

1-1-2008

Multi-Level Clustering Architecture And Protocol Designs For Wireless Sensor Networks

Barnabas C. Okeke
Ryerson University

Follow this and additional works at: <http://digitalcommons.ryerson.ca/dissertations>

 Part of the [Electrical and Computer Engineering Commons](#)

Recommended Citation

Okeke, Barnabas C., "Multi-Level Clustering Architecture And Protocol Designs For Wireless Sensor Networks" (2008). *Theses and dissertations*. Paper 1171.

This Thesis is brought to you for free and open access by Digital Commons @ Ryerson. It has been accepted for inclusion in Theses and dissertations by an authorized administrator of Digital Commons @ Ryerson. For more information, please contact bcameron@ryerson.ca.

010015101
TK
7872
D48
044
2008

**MULTI-LEVEL CLUSTERING ARCHITECTURE AND PROTOCOL
DESIGNS FOR WIRELESS SENSOR NETWORKS**

by

Barnabas C. Okeke

B.Eng., Federal University of Technology, Owerri, Nigeria, 2002
M.Sc., Blekinge Institute of Technology, Karlskrona, Sweden, 2007

A thesis submitted in conformity with the requirements
for the degree of Master of Applied Science
Graduate Department of Electrical and Computer Engineering
Ryerson University

Toronto, Ontario, Canada, 2008

© Barnabas C. Okeke, 2008

Author's Declaration

I hereby declare that I am the sole author of this thesis or dissertation. I authorize Ryerson University to lend this thesis or dissertation to other institutions or individuals for the purpose of scholarly research.

I further authorize Ryerson University to reproduce this thesis or dissertation by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

Abstract

Multi-level Clustering Architecture and Protocol Designs for Wireless Sensor Networks

Barnabas C. Okeke

Master of Applied Science

Graduate Department of Electrical and Computer Engineering

Ryerson University

2008

Wireless sensor network (WSN) consists of a number of sensors, which measure and gather data in a variety of environments. In a WSN, sensed data are collected at a centralized location, called sink, for processing and analysis. With limited transmission ranges, sensed data may require multiple relays to reach the sink. In this thesis, a novel system design for multi-level clustering (MLC) WSNs and its associated protocol operations are proposed. Cluster-heads in the proposed design form a tree with a goal to reach all sensor nodes in the network. Subsequently, all sensed data in the tree are delivered to the sink. Energy savings is improved by exploiting sensor node redundancy in the WSN. To validate the proposed design, thorough simulations have been carried out. Upon comparing to the LEACH protocol, it offers consistent wider coverage area and longer life span of a WSN with proper settings of system parameters.

Dedication

This work is dedicated to Almighty God for His mercies and guardians.

Contents

1	Introduction	1
1.1	Problem Formulation	2
1.2	Thesis Organization	4
2	Literature Review	5
2.1	Hierarchical Clustered Protocols	5
2.1.1	LEACH	6
2.1.2	DTRAP	10
2.1.3	Multi-level Cluster-heads	11
2.1.4	Optimization Algorithm based Protocols	13
2.2	Flat Clustered Protocols	14
2.3	WSN Physical Layer Properties	15
2.3.1	WSN Communication Patterns	16
2.3.2	WSN Energy Loss Activities	17
3	Multi-level Clustering (MLC) WSN	19
3.1	Network Model	21
3.2	Energy Model	22
3.3	MLC Neighbor Discovery Process	25
3.4	MLC Cluster-head Selections Process	28
3.4.1	Maximum Residual Energy Model	32
3.5	MLC Cluster Member Admittance Process	33
3.6	MLC Transmission Process	35
3.7	Redirection Protocol	36
3.7.1	Minimum Cost Model	37

3.8	MLC with Load Balancing (MLC-B)	38
3.8.1	MLC-B Cluster Member Admittance	41
4	Simulation and Results	45
4.1	MLC Simulations and Results	47
4.2	MLC-B Simulations and Results	50
5	Conclusions	56
A	MLC-B Parameters	57
	Bibliography	67

List of Tables

3.1	Important Node Properties	26
3.2	NEIGHBOR_TABLE	26
3.3	HELLO Message	26
3.4	NOTIFY message.	28
3.5	REQUEST/ADVERT Message	31
3.6	REPLY/JOIN/ACK/NACK Message	33
3.7	PARAM message.	35
3.8	Modified NOTIFY message.	42
3.9	ACCEPT/REJECT Message	43
4.1	Parameters for network <i>A</i> and network <i>B</i>	46
4.2	Reference Parameters	46

List of Figures

1.1	Hierarchical topology of a clustered network. Cluster- <i>A</i> shows an example of intra-cluster communications. None of the selected cluster-heads is within transmission ranges of the sink.	3
2.1	Hierarchical topology of the LEACH protocol. Cluster- <i>A</i> shows an example of intra-cluster communications. There is no inter-cluster communication.	7
2.2	Hierarchical topology of the multi-hop LEACH protocol (LEACH B+). Cluster- <i>A</i> shows an example of intra-cluster communications. There is cluster-head to cluster-head inter-cluster communication.	9
2.3	Multi-level Cluster-heads topology. Cluster- <i>C</i> shows an example of intra-cluster communications. The shaded regions indicate cluster-heads that belong to the same upper level cluster. There is cluster-head to cluster-head inter-cluster communication.	12
3.1	Multi-level cluster-heads structure for the MLC WSN.	20
3.2	The MLC architecture showing the four operational processes in the MLC WSN protocol designs.	20
3.3	Neighbor distance estimation using the power level received from the neighbor's HELLO message	23
3.4	Cluster-head selections: illustrations of the REQUEST, NOTIFY and REPLY messages. Only nodes within the shaded area will respond with REPLY messages when a REQUEST message is sent by cluster-head <i>s</i>	30
3.5	One round of transmission time line for a cluster with maximum cluster size, <i>M</i> and <i>T</i> transmissions.	35

3.6	Comparison of the flow patterns of MLC and MLC-B protocols for a worst case scenario where a Level-1 cluster-head has to forward flows from all Level-2 to Level- k cluster-heads. The diagram shows only a single chain for simplicity.	38
4.1	Change in active node percentage versus number of transmissions per round (time).	48
4.2	MLC WSNs with different settings of I and J , $T = 100$ in simulations.	51
4.3	Frames received at the sink versus amount of active nodes in %.	52
4.4	Comparison of MLC-B, MLC and multi-path LEACH protocols for percentage active nodes against number of transmissions (a function of how long the network lasted) and number of frames received at the sink. MLC-B Parameters: $I = 8$, $J = 4$ and $\beta = 2$ for Network A , and $I = 2$, $J = 4$ and $\beta = 2$ for network B . MLC Parameters: $I = 2$ and $J = 3$ for Network A , and $I = 10$ and $J = 4$ for network B . $T = 100$ Transmissions/Round in all simulations.	54
4.5	Comparison of the MLC protocol and the MLC-B algorithm introduced at different percentage Active Nodes in an MLC WSN. $I = 2$, $J = 3$, and $T = 100$ in simulations.	55
A.1	The MLC-B protocol for different values of I and J with $\beta = 1$ and $T = 100$ in each simulation.	57
A.2	The MLC-B protocol for different values of I and J with $\beta = 1$ and $T = 100$ in each simulation.	58
A.3	The MLC-B protocol for different values of I and J with $\beta = 2$ and $T = 100$ in each simulation.	59
A.4	The MLC-B protocol for different values of I and J with $\beta = 2$ and $T = 100$ in each simulation.	60

A.5	The MLC-B protocol for different values of I and J with $\beta = 3$ and T = 100 in each simulation.	61
A.6	The MLC-B protocol for different values of I and J with $\beta = 3$ and T = 100 in each simulation.	62
A.7	The MLC-B protocol for different values of I and J with $\beta = 4$ and T = 100 in each simulation.	63
A.8	The MLC-B protocol for different values of I and J with $\beta = 4$ and T = 100 in each simulation.	64
A.9	The MLC-B protocol for different values of I and J with $\beta = 5$ and T = 100 in each simulation.	65
A.10	The MLC-B protocol for different values of I and J with $\beta = 5$ and T = 100 in each simulation.	66

Chapter 1

Introduction

With the rapid technological advancements in both the wireless and sensing devices, the creation of wireless sensor networks (WSNs) may find a broad spectrum of applications. Examples of applications for wireless sensor networks include border surveillance, environment monitoring, industrial process monitoring, tactical systems, etc. In general, sizes of sensor nodes are small and their deployment environments may prohibit the uses of external power sources. It is therefore that the operating life duration of a sensor is at most equal to the life span of its installed battery. Energy optimization is then a critical factor in the system designs and deployments of wireless sensor networks. The more energy effective the system design is, the longer the operating life spans the sensor nodes and the associated wireless sensor network should be.

Signal transmissions in wireless communication medium may easily be corrupted with co-channel interferences and noises. This implies that signals in wireless medium usually suffer relatively high bit error rate (BER). In fact, with the physical size limitation and a reasonable signal-to-interference-and-noise ratio (SINR), the transmission range of a sensor node is limited. It is unrealistic to consider that the data collecting and processing station, the sink, is always within the transmission ranges of all sensor nodes in a wireless sensor network. As a result, it is important to design a wireless sensor network architecture which can operate with an energy effective routing protocol.

Given a fixed maximum transmission range, if a data transmitting sensor is close to a receiving sensor node, then transmitted data can be sent at a lower signal transmission power to save battery power. The communication distance between a sender and receiver can be estimated through pilot signals in the system. Consequently, the transmission

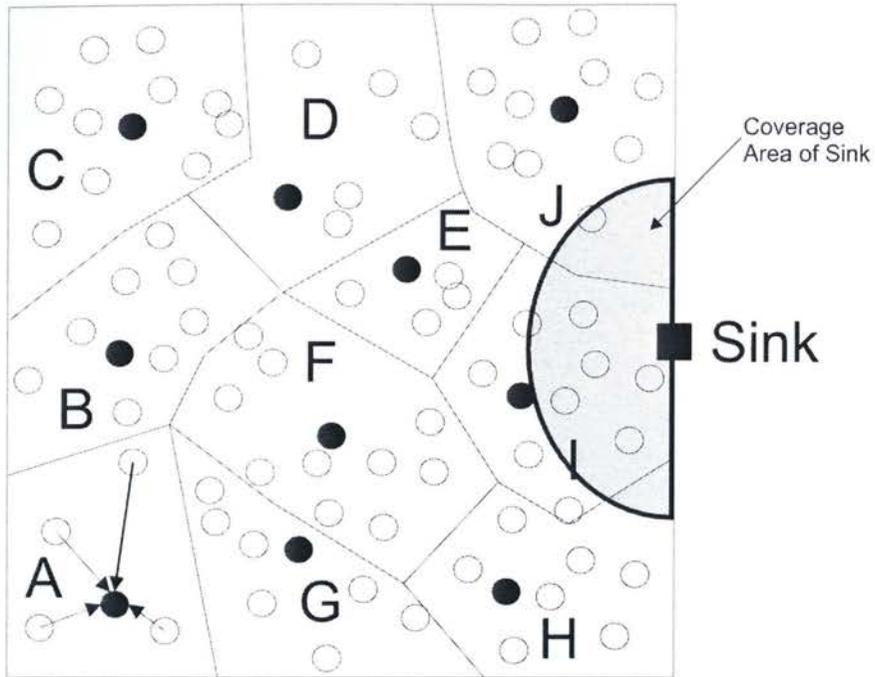
range can be adjusted by tuning to an agreed upon signal strength at the receiving node. This design technique is assumed in our system model which helps avoid wasting energy.

1.1 Problem Formulation

The density of a uniformly deployed wireless sensor network can be measured by the average number of neighboring nodes of a sensor in the network or by the number of nodes deployed per a unit area. In a high density network, a single node has many neighbors. Although, this guarantees better network coverage and longer network lifetime (if the network is power constrained), it has some drawbacks. For example, interference, energy waste, reduced spatial reuse, and MAC protocol overhead are major problems. Several topology control [19] techniques have been proposed (see Chapter 2) to minimize sensor node interference, MAC protocol overhead, and total energy per unit time consumed by the sensor network.

Cluster based topology controls have been shown to be effective [13, 19] but they do not guarantee connectivity of the network when the WSN deployment environment is huge enough that not all nodes are within the transmission range of a sink. Nodes within the sink's transmission ranges are not always selected as a cluster-head due to the randomized nature of cluster-head selection algorithms. Thus, an algorithm may sometimes generate a disconnected graph even though its input is a connected graph. Such a scenario is illustrated in Fig. 1.1 where none of the randomly selected cluster-heads can reach the sink.

Our goal in this thesis is to create an energy-efficient topology control algorithm and its associated protocol designs. If there is a connected graph in the WSN before the application of the algorithm there will always be a connected graph at the end of the algorithm. That is, if without clustering, the sensor nodes can reach the sink by multi-hop transmissions then, after clustering, the cluster-heads must be able to reach the sink by multi-hop transmissions. Thus, after clustering, each cluster-head must have at least



- Signal Flow from Sensor Node to Cluster-Head
- ➔ Signal Flow from Cluster-Head to Cluster-Head or Sink
- Sensor Node
- Cluster-Head
- A to J Clusters

Figure 1.1: Hierarchical topology of a clustered network. Cluster-A shows an example of intra-cluster communications. None of the selected cluster-heads is within transmission ranges of the sink.

one next-hop cluster-head on its path to the sink.

1.2 Thesis Organization

In the following, some related routing system designs for wireless sensor networks are discussed in Chapter 2. The Low-Energy Adaptive Clustering Hierarchy (LEACH) algorithm is the most popular clustering design and its design is reviewed in detail. In Chapter 3, the proposed multi-level clustering (MLC) system design and protocol operations are discussed. Also, a variation of the MLC, multi-level clustering with load balancing (MLC-B) is discussed in Section 3.8.1 of Chapter 3. MLC-B involves load-balancing techniques based on exploiting the redundancy in a densely deployed WSN. Thorough simulations have been carried out and results regarding the performance of the MLC system, MLC-B system, and protocol designs are elaborated in Chapter 4. The proposed design has demonstrated superior performance upon comparing to the LEACH design. Finally, conclusion can be found in Chapter 5.

Chapter 2

Literature Review

Several topology control techniques have been proposed over years to enhance the energy-efficiency of wireless sensor networks. Adaptive modulation [34, 9] has been proposed to help reduce the amount of energy consumed in transmitting one bit of information. This involves applying very low energy but low-bandwidth efficient modulation systems such as MFSK for short range transmissions. Then, higher energy but higher-bandwidth efficient modulation systems such as QPSK for long range transmissions. Also, power control [14, 30] has been proposed for controlling the transmission ranges of nodes in order to reduce the number of neighbors each node can support in a densely deployed wireless network. This technique has the advantage of reducing the amount of MAC protocol overhead required for network connectivity maintenance. However, none of these techniques is sufficient alone.

Dividing a WSN into clusters has been shown to be very effective as it does not only reduce MAC protocol overhead, it also reduces interference and improves spatial reuse. Individual clusters can have different network parameters and characteristics. The different proposed clustered protocols differ in the way the clusters are formed and the topology of the formed clusters: *hierarchical* and *flat*. They also differ on the technique for cluster-head selections. The cluster-head selection technique may be one of or a combination of the following: *distributed*, *dynamic*, *randomized*, *static* - based on sensor node locations, and *sink coordination* - based on an outcome of an optimization algorithm.

2.1 Hierarchical Clustered Protocols

Hierarchical clustered topologies are generally more suitable for WSN applications (especially, for large environment monitoring applications). They allow data fusion and other

common functions within a cluster. It minimizes total number of communications required in a network. Also, total network communication distances are minimized. These factors make hierarchical clustered topology more energy-efficient than flat clustered topology [30].

The topology of clusters in the hierarchy can be single-hop or multi-hop. In a single-hop cluster, a cluster-head is only one-hop away from each of its members. The member nodes form a *star topology* with their cluster-head. In a multi-hop cluster, a member node form a relay node for other member nodes to ensure connectivity between them and their cluster-head. The member nodes form a *multi-hop star topology* with their cluster-head.

The hierarchy in the network is based on the different functions of the sensor nodes undertake in the network. The sensor nodes may be homogenous or heterogenous. Member nodes of a cluster are at the lowest level of the hierarchy and then followed by the cluster-head. The sink can be considered to be at the highest level of the hierarchy. Member node performs the sensing task. A cluster-head coordinates the member nodes of its cluster, performs minimal frame processing, aggregation, and forwarding. Some popular hierarchical cluster-based protocols will be presented in the following subsections.

2.1.1 LEACH

Low-Energy Adaptive Clustering Hierarchy (LEACH) [13] is a popular energy-efficient group-based communications for wireless sensor networks. In LEACH, sensors are partitioned into groups called clusters, and time is partitioned into fixed-length intervals called rounds. Each round begins with a setup phase which includes three sub-phases: advertisement phase for cluster-head selection, cluster set-up phase for cluster member selection, and schedule creation phase for cluster member TDMA schedule creation and notification.

At the beginning of the setup phase, each sensor may become a cluster-head with some

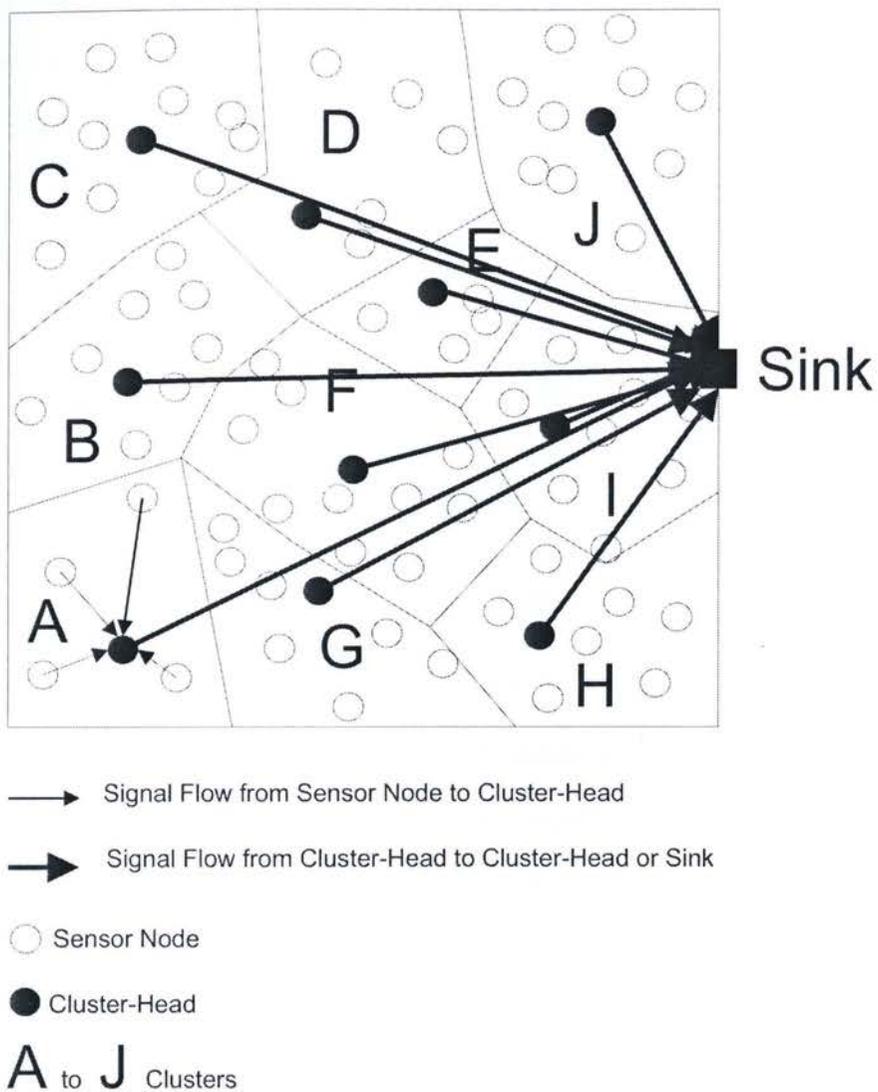


Figure 2.1: Hierarchical topology of the LEACH protocol. Cluster-A shows an example of intra-cluster communications. There is no inter-cluster communication.

predefined a priori probability. Thus, cluster-head selection is randomized, dynamic, and distributed. Cluster-heads broadcast messages to neighboring sensor nodes. If a node receives messages from multiple cluster-heads, it then joins a cluster by choosing the one marked in a received message with the strongest signal strength, i.e, the nearest cluster-head. Thus, they form single-hop clusters. During the round, cluster members send sensed data to their respective cluster-heads, which then aggregate, compress, and route the information to the sink directly. This clustering operation process repeats for every round. Since cluster-heads usually consume more energy than cluster members, this design enables dynamic setup of clusters and cluster-heads in the process. The protocol is designed to ensure that each node in the network becomes cluster-head within the life time of a LEACH WSN.

Unfortunately, LEACH does not consider factors such as the residual energy of sensor nodes, dead nodes, and the relative locations of sensor nodes while choosing cluster-heads. The distributed decision on whether to be a CH in a round does not always yield the optimal number of cluster-heads determined analytically for a network. This results in variation of the number of clusters [33, 29] from the computed optimal value in [12].

Fig. 2.1 shows the hierarchical topology of the LEACH protocol. The sensor nodes in a cluster perform the sensing tasks of an application, and then send the sensed data to their respective cluster-heads. The communication flow pattern in a cluster is illustrated in cluster-A of Fig. 2.1. The cluster-head fuses together all the received frames from its member sensor nodes and forwards it directly to the sink.

LEACH makes an assumption that all sensor nodes can reach the sink with one single hop. But in reality, all sensor nodes have limited transmission ranges, and sensed data may have to be relayed over multiple hops to reach the sink. Thus, it is possible that none of the sensor nodes selected in the randomized technique of the LEACH protocol can reach the sink directly.

There is a variation of the LEACH protocol called LEACH B+ [5] in which the cluster-

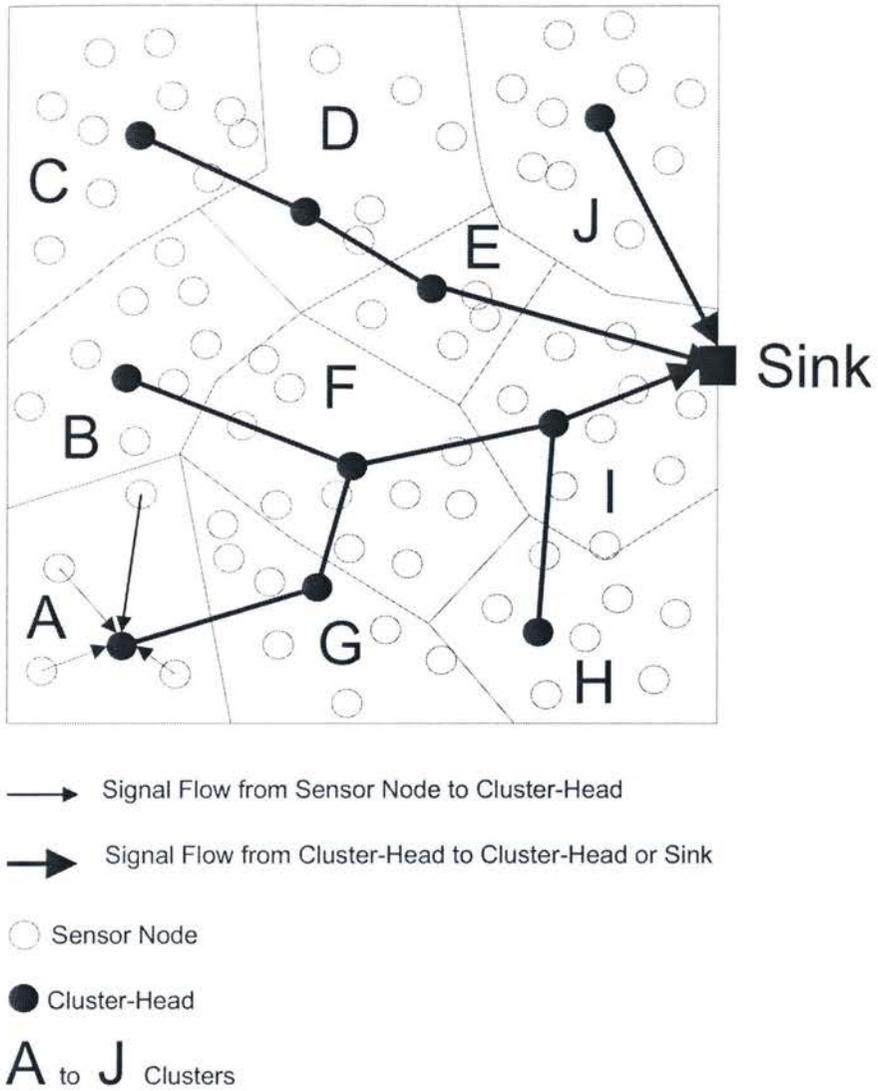


Figure 2.2: Hierarchical topology of the multi-hop LEACH protocol (LEACH B+). Cluster-A shows an example of intra-cluster communications. There is cluster-head to cluster-head inter-cluster communication.

heads use a combination of direct transmission to the sink and multi-hop cluster-head to cluster-head transmission to relay fused frames to the sink. The topology for the LEACH B+ is illustrated in Fig 2.2. A decision on whether to use direct transmission or forward to the next hop cluster-head is made on a hop by hop basis. A direct transmission is applied if the energy required for direct transmission is smaller. The energy required for multi-hop forwarding is the sum of the energy required for forwarding the frame and the energy required for receiving the frame. This gives better performance than the LEACH protocol. However, it is shown in [30] that ensuring a relaying node has better residual energy level than the node it is to relay its signal gives a better performance than simply choosing any node that can relay the signal.

In [21], the functionality of the normal single level cluster is extended with a proxy node. This proxy node is used by a cluster-head when it does not have enough power to transmit its fused signal to the sink. Any node within a cluster can be selected to serve as a proxy node to its cluster-head based on the node's residual energy. The proxy node approach improves the efficiency of the network by ensuring that signal from cluster-heads reach the sink. However, all sensor nodes must be able to transmit directly to the sink for this to be effective. The proxy node is required to transmit directly to the sink just as its associated cluster-head.

2.1.2 DTRAP

In [30], Dynamic Transmission Range Adjustment Protocol (DTRAP) is proposed to ensure every node effectively adjusts its transmission range to keep its neighbor size at a preset maximum neighbor size independent of the node distribution. This guarantees that the cluster size does not exceed the maximum neighbor size. At the end of each round, a cluster-head simply appoints another node within its cluster that has the highest residual energy to be the cluster-head in the next round. This re-clustering technique reduces the number of control messages involved in repeating a full clustering selection

process. The cluster-head selection mechanism in this case is distributed but it is not randomized.

However, this design is only effective where all nodes in the network are within the transmission range of the sink. Otherwise, it is possible that the appointed cluster-head is not in transmission ranges with other nodes in the cluster or the sink.

In [24], comparisons between the effect of residual energy of nodes and the relative positions of nodes in selecting cluster-heads were made. The simulation results showed that selecting cluster-heads with larger number of neighbors would be more important than those with more residual energy in creating energy-efficient clusters. However, neglecting the residual energy of a potential cluster-head might create *gray zones* in the network. The death of a cluster-head within a round means no sensing information from that cluster to the sink.

2.1.3 Multi-level Cluster-heads

In [3, 21], locations of sensor node are modeled as an independent homogeneous spatial Poisson process. The model is used to compute the optimal probability, P_0 , of a node becoming a cluster-head as in the LEACH protocol. In [3], the cluster-heads elect themselves to a new level of cluster-heads with probability, P_1 and the process continues to at most the n -th level. Sensed data are fused at each level and forwarded to the upper level and the n -th level cluster-heads send to the sink. At the first level of the hierarchy, nodes forward cluster-head advertisement messages to their neighbors up to a predetermined at most k number of hops away.

Thus, a multi-hop star cluster topology is formed. The topology is illustrated in Fig. 2.3. Cluster- C shows an example of intra-cluster communication. The shaded regions indicate cluster-heads that belong to the same upper level cluster in which its members are cluster-heads.

The proper values of n and k are not clearly defined. This design is a bottom-up

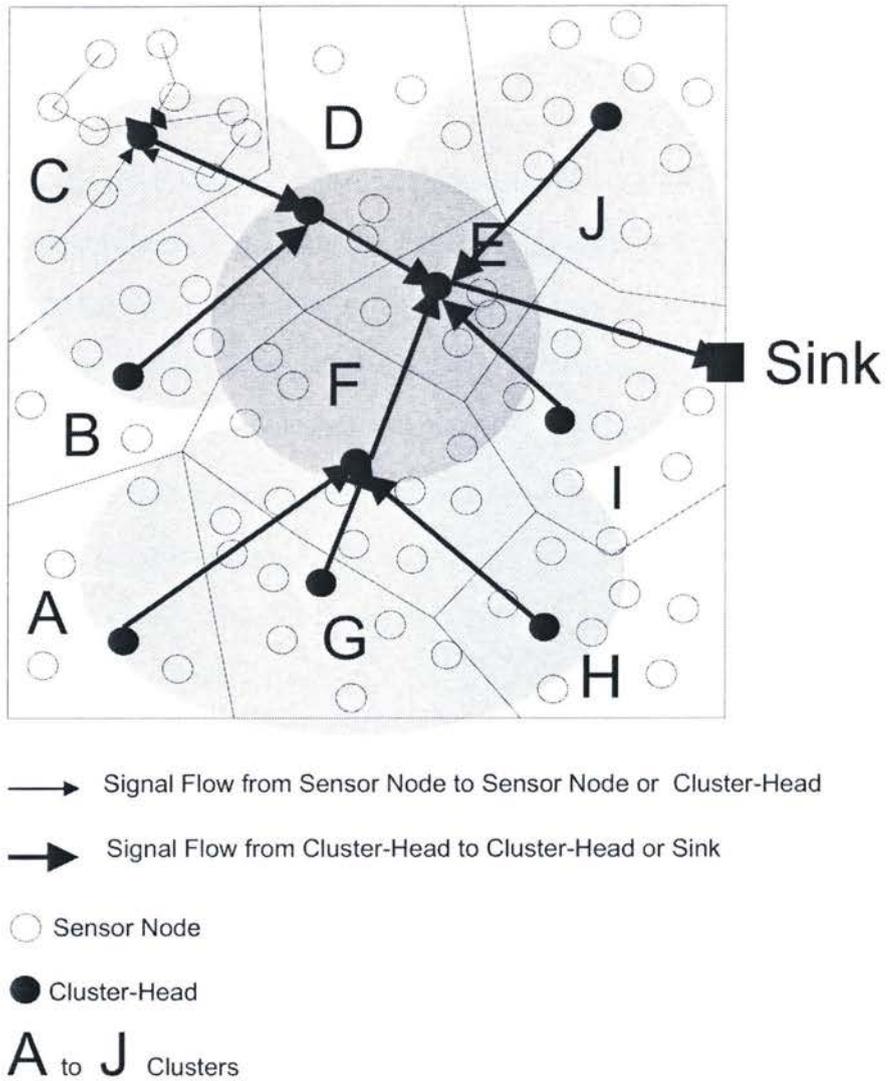


Figure 2.3: Multi-level Cluster-heads topology. Cluster-C shows an example of intra-cluster communications. The shaded regions indicate cluster-heads that belong to the same upper level cluster. There is cluster-head to cluster-head inter-cluster communication.

approach. Although it reduces the number of clusters in the network that has to forward signal to the sink, it is not suitable for a case where not all nodes can reach the sink. Since the multi-level cluster-heads form a pyramid structure, the cluster-heads at the peak of the pyramid is unlikely to reach the sink. Also, the intelligence of the message may be lost after multiple aggregation.

2.1.4 Optimization Algorithm based Protocols

Examples of optimization algorithms include genetic algorithm, simulated annealing, and differential evolution. Optimization of a model of a network can be carried out to determine the optimal parameters for the network, and the roles of nodes in the network: cluster-heads and member nodes. In some cases, there are more than two roles. The requirement of this technique is that the sink must have a method for obtaining current deployment statistics of nodes in the network. It must be able to identify when a node dies in order to adjust the input to the algorithm accordingly. In some protocol designs, nodes need to piggyback their current energy levels and other statistics in every message they send to the sink.

The use of optimization algorithm technique has produced the most energy-efficient protocols but the major drawback is that the network is not easily scalable. All nodes must be able to communicate with the sink directly.

In [20, 16, 18], the sink uses Genetic algorithm to coordinate the formation of energy-efficient clusters and the election of cluster-heads. In [16, 18], the nodes have identical properties and each node can be a cluster-head. A node is set as either 1 or 0. In each round, a node set as 1 is a cluster-head, while a node set as 0 is a sensor node. Thus, the functions of the cluster-heads are limited to receiving sensed frames from their respective members, fusing them, and then forwarding the fused signal to the sink. Single-hop clusters are formed in these protocols and the network topology is the same as in Fig. 2.1 or Fig. 2.2.

In [20], the WSN maintains four nodal states in each genetic generation: **powered off**, **cluster-head**, **inter-cluster router** and **sensor node**. The inter-cluster routers simply form a packet routing path to the sink for the cluster-heads. They do not participate in the sensing operation nor do they belong to any cluster.

2.2 Flat Clustered Protocols

The cluster topology of a flat clustered protocol is a spanning tree. The head of the tree is the cluster-head and it initiates the branching/clustering process. Thus, they form *multi-hop spanning tree* topology which has its root at the cluster-head. Communication pattern in the cluster is not necessarily to or from cluster-head as in a hierarchical clustered topology. It is dependent on the application. There is no fusing of signal at the cluster-head. All nodes in the network are operational and they all have the same sensing function.

Thus, the number of communication messages and the total network communication distances required in a flat clustered topology is higher than it is in a hierarchical clustered topology. The Rapid and Persistent protocols in [22] assign the initiator a budget α . This budget is the number of nodes (including the initiator) that can be supported by a cluster in the network.

In Rapid [22], the initiator accounts for itself and then distributes $(\alpha - 1)$ budget equally among its neighbor nodes. This becomes the number of nodes the respective neighbors can add to the cluster with themselves inclusive. Thus, in a similar fashion, they minus their assigned budget by one and distribute the rest equally to their neighbors. This process continues until the budget is exhausted in each subtree. The size of clusters created in the Rapid algorithm is usually smaller than the initial budget because some nodes end up receiving a budget of more than the number of neighbors available in their neighbor tree. But, Persistent was designed to solve this problem by redistributing unused budget in another tree.

However, if a node fails to receive a budget request after a predetermined waiting period, it initiates a new cluster formation. Thus, it is possible for the sink to trigger the initial cluster formation process to guarantee that sensor nodes within the sink coverage area is part of the cluster-heads. However, there is no guarantee that subsequent initiators will have previous initiators in their transmission ranges. Therefore, there is only a guarantee of inter-cluster connectivity if all the nodes in the network and the sink are within transmission ranges.

Moreover, there is better energy-efficiency in a hierarchial single-hop cluster than in the multi-hop spanning tree cluster. The total communication distances is larger in the multi-hop spanning tree cluster. Multi-hop spanning tree clustering technique is only attractive in a very low density network.

2.3 WSN Physical Layer Properties

WSN is expected to operate in the unlicensed ISM bands [25, 1]. System on Chip (SoC) devices can be implemented for any of these frequency bands. At higher frequency, much smaller SoC device can be implemented because of the advantage of smaller antenna structures at the same antenna gain when compared with lower frequency antenna. Despite advances in VLSI process, which addresses issues of difficulty in fabricating SoC at high frequencies, cost of such process tend to force researchers to favor lower frequency bands which has the advantage of lower cost of fabrications.

Mote2dot [7] operates on 916 MHz, 868 MHz, 433 MHz, and 315 MHz multichannel frequency bands with extended outdoor transmission range of 500 ft at 916 MHz/868 MHz and 1000 ft at 433 MHz/315 MHz. However, empirical results obtained from the experiment conducted by Anastasi *et al.* [2] showed that Mote2dot operating at 4 MHz can reach an outdoor transmission range of 135 m whereas Mote can only reach 55 m at the same frequency.

2.3.1 WSN Communication Patterns

Kulkarni [23] and Demirkol *et al.* [8] defined four common communication patterns for WSNs: *broadcast*, *converge-cast*, *local gossip*, and *multi-cast*.

Broadcast is defined as a type of communication pattern between sensor nodes and the sink. The sink broadcasts the network configurations such as cluster formation parameters or synchronization information periodically to the entire network (if all nodes can be reached from the sink) or a selected number of nodes. In many applications, this is the only communication pattern that exists from the sink to the sensor nodes. This is however different from broadcast frames sent from a sensor node to another sensor node or other sensor nodes.

Sensor nodes use broadcast frames for neighbor discovery. Such scenario is described in section 3.3. In cluster based routing protocols, limited range broadcast frame called multi-cast is used by a cluster-head to notify all nodes in its cluster the scrambling CDMA code to be used within the cluster.

Multi-cast is a type of communication pattern that exists between a cluster-head and its member nodes. The cluster-head uses it to communicate synchronization and other cluster information to its members.

Generally, message frames in a sensor network are either ultimately destined to the sink or from the sink. In environment monitoring applications, sensor nodes route their sensed data to the sink while in a military tactical network, the sensor nodes have a bi-directional communication with the sink. The nodes send sensed data to the sink. The sink performs data analysis as required by the application and advises the nodes on the necessary cause of action to take.

Converge-cast is a communication pattern in which a group of sensors transmit only to a single node. The single node can be a cluster-head or the sink. Thus, the sensed data from all the sensors in the group converge at the cluster-head for processing before

being forwarded towards the sink. However, the sensor nodes can send their sensed data directly to the sink. The method of converge-cast used in a network depends on routing protocol implementation, maximum sensor node transmission range, and network size.

Local gossip occurs when nodes only send sensed data through a broadcast frame just to a limited number of hops. This type of communication pattern is used in the directed diffusion [17], gossiping [11] and PEGASIS [26] routing protocols and their variants.

2.3.2 WSN Energy Loss Activities

Demirkol *et al.* [8] outlined five activities of a WSN that result in waste of energy: *collision*, *overhearing*, *control-packet overhead*, *ideal listening*, and *over-emitting*.

When packets from different nodes collide due to hidden terminal problem, the nodes are forced to retransmit resulting in energy waste. Power capture effect can be used to mitigate this problem. Application of adaptive transmission range using the power amplification technique is a good way to not only mitigate collision but also reduce energy consumption. With this technique, a node transmits with just enough energy required to achieve the acceptable receive sensitivity.

Overhearing occurs when a node receives a packet destined to another receiver. Ideal listening on other hand is a situation where the node continuously listens to the channel for a potential traffic. Both overhearing and ideal listening can be mitigated by duty-cycling the sensing operation which allows nodes to operate in different states.

The sensor node can be in low-energy (sleep mode) state when it has no task to perform and steady state when in full operation. However, the transient state between these two states provides another activity for energy loss - *start-up time*. High energy is required for a node to move from low energy operation state to the steady state. The start-up time is the amount of time spent in this transient state.

The amount of control frames required for the MAC and the routing protocol operations contribute to energy loss. Although the number and size of control frames required

depend on the MAC and the routing protocols, the modulation scheme can ensure that these frames are transmitted at much lower data rates. It can also ensure that it is transmitted with just enough energy since they are mainly meant to maintain neighbor connectivity and WSN is usually very dense.

Over-emitting is a situation where a node transmits when the intended recipient is not available. In the protocol design presented in this thesis, each cluster-head creates a TDMA schedule for its members. Members of a cluster transmit to their cluster-heads only on their TDMA time slots to avoid over-emitting. However, inter-cluster transmissions between cluster-heads are scheduled differently in another time schedule agreed upon by two adjacent cluster-heads.

Other areas of energy waste include *transmission time*, *transmission energy*, and *circuit energy consumption*. Although these three are more related to the implementation of the sensor node than the routing protocol, both the transmission time and the transmission energy are related to transmission distance. Our first goal is to minimize transmission distance.

The amount of time it takes a node to successfully transmit l bits from node A to B is referred to as the *transmission time*. Since the network is very dense and most sensor node to sensor node transmissions are over short distances, the energy required for both the electronics of the transmitter and receiver to maintain full operational state are almost the same. Thus, longer transmission time wastes energy both in the transmitter and the receiver.

The amount of energy the transmitter uses to deliver a bit of information over a unit distance in order to achieve an acceptable SINR, receive sensitivity and BER is referred to as the *transmission energy*. Our second goal is to minimize the energy consumed in frame delivery by minimizing the number of control messages, control message size and control message transmission distance in our protocol design.

Chapter 3

Multi-level Clustering (MLC) WSN

In this chapter, a novel top-down tree-based architectural design for creating multi-level clustering (MLC) wireless sensor networks is proposed. As shown in Fig. 3.1, the root of the MLC tree is always the sink of a wireless sensor network. Depending on the nature of an application, sensor nodes may be randomly and geographically located. And with limited transmission ranges of all sensor nodes and the sink, likely only some nodes can send data directly to the sink. Then in this group of sensor nodes, some of them can be picked to operate as the Level-1 cluster-heads. In other words, the sink is the only Level-0 cluster-head, which is the highest level node in the tree.

Sensor nodes may be spread in a wide geographical area. Multi-level clustering tree-based architecture can be constructed for relaying information from distant node to the sink. The Level-1 cluster-heads is responsible for relaying or aggregating, if exist, the information from the Level-2 clusters to the sink. Obviously, the traditional clustering techniques used in LEACH and its variants do not work effectively in networks requiring relaying. This is because the LEACH design is based on an assumption that all nodes can communicate with each other and the sink. But for most applications, it is impossible to make all nodes to reach the sink or every other node in the network. In such scenarios, LEACH-like protocols create *island clusters* which can be completely isolated, and form *gray zones* in sensing environments.

The communications within the MLC protocol run in rounds. Each round consists of four operational procedures, and they are the **neighbor discovery**, **cluster-head selection**, **cluster-head member admittance**, and **transmission** processes. The first three processes repeat every round and they are followed with T number of transmission

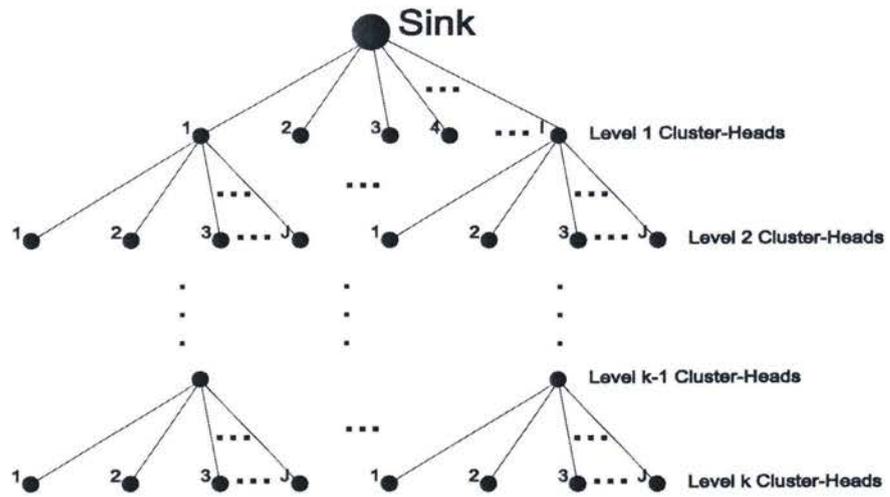


Figure 3.1: Multi-level cluster-heads structure for the MLC WSN.

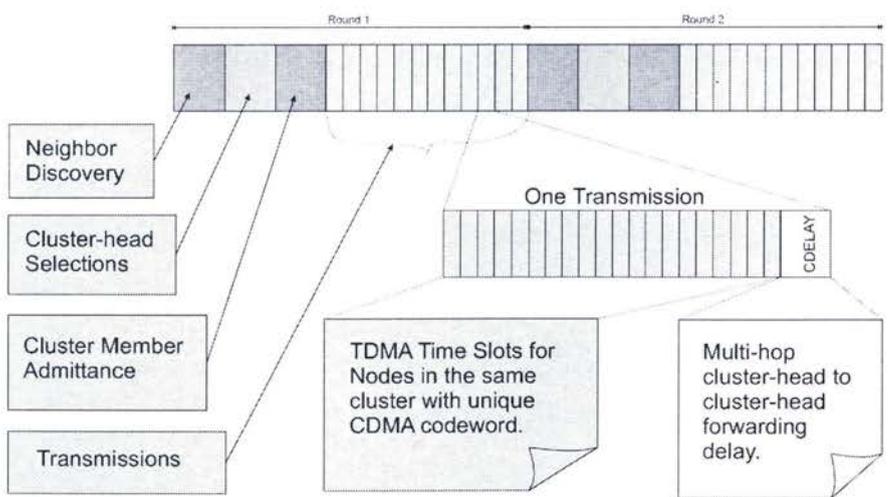


Figure 3.2: The MLC architecture showing the four operational processes in the MLC WSN protocol designs.

operations. This is illustrated in the MLC architecture shown in Fig. 3.2. The cluster delay, `CDELAY` is the time delay it takes a cluster-head to perform the following operations:

- Fuse the signals it has received from its members.
- Forward the fused signal through its `NEXT_HOP` cluster-head to the sink.
- Receive and forward fused signals from its upstream cluster-heads through its `NEXT_HOP` cluster-head to the sink.

The value of T is specified in the `NOTIFY` (Table 3.4) message sent by the sink, and it is subsequently copied into the `PARAM` (Table 3.7) messages forwarded down the tree by the cluster-heads. A combination of TDMA/CDMA schemes can be used to minimize inter- and intra-cluster interferences.

3.1 Network Model

The network model studied in this thesis is an independent randomly distributed sensor network deployed in a 2-dimensional plane. The network can be approximated to be a point spatial poisson process with characteristic:

$$P(N = i) = \frac{(\lambda\pi r^2)^i}{i!} e^{-(\lambda\pi r^2)} \quad (3.1)$$

where $\lambda > 0$ in *nodes/m²* is the node density of the network, N is the total number of nodes in the network, πr^2 is the area covered by each sensor node, and $r = \text{TX_RANGE}$ is the transmission range of each sensor node. Thus, we assume that each node uses a boolean sensing model and that the sensing region of a node is a circle with radius, r . In a boolean sensing model [27, 28], the sensing environment of a sensor node is considered constant and equal to the transmission range of the sensor node. An object within a node's transmission range is considered sensed and not sensed if it is outside the node's transmission range. Boolean sensing model is used in the work reported in this thesis.

The probability of finding a node within an area πr^2 in the network follows a poisson distribution:

$$f_a = 1 - e^{-(\lambda\pi r^2)} \quad (3.2)$$

This is also the percentage area coverage of the network [27, 28]. Thus, in a $1000 \times 1000 \text{ m}^2$ network with $r = 152.4 \text{ m}$ and $N = 2000 \text{ nodes}$ i.e. $\lambda = 0.002 \text{ nodes/m}^2$, $f_a = 1$ or 100%. However, with $N = 200 \text{ nodes}$ i.e. $\lambda = 0.0002 \text{ nodes/m}^2$, $f_a = 99.99995\%$.

However, the amount of redundancy in the network can be expressed as the number of sensors whose sensing areas are fully covered by another sensor or collections of other sensors. This has a direct effect on the performance of energy-efficient protocols which turn off redundant sensors while preserving area coverage. Tian *et al.* [32] used geometry to determine the percentage node coverage (redundancy) in a network. Because of the difficulty in expressing a closed form mathematical model for this problem, they represented their result via simulations.

Assuming boolean sensing model and that all nodes have equal sensing range, a node can only cover another node's sensing area completely if they are both deployed at the same exact position. It requires at least three nodes to completely cover the coverage area of a node if otherwise. This redundancy is exploited in the improvement on MLC aimed at further extending the lifetime of a MLC WSN.

3.2 Energy Model

The energy model used in simulations is based on the one discussed in [12] and it is briefly reviewed. Power attenuation depends on the traveling distance, d , from a sender to a receiver. In free space or short distance without any obstacles, signal power follows the inverse-square law, $\propto 1/d^2$; but for longer distance with multi-path fading, signal power follows $\propto 1/d^4$.

For the simulations carried out as described in Chapter 4, both Friss free space model

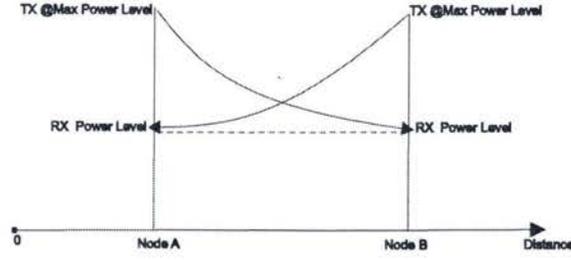


Figure 3.3: Neighbor distance estimation using the power level received from the neighbor's HELLO message

and multi-path fading model are used based on the distance parameter, d . The received power $P_r(d)$ is

$$P_r(d) = \begin{cases} \frac{P_t G_t G_r \lambda^2}{(4\pi d)^2 L} & \text{for } d < d_{crossover} \\ \frac{P_t G_t G_r h_t^2 h_r^2}{d^4} & \text{for } d \geq d_{crossover} \end{cases} \quad (3.3)$$

where P_t is transmitted signal power, G_t and G_r are the transmitter and receiver antenna gains, λ is the wavelength of the carrier frequency, and $L \geq 1$ is the system loss factor not relating to propagation loss. The h_t and h_r are the transmitter and receiver antenna heights, respectively. And the crossover point of the two models is set at

$$d_{crossover} = \frac{4\pi\sqrt{L}h_t h_r}{\lambda}. \quad (3.4)$$

At a sensor node, energy $E_{tx}(l, d)$ is consumed for transmitting l -bit frame through a distance, d , and it is:

$$E_{tx}(l, d) = \begin{cases} l(E_e + d^2 \epsilon_{friss}) & \text{for } d < d_{crossover} \\ l(E_e + d^4 \epsilon_{two-ray}) & \text{for } d \geq d_{crossover} \end{cases} \quad (3.5)$$

where E_e is energy consumed per bit (J/bit) by the transceiver electronics, ϵ_{friss} is the free space power amplification coefficient in $J/bit/m^2$ and $\epsilon_{two-ray}$ is the multi-path fading power amplification coefficient in $J/bit/m^4$. Similarly, the energy $E_{rx}(l)$, consumed upon receiving l -bit frame, is:

$$E_{rx}(l) = l \times E_e. \quad (3.6)$$

If data rate is R_b , the transmit power P_t is:

$$P_t = \frac{E_{tx}(l, d)}{l} R_b. \quad (3.7)$$

Using Eqn. (3.3), (3.5), and (3.7), we can obtain parameters ϵ_{friss} and $\epsilon_{two-ray}$,

$$\epsilon_{friss} = \frac{P_{r-sensitivity}(4\pi)^2}{R_b G_t G_r \lambda^2} \quad (3.8)$$

$$\epsilon_{two-ray} = \frac{P_{r-sensitivity}}{R_b G_t G_r h_t^2 h_r^2} \quad (3.9)$$

With the parameter settings¹ used in our simulations, from Eqn. (3.8) and (3.9), we obtain: $\epsilon_{friss} = 6 \text{ fJ/bit/m}^2$ and $\epsilon_{two-ray} = 0.815 \text{ aJ/bit/m}^4$ where $f = 10^{-15}$ and $a = 10^{-18}$.

In general, a cluster-head consumes more energy than its members to execute its operations as it administers, receives transmissions from all cluster members, and aggregate them into a frame for forwarding. If $E_{CH}(l, n)$ is the energy consumed by a cluster-head with n clustered members, then it is:

$$E_{CH}(l, n) = n \times [E_{rx}(l) + E_f] \quad (3.10)$$

where E_f is the energy required for the fusing operation. Furthermore, a cluster-head may need to relay fused information frames from cluster-heads at lower levels to the sink. Suppose that there are in total F fused data flows including the local cluster required forwarding. Then the total energy, $E_T(l, n)$ required by a cluster-head in one transmission process is:

$$E_T(l, n) = E_{CH}(l, n) + (F - 1)E_{rx}(l) + \sum_{f=1}^F E_{tx}(l, d_f) \quad (3.11)$$

where the d_f is the transmission distance for a cluster-head forwarding flow f . If a cluster-head always has energy to reach its NEXT_HOP cluster-head, then the d_f is constant throughout a transmission process. But with insufficient residual energy, the cluster-head

¹ $G_t = G_r = 1 \text{ m}$, $h_t = h_r = 1.5 \text{ m}$, $R_b = 38.4 \text{ kb/s}$, $\lambda = 0.328 \text{ m}$, $P_{r-sensitivity} = -98 \text{ dBm}$ or 158 fW

may need to invoke a redirection process which changes the distance d_f for the flow upon finding a new re-director.

The first term in eqn. 3.11 is expressed in eqn. 3.10. The second term is the energy consumed in receiving aggregated signals from the cluster-head's upstream cluster-heads. The last term is the energy consumed by the cluster-head in forwarding its own signal and forwarding the signals received from its upstream cluster-heads.

In the design, the required minimum energy operating level, `MIN_ENERGY` of a node is fixed at twice the amount of energy required to transmit the maximum frame size at the maximum transmission range. It switches off itself when its residual energy is below this level. This ensures that a node, low in stored energy, does not get involved in the neighbor discovery process (Section 3.3) and end up polluting the `NEIGHBOR_TABLE` of other nodes. This is because such node with low energy level ends up sending `HELLO` message with a transmission power lower than what is required to reach the maximum transmission range. Such a situation causes those nodes, receiving the `HELLO` message, make incorrect estimates of the distances that this node is away from. Thus,

$$\text{MIN_ENERGY} = 2 \times E_{tx}(l, \text{TX_RANGE}). \quad (3.12)$$

where `TX_RANGE` is the maximum transmission range.

3.3 MLC Neighbor Discovery Process

In the proposed multi-level clustering protocol design, a sensor node selects a random number between $0 - u$ to determine the number of seconds to wait before sending a `HELLO` broadcast message covering its maximum transmission range². u is set to 9 in simulations. This operation is to notify all neighboring nodes regarding the existence of this specific node. During the waiting period, a sensor node listens for neighboring nodes' discovery messages.

²A sensor node transmits at its maximum power to reach sensor nodes at its maximum coverage area with acceptable receive sensitivity. The coverage area of a node is modeled as a perfect circle.

The fields in HELLO message are shown in Table 3.3. The NODEID field is the unique identity of the node sending the message. The ENERGY field indicates the current residual energy level of the node. The NODEID of a node is unique but the ENERGY level of a node decreases with time elapses. However, this condition does not apply to the sink, because the sink is assumed to have energy renewing source.

Table 3.1: Important Node Properties

Property	Description
NODEID	The unique identity of a node.
ENERGY	Current residual energy of a node.
TX_RANGE	Maximum Transmission Range of a node. This value is the same for all nodes including the sink.
CLUSTERID	The unique identity of a cluster. This is equal to the NODEID of the cluster-head of a cluster.
NEXT_HOP	The next-hop cluster-head in the cluster-head's route to the sink.

Table 3.2: NEIGHBOR_TABLE

NODEID	ENERGY	DISTANCE	CLUSTERID	NEXT_HOP
--------	--------	----------	-----------	----------

Table 3.3: HELLO Message

NODEID	ENERGY
--------	--------

Upon receiving a HELLO message from another node, a node registers the received NODEID and the latest ENERGY in its NEIGHBOR_TABLE. The fields for each entry in the NEIGHBOR_TABLE is shown in Table 3.2. If a HELLO with a new NODEID is received for the first time, it randomly picks a number between $0 - u$, and waits for that length of time before replying with a HELLO message. If the received HELLO message is from a

registered node, it simply updates the latest ENERGY in the NEIGHBOR_TABLE. The reason for replying with a HELLO message is to create a bidirectional link between the two nodes and secondly to update other neighbors of its current battery status.

Since HELLO message is sent with transmission powers to reach the maximum transmission ranges, a message receiving node can thus make an estimate of the DISTANCE, d , of the message sending neighbor. For example, from Fig. 3.3, the distance can be interpreted with the energy model discussed in Section 3.2. With the values of parameters used in simulations, the received power at the crossover point is 171 pW. With d 's set to TX_RANGE in Eqn. (3.3) and (3.7), we can obtain the DISTANCE, d , as

$$d = \begin{cases} \sqrt{\frac{P_t G_t G_r \lambda^2}{P_r L (4\pi)^2}} & \text{for } P_r > 171 \text{ pW} \\ \sqrt[4]{\frac{P_t G_t G_r h_t^2 h_r^2}{P_r}} & \text{for } P_r \leq 171 \text{ pW} \end{cases} \quad (3.13)$$

At the end of each round, every node including the sink purges its NEIGHBORS_TABLE and begins a new neighbor discovery process. This takes care the scenario that some nodes might have switched off due to low battery powers based on Eqn. (3.12).

Another technique that can be applied to take care of this effect is for the nodes to periodically send HELLO messages to maintain connectivity. This technique is used in conventional wireless ad-hoc routing protocols such as Ad Hoc On Demand Distance Vector (AODV) Routing protocol [31]. AODV sends three consecutive "Hello" messages back-to-back periodically (every 3 seconds). However, this approach has four drawbacks in an energy-constrained densely deployed WSN:

- The periodic sending of HELLO messages further depletes the energy of the nodes since the neighbor size is large.
- Since nodes of different clusters use different CDMA_CODE for transmission within their cluster, sending of HELLO messages during a transmission process requires a node to switch from its cluster specific CDMA_CODE to the general CDMA_CODE.

- Such high energy transmission results to higher power capture effect and increased intra- and inter-cluster interference.
- The probability of the HELLO messages of two sensor nodes deployed at the same time within transmission range of each other to collide is 10% in this implementation and about 100% if AODV technique is implemented. However, their technique works with better collision probability in IEEE 802.11 products because of the contentious nature of the IEEE 802.11's MAC protocol when working in ad-hoc mode.

3.4 MLC Cluster-head Selections Process

After the neighbor discovery process, a cluster-head can select cluster-heads at the next level. For example, at the Level-0, the sink picks at most I number of nodes from its NEIGHBOR_TABLE as the next level cluster-heads and broadcasts the NOTIFY message. The transmission power level of sending a NOTIFY message depends on the DISTANCE estimates of selected nodes recorded in the sink's NEIGHBOR_TABLE. The selected node with the largest DISTANCE, d , value is used for setting the transmission power of the NOTIFY message, that is,

$$\text{NOTIFY}_{tx_range} = \max_{1 \leq i \leq I} [\text{DISTANCE}]. \quad (3.14)$$

Upon receiving the NOTIFY message, a node firstly checks to see if its NODEID is listed in the message. It simply discards the message if it is not selected.

Table 3.4: NOTIFY message.

NODEID[1]	...	NODEID[I]	T	A	Z	F	J	CLUSTERID
-----------	-----	-----------	---	---	---	---	---	-----------

The design can control the transmission power upon sending the NOTIFY message and help conserve overall network energy. Certainly, reducing the number of nodes that receive the message may potentially extend the life of a node, but may also minimize

the coverage area of the sensor network. We will explore the system performance with regard to the number of cluster-heads selected in the next level in later section.

A node is selected as a cluster-head using a *Maximum Residual Energy Model* which will be discussed in detail in Section 3.4.1. For example, at the Level-0, the sink selects the first I nodes with the lowest COST values. These I nodes form the Level-1 cluster-heads as shown in Fig. 3.1. If the sink has fewer than I neighboring nodes, then the sink may select all of them. In the protocol design, the sink has choices to require confirmation reply messages from these selected I nodes which indicates the acceptance or rejection of the task. The F field in the NOTIFY message is used for this purpose. When F is set to 1, a reply acknowledgement is required. When it is set to 0, the acknowledgement is not needed. An $F = 0$ also indicates that the selected nodes must accept the task of being cluster-heads. The CLUSTERID field is the NODEID of the node initiating the NOTIFY message.

A listed node, upon receiving the NOTIFY message, registers the CLUSTERID field value in its NEIGHBOR_TABLE as the NODEID of its NEXT_HOP node to the sink. Then, it broadcasts a REQUEST message at its peak power to all its neighbors with the CLUSTERID field updated with its NODEID. That is, the CLUSTERID of a cluster-head is the same as its NODEID.

Notice that some neighbors of the Level-1 cluster-heads may include the neighbors of the sink. Therefore, for the nodes within this cluster-head's neighborhood that have not received any REQUEST, NOTIFY or ACK messages from any other nodes, they respond with REPLY messages. This is designed to allow those nodes that are in front of the cluster-head to respond. Or, at least, most of the replying nodes will be in front of the cluster-head. The goal of the algorithm is to select only nodes in front of a cluster-head as members of its next level cluster-heads. Thus, ensuring that the structure in Fig. 3.1 is achieved.

Suppose in Fig. 3.4, the sink has selected a Level-1 cluster-head s . Suppose for simplic-

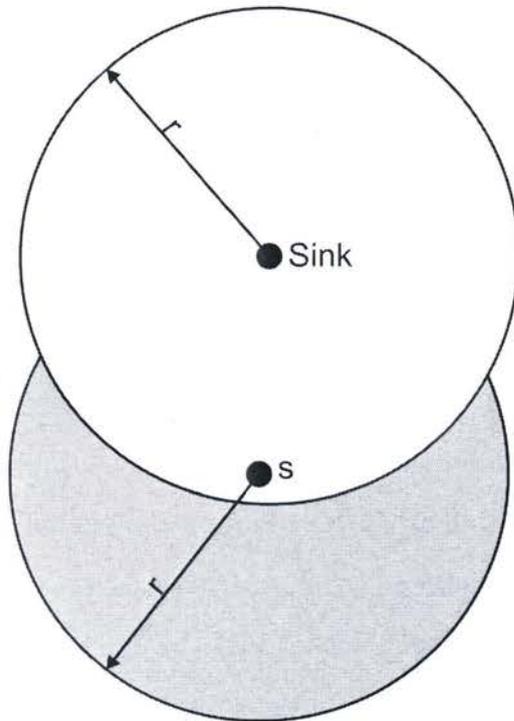


Figure 3.4: Cluster-head selections: illustrations of the **REQUEST**, **NOTIFY** and **REPLY** messages. Only nodes within the shaded area will respond with **REPLY** messages when a **REQUEST** message is sent by cluster-head s .

ity that the cluster-head s is located at a distance close to sink's maximum transmission range. Suppose also that the cluster-head s has sent out a **REQUEST** message at its peak power in invitation of qualified candidates for next level of cluster-heads. Assume the unshaded circle represent the coverage area of the sink and the shaded circle represent part of the coverage area of the cluster-head s that is not covered by the sink. It is only nodes located in the shaded area that will respond with **REPLY** messages. Notice that these nodes are in front of the cluster-head s .

The **NODEID** in the **REQUEST** message is that of the cluster-head sending it. The receiving node uses this field to identify the sender of the message. The **NODEID** and **CLUSTERID** fields in the **REPLY** message are the identities of the node replying it and the cluster-head that has initiated the **REQUEST** message, respectively. The cluster-head

uses the `CLUSTERID` field to identify the unicast `REPLY` message that is intended for it. While it uses the `NODEID` field to identify the node that has sent the reply. The `REPLY` message also contains the current `ENERGY` level of the replying node, and the cluster-head can update it properly in its `NEIGHBOR_TABLE`. The current `ENERGY` levels are critical in selecting the next level cluster-heads.

Table 3.5: `REQUEST/ADVERT` Message

<code>NODEID</code>	<code>TYPE</code>
---------------------	-------------------

<code>TYPE</code>	Message	Description
0	<code>REQUEST</code>	Used by a k -th level cluster-head to send invitation for $(k + 1)$ -th level cluster-heads.
1	<code>ADVERT</code>	Used by a cluster-head to send invitation for member nodes.

A Level- k cluster-head ($k \neq 0$) may select up to J number of next level Level- $(k + 1)$ cluster-heads. The settings of both I and J parameters will be studied in Chapter 4. The value of J is specified in the `J` field of the `NOTIFY` message. Using the maximum residual energy model, J number of nodes are selected from the set of replied nodes, and a `NOTIFY` message is broadcast with a power level determined based on Eqn. (3.14), where in this case I is changed to J . When the `F` field in the `NOTIFY` message is set to 1, an `ACK` or `NACK` message is replied to the cluster-head that has sent the `NOTIFY` message to the intended next level cluster-heads.

When a selected node finds its `NODEID` in the list within the `NOTIFY` message, it replies an `ACK` message to the cluster-head if and only if it has not overheard `ACKs` from other nodes from the time that it sent the `REPLY` message to the time that it received the `NOTIFY` message. This ensures that the selected cluster-heads are not concentrating in one location. Also, this design ensures that proper number of nodes are selected as members of k -th level cluster-heads by a $(k - 1)$ -th level cluster-head, as described in

Fig. 3.1. This guarantees effective area coverage of a network that can reach the sink with a good number of cluster-heads.

If some nodes replied with **NACK** messages, a cluster-head then has to select another set of nodes equal to the number of receiving **NACK** messages using the maximum residual energy model. The cluster-head then sends a **NOTIFY** message to them. Indeed, **ACK** and **NACK** messages may be returned by some or all the nodes again. This process continues until J number of nodes have accepted the task of next level cluster-heads or the list of replied nodes is exhausted. Notice that even though the **J** field is the same in all **NOTIFY** messages, the number of lower level cluster-heads created by each cluster-head in the same level is not always the same.

The newly selected k -th level cluster-heads repeats the same process of selecting $(k + 1)$ -th level cluster-heads until such a time that no more node replies to a cluster-head's **REQUEST** message. At each level, every Level- $(k + 1)$ cluster-head registers the **NODEID**³ of its k -th level cluster-head in its **NEXT_HOP** field of its **NEIGHBOR_TABLE**. This protocol design has an advantage of potentially creating a spanning tree that can be used to forward all sensed data in a network through the tree to the sink consistently.

3.4.1 Maximum Residual Energy Model

The **COST(node)** function of a node, **node** located at an estimated distance, **DISTANCE** from a certain cluster-head which is its neighbor is:

$$\text{COST}(\text{node}) = \frac{E_{ini}}{\text{ENERGY}(\text{node})} \quad (3.15)$$

where E_{ini} is the initial energy state of a node when it is firstly deployed. The **ENERGY(node)** is the latest residual energy level of the node. To simplify the computation, the numerator and denominator in Eqn. (3.15) can be switched, and the division can become a simple bit-wise shift operation; in this case, the node with the maximum residual energy

³This is the **CLUSTERID** field in the **NOTIFY** message.

Table 3.6: REPLY/JOIN/ACK/NACK Message

NODEID	CLUSTERID	ENERGY	TYPE
--------	-----------	--------	------

TYPE	Message	Description
00	ACK	Used by a node to accept the task of cluster-head.
01	NACK	Used by a node to reject the task of cluster-head.
10	REPLY	Used by a node to respond to a REQUEST message indicating that it is available to accept the task of cluster-head.
11	JOIN	Sent by a node to a cluster-head to indicate that it is willing to join its cluster.

should be picked. In other words, the inverse of the $COST(node)$ function is the output of the maximum residual energy model.

Suppose that a Level- k cluster-head has sent out a REQUEST message, if G nodes out of the P neighbors of the cluster-head responded with a REPLY message, then J among these G sensor nodes can be selected as the Level- $(k + 1)$ cluster-heads.

3.5 MLC Cluster Member Admittance Process

Once the cluster-head selection process has been completed, each selected cluster-head broadcasts an ADVERT message at its peak power to all its neighbors. Node then replies with a JOIN message to the cluster-head that it has received with the strongest signal strength. This likely should be the closest cluster-head. In the MLC cluster member admittance algorithm, a cluster-head simply accepts all JOIN messages by sending node-specific ACCEPT messages to all joining nodes.

However, in the modified version of MLC in Section 3.8.1, the parameters A and Z in NOTIFY messages will be used to control the minimum and maximum member nodes a cluster-head can support. A cluster-head may reject some JOIN replies. With fewer

member nodes than the preset minimum number, a cluster-head informs its current members to join another cluster-head. But the cluster-head cannot abandon its role unless its higher level cluster-head disappears in the tree. This technique may reduce the waiting time of cluster-heads with small member size before initiating aggregate message forwarding as described in the next section. Furthermore, it may improve the life span of the network. Smaller number of flows is expected to be forwarded by cluster-heads at the lower levels.

Cluster-heads should cooperate with the immediate upstream, immediate downstream, and same level neighboring cluster-heads to create orthogonal CDMA codewords for their respective clusters. Each cluster-head uses a specific codeword for all intra-cluster communications, while a dedicated codeword is used for cluster-head to cluster-head for multi-hop frame forwarding. This dedicated codeword should be known to all nodes in network.

After the establishments of intra-cluster codewords, cluster-heads set up TDMA schedules for all associated members and notify them through broadcast PARAM messages. The transmission power level of an intra-cluster communication is based on the Eqn. (3.14), where I is changed to SIZE, the membership size of the cluster. The CLUSTERID field in the PARAM message is the NODEID of cluster-head which the nodes use to update their respective CLUSTERID fields.

The NEXT_HOP and CLUSTERID fields of NEIGHBOR_TABLE are usually null for all nodes just after the neighbor discovery process. At the end of the cluster-head selections process, the NEXT_HOP field of cluster-heads contain the NODEID of their respective immediate downstream next hop cluster-head. While the NEXT_HOP field of every other node remains null. However, at the end of the cluster member admittance process, the CLUSTERID field of all nodes that are part of a cluster (including cluster-heads) contain the NODEID of their respective cluster-head. The CLUSTERID field of *inactive nodes* (i.e. nodes that are not part of any cluster) remains null throughout a round.

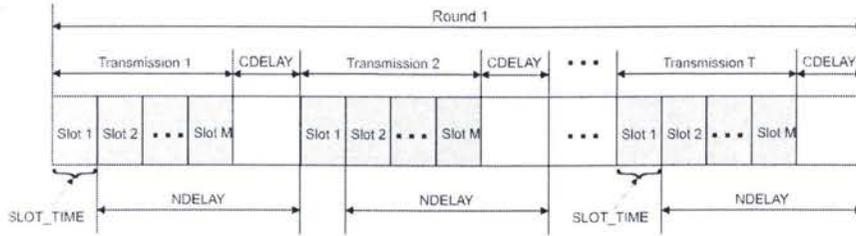


Figure 3.5: One round of transmission time line for a cluster with maximum cluster size, M and T transmissions.

3.6 MLC Transmission Process

In a transmission cycle, each member node transmits its sensed data to its cluster-head in its allocated TDMA SLOT. `SLOT_TIME` is the amount of time allocated to a member node by its cluster-head. The `SLOT_TIME` is constant in simulations because all member nodes are set to send identical l -bit data frame per transmission operation.

Table 3.7: PARAM message.

CLUSTERID	SLOT	SLOT_TIME	CODE	T	SIZE
-----------	------	-----------	------	---	------

Once a cluster-head has received data from all its member sensor nodes, it aggregates the sensed data. It waits for a pre-assigned time until its `NEXT_HOP` cluster-head is ready to receive its aggregate data frame with the cluster-head to cluster-head dedicated codeword. Also with possibility, cluster-head may receive aggregate data from upstream cluster-heads towards the sink, and need to schedule them for forwarding. In the design, its own signal always takes precedence. In the simulations, all nodes and cluster-heads are assumed to operate at constant data rate for simplicity.

For a sensor node, the transmission process may involve T number of transmissions, which is specified in the `T` field of `PARAM` message in a round, r . As depicted in Eqn. (3.16), a node waits for `NDELAY(node, r)` after each frame transmission inside the round r before

transmitting another frame as shown in Fig. 3.5, i.e.,

$$\text{NDELAY}(\text{node}, r) = [\text{SIZE}(h) - 1]\text{SLOT_TIME} + \text{CDELAY}(h) \quad (3.16)$$

where the cluster delay, $\text{CDELAY}(h)$ is the time required to aggregate signal, received aggregated signals from upstream cluster-heads and then propagate each of the signals from a cluster-head h to the sink, and $\text{SIZE}(h)$ is the number of member nodes of h . The $\text{CDELAY}(h)$ largely depends on the estimate of the cluster-sizes of all clusters along the route to the sink and number of flows on the route.

At the end of the T -th transmission, all nodes including the cluster-head purge the `NEIGHBOR_TABLES` and switch to the common codeword. A new neighbor discovery process begins which is followed by a new multi-level clustering process. This cycle continues until the time that the network is no longer functional. Recall that a node switches off when its residual energy is lower than the `MIN_ENERGY` in Eqn. (3.12). Also, when a cluster-head does not have enough energy to transmit its aggregate signal from its members in any operating process in a round, it initiates a redirection protocol.

3.7 Redirection Protocol

In some cases, a cluster-head h has enough energy to successfully receive signals from its members and form a single fused signal. However, it does not have enough energy to transmit it to the `NEXT_HOP` cluster-head. The cluster-head uses the minimum cost model defined in Eqn. (3.17) to choose a re-director.

The re-director node must be a member of the cluster-head with higher residual energy than the cluster-head. This ensures that the re-direction operation balances energy consumption within the cluster.

3.7.1 Minimum Cost Model

$\text{COST}(m)$ is the cost function of a member node, m , of a cluster-head, h in performing the task of a re-director for its cluster-head, h . The cost function is

$$\text{COST}(m) = E_{tx}(l, d_{h-m}) + E_{rx}(l) + E_{tx}(l, d_{m-\text{NEXT_HOP}}), \quad (3.17)$$

where d_{h-m} is the distance between the cluster-head h and the member node m . The $d_{m-\text{NEXT_HOP}}$ is the distance between this m and the NEXT_HOP cluster-head. Thus, energy $E_{tx}(l, d_{h-m})$ is used by h to forward a fused signal to m , and then $E_{rx}(l)$ is the energy consumed by m to receive the forwarded fused signal from h . Also, $E_{tx}(l, d_{m-\text{NEXT_HOP}})$ is the energy consumed by node m for forwarding fused signal to the NEXT_HOP cluster-head.

The node m may possibly serve as a redirecting node for its cluster-head in a transmission process if and only if the following conditions hold true:

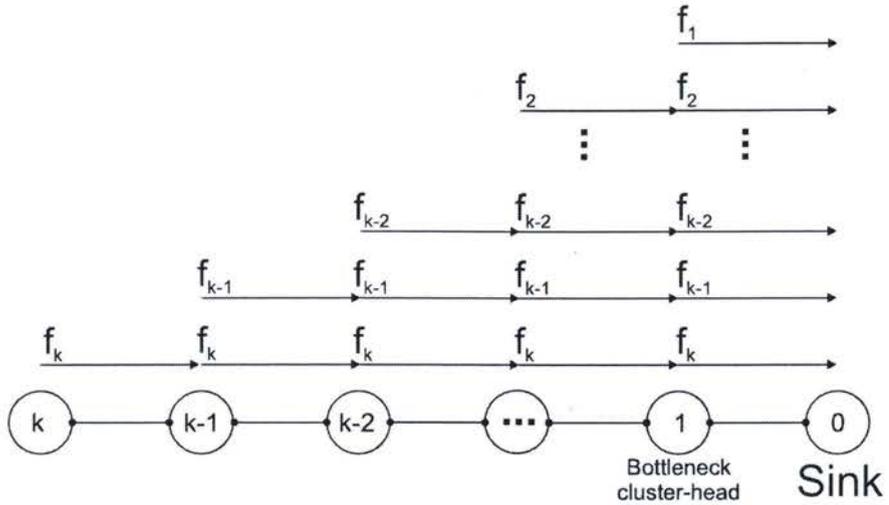
- m is in the NEIGHBOR_TABLE of the NEXT_HOP cluster-head of h .
- The residual energy of h is greater than the energy required for it to reach m ; i.e. $\text{ENERGY}(h) > E_{tx}(l, d_{h-m})$.
- The residual energy of m is greater than the residual energy of h ; i.e. $\text{ENERGY}(m) > \text{ENERGY}(h)$.
- The residual energy of m is greater than the energy required at m in receiving the fused signal from its cluster-head and then forwarding it to its cluster-head's NEXT_HOP cluster-head; i.e. $\text{ENERGY}(m) > E_{rx}(l) + E_{tx}(l, d_{m-\text{NEXT_HOP}})$.
- $\text{COST}(m)$ is the minimum among all nodes that has satisfied the first four conditions.

Therefore, selected re-director, m , for the cluster should be

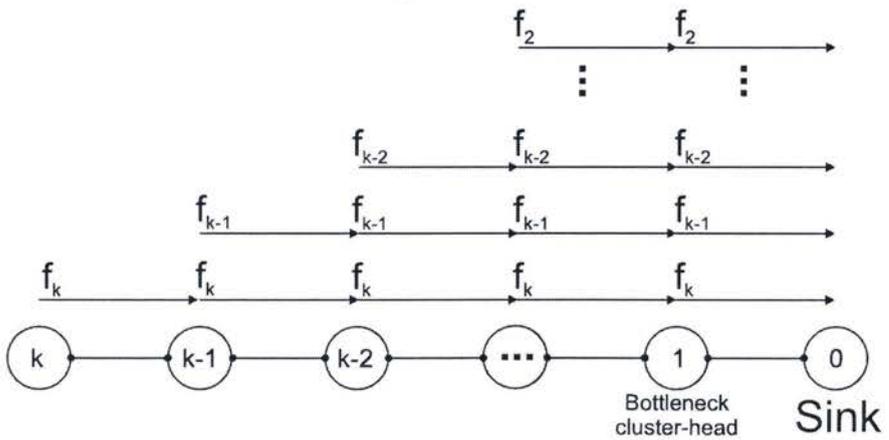
$$m = \arg \min_{1 \leq m \leq Q} \text{COST}(m), \quad (3.18)$$

where Q is the set of nodes with residual energies higher than that of the cluster-head h and can function as a re-director node.

3.8 MLC with Load Balancing (MLC-B)



(a) MLC



(b) MLC-B

Figure 3.6: Comparison of the flow patterns of MLC and MLC-B protocols for a worst case scenario where a Level-1 cluster-head has to forward flows from all Level-2 to Level- k cluster-heads. The diagram shows only a single chain for simplicity.

Although the MLC protocol performed much better than the multi-path LEACH

protocol, it can be seen from Fig. 4.1(a) that 40% (the knee of the graph) of the originally deployed nodes in Network *A* are still alive when the network suddenly became unusable. This is because all nodes within the sink region is dead and all other 40% remaining nodes cannot reach the sink. The nodes in the sink's `NEIGHBOR_TABLE` which form the Level-1 cluster-heads are bottleneck cluster-heads in the network. They perform the task of cluster-head and as well forward their own flow plus all other flows of their upstream cluster-heads.

In a worst case scenario, if all the tree branches of a Level-1 cluster-head is complete in a round up until its Level-*k* cluster-heads, the total number of flows, $F(h)$ the Level-1 cluster-head, h forwards in one transmission processes of that round is:

$$F(h) = \sum_{i=0}^{k-1} J^i \quad (3.19)$$

where J is specified in the `NOTIFY` (Table 3.4) message as the maximum number of lower level cluster-heads an upper level cluster-head can select. Eqn. 3.19 is illustrated in Fig. 3.6(a). However, only a single chain is represented for simplicity - one cluster-head is shown for each level.

Assuming equal cluster sizes, that all flows are forwarded at equal transmission distances, and that all intermediate cluster-heads have enough energy to forward all upstream flows, then $E_{CH}(l, n)$ and $E_T(l, n)$ from Eqns. (3.10) and (3.11) respectively are constants for each cluster-head in a transmission process. Thus, if $E_T(k)$ is the total energy consumed by a Level-*k* cluster-head in one transmission process of a round, the total energy consumed by its Level-1 cluster-head, $E_T(1)$ in one transmission process of that round is:

$$E_T(k) = E_{CH} + E_{tx} \quad (3.20)$$

$$E_T(1) = [E_T(k) - E_{CH}]F(h) + E_{CH} + [F(h) - 1]E_{rx} \quad (3.21)$$

From Eqn. (3.21), the energy consumption of the cluster-heads reduces asymptotically as their distance from the sink increases. Therefore, a model that will result in better load balancing must reflect this situation.

However, WSN node energy consumption balancing and cooperation techniques proposed for LEACH like protocols and all other protocols for WSNs that assume that all nodes can reach the sink directly cannot work for this scenario because they are based on optimizing the minimum number of hops the fused signal must be forwarded before forwarding it directly to the sink. This requires that the node at the optimum hop must be able to reach the sink directly.

However, some other energy balancing techniques [4, 15] rely on the sensor network node characteristic distance, d_{char} in determining optimum number of hops to the sink. Howitt *et al.* in [15] also used shorter hop distances for frame forwarding closer to the sink than away from the sink. Although these techniques reduce energy consumption at per node level, it does not result in better network performance in addressing the issue of bottleneck that occurs at the cluster-heads close to the sink.

Haenggi *et al.* [10] proposed several energy balancing techniques based on avoiding shorter multiple multi-hop transmissions in favor of long haul transmissions to use as few number of hops as possible in reaching the sink. However, the nodes within the sink region will still be overloaded unless if all nodes in the network can reach the sink. In which case, nodes far away from the sink can alternate between sending directly to the sink and sending through multiple multi-hop transmissions. A cluster-head uses direct transmission when its residual energy is higher than that of other cluster-heads it should have used for multi-hop transmissions. Also, it uses direct transmission when the energy required for direct transmission is smaller than the energy required for multiple multi-hop transmissions. In order to determine this energy, a synchronized link state information must be propagated regularly across the network. Technique for propagating link state information to all the nodes is not described in [10].

The proposed algorithm (MLC-B) is aimed at reducing the number of upstream flows the nodes close to the sink has to forward in each transmission process. In the MLC-B algorithm, the Level-1 cluster-heads in Fig. 3.1 only serve as *Inter-Cluster-Routers* (ICR). They do not advertise for members but only serve to forward flows from their upstream cluster-heads to the sink. Since in a dense network, not all the nodes in the NEIGHBOR_TABLE of the sink become cluster-heads, other nodes in the neighborhood of the sink and the Level-2 cluster-heads can join a Level-2 cluster-head. This ensures that there is sensing coverage within the sink's region.

However, relieving the Level-1 cluster-heads the task of coordinating a cluster saves them both the energy required to receive sensed data from member nodes, fusing them and then forwarding the aggregate signal to the sink. It follows that Eqns. (3.19) and (3.21) can be re-written as:

$$F(h) = \sum_{i=1}^{k-1} J^i \quad (3.22)$$

and

$$E_T(1) = [E_T(k) - E_{CH} + E_{rx}]F(h). \quad (3.23)$$

Comparing Eqns. (3.21) and (3.23) indicates the amount of energy gained in the Level-1 cluster-heads. The output of the comparison is $E_T(k)$ which is illustrated in Eqn. (3.20). Eqn. 3.22 is illustrated in Fig. 3.6(b). However, only a single chain is represented for simplicity - one cluster-head is shown for each level.

3.8.1 MLC-B Cluster Member Admittance

The cluster member admittance process in MLC-B protocol is in two stages. The first stage is similar to the cluster member admittance process in the MLC protocol (Section 3.5). The differences are as follows:

- The maximum number of members a cluster-head can support is controlled by Eqn. (3.24) in the MLC-B protocol.
- Level-1 cluster-heads have zero member sizes. Thus, Level-1 cluster-heads in the MLC-B WSN serve only as relay sensor nodes called ICRs. They do not perform sensing operation but simply forward signal from their upstream cluster-heads to the sink. However, other sensor nodes in sink's neighborhood not chosen as cluster-heads can join a Level-2 cluster-head (if possible) to ensure that the sink's region is covered.
- The minimum supported cluster-size in the MLC-B WSN is indicated in the A field of the NOTIFY messages.

However, in the second stage, a cluster-head that has member nodes lower than the value in A field of the NOTIFY messages must relinquish its members but it does not abandon its task as a cluster-head. Thus, it serves only as ICR for its upstream cluster-heads. However, a cluster-head abandons its task as a cluster-head if and only if it does not have an upstream cluster-head connected to it.

A cluster-head that has relinquished its task of cluster-head must initiate a JOIN message to its downstream cluster-head. It becomes an inactive node in that round if the join attempt fails. Also, its previous member nodes must attempt to join their respective next cluster-head they received stronger ADVERT messages from by sending a JOIN message to the cluster-head.

Table 3.8: Modified NOTIFY message.

NODEID[1]	...	NODEID[I]	T	A	Z	F	J	K	CLUSTERID
-----------	-----	-----------	---	---	---	---	---	---	-----------

Two new message types are defined for managing the cluster admittance process in MLC-B protocol. The ACCEPT and REJECT messages (Table 3.9). A node receives an ACCEPT message if its join request is successful. Otherwise, it receives a REJECT message.

Table 3.9: ACCEPT/REJECT Message

NODEID	CLUSTERID	TYPE
--------	-----------	------

TYPE	Message	Description
0	REJECT	Sent by a cluster-head to a sensor node in response to a JOIN message to indicate its unwillingness to admit it as a member of its cluster in that particular round.
1	ACCEPT	Sent by a cluster-head to a sensor node in response to a JOIN message to indicate its willingness to admit it as a member of its cluster in that particular round.

If a node receives a REJECT message, it simply sends JOIN message to the next cluster-head from which it received high signal strength ADVERT message. However, if a node fails to successfully join a cluster-head before running out of the list of cluster-heads from which it received ADVERT messages, it simply becomes an inactive node in that round. A cluster-head also uses the REJECT message in notifying its members to join another cluster when it has cluster member size smaller than A .

Similarly, a cluster-head that has member size equal to its maximum cluster size must reject all subsequent join attempts. Thus, cluster admittance process in the MLC-B protocol is on a first come first serve basis (FCFS). The upper bound of the maximum cluster size in the MLC-B WSN is specified in the Z field of the NOTIFY messages. However, the actual maximum cluster size of a cluster-head is dependent on the Level the cluster-head belongs. Thus, the maximum cluster size, $M(h)$ of a cluster-head, h in a round, r is:

$$M(h) = \min[(A + \beta K), Z] \quad (3.24)$$

where K is the level the cluster-head belongs and β is a non-negative integer weighting factor that determines the distribution of cluster sizes in the MLC-B WSN. If $\beta \geq 1$, the

distribution of $M(h)$ is arithmetic with the last term less than or equal to Z . However, at $\beta = 0$, $M(h) = A$ which is constant for all cluster-heads in the network except for Level-1 cluster-heads that have a different condition. Level-1 cluster-heads have zero member size in a MLC-B WSN. We set $A = 4$ and $Z = 20$ in simulations.

Eqn. 3.24 ensures that cluster sizes in the MLC-B WSN has a lower bound of A and an upper bound of Z . The cluster sizes gradually increases from Level-2 cluster-heads to Level- k cluster-heads. This ensures reduction in cluster related load on higher level cluster-heads (cluster-heads close to the sink) while increasing that of lower level cluster-heads. Cluster related loads include: cluster member admittance messages, TDMA schedule creation, receiving signals from cluster members, fusing of the received signals, forwarding the received the fused signal towards the sink through its NEXT_HOP cluster-head, and possible receiving and forwarding fused signals from its upstream cluster-heads. The smaller the cluster size, the smaller the amount of energy consumed in performing these tasks. However, enforcing a lower bound cluster size reduces the number of flows forwarded by higher level cluster-heads.

The NOTIFY message has been modified (Table 3.8) to include K field which holds the information of the level of the node receiving the message. Thus, a Level- $(k-1)$ cluster-head updates the K field to k before sending NOTIFY message.

Chapter 4

Simulation and Results

As shown in Table 4.1, two wireless sensor network models are used in simulations. The network A is a relatively dense network model, while network B is a relatively sparse network model. Network A is randomly generated with a total of $N = 2000$ uniformly positioned sensor nodes in an $1000 \times 1000 \text{ m}^2$ square-shaped area. The node density is 0.002 nodes/m^2 . The maximum transmission radius is 152.4 m [7], and an average number of neighboring nodes is 146 nodes.

Similarly, the second network, network B , is also randomly generated with a total of $N = 200$ uniformly positioned micro-sensor nodes in an $1000 \times 1000 \text{ m}^2$ square-shaped area. The node density is 0.0002 nodes/m^2 . The average number of neighboring nodes is 15 nodes with the maximum transmission range of 152.4 m . There are 63 neighboring nodes to the sink in network A , and it is only 4 nodes in network B . The sink is located on the right side at an x-y coordinate (1000, 500) in metres in both networks.

An initial energy of 2 *Joules* is assigned to each micro-sensor node. Other sensor node parameters are based on Mica2dot¹ specifications [7] (see Table 4.2).

In our simulations, a network is considered unusable when only fewer than 10% of all sensor nodes are still alive, or if there are no nodes that can reach the sink. A sensor node is considered dead and switches off when its residual energy is less than the `MIN_ENERGY` value in Eqn. (3.12).

A sensor node may be switched on but *inactive*. In a round, an inactive node is not linked to any cluster-head, and hence, it does not participate in the sensing operations. Hence, a node can go into *sleep mode* and wake up at the end of the round. From Fig. 3.5

¹Mica2dot is a third generation wireless micro-sensor node designed specifically for deeply embedded wireless sensor network.

Table 4.1: Parameters for network *A* and network *B*.

Parameter	Network <i>A</i>	Network <i>B</i>
Total number of nodes, N	2000	200
Network dimension (in metres)	1000×1000	1000×1000
Node density (nodes/ m^2)	0.002	0.0002
Average number of neighbor nodes	146	15
(x, y) position of sink (metres)	(1000, 500)	(1000, 500)
# of sink's neighboring nodes	63	4
TX_RANGE (in metres)	152.4	152.4

Table 4.2: Reference Parameters

Parameter	Value	Parameter	Value
$\epsilon_{friss-amp}$	6 fJ/bit/m^2	E_e	50 nJ/bit
$\epsilon_{two-ray-amp}$	0.815 aJ/bit/m^4	E_f	5 nJ/bit
$G_t = G_r$	1 m	R_b	38.4 kbps
$h_r = h_t$	1.5 m	L	1
$d_{crossover}^2$	87 m	λ	0.328 m
TX_RANGE	152.4 m	l	4000 bits

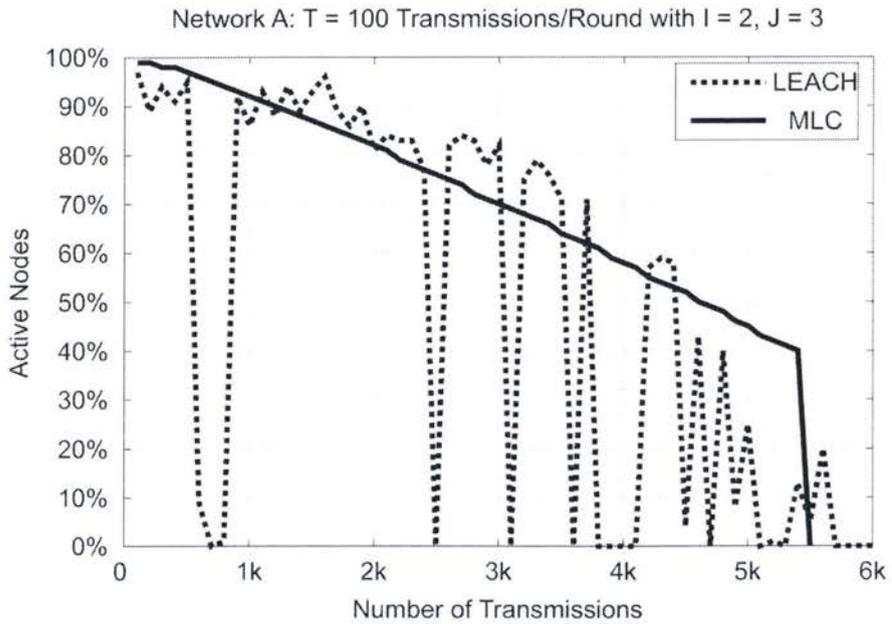
which shows the time line of transmission operations, a node can estimate the length of time it can spend in the sleep mode.

Furthermore, clusters may form but sensed data from their associated members cannot reach the sink. This is because no path can be established to the sink. These clusters are called *island clusters*, within which the nodes are also considered inactive. That is, the cluster-head in an island cluster does not have a NEXT_HOP cluster-head, and the cluster-head is not in the NEIGHBOR_TABLE of the sink. Island cluster is a worse scenario because energy has been consumed but wasted in sensing information, as the sensed signal does not reach the sink.

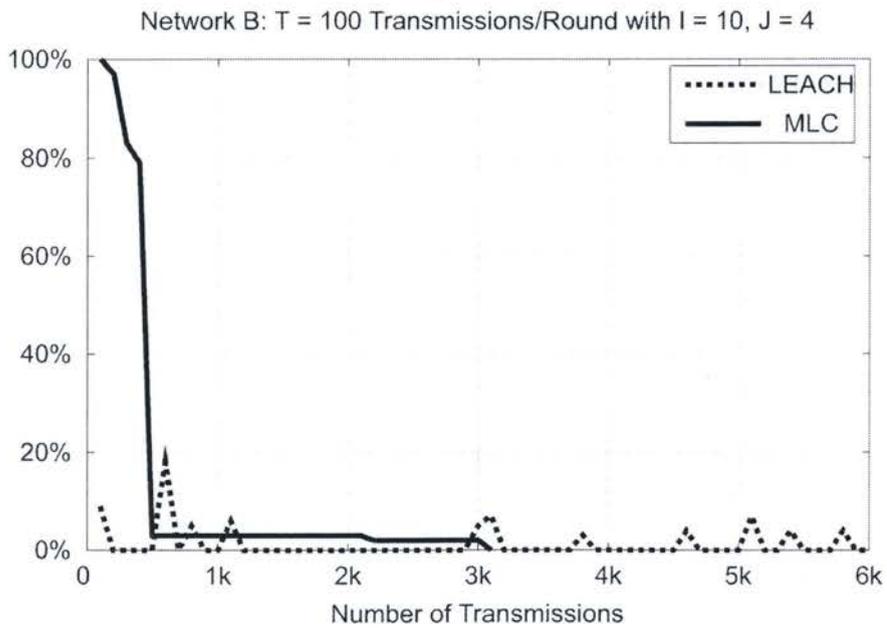
4.1 MLC Simulations and Results

From the plotted results shown in Fig. 4.1, island clustering does not occur in the proposed MLC protocol. The proposed clustering technique forms a tree from the sink, and cluster-heads are generated as by-products. However, the island clustering problem is noticeable in the LEACH protocol. The effect is more prominent in a relatively sparse network, as shown in Fig. 4.1(b). This is because the randomized nature of cluster-head selection in multi-hop LEACH does not guarantee that a node within range of the sink is selected as cluster-head in a round. In sparse network, the probability of a node in the sink NEIGHBOR_TABLE becoming a cluster-head is much smaller.

Even if a few nodes within the range of the sink is selected, they might still be not enough to sustain the network for the entire period of the round. There are multiple occurrences that multi-hop LEACH protocol recorded zero number of transmissions in Fig. 4.1(a). It means that there are no cluster-heads in the neighborhood of the sink. The percentage active nodes is zero in this case. In Fig. 4.1(b), this happens always for the multi-hop LEACH protocol in network *B*. The results indicate that the MLC protocol ensures good coverage of the network provided, and it makes good use of the sensor nodes within the transmission range of the sink.



(a)



(b)

Figure 4.1: Change in active node percentage versus number of transmissions per round (time).

Certainly, the performance of MLC protocol may depend on the parameters of I field and J field as shown in Fig. 4.2. It showed that for both sparse and dense networks, $J = 1$ gives erratic result with higher variation in a more dense network. There is low network coverage at $J = 1$.

Recall from cluster-head selections process in Section 3.4 that a Level- $(k-1)$ CH selects only nodes in front of itself as members of its Level- k cluster-heads. Observing the graphical user interface of the simulator showed that when cluster-heads are selecting just one node (i.e. $J = 1$), there is high probability that the coverage area of a newly selected CH is already covered by other higher level cluster-heads. Thus, the CH will not receive any **REPLY** message in response to its **REQUEST** message. This causes the algorithm to terminate prematurely and it results in low network coverage. Since, there is higher node redundancy at higher network density, the effect of this increases with increase in the sensor network density. However, lower values of I performs better in a dense network in Fig. 4.2(a). Larger value of I performs better in a sparse network (Fig. 4.2(d) and 4.2(f)).

Notice that Fig. 4.2(d) and 4.2(f) are the same despite their different values of I . The reason is that there are only 4 nodes within the transmission range of the sink (see Table 4.1) in network B . Based on the maximum residual energy model algorithm in Section 3.4.1, all neighboring nodes of the sink are selected as Level-1 cluster-heads if $I \geq P$, where P is the number of nodes in the **NEIGHBOR_TABLE** of the sink. Thus, $I = 4$ is the actual value used by the protocol for both Fig. 4.2(d) and 4.2(f) despite different values of I are set.

Fig. 4.3 shows the number of aggregate frames sent from active nodes received by the sink. The result further depicts that the multi-hop LEACH protocol is unreliable. In some rounds, no frame can arrive at sink. The performance improvement is 17% in terms of the total number of frames received at the sink by the MLC protocol in network A upon comparing to the multi-hop LEACH protocol (see Fig. 4.3(a)). A 300% performance

increase is achieved in network B (see Fig. 4.3(b)) for the MLC design. The difference in the area under the graphs indicates that the MLC protocol guarantees better coverage and more reliability in delivering sensed data to the sink.

4.2 MLC-B Simulations and Results

Extensive simulations were performed with different network parameters of the MLC-B protocol from which we determined the optimal values of the fields of the NOTIFY messages to be: $I = 8$ and $J = 4$ for Network A (see Fig. A.4(a)), and $I = 2$ and $J = 4$ for network B (see Fig. A.3(b)). Interestingly, we discovered that an optimal value of $\beta = 2$ is independent of density of the network. Details of the performance of MLC-B protocol with different parameters can be found in Appendix A.

Simulation results shown in Fig. 4.4 gives a summary of the performance of the various protocols discussed in this report. It can be seen from Fig. 4.4(a) that the MLC-B lasted much more longer than MLC protocol while utilizing 20% (compare the knee points on the graph) more nodes than the MLC protocol. MLC-B recorded 58% performance improvement on the life time of the network A when compared with MLC. MLC-B also recorded 75% performance increase (see Fig. 4.4(c)) on the number of frames that is received at the sink when compared with the MLC protocol in network A . However, there is a performance decrease in the network B case. This shows that the performance of the load balancing algorithm is dependent on the density of the network.

The MLC-B protocol ensures that the cluster sizes of higher Level cluster-heads are smaller than that of lower Level cluster-heads. Eliminating unnecessary proliferation of clusters in the network while increasing the number of inter-cluster-routers reduced the amount of flows that the Level-1 and other lower level cluster-heads has to forward. These resulted in improved life-time of the MLC-B WSN. Note that it is necessary to have a progressive decrease in the number of flows generated and energy consumed across the network rather than simply concentrating on the nodes just in the NEIGHBOR_TABLE

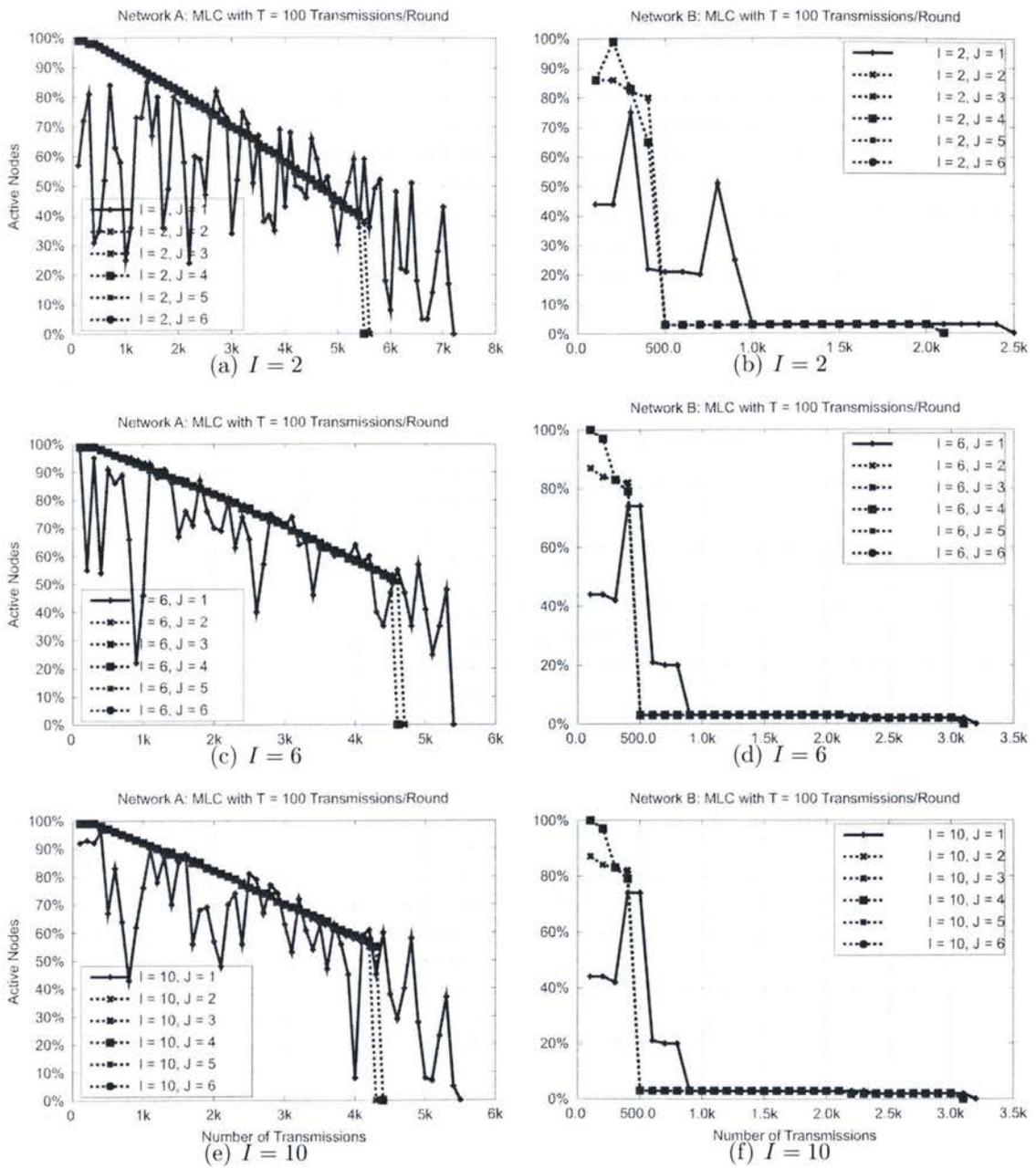
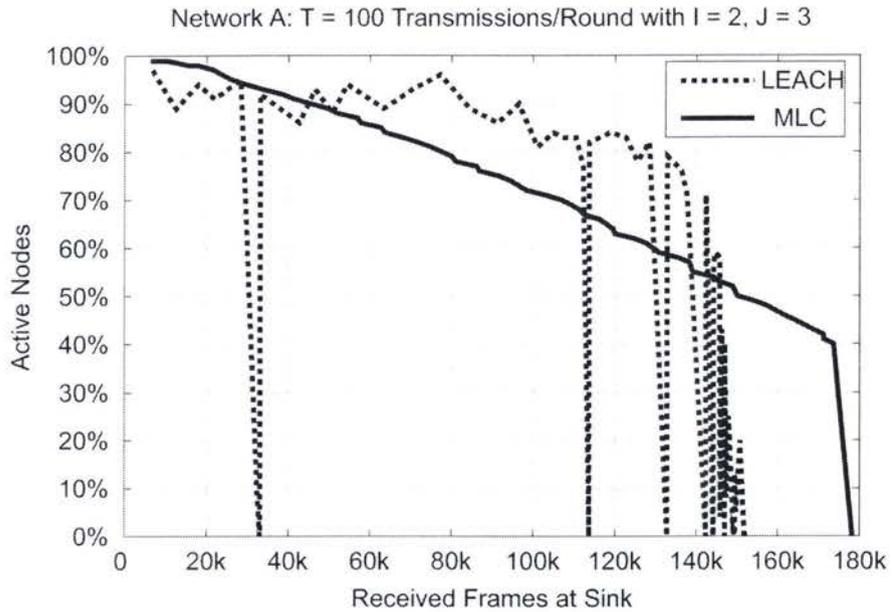
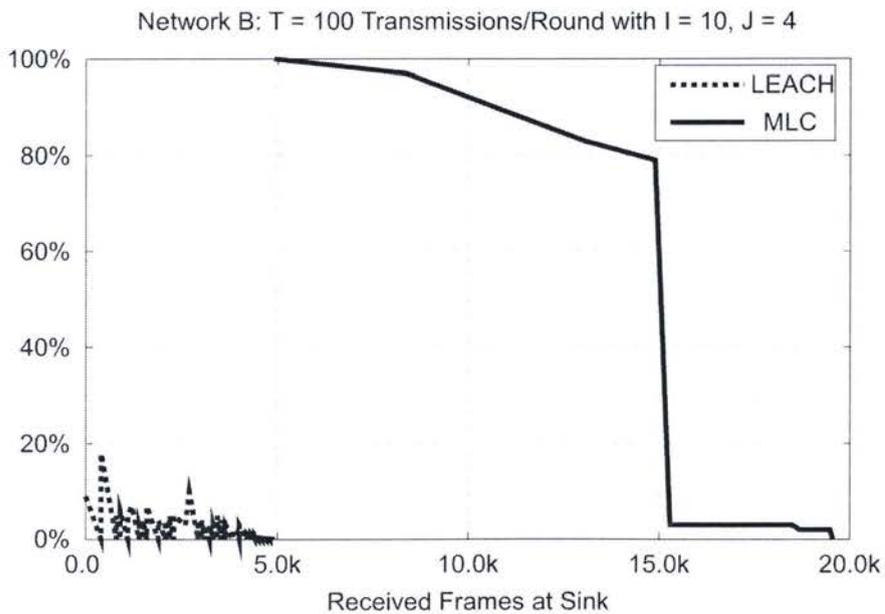


Figure 4.2: MLC WSNs with different settings of I and J , $T = 100$ in simulations.



(a)



(b)

Figure 4.3: Frames received at the sink versus amount of active nodes in %.

of the sink. The reason is that if any Level cluster-heads are over loaded, then that Level becomes the new bottleneck Level.

MLC-B algorithm follows the ideology in an energy-saving technique, Span [6]. Span is proposed for IEEE 802.11 as a layer 2.5 algorithm which conventional routing protocols such as Dynamic Source Routing (DSR) or AODV can run on seamlessly to conserve the energy consumption of individual IEEE 802.11 nodes and that of the wireless ad-hoc network. Span is based on the observation that an area of a shared-channel wireless network with a higher node density than what is required to cover the area can operate at an optimum node density to coverage area ratio while achieving node energy conservation and improved network lifetime. In Span, nodes use a distributed, randomized algorithm to make a local decision on whether to sleep, or join a forwarding backbone.

Network lifetime in Span is shown to increase as the ratio of idle-to-sleep energy consumption increases and as the density of the network increasing. In agreement with Span, the MLC-B resulted in a WSN lifetime performance improvement of 58% in Network A which has a high network density of $0.002 \text{ nodes}/m^2$ but it resulted in a WSN lifetime performance decrease in Network B which has low network density of $0.0002 \text{ nodes}/m^2$. Simulation results in Fig. 4.5 shows that as the point where the energy-saving algorithm of MLC-B is introduced increases from 50% Active Nodes (see Fig. 4.5(b)) to 70% Active Nodes (Fig. 4.5(d)), the WSN lifetime performance also increases. This also agrees with the observation in Span.

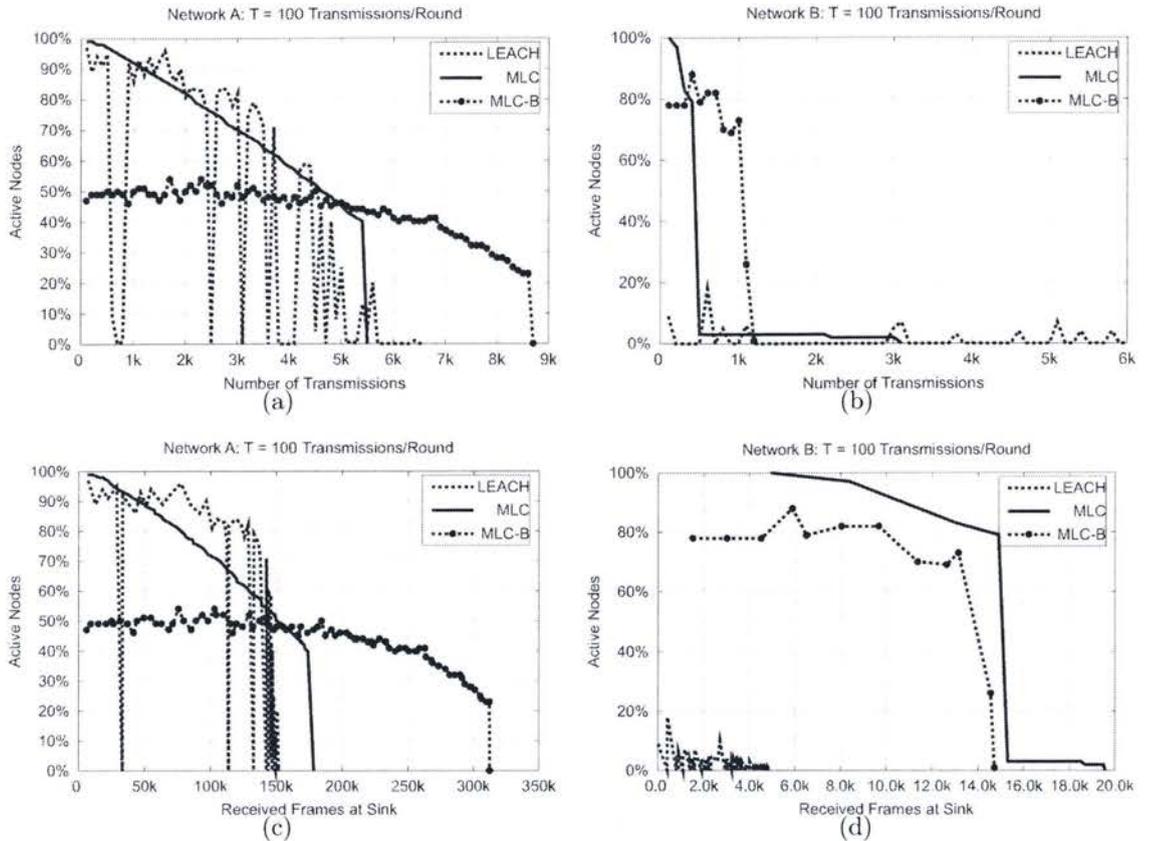


Figure 4.4: Comparison of MLC-B, MLC and multi-path LEACH protocols for percentage active nodes against number of transmissions (a function of how long the network lasted) and number of frames received at the sink. **MLC-B Parameters:** $I = 8$, $J = 4$ and $\beta = 2$ for Network A, and $I = 2$, $J = 4$ and $\beta = 2$ for network B. **MLC Parameters:** $I = 2$ and $J = 3$ for Network A, and $I = 10$ and $J = 4$ for network B. $T = 100$ Transmissions/Round in all simulations.

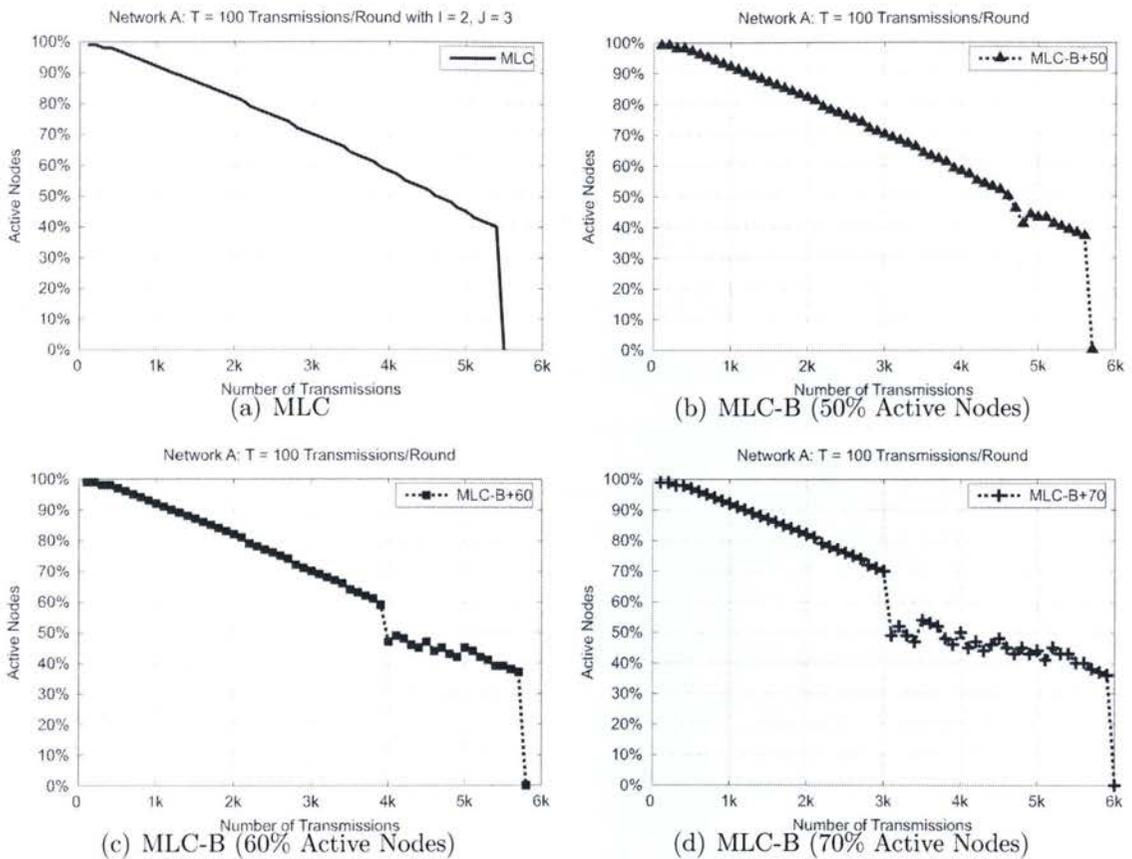


Figure 4.5: Comparison of the MLC protocol and the MLC-B algorithm introduced at different percentage Active Nodes in an MLC WSN. $I = 2$, $J = 3$, and $T = 100$ in simulations.

Chapter 5

Conclusions

In this thesis, a novel design of multi-level clustering (MLC) architecture and protocol have been proposed for interconnecting sensor nodes in a wireless sensor network. The design is especially useful in WSN when some sensors do not reach the sink directly. With the energy constraints in WSNs, traditional wireless multi-hop routing protocols are not suitable. But the proposed design can create a highly interconnected tree graph in the network, and the sensed data can get to the sink from far away. Simulation results indicate that the uses of other LEACH alike designs are not appropriate because majority of the frames generated by clusters can not reach the sink. The proposed design offers a sensor network a wide-area coverage, and generated frames can reach the sink.

Appendix A

MLC-B Parameters

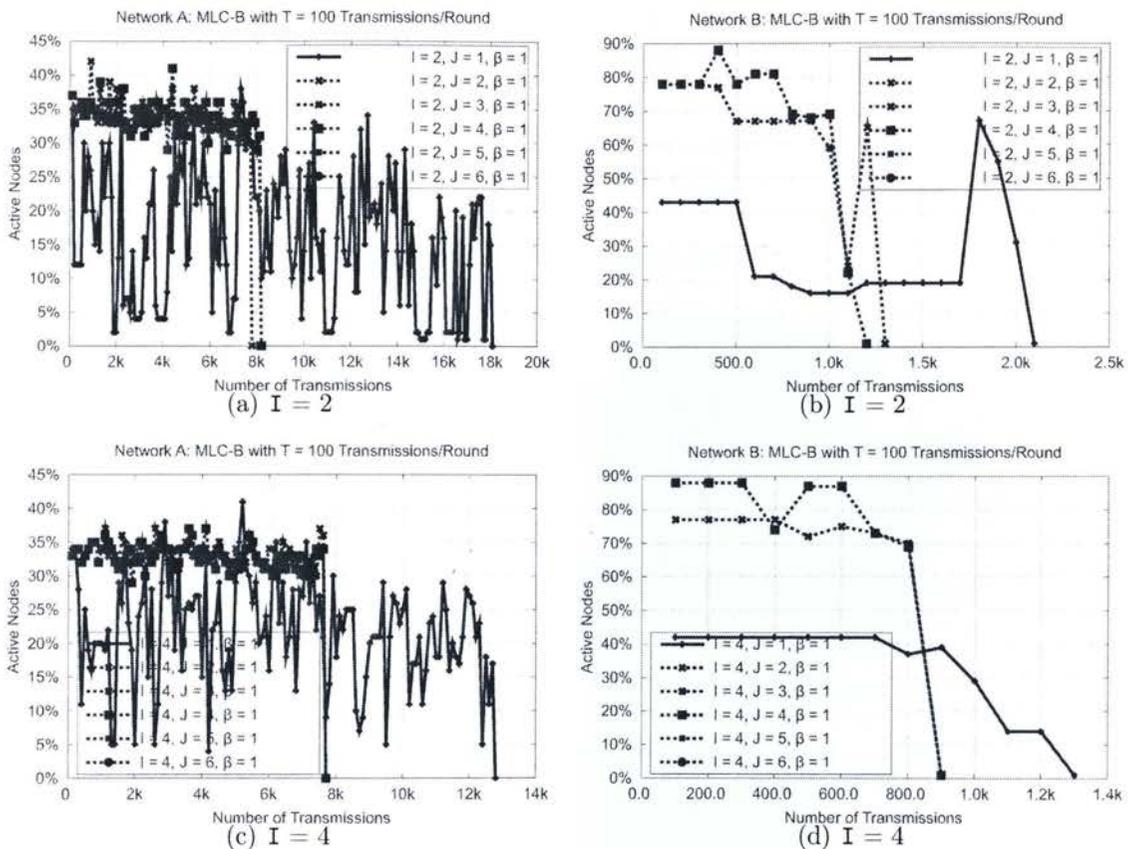


Figure A.1: The MLC-B protocol for different values of I and J with $\beta = 1$ and $T = 100$ in each simulation.

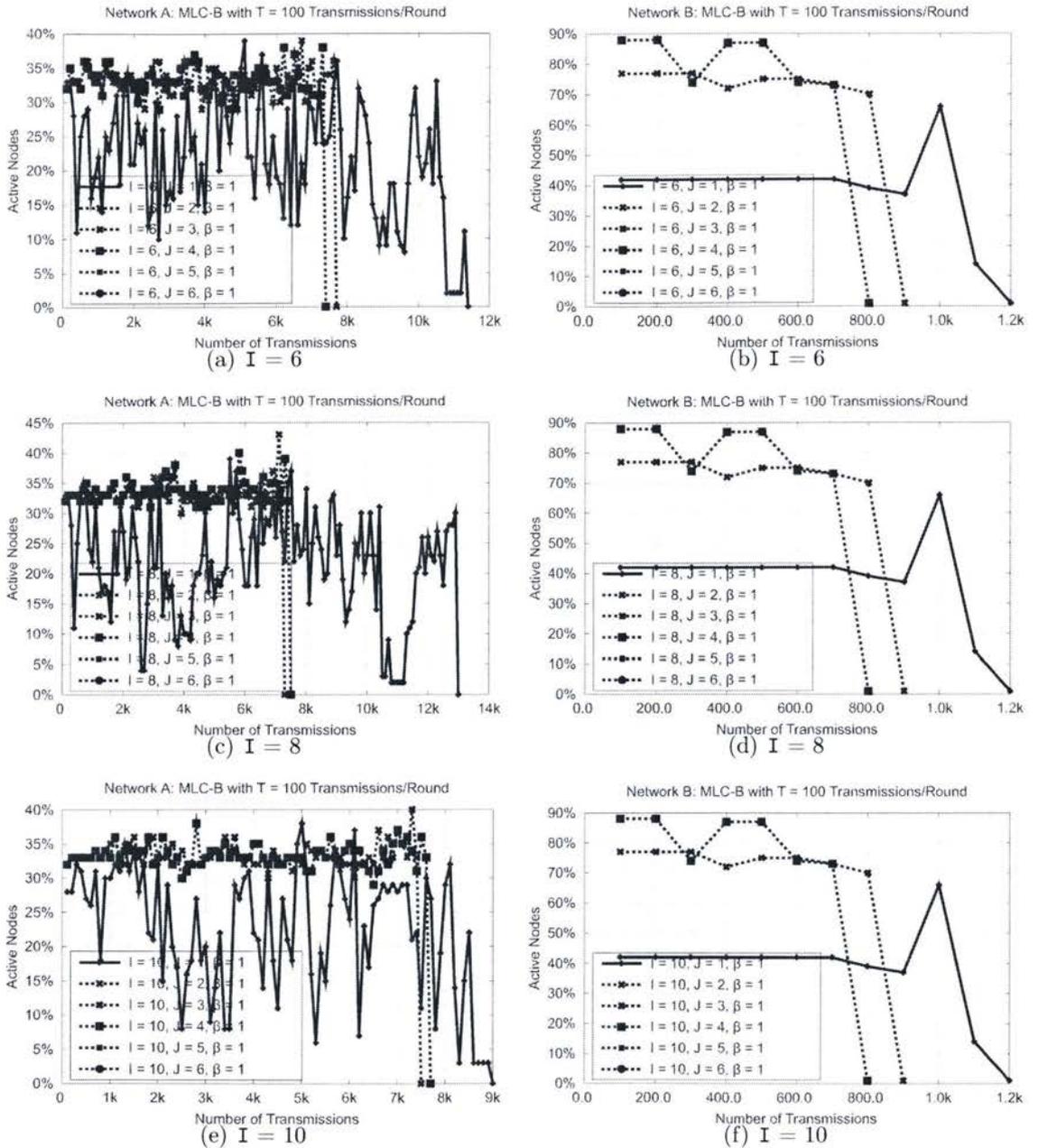


Figure A.2: The MLC-B protocol for different values of I and J with $\beta = 1$ and $T = 100$ in each simulation.

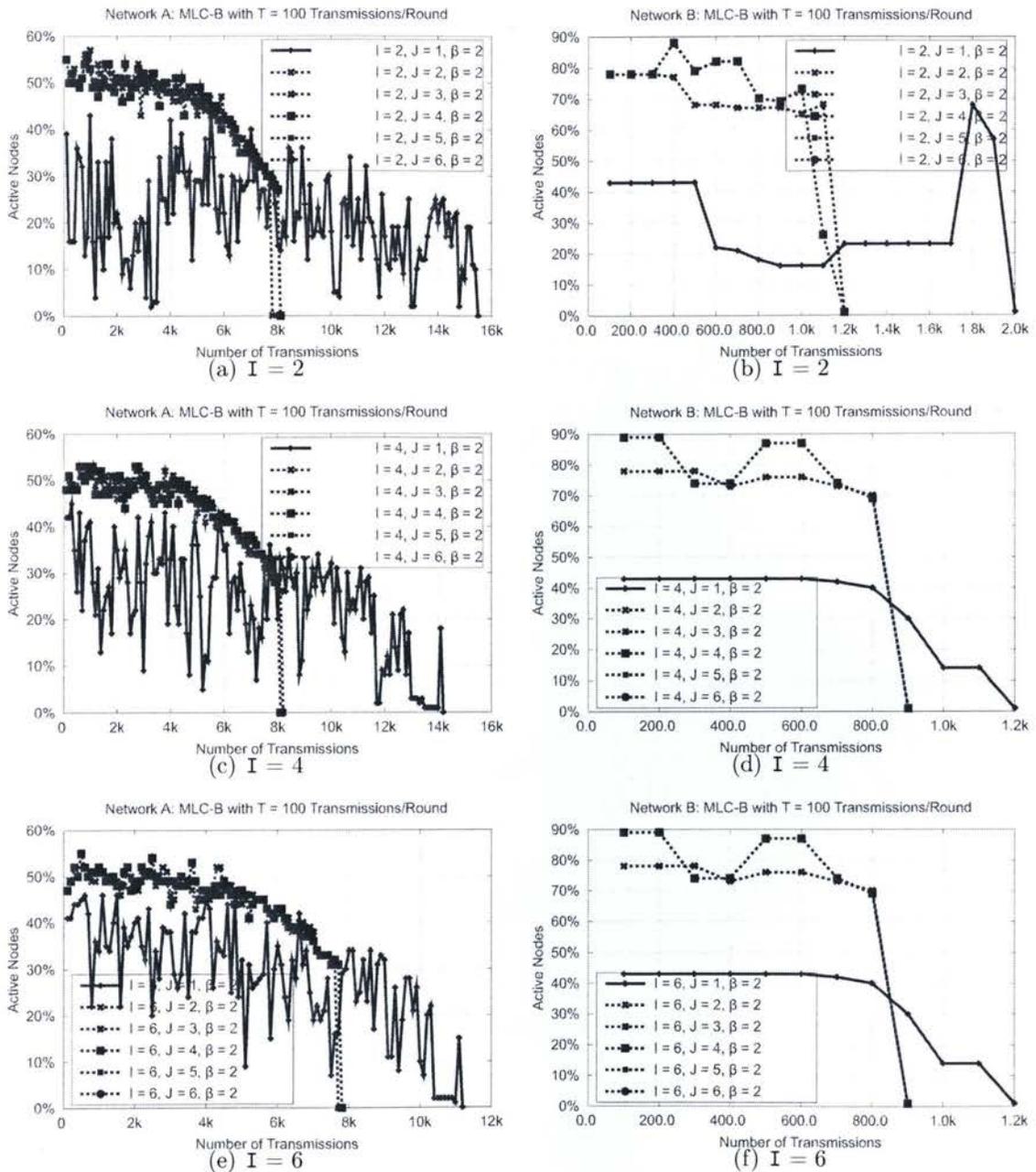


Figure A.3: The MLC-B protocol for different values of I and J with $\beta = 2$ and $T = 100$ in each simulation.

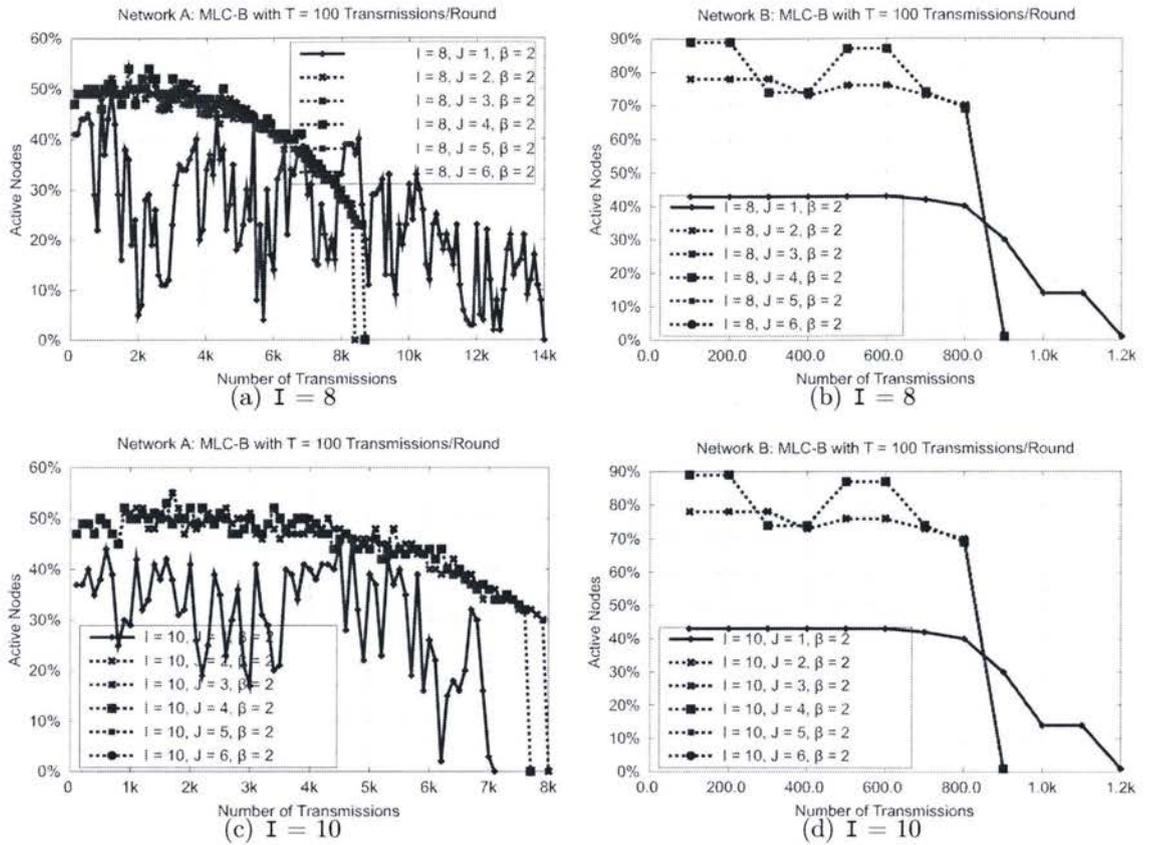


Figure A.4: The MLC-B protocol for different values of I and J with $\beta = 2$ and $T = 100$ in each simulation.

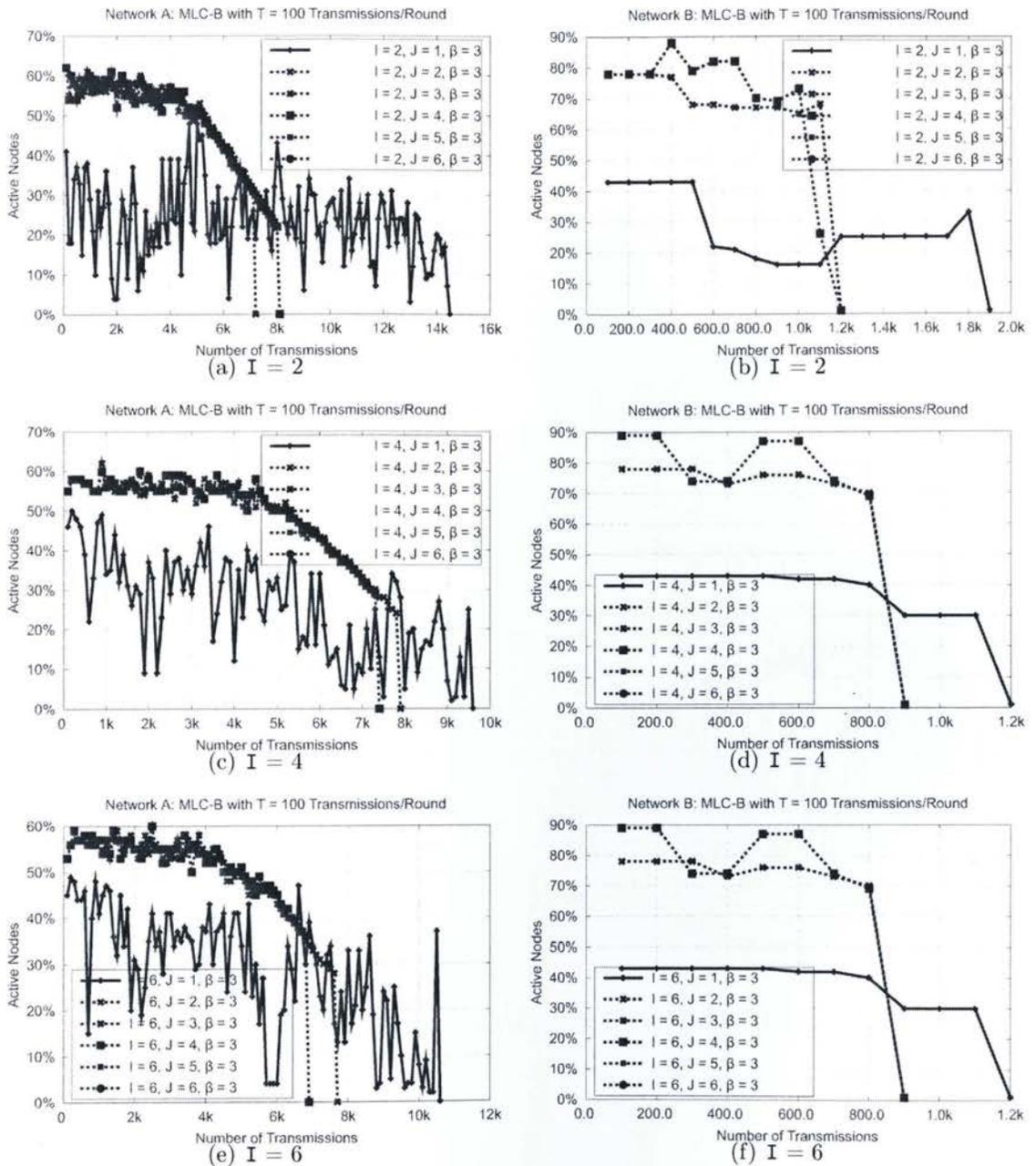


Figure A.5: The MLC-B protocol for different values of I and J with $\beta = 3$ and $T = 100$ in each simulation.

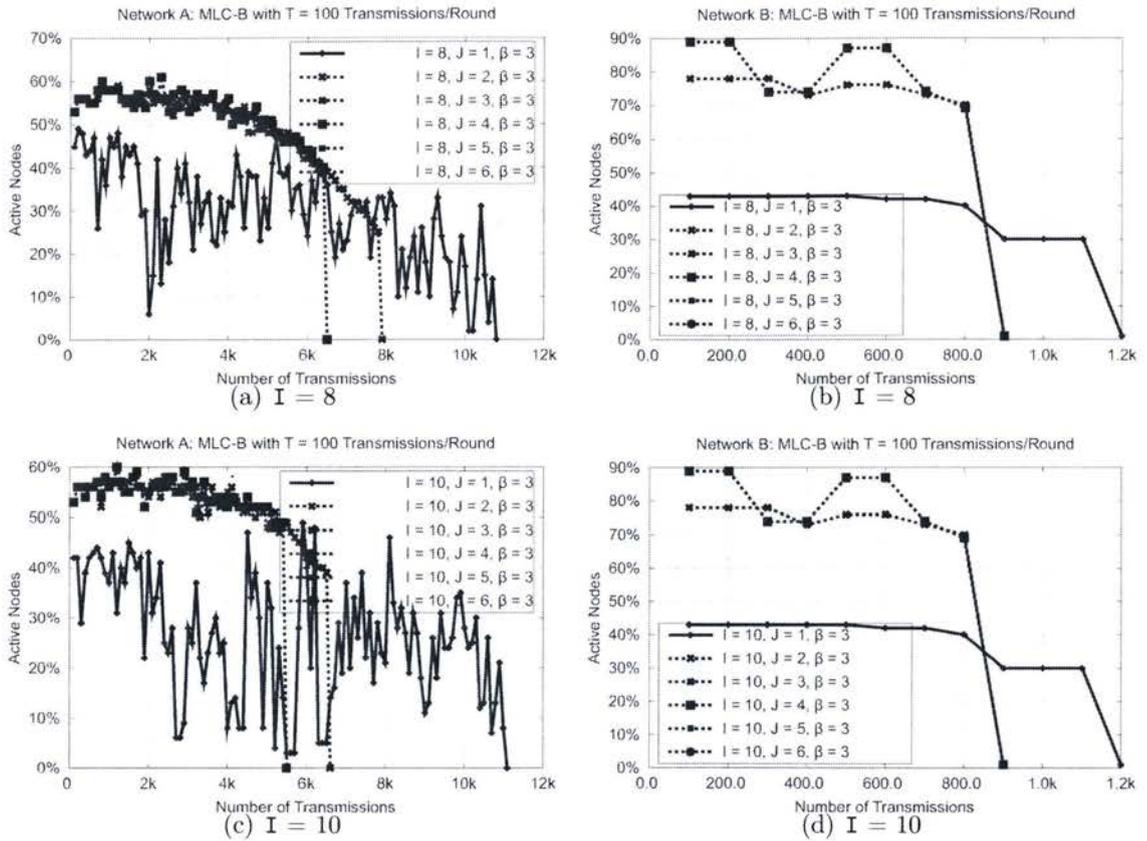


Figure A.6: The MLC-B protocol for different values of I and J with $\beta = 3$ and $T = 100$ in each simulation.

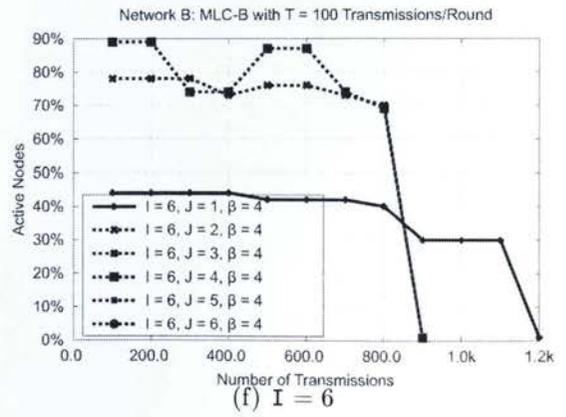
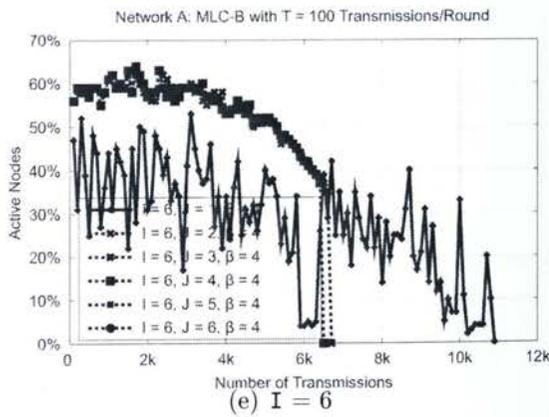
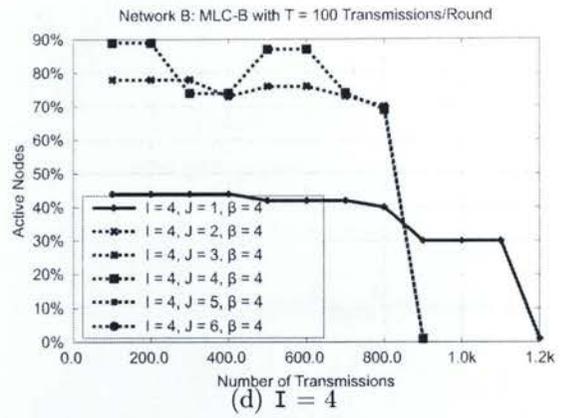
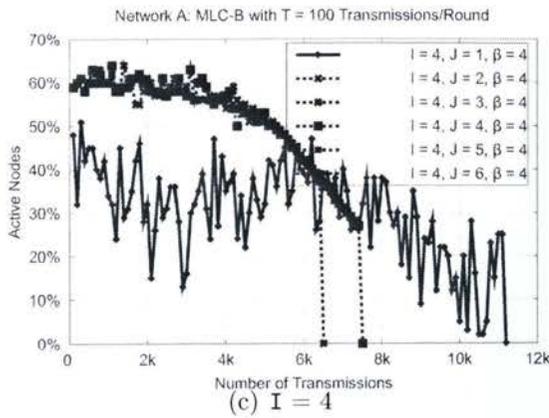
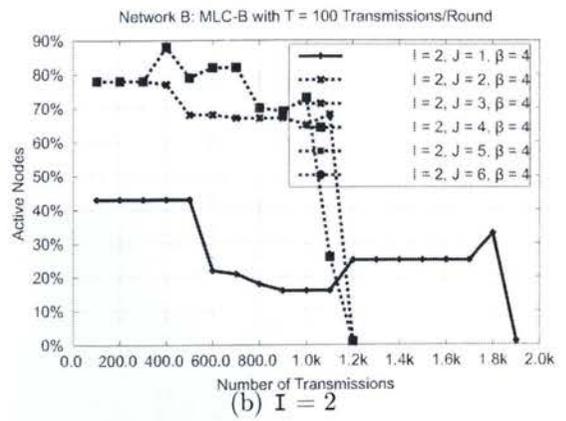
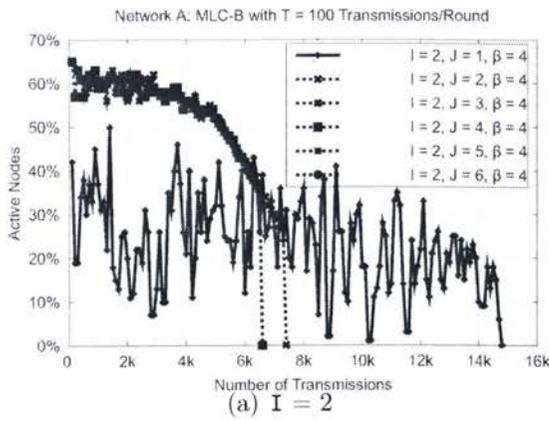


Figure A.7: The MLC-B protocol for different values of I and J with $\beta = 4$ and $T = 100$ in each simulation.

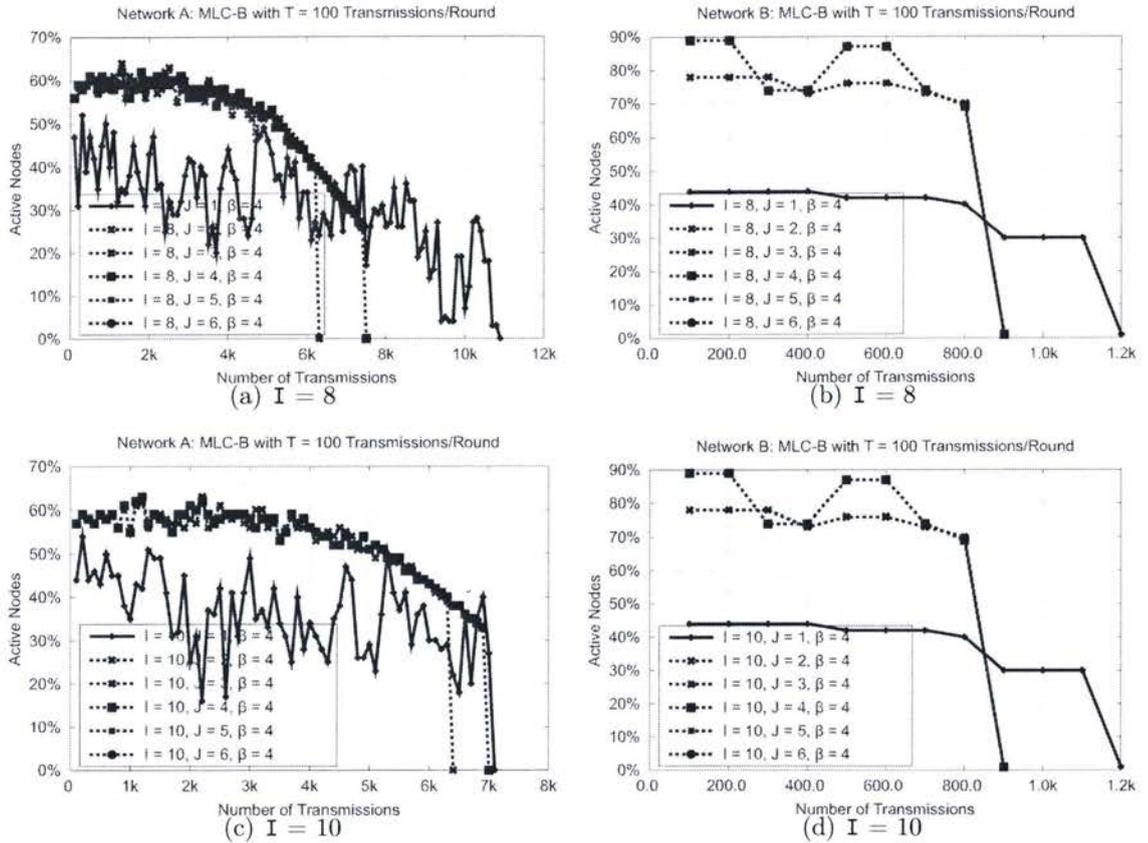


Figure A.8: The MLC-B protocol for different values of I and J with $\beta = 4$ and $T = 100$ in each simulation.

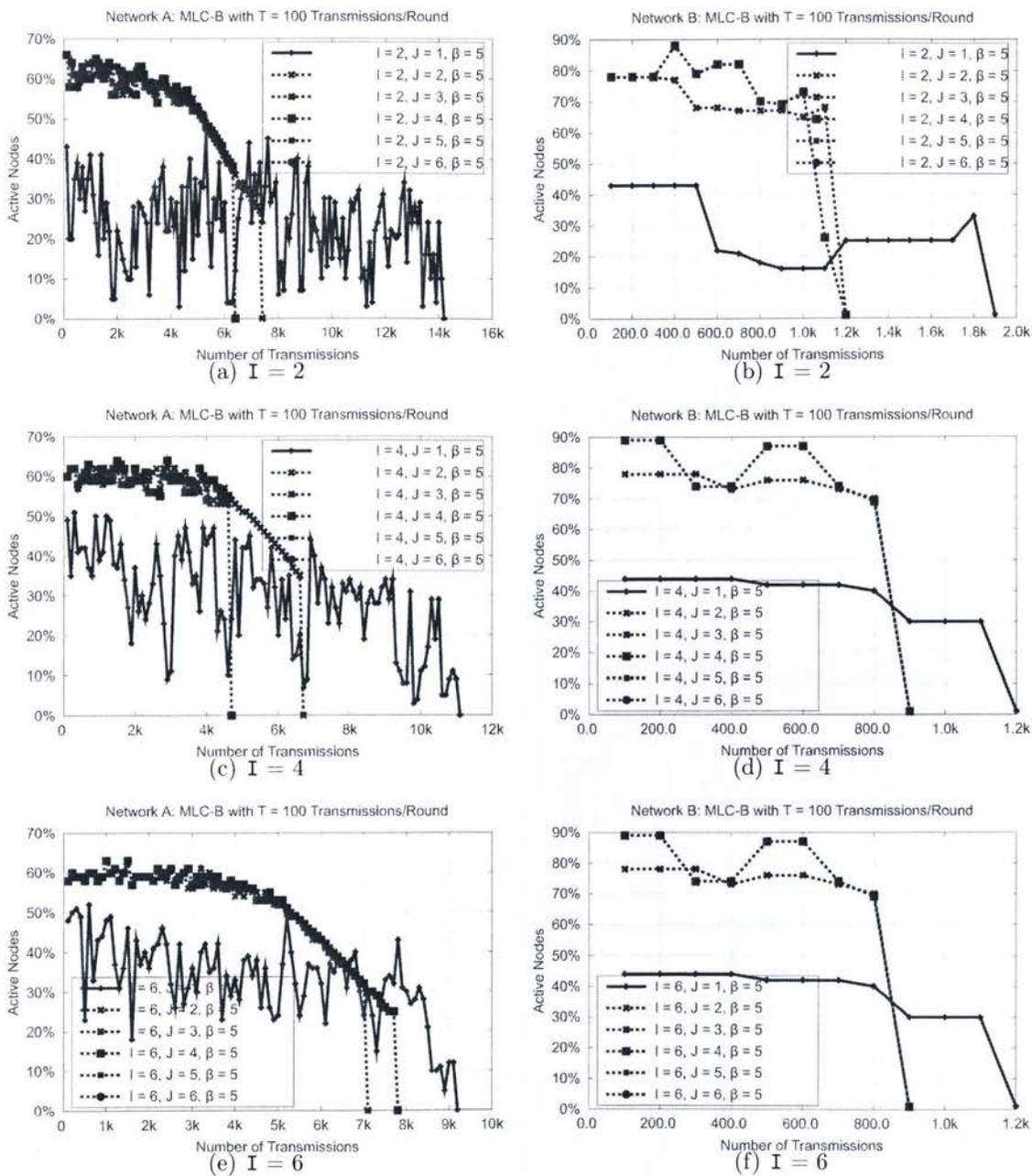


Figure A.9: The MLC-B protocol for different values of I and J with $\beta = 5$ and $T = 100$ in each simulation.

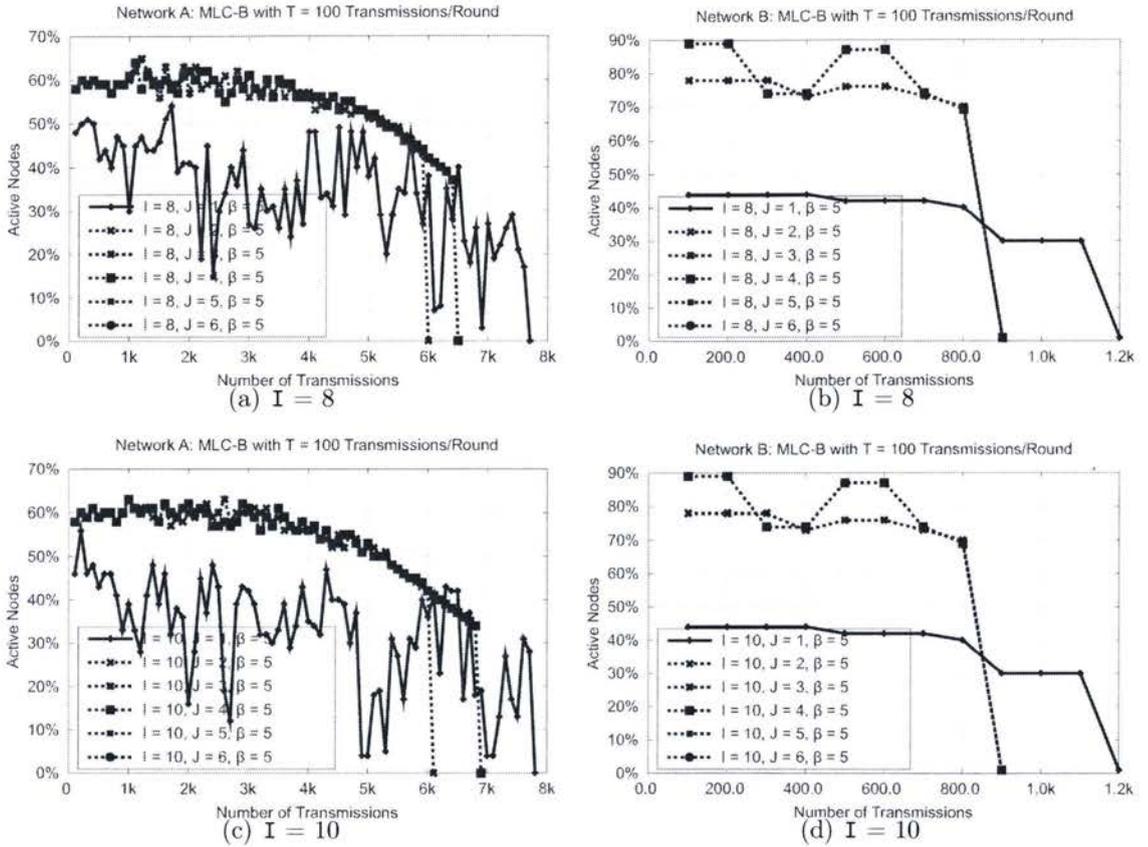


Figure A.10: The MLC-B protocol for different values of I and J with $\beta = 5$ and $T = 100$ in each simulation.

Bibliography

- [1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. A survey on sensor networks. *IEEE Communications Magazine*, 2002.
- [2] G. Anastasi, A. Falchi, A. Passarella, M. Conti, and E. Gregori. Performance measurements of motes sensor networks. In *Proc. 7th ACM international symposium on Modeling, analysis and simulation of wireless and mobile systems*, pages 174–181, 2004.
- [3] S. Bandyopadhyay and E. J. Coyle. An energy-efficient hierarchical clustering algorithm for wireless sensor networks. In *Proc. IEEE INFOCOM'03*, 2003.
- [4] M. Bhardwaj, T. Garnett, and A. P. Chandrakasan. Upper bounds on the lifetime of sensor networks. In *Proc. IEEE International Conference on Communications (ICC 2001)*, volume 3, pages 785–790, 2001.
- [5] C. Buratti, A. Giorgetti, and R. Verdone. Cross-layer design of an energy-efficient cluster formation algorithm with carrier-sensing multiple access for wireless sensor networks. *EURASIP Journal on Wireless Communications and Networking*, 5:672–685, 2005.
- [6] B. Chen, K. Jamieson, H. Balakrishnan, and R. Morris. Span: An energy-efficient co-ordination algorithm for topology maintenance in ad hoc wireless networks. In *Proc. MOBICOM'01*, pages 85–96, 2001.
- [7] Crossbow Technology, Inc., http://www.xbow.com/products/Product_pdf_files/Wireless_pdf/MICA2DOT_Datasheet.pdf. *Mica2dot Datasheet*, document part number: 6020-0043-05 rev a edition, June 2008.

- [8] I. Demirkol, C. Ersoy, and F. Alagoz. MAC protocols for wireless sensor networks: A survey. *IEEE Communications Magazine*, pages 115–121, April 2006.
- [9] J. J. E. Garzas, C. B. Calzon, and A. G. Armada. An energy-efficient adaptive modulation suitable for wireless sensor networks with SER and throughput constraints. *EURASIP J. on Wireless Communications and Networking*, 2007.
- [10] M. Haenggi. On routing in random rayleigh fading networks. *IEEE Trans. on Wireless Communications*, 4(4):1553–1562, July 2005.
- [11] S. M. Hedetniemi, S. T. Hedetniemi, and A. L. Liestman. A survey of gossiping and broadcasting in communication networks. *Networks*, 18(4):319–349, 1988.
- [12] W. R. Heinzelman. *Application-Specific Protocol Architectures for Wireless Networks*. Ph.d., Massachusetts Institute of Technology, Cambridge, MA, June 2000.
- [13] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. In *Proc. 33rd Hawaii International Conference on System Sciences*, January 2000.
- [14] T. Hou and V. O. K. Li. Transmission range control in multihop packet radio networks. *IEEE Transactions on Communications*, 34(1):38–44, January 1986.
- [15] I. Howitt and J. Wang. Energy balanced chain in distributed sensor networks. In *Proc. IEEE Wireless Communications and Networking Conference (WCNC. 2004)*, volume 3, pages 1721– 1726, March 2004.
- [16] S. Hussain, A. W. Matin, and O. Islam. Genetic algorithm for hierarchical wireless sensor networks. *Journal of Networks*, 2(5):87–97, September 2007.
- [17] C. Intanagonwiwat, R. Govindan, and D. Estrin. Directed diffusion: A scalable and robust communication paradigm for sensor networks. In *Proc. ACM MobiCom'00*, pages 56–67, Boston, MA, 2000.

- [18] S. Jin, M. Zhou, and A. S. Wu. Sensor network optimization using genetic algorithm. In *7th WMSCI'03*, 2003.
- [19] H. Karl and A. Willig. *Protocols and Architectures for Wireless Sensor Networks*. John Wiley & Sons, Ltd, England, 2005.
- [20] R. Khanna, H. Liu, and H. Chen. Self-organization of sensor networks using genetic algorithms. *IEEE ICC '06*, 8:3377–3382, June 2006.
- [21] K. T. Kim, H. Kim, and H. Y. Youn. Optimized clustering for maximal lifetime of wireless sensor networks. In *EUC Workshops 2006*, pages 465–474, 2006.
- [22] R. Krishnan and D. Starobinski. Efficient clustering algorithms for self-organizing wireless sensor networks. In *Ad Hoc Networks*, volume 4, pages 36–59, 2006.
- [23] S. S. Kulkarni. TDMA services for sensor networks. In *Proc. 24th International Conference on Distributed Computer Systems Workshop*, pages 604–609, March 2004.
- [24] H. S. Lee, K. T. Kim, and H. Y. Youn. A new cluster head selection scheme for long lifetime of wireless sensor networks. In *ICCSA 2006*, pages 519–528, 2006.
- [25] T. Lin, W. J. Kaiser, and G. J. Pottie. Integrated low-power communication system design for wireless sensor networks. *IEEE Communications Magazine*, 42(12):142–150, December 2004.
- [26] S. Lindsey and C. S. Raghavendra. PEGASIS: Power efficient gathering in sensor information systems. In *Proc. IEEE Aero. Conf.*, March 2002.
- [27] B. Liu and D. Towsley. On the coverage and detectability of large scale wireless sensor networks. In *Proc. of WiOpt'03: Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks*, 2003.

- [28] B. Liu and D. Towsley. A study on the coverage of large-scale sensor networks. In *Proc. of the 1st IEEE International Conference on Mobile Ad-hoc and Systems (MASS'04)*, 2004.
- [29] H. Liu, L. Li, and S. Jin. Cluster number variability problem in LEACH. In *Lecture Notes in Computer Science*, 2006.
- [30] J. Liu and C. R. Lin. Energy-efficient clustering protocol in wireless sensor networks. *Ad Hoc Networks*, 3, 2005.
- [31] C. E. Perkins, E. M. Belding-Royer, and S. Das. Ad hoc on demand distance vector (AODV) routing. IETF RFC 3561, July 2003.
- [32] D. Tian and N. D. Georganas. A coverage-preserving node scheduling scheme for large wireless sensor networks. In *Proc. of 1st ACM International Workshop on Wireless Sensor Networks and Applications*, pages 32–41, 2002.
- [33] Q. Wang, K. Xu, H. Hassanein, and G. Takahara. SWATCH: A stepwise adaptive clustering hierarchy in wireless sensor networks. In *Proc. 4th IFIP-TC6 International Conferences on Networking*, 2005.
- [34] Y. Yuan, Z. Yang, and J. He. An adaptive modulation scaling scheme for quality of services ensurance in wireless sensor networks. *American Journal of Applied Sciences*, 2(3):734–738, 2005.

2005