nodes will be in sleeping state to conserve energy. In the next time slot, they may be active and the previously activated nodes may go to sleep state.

We propose two Linear Programming models called the Minimax model and the Minimum Cost Network Flow model which consider heterogeneous property of sensor network. In the Minimax model, we consider different transmission ranges of sensor and relay nodes and in the Minimum Cost Network Flow model, we incorporate different battery power supply of network nodes also. Both of our models can check the feasibility of the problem domain also.

We perform experiment on extensive sets of data. It is observed that, in the Minimax model relay nodes are in sleeping state for on an average 32% of total time when we consider different transmission ranges. When relay nodes are in sleeping state, it can save energy which prolongs total network lifetime. We incorporate battery power also in the Minimum Cost Network Flow model. Our test result shows on an average of 16.4% of the total supplied energy can be saved in the Minimum Cost Network Flow model. We present the experimental results in form of tables and graphs.

5.2 Future Work

Ensuring k-connectivity: We work on single connectivity that is our models provide optimal solution maintaining k-connectivity when $k = 1$ for heterogeneous sensor network. In future, we will extend our model for $k > 1$.

Using different LP techniques: We give the Minimax model and the Minimum Cost Network Flow model. Other Linear Programming solving techniques may be applied and computing time may be compared. For example, we can try assignment model, transportation model etc. to schedule multiple paths between two sensors to prolong network lifetime.

Weighed cost: Our model is static with the fixed cost. We can make the model dynamic with cost adjusting.

References

[1] A. Silberschatz and P. B. Galvin, "Operating System Concepts", 5th ed., Addison Wesley, 1998, pp. 129 – 137.

[2] B. Hao, J. Tang and G. Xue, "Fault-tolerant relay node placement in Wireless Sensor Networks: formulation and approximation", *IEEE Workshop on High Performance Switching and Routing (HPSR)*, 2004, pp. 246-250.

[3] B. Karp and H. T. Kung, "GPRS: Greedy Perimeter Stateless Routing for Wireless Networks", *Proc. The 6th Annual International Conference on Mobile Computing and Networking (MobiCom)*, Boston MA, USA, 2000, pp. 243-254.

[4] C. Gueret, S. Helpcke, C. Prins and M. Sevaux., (2000), *Application of optimization with Xpress – MP*. (1st ed.). [Online], pp. 28.
Available : http://www.dashoptimiztion.com/home/downloads/book/booka4.pdf

[5] *CPU scheduling*. [Online].
Available: http://www.cis.upenn.edu/~lee/07cis505/Lec/os-schedule-v2.pdf

[6] D. Chen, D.Z. Du, X.D. Hu, G. Lin, L. Wang and G. Xue, "Approximation for Steiner trees with minimum number of Steiner points", *Journal of Global Optimization*, 2000, pp. 17-33.

[7] D. Du, L. Wang and B. Xu, "The Euclidean Bottleneck Steiner Tree and Steiner Tree with Minimum Number of Steiner Points", *Computing and Combinatorics, 7^{th} Annual International Conference COCOON*, Guilin, China, Proceedings, 2001.

[8] D. S. Hochbaum and Maass, "Approximation schemes for covering and packing problems in image processing and VLSI", *Journal of the ACM*, 1985, pp. 130-136.

[9] E. L. Lloyd and G. Xue, "Relay Node Placement in Wireless Sensor Networks", *IEEE transaction on Computers*, 2007, pp. 134-138.

[10] *Earlier deadline first*. [Online].
Available : http://retis.sssup.it/~lipari/courses/str06/10.edf.pdf

[11] F. Zhao and L. Guibas, "Wireless Sensor Networks: An information processing approach", *Boston: Elsevier-Morgan Kaufmann*, 2004.

[12] G. Lin and G. Xue, "Steiner tree problem with minimum number of Steiner points and bounded edge-length", *Information Processing Letters*, 1999, pp. 53-57.

[13] H. Lin, A. Nayak and I. Stojmenovic, "On Fault Tolerance in Wireless Sensor Networks", 2008.

[14] H. Liu, P. Wan and X. Jia, "On optimal placement of relay nodes for reliable connectivity in Wireless Sensor Networks", *Journal of Combinatorial Optimization*, 2006, pp: 249 – 260.

[15] Ian F. Akyildiz, Weilian Su, Yogesh Sankarasubramaniam and Erdal Cayirci, "A Survey on Sensor Networks", *IEEE Communication Magazine*, 2002, pp. 102-114.

[16] J. Blazewicz, W. Domschke, and E. Pesch, "The job shop scheduling problem: conventional and new solution techniques", *European Journal of Operational Research*, Vol. 93, No. 1, 1996, pp. 1-33.

[17] J. Tang, B. Hao and A. Sen, "Relay node placement in large scale Wireless Sensor Networks", *Computer Communications*, 2006, pp. 490-501.

[18] L. Doherty , K. S. J. Pister and L. E. Ghaoui, "Convex Position Estimation in Wireless Sensor Networks", *Proc. 20th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, 2001, pp. 1655-1663.

[19] *Linear programming*. [Online]. Available: http://www.math.ucla.edu/~tom/LP.pdf

[20] *Linear Programming*. [Online].
Available: http://mathworld.wolfram.com/LinearProgramming.html

[21] M. Naznin, X. Du and K. E. Nygard, "Scheduling duty cycles in differentiated sensor networks", *Proc. of IASTED International Conference on Knowledge Sharing and Collaborative Engineering (KSCE)*, St. Thomas Virgin Islan, USA, 2006.

[22] M. Naznin and Kendall E. Nygard, "Handling Multiplicities in dynamic network flow model", *The 40th Annual Midwest Instructional and Computing Symposium (MICS)*, hosted by University of North Dakota (UND), Grand Forks, ND, USA, 2007.

[23] N. Bulusu, J. Heidermann and D. Estrin, "GPS-less Low-cost Outdoor Localization for Very Small Devices", *IEEE Pers. Communication*, 2000, pp. 28-34.

[24] P.K. Gupta and D.S. Hira, "Operations research", 1st ed., S. Chand and Company Ltd., 1976, pp. 39-102.

[25] R.K. Ahuja, T.L. Magnalti and J.B. Orlin, "Network Flows Theory, Algorithms, and Applications", Prince-Hall Inc., 1993.

[26] *Real time scheduling*. [Online].
Available: http://www.cs.auckland.ac.nz/compsci340s2c/lectures/lecture09.pdf

[27] S. Khuller and B. Raghavachari, "Improved approximation algorithms for uniform connectivity problems" *Journal of Algorithm*, 1996, pp. 214-235.

[28] S. Meguerdichian and M. Potkonjak, "Low Power 0/1 Coverage and Scheduling Techniques in Sensor Networks", Technical Report 030001, UCLA, 2003.

[29] *Spanning tree protocol in IEEE 802.1D*. [Online].
Available: http://www.javvin.com/protocolSTP.html

[30] *The Minimum Cost Network Flow Problem*. [Online].
Available: http://lyle.smu.edu/~olinick/emis8374/lectures/mcnfp_review.pdf

[31] V. P. Mhatre, C. Rosenberg, D. Kofman, R. Mazumdar and N. Shroff, "A minimum cost heterogeneous sensor network with a lifetime constraint" *IEEE Transactions on Mobile Computing*, 2005, pp. 4-15.

[32] W. Zhang, G. Xue and S. Misra, "Fault-Tolerant Relay Node Placement in Wireless Sensor Networks", *Problems and Algorithms, 26th IEEE international conference on Computer Communications (INFOCOM)*, 2007, pp. 1649-1657.

[33] X. Cheng, D. Z. Du, L. Wang and B. Xu, "Relay node placement in Wireless Sensor Networks", *ACM/Springer Journal of Wireless Networks*, 2008, pp. 347-355.

[34] X. Han, X. Cao, E. L. Lloyd and C. Shen, "Fault-tolerant Relay Node Placement in Heterogeneous Wireless Sensor Networks", *26th IEEE international conference on Computer Communications (INFOCOM)*, 2007, pp. 1667-1675.

[35] Y. Zou and K. Chakrabarty, "Energy-Aware Target Localization in Wireless Sensor Networks", *Proc. International Journal of Wireless Ad Hoc and Sensor Networks*, 2004.

[36] *Wireless Sensor Networks*. [Online].
Available: http://www.alicosystems.com/wireless%20sensor.htm

[37] *Project SmartMotes*. [Online].
Available: http://www.tecnologico.deusto.es/projects/smartmotes/

[38] *Volcano monitoring*. [Online]. Available: http://fiji.eecs.harvard.edu/Volcano

[39] Welcome to Dynamic Sensor Networks. [Online].
Available: http://dsn.east.isi.edu/welcome.html

Appendix A

Used formula:

$$\% \text{ of relay nodes in sleeping state} = \frac{(\text{Total number of relay nodes} - \text{Number of active relay nodes})}{\text{Total number of relay nodes}} \times 100 \quad \ldots \quad A1$$

$$\text{Average} = \frac{\sum_{i=1}^{n} x_i}{n} \quad \ldots \quad A2$$

$$\text{Standard Deviation} = \sqrt{\frac{\sum_{i=1}^{n}(x_i - Average)^2}{n}} \quad \ldots \quad A3$$

Table - A1 : Standard deviation and average of the % of relay nodes in sleeping state using the Minimax model for Time Slot = 2

Sensors	Relay nodes	% of relay nodes in sleeping state	Average	Standard Deviation
35	10	30	33	2.74
		30		
		35		
		35		
		35		
	15	53.33	56	3.65
		53.33		
		53.33		
		60		
		60		
	20	60	64	4.18
		60		
		65		
		65		
		70		

Table - A1 : Standard deviation and average of the % of relay nodes in sleeping state using the Minimax model for Time Slot = 2 (Continued)

Sensors	Relay nodes	% of relay nodes in sleeping state	Average	Standard Deviation
40	10	28	30	2
		28		
		30		
		32		
		32		
	15	50	56	3.74
		56		
		56		
		58		
		60		
	20	60	63.5	4.18
		60		
		62.5		
		65		
		70		
45	10	32	36	3.08
		35		
		35		
		38		
		40		
	15	53.55	58.71	5.53
		53.33		
		60		
		60		
		66.67		
	20	62.5	67	4.11
		65		
		65		
		70		
		72.5		
	25	72	76.8	3.35
		76		
		76		
		80		
		80		

Table - A1 : Standard deviation and average of the % of relay nodes in sleeping state using the Minimax model for Time Slot = 2 (Continued)

Sensors	Relay nodes	% of relay nodes in sleeping state	Average	Standard Deviation
50	10	25	27	2.74
		25		
		25		
		30		
		30		
	15	45	51	4.18
		50		
		50		
		55		
		55		
	20	55	61	4.18
		60		
		60		
		65		
		65		
	25	72	74.8	3.03
		72		
		74		
		78		
		78		
55	10	30	32	2.74
		30		
		30		
		35		
		35		
	15	50	52	1.82
		50		
		53.33		
		53.33		
		53.33		
	20	60	70	6.12
		60		
		70		
		70		
		75		

Table - A1 : Standard deviation and average of the % of relay nodes in sleeping state using the Minimax model for Time Slot = 2 (Continued)

Sensors	Relay nodes	% of relay nodes in sleeping state	Average	Standard Deviation
55	25	72 72 76 76 80	75.2	3.35
	30	76.67 80 80 83.33 83.33	80.67	2.79
60	15	42 46.67 46.67 48 53.33	47.33	4.05
	20	50 55 55 55 60	56	4.18
	25	56 60 64 68 68	63.2	5.22
	30	75 80 85 85 85	82	4.47
65	15	50 50 50 55 60	53	4.47

Table - A1 : Standard deviation and average of the % of relay nodes in sleeping state using the Minimax model for Time Slot = 2 (Continued)

Sensors	Relay nodes	% of relay nodes in sleeping state	Average	Standard Deviation
65	20	70	75	3.54
		75		
		75		
		75		
		80		
	25	80	82.8	1.79
		82		
		84		
		84		
		84		
	30	85	85	0
		85		
		85		
		85		
		85		
	35	86	89.47	2.28
		88.57		
		91.43		
		91.43		
		90		
70	15	50	54	4.18
		50		
		55		
		55		
		60		
	20	60	68	5.58
		66.67		
		66.67		
		73.33		
		73.33		
	25	65	74	5.52
		70		
		75		
		80		
		80		

Table - A1 : Standard deviation and average of the % of relay nodes in sleeping state using the Minimax model for Time Slot = 2 (Continued)

Sensors	Relay nodes	% of relay nodes in sleeping state	Average	Standard Deviation
70	30	80	82	1.82
		80		
		83.33		
		83.33		
		83.33		
	35	90	89.64	1.2
		88.57		
		88.57		
		90		
		91.43		

Table – A2 : Standard deviation and average of the % of relay nodes in sleeping state using the Minimax model for Time Slot = 3

Sensors	Relay nodes	% of relay nodes in sleeping state	Average	Standard Deviation
35	10	28	32	3.08
		30		
		32		
		35		
		35		
	15	53.33	57.33	3.65
		53.33		
		60		
		60		
		60		
	20	55	65	5.7
		60		
		65		
		65		
		70		
40	10	26	28	2
		26		
		28		
		30		
		30		

Table – A2 : Standard deviation and average of the % of relay nodes in sleeping state using the Minimax model for Time Slot = 3 (Continued)

Sensors	Relay nodes	% of relay nodes in sleeping state	Average	Standard Deviation
40	15	50	54.67	3.8
		53.33		
		53.33		
		56.67		
		60		
	20	55	62	5.42
		60		
		62.5		
		62.5		
		70		
45	10	30	34	2.65
		33		
		35		
		35		
		37		
	15	53.55	60.04	4.64
		60		
		60		
		60		
		66.67		
	20	62.5	68	4.11
		65		
		70		
		70		
		72.5		
	25	72	76	4
		72		
		76		
		80		
		80		
50	10	27	28	2.12
		30		
		25		
		28		
		30		

Table – A2 : Standard deviation and average of the % of relay nodes in sleeping state using the Minimax model for Time Slot = 3 (Continued)

Sensors	Relay nodes	% of relay nodes in sleeping state	Average	Standard Deviation
50	15	47	52	4.42
		50		
		55		
		50		
		58		
	20	55	60	5
		55		
		60		
		65		
		65		
	25	72	75.2	2.68
		74		
		74		
		78		
		78		
55	10	30	33	2.45
		32		
		32		
		35		
		36		
	15	50	54	4.18
		50		
		53.33		
		58.33		
		58.33		
	20	60	68	5.7
		65		
		70		
		70		
		75		
	25	72	76	4
		72		
		76		
		80		
		80		

Table – A2 : Standard deviation and average of the % of relay nodes in sleeping state using the Minimax model for Time Slot = 3 (Continued)

Sensors	Relay nodes	% of relay nodes in sleeping state	Average	Standard Deviation
55	30	76.67	80.67	2.79
		80		
		80		
		83.33		
		83.33		
60	15	40	45.33	4.19
		42		
		46.67		
		48		
		50		
	20	50	55	5
		50		
		55		
		60		
		60		
	25	56	62.4	4.56
		60		
		64		
		64		
		68		
	30	75	81	4.18
		80		
		80		
		85		
		85		
65	15	46	52	4.69
		48		
		54		
		56		
		56		
	20	70	74	4.18
		70		
		75		
		75		
		80		

Table – A2 : Standard deviation and average of the % of relay nodes in sleeping state using the Minimax model for Time Slot = 3 (Continued)

Sensors	Relay nodes	% of relay nodes in sleeping state	Average	Standard Deviation
65	25	80	82.4	2.19
		80		
		84		
		84		
		84		
	30	85	85	0
		85		
		85		
		85		
		85		
	35	86	88.97	2.78
		86		
		91.43		
		91.43		
		90		
70	15	50	54	4.18
		50		
		55		
		55		
		60		
	20	65	69.33	4.5
		66.67		
		66.67		
		73.33		
		75		
	25	65	74	6.52
		70		
		75		
		80		
		80		
	30	80	81.33	1.82
		80		
		80		
		83.33		
		83.33		

Table – A2 : Standard deviation and average of the % of relay nodes in sleeping state using the Minimax model for Time Slot = 3 (Continued)

Sensors	Relay nodes	% of relay nodes in sleeping state	Average	Standard Deviation
70	35	90	90	1.43
		88.57		
		88.57		
		91.43		
		91.43		

Table – A3 : Standard deviation and average of the % of relay nodes in sleeping state using the Minimax model for Time Slot = 4

Sensors	Relay nodes	% of relay nodes in sleeping state	Average	Standard Deviation
35	10	26	30	3.16
		28		
		30		
		32		
		34		
	15	53.33	56	3.65
		53.33		
		53.33		
		60		
		60		
	20	57	64	5.43
		60		
		65		
		68		
		70		
40	10	28	29	2.65
		25		
		30		
		30		
		32		
	15	46.67	53.33	4.71
		53.33		
		53.33		
		53.33		
		60		

Table – A3 : Standard deviation and average of the % of relay nodes in sleeping state using the Minimax model for Time Slot = 4 (Continued)

Sensors	Relay nodes	% of relay nodes in sleeping state	Average	Standard Deviation
40	20	60	63	4.12
		60		
		62.5		
		62.5		
		70		
45	10	32	36	3.16
		34		
		40		
		36		
		38		
	15	53.55	58.71	5.52
		53.33		
		60		
		60		
		66.67		
	20	62.5	68.5	4.54
		65		
		70		
		72.5		
		72.5		
	25	72	77.6	3.58
		76		
		80		
		80		
		80		
50	10	28	29	2.65
		30		
		25		
		30		
		32		
	15	45	51	4.18
		50		
		55		
		55		
		50		

Table – A3 : Standard deviation and average of the % of relay nodes in sleeping state using the Minimax model for Time Slot = 4 (Continued)

Sensors	Relay nodes	% of relay nodes in sleeping state	Average	Standard Deviation
50	20	55	61	4.18
		60		
		60		
		65		
		65		
	25	72	76	2.83
		74		
		78		
		78		
		78		
55	10	32	34	2.45
		34		
		34		
		32		
		38		
	15	50	55.66	5.22
		53.33		
		53.33		
		58.33		
		63.33		
	20	60	69	6.52
		65		
		70		
		75		
		75		
	25	72	75.2	3.35
		72		
		76		
		76		
		80		
	30	76.67	81.33	2.98
		80		
		83.33		
		83.33		
		83.33		

Table – A3 : Standard deviation and average of the % of relay nodes in sleeping state using the Minimax model for Time Slot = 4 (Continued)

Sensors	Relay nodes	% of relay nodes in sleeping state	Average	Standard Deviation
60	15	45	48.66	3.2
		46.67		
		50		
		48.33		
		53.33		
	20	50	57	4.47
		55		
		60		
		60		
		60		
	25	56	62.4	6.07
		56		
		64		
		68		
		68		
	30	75	81	4.18
		80		
		80		
		85		
		85		
65	15	50	52	3.08
		48		
		52		
		55		
		55		
	20	70	76	4.18
		75		
		75		
		80		
		80		
	25	80	82.4	1.67
		82		
		82		
		84		
		84		

Table – A3 : Standard deviation and average of the % of relay nodes in sleeping state using the Minimax model for Time Slot = 4 (Continued)

Sensors	Relay nodes	% of relay nodes in sleeping state	Average	Standard Deviation
65	30	85	84	2.24
		85		
		80		
		85		
		85		
	35	86	89.2	2.05
		88.57		
		91.43		
		90		
		90		
70	15	50	52	4.47
		50		
		50		
		50		
		60		
	20	65	69.33	4.5
		66.67		
		66.67		
		73.33		
		75		
	25	65	73	5.7
		70		
		75		
		75		
		80		
	30	80	82	1.82
		80		
		83.33		
		83.33		
		83.33		
	35	90	90.57	1.28
		88.57		
		91.43		
		91.43		
		91.43		

Table – A4 : Standard deviation and average of the % of relay nodes in sleeping state using the Minimum Cost Network Flow model for Time Slot = 2

Sensors	Relay nodes	% of relay nodes in sleeping state	Average	Standard Deviation
10	3	33.04	38.31	3.64
		36.48		
		39.39		
		40.18		
		42.45		
	4	42.86	48.08	3.28
		42.86		
		45.95		
		48.72		
		50		
	5	57.45	61.65	2.97
		60		
		62.26		
		63.64		
		64.91		
12	3	22.7	25.51	2.41
		23.6		
		25.5		
		27.42		
		28.33		
	4	33.2	35.1	2.74
		31.7		
		35.14		
		38.46		
		37		
	5	48.94	54	3.55
		52.1		
		54.72		
		56.36		
		57.89		
	6	61.29	65.29	3.16
		63.08		
		65.71		
		67.12		
		69.23		

Table – A4 : Standard deviation and average of the % of relay nodes in sleeping state using the Minimum Cost Network Flow model for Time Slot = 2 (Continued)

Sensors	Relay nodes	% of relay nodes in sleeping state	Average	Standard Deviation
14	3	12.45	13.63	1.23
		12.68		
		13.15		
		14.65		
		15.22		
	4	24	23.87	1.82
		22.22		
		21.92		
		26.21		
		25		
	5	44	46.31	4.16
		47.17		
		49.09		
		50.88		
		20		
	6	54.84	59.5	3.67
		56.92		
		60		
		61.64		
		64.1		
	7	63.16	67.25	3.29
		65		
		67.44		
		69.23		
		71.43		
16	3	4	4	1.17
		3		
		3.03		
		5.88		
		4.11		
	4	12.57	13.72	1.39
		12.57		
		13.51		
		15.95		
		14		

Table – A4 : Standard deviation and average of the % of relay nodes in sleeping state using the Minimum Cost Network Flow model for Time Slot = 2 (Continued)

Sensors	Relay nodes	% of relay nodes in sleeping state	Average	Standard Deviation
16	5	37.91	38.64	2.25
		36		
		37.62		
		39.82		
		41.86		
	6	48.39	53.72	4.21
		50.77		
		54.29		
		56.16		
		58.97		
	7	57.89	62.57	3.76
		60		
		62.79		
		64.84		
		67.35		
	8	62.79	67.28	3.29
		65.59		
		67.35		
		69.52		
		71.17		
18	4	3	4.08	0.8
		4		
		3.7		
		4.69		
		5		
	5	26.4	30.97	2.91
		30		
		32.08		
		32.55		
		33.84		
	6	41.94	47.93	4.74
		44.62		
		48.57		
		50.68		
		53.85		

Table – A4 : Standard deviation and average of the % of relay nodes in sleeping state using the Minimum Cost Network Flow model for Time Slot = 2 (Continued)

Sensors	Relay nodes	% of relay nodes in sleeping state	Average	Standard Deviation
18	7	52.63	57.9	4.23
		55		
		58.14		
		60.44		
		63.27		
	8	58.14	63.2	3.7
		61.29		
		63.27		
		65.71		
		67.57		
	9	58.62	64.47	5.12
		61.29		
		62.89		
		68.93		
		70.64		

Table – A5 : Standard deviation and average of the % of relay nodes in sleeping state using the Minimum Cost Network Flow model for Time Slot = 3

Sensors	Relay nodes	% of relay nodes in sleeping state	Average	Standard Deviation
10	3	8.4	11.08	1.59
		10.67		
		11.76		
		12.67		
		10.92		
	4	26.4	24.52	1.44
		23.08		
		23.08		
		24.83		
		25.23		
	5	40	45.21	4.12
		42.31		
		45.45		
		48.28		
		50		

Table – A5 : Standard deviation and average of the % of relay nodes in sleeping state using the Minimum Cost Network Flow model for Time Slot = 3 (Continued)

Sensors	Relay nodes	% of relay nodes in sleeping state	Average	Standard Deviation
12	3	0	1.08	0.68
		1		
		1		
		1.7		
		1.7		
	4	20.2	21.39	1.41
		19.8		
		21.4		
		23		
		22.57		
	5	33	34.25	2.64
		30.77		
		34.55		
		37.93		
		35		
	6	44.62	49.61	4
		47.06		
		49.3		
		52.63		
		54.43		
14	4	4.66	4.12	0.5
		3.83		
		3.63		
		4.66		
		3.83		
	5	22	23.29	1.72
		21.23		
		23.64		
		25.59		
		24		
	6	37.38	41.21	3.8
		38.24		
		40.85		
		42.74		
		46.84		

Table – A5 : Standard deviation and average of the % of relay nodes in sleeping state using the Minimum Cost Network Flow model for Time Slot = 3 (Continued)

Sensors	Relay nodes	% of relay nodes in sleeping state	Average	Standard Deviation
14	7	46.84	52	4.11
		49.45		
		51.72		
		54.84		
		57.14		
16	5	12.26	13.02	0.85
		12.16		
		13.46		
		14.18		
		13.05		
	6	32.42	34.23	2.95
		30.77		
		33.82		
		38.36		
		35.79		
	7	42.24	45.13	2.18
		45.17		
		44.83		
		48.39		
		45.02		
	8	46.66	50.92	3.46
		48.94		
		50.52		
		52.94		
		55.56		
18	5	4.18	5.23	0.72
		5.9		
		5.18		
		5.9		
		5		
	6	22.92	24.41	2.03
		22.59		
		23.94		
		24.95		
		27.65		

Table – A5 : Standard deviation and average of the % of relay nodes in sleeping state using the Minimum Cost Network Flow model for Time Slot = 3 (Continued)

Sensors	Relay nodes	% of relay nodes in sleeping state	Average	Standard Deviation
18	7	35.65	38.27	3.1
		34.94		
		37.93		
		41.94		
		40.9		
	8	40	46.07	4.04
		44.55		
		46.53		
		49.06		
		50.21		
	9	41.94	46.72	3.97
		43.75		
		47.06		
		49.06		
		51.79		

Table – A6 : Standard deviation and average of the % of relay nodes in sleeping state using the Minimum Cost Network Flow model for Time Slot = 4

Sensors	Relay nodes	% of relay nodes in sleeping state	Average	Standard Deviation
10	4	5.44	6	0.9
		5.44		
		6.98		
		6.98		
		5.16		
	5	27.53	30.29	2.07
		30.27		
		29.82		
		33.33		
		30.51		
12	5	14.6	16.2	1.71
		14.4		
		16.2		
		18		
		17.81		

Table – A6 : Standard deviation and average of the % of relay nodes in sleeping state using the Minimum Cost Network Flow model for Time Slot = 4 (Continued)

Sensors	Relay nodes	% of relay nodes in sleeping state	Average	Standard Deviation
12	6	30.27	35.17	3.04
		36.62		
		35.14		
		35.37		
		38.46		
14	5	6.67	7.18	1.48
		9.68		
		5.75		
		6.67		
		7.11		
	6	21.15	23.92	2.27
		22.13		
		24.32		
		25.27		
		26.71		
	7	35.11	37.24	2.3
		34.88		
		37.78		
		40.43		
		38		
16	6	12.03	13.05	0.7
		13.86		
		13.51		
		12.88		
		12.95		
	7	24.99	28.27	2.63
		26.58		
		28.88		
		31.91		
		29		
	8	32.67	37.35	3.41
		35.69		
		37.25		
		41.28		
		39.86		

Table – A6 : Standard deviation and average of the % of relay nodes in sleeping state using the Minimum Cost Network Flow model for Time Slot = 4 (Continued)

Sensors	Relay nodes	% of relay nodes in sleeping state	Average	Standard Deviation
18	6	5.7	6.12	1.63
		6.49		
		3.7		
		6.49		
		8.2		
	7	17.86	19.31	1.35
		18.28		
		19		
		20.4		
		21		
	8	25.88	29.52	2.52
		28.53		
		29.41		
		31.94		
		31.84		
	9	30	31.67	2.26
		29.41		
		31.43		
		35.14		
		32.39		