IMPACT OF NEIGHBORHOOD DISCOVERING AND ADAPTIVE SAMPLING IN WIRELESS SENSOR NETWORKS

BY EUN KYUNG LEE

A thesis submitted to the

Graduate School—New Brunswick

Rutgers, The State University of New Jersey

in partial fulfillment of the requirements

for the degree of

Master of Science

Graduate Program in Electrical and Computer Engineering

Written under the direction of

Professor Dario Pompili

and approved by

_____

_____

_____

New Brunswick, New Jersey

January, 2009

ABSTRACT OF THE THESIS

Impact of Neighborhood Discovering and Adaptive Sampling in Wireless Sensor Networks

by Eun Kyung Lee

Thesis Director: Professor Dario Pompili

Wireless Sensor Networks (WSNs) are networks characterized by a dense deployment of sensor nodes. Because of the dense deployment, sensors can make interference when exchanging data messages. Besides these data messages, in location-based routing that uses geographical positions to route messages, there is a Neighborhood Discovery Protocol (NDP). It should periodically broadcast "Hello" packets to discover neighboring nodes and maintain routing tables updated. This is due to the uncertainty of the wireless environment such as varying radio interference and mobility. Due to the overhead caused by these periodic broadcasts from many nodes in certain radio range, however, NDP may heavily impact on the performance of the routing scheme itself, which in turn could affect end-to-end performance. Although this is an important and challenging problem in WSNs, this impact and the associated tradeoffs have not been fully explored in the literature. Hence, in the first half of this thesis, an analytical and experimental study is conducted to determine how parameters such as power and transmission frequency of neighborhood discovery packets affect the communication process in static and mobile environments.

In addition, WSNs are used to monitor and reliably estimate a phenomenon from the collective information provided by its constituent sensor nodes. Due to the high density of the sensor nodes, the data obtained from them are usually correlated in both space and time. Adaptive sampling is a method that employs this spatio-temporal correlation inherent in WSNs to obtain an energy-efficient estimate of the field. In the second half of this thesis, a distributed, hierarchical, cluster-based adaptive sampling framework is proposed using multiple manifestations for field estimation in three-dimensional environment. Nodes sensing highly correlated values in space are grouped to form clusters and these clusters are modified based on variation in sensor data over time. Energy efficiency is achieved through minimization of communication costs by restricting data communication to the local domain (within clusters) and by applying sleep mode. Moreover, a phenomenon is more reliably captured by using multiple manifestations than by using a single manifestation. It ensures joint optimization by adaptively varying the sampling rates in both space and time domains.

Acknowledgements

"Time flies, doesn't it?" one of my friends often says on the phone. With him, I would say that my time at Rutgers pursuing an M.S. degree flew by.

Doing coursework, working on assignments and conducting research were not simple but hard for a foreign student. It seems that I did nothing but study for two years. However, after having gone through the demanding process of research leading up to my masters dissertation, I now appreciate the sentiments expressed in the maxim "No pain, no gain." My work would not have borne completion without the guidance of my advisor Professor Pompili. His amazing enthusiasm for research and perception into problems have taught me a lot about how to conduct research. He has always motivated me with good discipline whenever we had discussions, providing research-oriented guidance. His guidance helped me not only generate novel ideas but also understand how research changes in this fast-paced world and how to tackle such research challenges. I have learned much about the principles and practical details of research from him and I look forward to the coming years of my doctoral research under him.

I would also like to thank Prof. Dipankar Raychaudhuri, whose lecture "Communication Network II" inspired my interest in networks and provided me with a firm understanding of network. The research work with him also gave me insights into the problems of current wireless networks. He was exceedingly patient and always willing to share his vast analytical and practical engineering knowledge with me. Other than research, he had given me personal advice when I was struggling with my thesis research topic. His professionalism encouraged me to complete my M.S studies. Special thanks to all the professors at Rutgers who have offered stimulating courses on communications, computer engineering, networking, programming and mathematics.

I have been always lucky to have supportive friends and I would like to thank all of

them. I thank Kyung Su for rescuing me whenever I was stuck with personal problems, John Paul for being the most devoted research partner and consuming many a chocolate chip cookies, and Eui Young for helping with the revision process of my thesis work and accepting cups of coffee as compensation.

Finally, I thank my parents for the way they have brought me up over the years, for the way they have always urged me to excel in every sphere of life. And the last one goes my brother, who, apart from being my best buddy, has often advised me on the priorities of life.

Dedication

To my mother Mrs. Kum Ryun Kim

my father Mr. Kun Chang Lee

and my brother Mr. Yang Kyung Lee

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Wireless Sensor Networks (WSNs) are composed of autonomous devices that cooperatively monitor physical or environmental conditions and track specific targets and communicate with each other to route data back to the sink. There are many possible applications of this technology, expected to be used in the military, private and business sectors, among others. Based on some local decision processes, they can transmit the sensed data to the user. Smart sensor nodes are equipped with multiple sensors and processors with low power consumption. A variety of sensors such as biological, thermal, chemical, optical, and mechanical sensors may be attached to the node to measure environmental properties and send the data to the user. In order to utilize such functionalities of WSNs, some constraints should be addressed. Unlike traditional networks, a WSN has its characteristics such as a different network design (ad hoc) and resource constraints. They provide research issues for making WSN reliable and feasible.

1.1 Challenges in WSNs

In a cellular system (i.e., cell-phone infrastructure), the base stations form a wired backbone. A mobile node is only a single hop away from the nearest base station. This type of network is also referred to as infrastructure-based in literature. Unlike this infrastructure-based networks, a WSN has its own architecture design in that hops are randomly placed within the network. This is the so-called ad hoc network, illustrated in Fig(1.1). Data from the sensors are collected by the sink nodes and transmitted to the user via internet, satellite and unmanned arial vehicle (UAV) in ad hoc fashion.

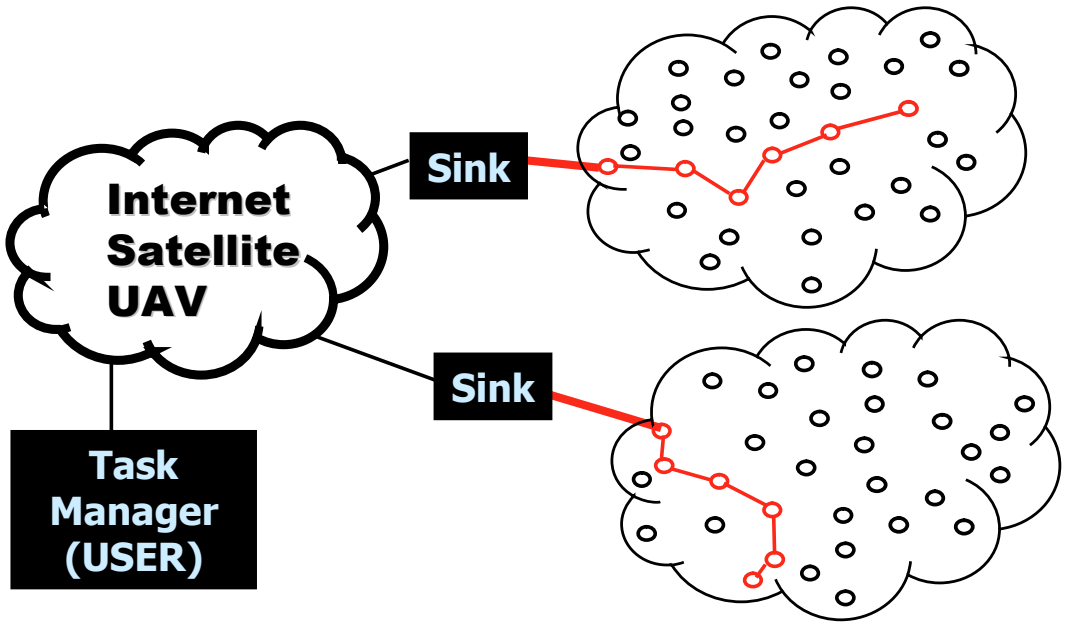


Figure 1.1: Sensor network architecture

Therefore, WSNs have their own specific design. Design constraints arise from application dependency and the need to adapt to differences in the monitored environment. The environment plays a crucial role in deciding the size of the network, the number of nodes to be deployed, the deployment scheme, mobility, and network topology. The size of the network varies and also the number of nodes to be deployed varies with the monitored environment. For indoor environments, fewer nodes are required to form a network in a limited space, whereas outdoor environments may require more nodes to cover a larger area. For mobility environments such as cars equipped with sensors, a large number of nodes are required to capture their movement. Mobile nodes must have the ability to reposition and organize themselves consistently while moving in order to maintain network connectivity and hierarchy. An ad hoc deployment is preferred over pre-planned deployment when the environment is inaccessible by humans or when the network is composed of hundreds to thousands of nodes. Obstacles in the environment can also confine the exchange of messages between nodes, which in turn affects the network connectivity and performance (or topology).

Moreover, there are some resource constraints because of the imperfect capacity of the sensors. Resource constraints include short communication range, limited amount

of power, low bandwidth, and limited processing and storage in each node. Properly allocating resources such as power and bandwidth maximizes utilization. In controlling these resources, coverage, a state in which the monitored region is completely covered with a high degree of reliability, can be achieved [1]. Coverage is important because it affects the number of sensors to be deployed, the placement of these sensors, connectivity, and energy.

Short communication range requires a number of sensor nodes working together to cover a large region to obtain data about the environment. Because of the number of nodes, interference increases. When excessive nodes are deployed, radio from one node disturbs those from the others communicating with the same channel. On the other hand, connection could be lost when not enough nodes are deployed to fulfill the task requiring a larger number of nodes. Optimization of the number of nodes, management of channels, and transmission scheduling issues arise here. Depending on the application, a higher degree of coverage may be required to increase the accuracy of the sensed data.

Table 1.1: Sensor node features

Feature	Imote (2003)	Mica2 (2003)	Micaz (2004)	Telos (2005)	Imote2
CPU type@ [MHz]	32bit ARM @	8bit Atmel @8	8bit Atmel @8	16bit TI @8	32b XS@13(104)
SRAM [KB]	34	4	4	10	256/32,000
FLASH [KB]	512	125+512	125+512	48/1024	32,000
Radio	BT	300-900MHz	802.15.4	802.15.4	15.4 (BT/802.11)
Bandwidth [Kb/s]	720	15	250	250	250(720,11000)
Power(Rx/Tx) [mA]	24/24	10/27	20/18	20/18	20/18
Power sleep [uA]	1-250	19	27	6	1-100
OS support	TinyOS	TinyOS	TinyOS	TinyOS	TinyOS

Conserving energy is a key objective in WSNs in order to increase the lifetime of the network because of the limited amount of power in the nodes. Reliable wireless communication protocols are proposed in the literatures to reduce retransmissions which waste energy. Intelligent sensor placements are suggested to achieve adequate coverage, security, and efficient storage management. Data aggregation and data compression are proposed in order to save energy for the network by reducing data size delivered to the user.

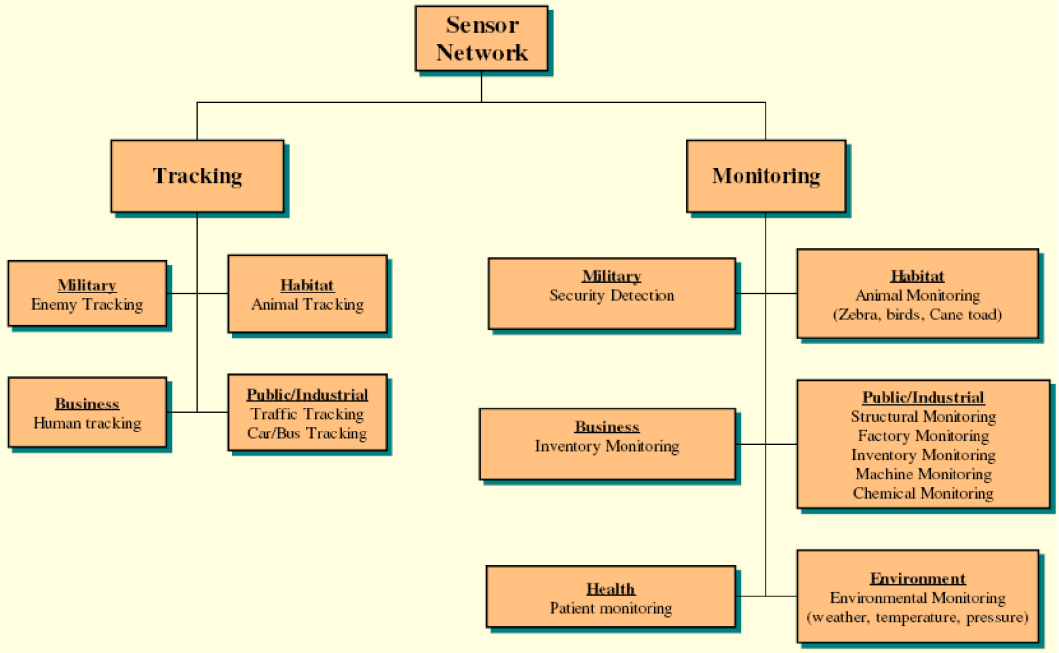


Figure 1.2: Application tree

Low bandwidth is one of the constraints in WSNs. Table 1.1 shows bandwidth of the current sensors. They all have limited bandwidth inadequate for sending large data such as images, sounds and videos.

Limited processing and storage of a sensor require efficient protocol and extra capacity of the node. Since a wholesale replacement of the sensors would be too costly, a revision of the hierarchy would be more cost-efficient. Cluster head, which aggregates the data or may have extra functionality, helps to make up for the weakness of the sensors. Regulating the process and storage load to the CH provides a simple solution to the hardware constraints.

Management and control of these constraints in response to application requirements play a key role in designing WSNs.

1.2 Applications for WSNs

Applications for WSNs have great potential in such scenarios as military target tracking and surveillance, natural disaster relief, biomedical health monitoring, and hazardous

environment exploration, and seismic sensing. WSNs are mainly divided into two categories: tracking and monitoring. Fig. 1.2 shows where detailed applications belong between the two categories [1].

1.3 Thesis Overview

This thesis deals with two different, but related, topics: "On the Impact of Neighborhood Discovery on Location-Based Routing in Wireless Sensor Networks" and "Adaptive Sampling in Wireless Sensor Networks Measuring Multiple Phenomenal Manifestations." The former explores the impact of neighborhood discovery protocol (NDP), which is periodically needed to exchange control messages such as "hello" packet. This packet is mainly required to capture the uncertain wireless environment and the mobility of the nodes for routing purposes. In order to verify the impact of NDP, the following questions will be explored: How far should a node know about its neighbors? How often should a node update the neighborhood information? And how much information should a node require? Analytical and experimental study will be conducted to determine how NDP parameters such as power, frequency, and packet size affect the communication process.

The second topic builds on the first to optimize network performance. Data aggregation in time and space is proposed to save overall energy consumption of the network. Due to the high density of sensor deployment, the degree of "similarity" among spatially proximal sensor observations increases as the inter-node distance decreases. Also, the degree of "similarity" between consecutive sensor measurements varies according to the temporal characteristics of the phenomenon's manifestation. Energy-efficient estimation of the phenomenon can be performed by leveraging the above mentioned spatio-temporal correlation among the sensor data. In this research effort, adaptive distributed field estimation techniques will be developed to prolong the lifetime of 3D WSNs and/or to ensure effective utilization of resources, whereas existing solutions perform spatial and temporal adaptive sampling separately. By grouping sensors in clusters and electing cluster heads (CHs) that will report the data on behalf of the nodes in their clusters, communication cost can be minimized. This overhead reduction

occurs because cluster heads do not need to collect data from all the nodes in the cluster as it is normally proposed in the literature when in-network processing is advocated. Obviously, efficient distributed clustering and resizing algorithms need to be developed so to assure data consistency as the phenomenon evolves in time and space. Chapter 3 discusses this issue in detail.

Chapter 2

On the Impact of Neighborhood Discovery on Location-based Routing in Wireless Sensor Networks

2.1 Introduction

It has been pointed out in [2] that energy efficiency in sensor networks can be improved by designing algorithms following cross-layer approach, i.e., taking into account the interaction between different layers of the communication process thereby minimizing the energy consumption of the entire network. In addition, in order to achieve energy efficiency, network algorithms and protocols need to be scalable, i.e., they must perform well for any arbitrary number of nodes. Scalable algorithms are normally localized/distributed and exchange information only with neighboring nodes. The notion of scalability is related to that of localization. As a result, local routing decisions help achieve minimum latency and energy consumption without having a global perspective of the network.

It has been shown that geographical routing schemes (i.e., based on node physical position) [3] are scalable. However, gathering only geographical position information from the neighboring nodes may not improve the energy efficiency. With the availability of high sensitivity receivers and multi-faceted functionalities in the sensor nodes currently available on the market, collecting more information such as channel quality and residual energy information from the neighboring nodes might make the routing tasks more energy efficient. However, the question lies in how much information should be collected from the neighboring nodes without incurring in excessive overhead.

Each node collects the information using a Neighborhood Discovery Protocol (NDP), which involves signaling of control packets between nodes. The more the information collected from the neighboring nodes, the better can the routing decisions be, but at the

cost of an increase in length of the signaling packets, which leads to higher overhead in terms of energy, bandwidth used, and processing complexity. In fact, the higher the transmission power used for sending the Neighborhood Discovery (ND) packet, the higher would be the number of neighboring nodes perceived, but this would come at the penalty of more collisions of signaling packets between nodes as well as higher energy consumption. Moreover, the higher the frequency of ND packets being exchanged, the more overhead induced in the network due to increase in signaling packets. Last but not least, mobility of the sensor nodes can also affect the routing decisions. When nodes move, they need to frequently update their routing table. In addition, even if the sensor nodes do not move, the topology of the network can change because nodes may die due to expended battery or any hardware failure. Therefore, selecting a suitable NDP update time interval is crucial for selecting the best next hop, even if the nodes are immobile.

Hence, through five routing schemes - *Most Advance Routing* [4], *Energy-aware Routing*, *Channel-aware Routing*, *Compass Routing* [5] and *Partial Topology Knowledge Forwarding* (PTKF) [2], and various parameters such as power and frequency of NDP, this paper addresses the following questions: "How much information is required from the neighboring sensor nodes?", "How 'far' should a sensor node know about the network?", and "How often should a node collect the information from the neighborhood?". This paper also analyzes the effect of mobility on routing decisions.

The remainder of the Chapter is organized as follows. In Sect. 2.2, we review some of the existing forwarding schemes and cross-layer approach proposed for WSNs. In Sect. 2.3, we describe the neighborhood discovery protocol and the routing schemes considered. In Sect. 2.4, we provide the performance evaluation, while in Sect. 2.5, we conclude the paper and draw the main conclusions.

2.2 Related Work

In this section, we present some of the existing geographical forwarding strategies used to find the best next hop as well as some of the existing cross-layer solutions that may

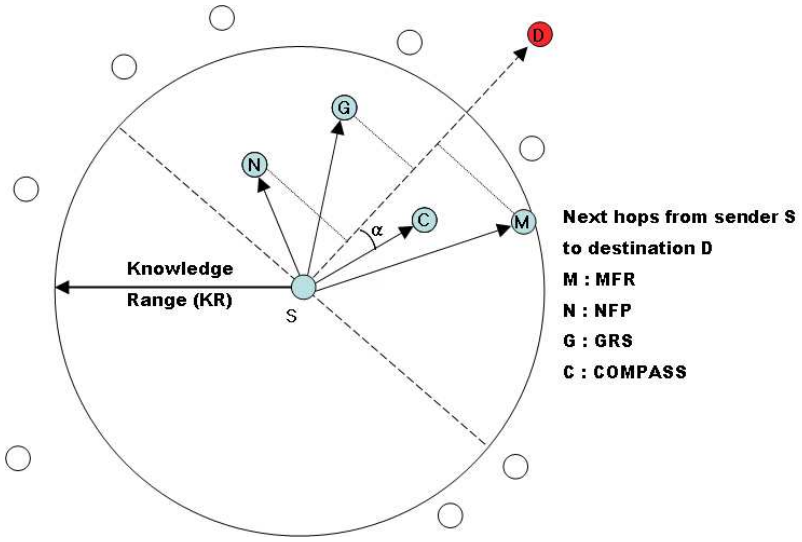


Figure 2.1: Different Forwarding Strategies

help improve the geographical routing by using information from both physical layer as well as data link layer.

2.2.1 Forwarding Strategies

There are several existing forwarding strategies to improve the performance of geographic routing. These forwarding strategies can be divided into two categories: *distance-based* and *reception-based* [6]. In WSNs, a node communicates only with nodes that are in its radio range. In this way, every node gets to know each other by signaling and exchanging position information. For distance-based forwarding, a node only knows the distance of its neighbors while in reception-based forwarding the packet reception rate of its neighbors is also known [6]. With reference to Fig. 2.1, we introduce the following definitions as given in [2]. Given a sender node S and a destination node D, the *progress* of a generic node X is the orthogonal projection of the line connecting S and X onto the line connecting S and D. Given a sender node S and a destination node D, the *advance* of a generic node X is the distance between S and D minus the distance between X and D.

In [4], Takagi and Kleinrock proposed a geographical routing strategy, using the notion of progress. In their Most Forward within Radius (MFR) scheme [4], the message

is forwarded to the maximum progress neighbor. With reference to Fig. 2.1, note that, although node G is closer to the destination, MFR forwards the packet to M because the progress of M is larger than that of G. Hou and Li [7] discuss the Nearest Forward Progress (NFP) method, which selects the minimum progress neighbor within the topology Knowledge Range (KR) of S, i.e., node N in Fig. 2.1. Knowledge range is defined as how far from itself a node can perceive its neighboring nodes. Finn [8] proposes the Greedy Routing Scheme (GRS) based on geographical distance in which a node selects among its neighbors the one closest to the destination, i.e., node G in Fig. 2.1. In the so-called compass routing method [5], the message is forwarded to a neighbor, C in Fig. 2.1, such that the angle $\angle CSD$ is minimum, i.e., the direction SC is the 'closest' to the direction SD. Whereas, the Random Progress Forwarding (RPF) method selects a random next hop among the neighbors.

The reception-based forwarding schemes are based on the idea that only distance information is not enough for routing decision in realistic condition because of the unreliability of the wireless network. Previously proposed geographic routing protocols perform poorly on lossy links. Packet Reception Rate (PRR) is considered for routing decision in reception-based forwarding. In [9], the authors suggest blacklisting/link-selection strategies. These strategies select next best hop by calculating the product of the packet reception rate and the distance traversed towards destination. By blacklisting bad links or nodes, the number of disconnections is reduced thereby achieving higher delivery rate. However, this scheme has an overhead of requiring low rate of control traffic to calibrate link quality. While this approach can detect good links, it adapts slowly to changes in link quality because reasonable number of packets need to be sent to measure PRR.

2.2.2 Cross-layer Solutions

Cross-layer approaches have gained importance due to the limitations of layered protocols. In this section, we discuss some of the cross-layer information, such as received signal strength (RSS), link quality indicator (LQI), residual energy and transmission power and path gain, that can be effectively used in the routing process to improve

end-to-end networking performance.

Received Signal Strength Indicator (RSSI)

Channel quality can be assessed by the received signal strength (RSS). In [10] [11], each neighbor obtains channel quality by using periodic beacon signal. In [12], the authors propose to use RSSI for calculating the Signal-to-Noise-Ratio (SNR) and Packet Reception Rate (PRR). In [9], it has been shown that the product of PRR and distance d traversed towards destination is the optimal forwarding metric in a lossy medium.

Link Quality Indicator (LQI)

In addition to RSSI, the radios such as CC2420 implement a parameter called Link Quality Indicator (LQI). The conventional opinion in the wireless networking community is that compared to RSSI [13], LQI is a better indicator. This is due to the existence of asymmetric links and external factors such as noise, multi-link interferences and has been proved based on experimental works [14]. LQI is featured by CC2420 radios (TMote, TelosB, Micaz) based on chip error rate ranging from the value of 50 to 110 (Minimum and maximum quality respectively).

Residual Energy

Residual energy can be utilized to increase entire network lifetime and energy balance. PAMAS protocol [15] is based on metrics to minimize energy consumed/packet (e_j) and cost/packet (c_j). The metric for calculating cost/packet is done by assigning node weights (f_i) to each node and then computing the shortest path at the routing layer. Generally, f_i is tailored to reflect residual energy of the battery of intermediate nodes using predefined characteristics.

Transmission Power and Path Gain

[16] implements a cross layer protocol between transport and physical layers to increase end-to-end throughput and maintain energy efficiency in multi-hop networks. CDMA based medium access is used. Jointly Optimal Congestion-Control and Power-Control

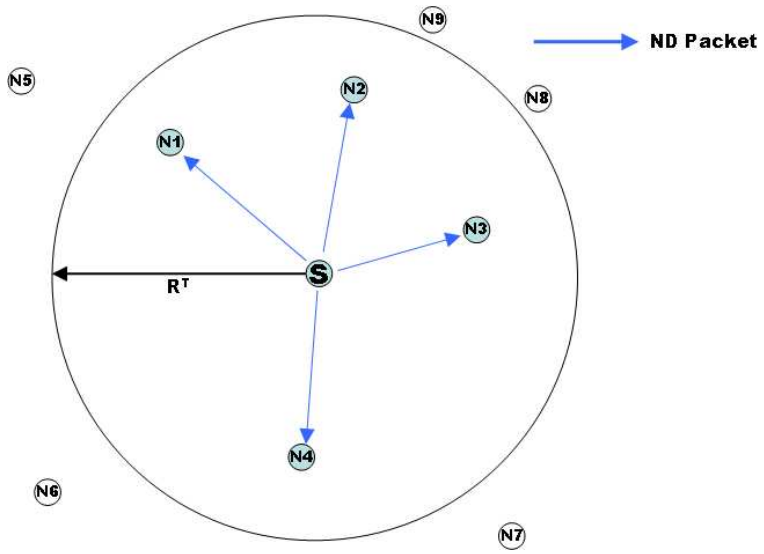


Figure 2.2: Oneway Neighborhood Discovery Protocol

Algorithm (JOCP) updates weighted queueing delay (λ_1) at each node at every time slot, t in a synchronized network. In the second step, the delay (D_s) measured from TCP protocol and TCP window size (W_s) is updated. As the third step, a parameter m_j is calculated based on locally measurable quantities at nodes such as signal to interference ratio (SIR), transmit power (P_j) and path gain G_{ij}. Finally, all the nodes update power based on locally measurable quantities and received messages from neighbors.

2.3 Neighborhood Discovery and Routing

In this section, we first describe the basic neighborhood discovery schemes, which allows each node to gather information from the neighboring nodes. Then, we introduce a mathematical relationship between neighborhood discovery time period and data time period. Further, we discuss about the mobility case. Finally, we discuss the various routing strategies that are considered.

2.3.1 Handshake Neighborhood Discovery

We analyze a basic type of neighborhood discovery scheme: Handshake. In this scheme, node S periodically sends Neighborhood Discovery (ND) packets to gather the information from the neighboring nodes at a power level to be received by all the nodes

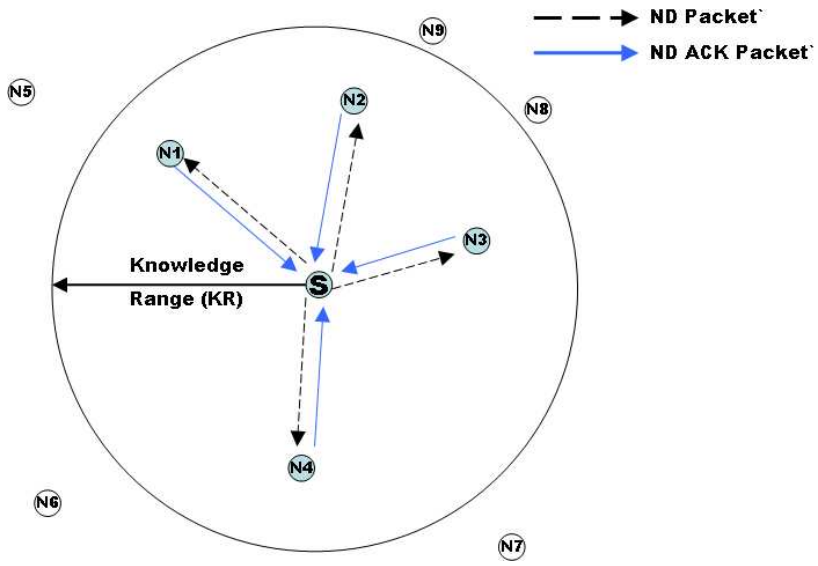


Figure 2.3: Handshake Neighborhood Discovery Protocol

within its chosen Knowledge Range (KR), as shown in Fig. 2.3. As a result, nodes $N1$, $N2$, and $N3$ receive the ND packet while the farther nodes do not. All the nodes that receive the ND packet would reply with neighborhood discovery acknowledgement packets (ND ACK packet) piggy-backed with information such as residual energy of the node, packet error rate, etc. It is intuitive that increasing the KR or collecting information more frequently from the neighboring nodes may result in better routing decisions. However, this causes higher energy consumption as a result of more power and exchange of signaling traffic.

2.3.2 Data Period vs NDP Period

Data and neighborhood discovery traffic are the major traffic sources of the sensor network which has limited bandwidth. Therefore, we need to verify whether data sending period (T_D) and NDP sending period (T_{ND}) are strongly related rather than considering them separate, before introducing the routing schemes. T_D is the time interval for sending data packets and T_{ND} is the time interval for sending NDP packets. In order to find the relation between T_D and T_{ND}, we use $\overline{\Delta}$, the average waiting time to send data packets with latest information from the NDP. Hence, there are two cases to be considered: one in which $K = 1$ (Fig. 2.4), and the other in which $K \geq 2$ (Fig.

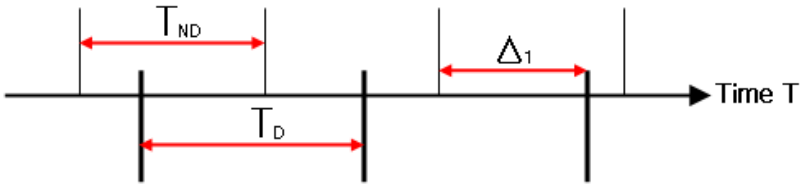


Figure 2.4: Δ when K = 1

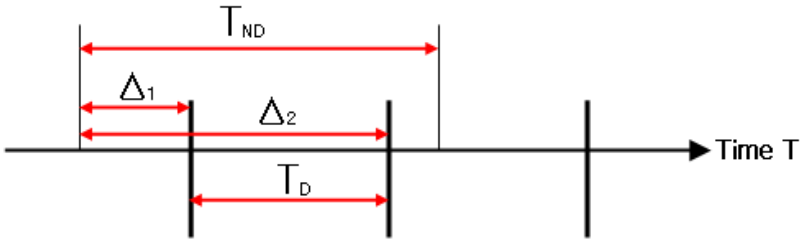


Figure 2.5: Δ when K = 2

2.5), where K is defined as,

$$\left\lceil \frac{T_{ND}}{T_D} \right\rceil = K, \tag{2.1}$$

$$\overline{\Delta} = \Delta_1 + \frac{T_D}{K} \cdot \sum_{i=1}^{K-1} i = \frac{T_D}{2} \cdot \left\lceil \frac{T_{ND}}{T_D} \right\rceil \leq \overline{\Delta}_{TH}. \tag{2.2}$$

The generalized equation (2.2) proves that the two values T_D and T_{ND} are strongly related and directly depends on K. Larger the value of K, larger would be the $\overline{\Delta}$. Once the $\overline{\Delta}$ crosses the threshold $\overline{\Delta}_{TH}$, the packet delivery ratio will drop drastically.

2.3.3 Mobility

In addition to the relation between T_D and T_{ND}, we need to consider the effect of mobility on routing decisions. Once the information of the neighboring nodes is known, a node would adjust its radio range to a value R_T so as to send data packets to reach the best next hop, where $R^T \leq R_{NDP}$ and R_{NDP} is the KR for the NDP.

In static case, when nodes do not move, the next best hop would be correct as conveyed by the routing algorithm. However, in the case of mobility, things change and the best next hop chosen by routing using the latest NDP information may not be the best next hop while sending the data packets. For example, with reference to Fig. 2.6 where nodes are mobile, routing would fail as node $N3$ has gone out of range

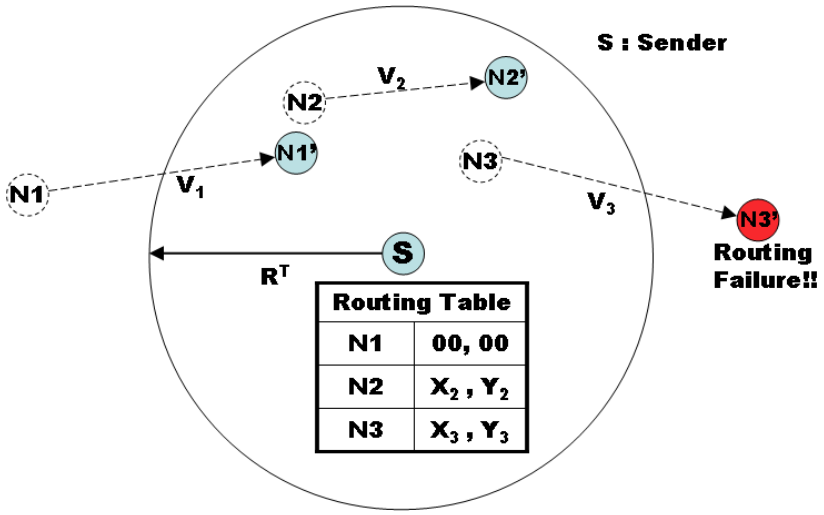


Figure 2.6: Mobile nodes

of the sender S while it is sending the data packet. The following case can also occur: sender S can route the packet as the radio range of S could capture the movement of nodes as depicted by node $N2$. However, node $N2$ has moved to a new geographical position still within the KR of sender S and it may be possible that another node $N1'$, which is not yet in the routing table of S, has moved to previous position of node $N2$. Moreover, if the routing is ID-driven, it will not be able to capture the effect of mobility as compared to position-driven routing.

The faster the nodes move, the less possibility that that sender can capture the mobility of the nodes. Hence, to capture the effect of mobility, the transmission power of sender S has to be increased resulting in more wastage of energy compared to static case just to capture the effect of mobility and to thereby increase the packet delivery ratio.

2.3.4 Routing Schemes

In this section, we introduce five routing schemes based on the functionalities they use for making the routing decisions. Each routing scheme needs different type of information from the neighboring nodes to make routing decisions.

Most Advance Routing

In this routing scheme, the next hop is decided based on geographical position of the nodes only. The next hop node will be the one which has maximum progress towards the destination node subject to the condition that it should be in positive advance towards the destination node. Hence, node i will select node j^* as the best next hop if

$$j^* = argmax_{j \epsilon N_i \cap P_i^d}[d_{ij} \cdot Cos(\theta_{ij})], \tag{2.3}$$

subject to condition for positive advance

$$\theta_{id} - 90^\circ < \theta_{ij} < \theta_{id} + 90^\circ, \tag{2.4}$$

where

- N_i set of neighboring nodes of i;
- P_i^d set of neighboring nodes of i having positive advance towards destination node d;
- d_{ij} [m] distance from node i to node j;
- θ_{id} [rad] angle of the line joining node i and destination node d;
- θ_{ij} [rad] angle of the line joining node i and node j .

Energy-aware Routing

This routing scheme makes routing decisions not just based on the geographical position but also on the available energy of the nodes. Hence, node i will select node j^* as the best next hop if

$$j^* = argmax_{j \epsilon N_i \cap P_i^d}[E_j^{Av} - (E_{ij}^T - E_j^R - E_{jd}^T) \cdot L_{Data}], \tag{2.5}$$

subject to the condition of positive advance given in (2.4), where

- N_i set of neighboring nodes of i;

- P_i^d set of neighboring nodes of i having positive advance towards destination node d;
- E_j^{Av} [J] available energy at node j;
- E_{ij}^T [J/bit] energy required to transmit one bit from i to j;
- E_j^R [J/bit] energy required to receive one bit from i;
- E_{jd}^T [J/bit] energy required to transmit one bit from j to destination node d;
- L_{Data} [bit] length of the Data packet.

Channel-aware Routing

This routing scheme takes into account the packet error rate (PER) [17] on the link connecting the nodes and the geographical position of the neighboring nodes for making the decisions. In this routing, node i will select node j as the best next hop if

$$j^* = argmin[E_{ij}^T \cdot \widehat{N}_{ij}^T \cdot \widehat{N}_{ij}^{Hop}], \tag{2.6}$$

subject to the condition of positive advance given in (2.4), where

- $\widehat{N}_{ij}^T = \frac{1}{1-PER_{ij}}$,
- $\widehat{N}_{ij}^{Hop} = \max(\frac{d_{ij}}{<d_{ij}>_{id}}, 1)$,
- $PER = 1 - (1 - BER)^{L_{NDP}}$,
- $BER_{ij} = \Phi^M(SNR_{ij})$.

Specifically,

- E_{ij}^T energy required to transmit one bit from i to j [J/bit];
- $\widehat{N}_{ij}^T$ average number of transmissions by node i such that the packet is correctly decoded by node j;
- $\widehat{N}_{ij}^{Hop}$ estimated number of hops from node i to the destination node d when j is selected as next hop;

- PER_{ij} packet error rate associated with the link (i,j);
- BER_{ij} bit error rate associated with the link (i,j);
- SNR_{ij} signal to noise ratio associated with link (i,j);
- M is the modulation scheme;
- L_{NDP} [bit] length of ND Packet;
- d_{ij} [m] distance between node i and node j;
- $< d_{ij} >_{id}$ projection of d_{ij} onto the line connecting node i with the destination node d.

Compass Routing

In this routing scheme, the next hop is decided based on geographical position of the nodes only. The next hop node will be the one which subtends minimum angle at the source node, subject to the condition that it should have positive advance towards the destination node. Hence, node i will select node j as the best next hop iff

$$j^* = argmin_{j \epsilon N_i \cap P_i^d}[|\theta_{ij} - \theta_{id}|], \tag{2.7}$$

subject to the condition of positive advance in (2.4), where

- N_i set of neighboring nodes of i;
- P_i^d set of neighboring nodes of i having positive advance towards destination node d;
- θ_{id} [rad] angle of the line joining node i and destination node d;
- θ_{ij} [rad] angle of the line joining node i and node j.

Partial Topology Knowledge Forwarding

This routing scheme [2] finds the next hop based on minimum energy path required to reach the destination within its limited topology knowledge, i.e, shortest weighted path

takes into account only nodes in its KR and destination, as other nodes are unknown to it. Energy calculation is based on the link metric in [2], as

$$E = E_{elec} + \beta d^{\alpha}, \tag{2.8}$$

where

- E_{elec} [J/bit] energy consumed by the electronic circuit;
- α path loss ($2 \leq \alpha \leq 5$);
- β [J/(bit $\cdot$ m$^{\alpha}$)] is a constant;
- βd^{α} [J/bit] accounts for radiated power necessary to transmit over a distance d between source and destination;

2.4 Performance Evaluation

2.4.1 Simulation Results

We implemented the forwarding schemes described in Sects. 2.2.1 and 2.3.4 and tested them in both static and mobility cases. Simulations were done using TOSSIM 2.x, a TinyOS simulator. TOSSIM 2.x currently supports only MicaZ radio components (IEEE 802.15.4). We are interested in scenarios where the density of nodes is high. However, due to the computational complexity of the problem and large amount of the data to process, simulations have been done with at most 49 nodes in 100 by 100 m^2 area to reproduce a high-density environment. The radio propagation model described in [18] was used for our simulation. In addition to the radio propagation model, TOSSIM also simulates the RF noise and interference with other nodes. TOSSIM uses Closest Pattern Matching (CPM) algorithm [19], which takes a noise trace as input and generates a statistical model from it.

We present simulation results for the scenarios illustrated in Table 2.1 with five routing schemes: Most Advance, Energy Aware, Channel Aware, Compass, and PTKF. The sink node was placed at the corner of the grid to collect the data packets from each

nodes at a data frequency of 2 Hz. Packet Delivery Ratio and Energy Drain have been used as metrics with respect to NDP power as well as NDP frequency for the evaluation of our results.

Table 2.1: Parameters of the Model Used for Simulations

Physical deployment parameters	
Terrain dimension	100 X 100 [m^2]
Topology	Uniform random
Mobility	(2, 4, 6 ,8) [m/s]
Number for nodes	25, 36, 49
Channel parameters	
Path loss exponent	3.5
Shadowing standard deviation	3.2
Reference distance(D0)	1 m
Path loss at reference distance	-30 [dBm]
Radio Parameters	
Noise	Thermal noise (stdev = 4)
Radio Noise floor	-105 [dBm]
Hardware Variance (Highly Asymmetric)	
covariance matrix	S = [S11 S12 ; S21 S22]
S11 (variance of noise floor)	3.7
S12 (covariance btw S1 and S1)	-3.3
S21 (same as S12)	-3.3
S22 (variance of output power)	6.0

First, we answer the question "How far should a sensor node know about the network?"To resolve this question, simulations were performed on power values of $(0, -1, -3, -5, -7, -10, -15, -25)$ dBm for each routing algorithms with NDP frequency set to 1 Hz. In Fig. 2.7, 2.8, and 2.9, we show the packet delivery ratio at various power values, as mentioned above, across three different number of nodes (25, 36, and 49). It can be inferred from the figures that, as the node densities increase the packet delivery ratio drops and also a decrease in the optimal power value required for NDP. From Fig. 2.7, 2.8, and 2.9, the optimal power value (power at which packet delivery ratio is maximum or energy drain is minimum) for 25 nodes is 0 dBm, for 36 nodes it is -3 dBm while for 49 nodes it is -5 dBm. For the energy drain vs. NDP power, in Fig. 2.10, 2.11, and 2.12, the optimal power turns out to be -5 dBm for 25 nodes, -7 dBm for 36 nodes while -10 dBm for 49 nodes. Therefore, as the node density increases less power is required for NDP.

For the mobility case, all the nodes are moving with velocity as given in Table 2.1.

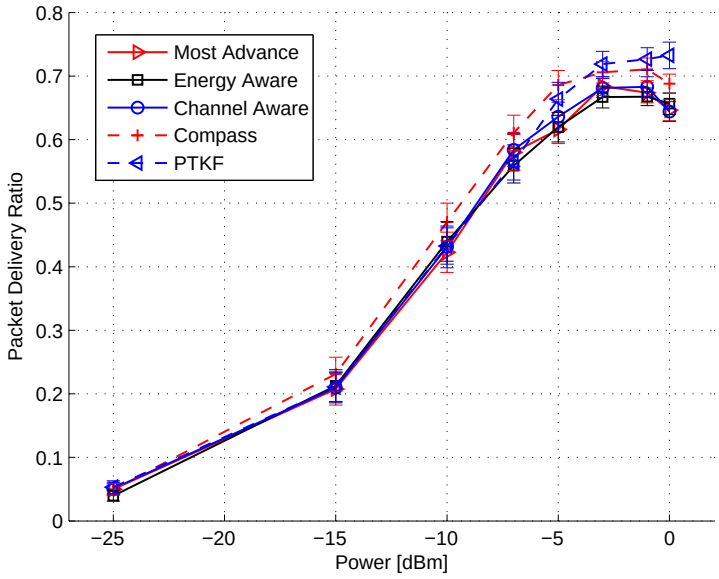


Figure 2.7: Packet Delivery Ratio vs. NDP power of 25 Nodes

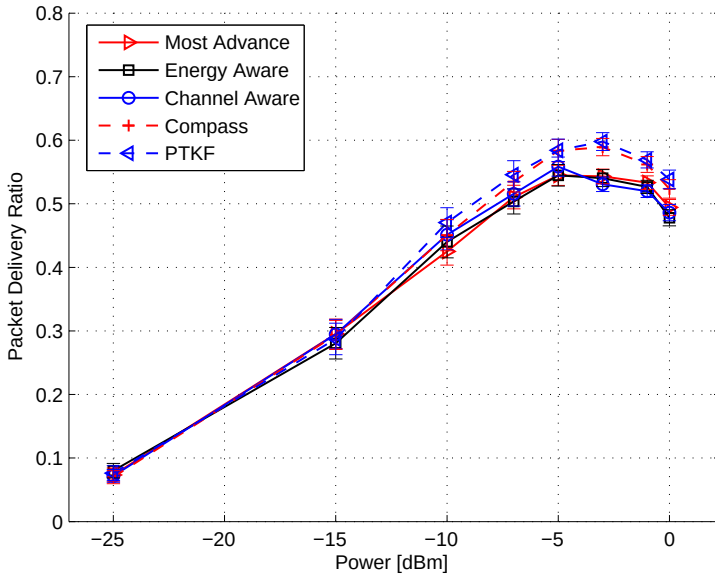


Figure 2.8: Packet Delivery Ratio vs. NDP power of 36 Nodes

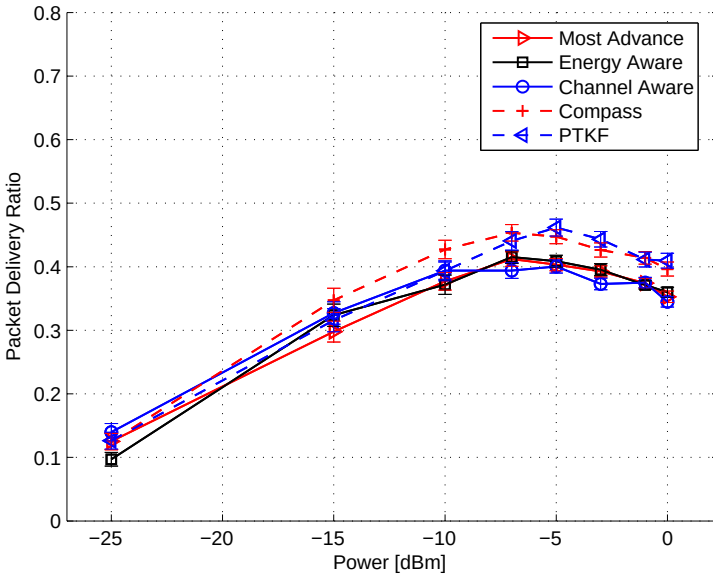


Figure 2.9: Packet Delivery Ratio vs. NDP power of 49 Nodes

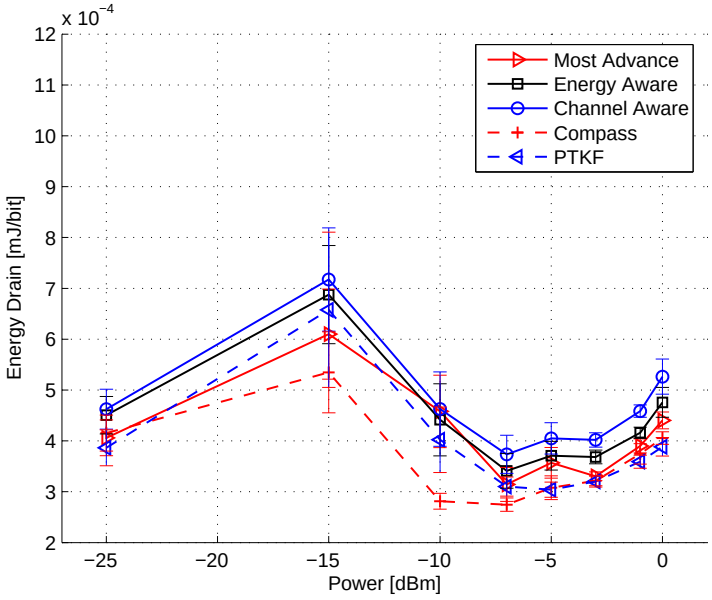


Figure 2.10: Energy Drain vs. NDP power of 25 Nodes

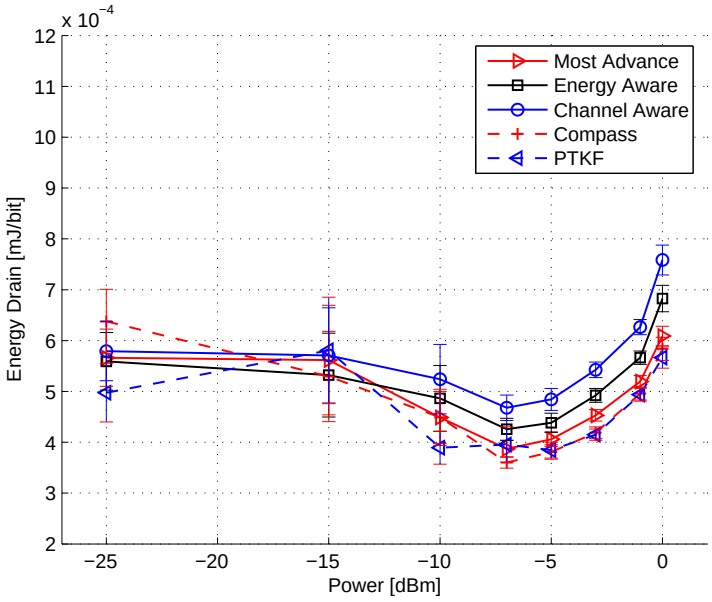


Figure 2.11: Energy Drain vs. NDP power of 36 Nodes

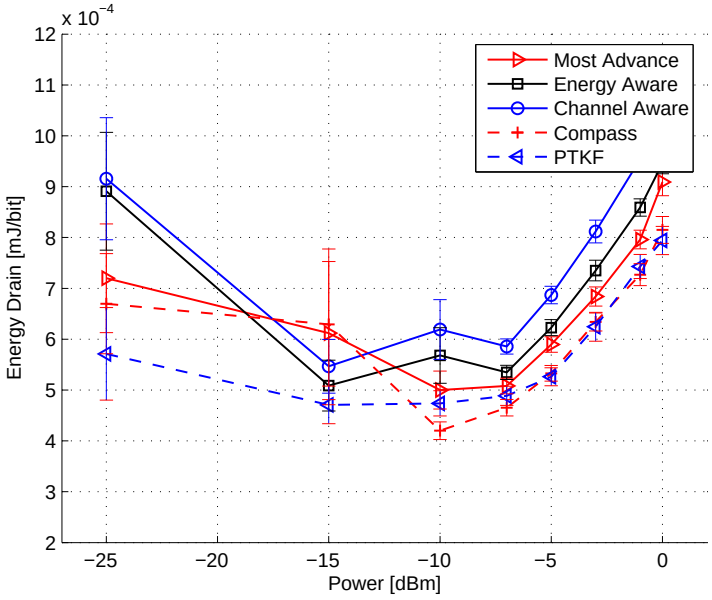


Figure 2.12: Energy Drain vs. NDP power of 49 Nodes

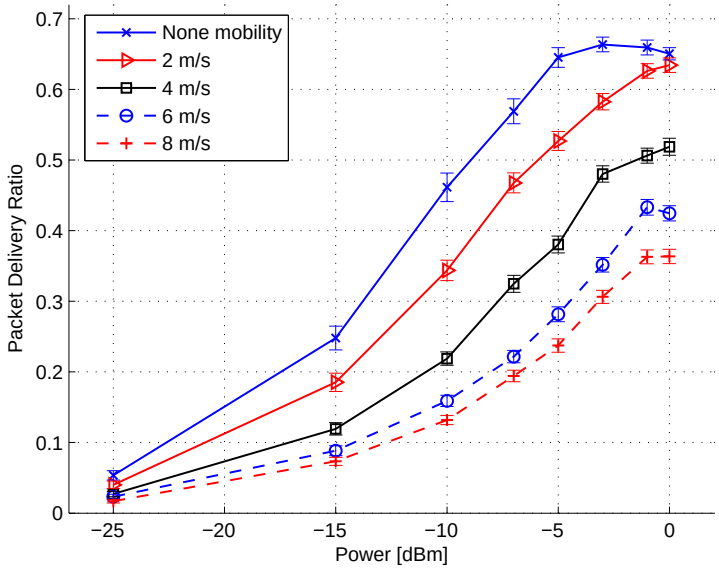


Figure 2.13: Packet Delivery Ratio in mobility (Most Advance Routing) for 25 nodes

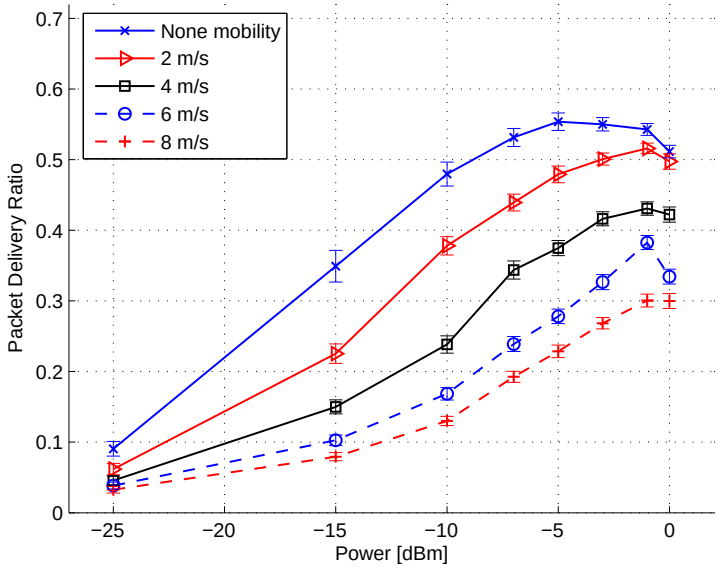


Figure 2.14: Packet Delivery Ratio in mobility (Most Advance Routing) for 36 nodes

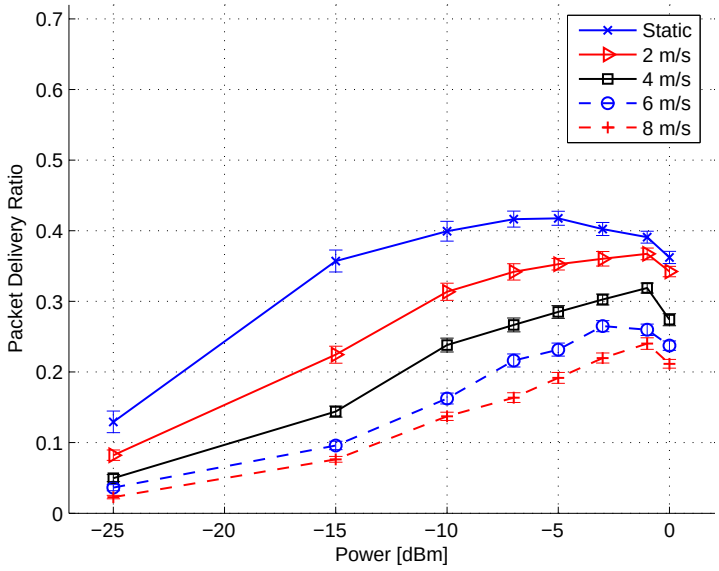


Figure 2.15: Packet Delivery Ratio in mobility (Most Advance Routing) for 49 nodes

It is intuitive that if mobility increases, the packet delivery ratio decreases because nodes will not be getting the updated information from the neighbors because all the nodes are moving around, even if they are receiving ND packets from the neighbors. Therefore, routing algorithm cannot perform well. From Fig. 2.13, 2.14 and 2.15, we can infer that, when the modes are moving faster, higher power is needed to capture the mobility. Fig. 2.16 shows the optimal power value across various velocities. It can be inferred from Fig. 2.16 that optimal power increases as velocity increases and also as node density increase the power required to capture mobility decreases as nodes are closer.

The second issue we faced is, "How often should a node collect the information from the neighborhood?" This question is answered by varying NDP update frequencies - $(0.02, 0.42, 0.82, 1.22, 1.62, 2.02, 2.42)$ Hz. For static case, Fig. 2.18 shows that packet delivery ratio decreases as NDP frequency increases because all the nodes frequently send NDP packets, which increases the traffic in the network thereby causing congestion around the sink. Again, this figure shows that higher power of the nodes does not guarantee higher packet delivery ratio. The power (-5 dBm) is optimal power for the network. For the mobility case, when the nodes are moving, a node needs information

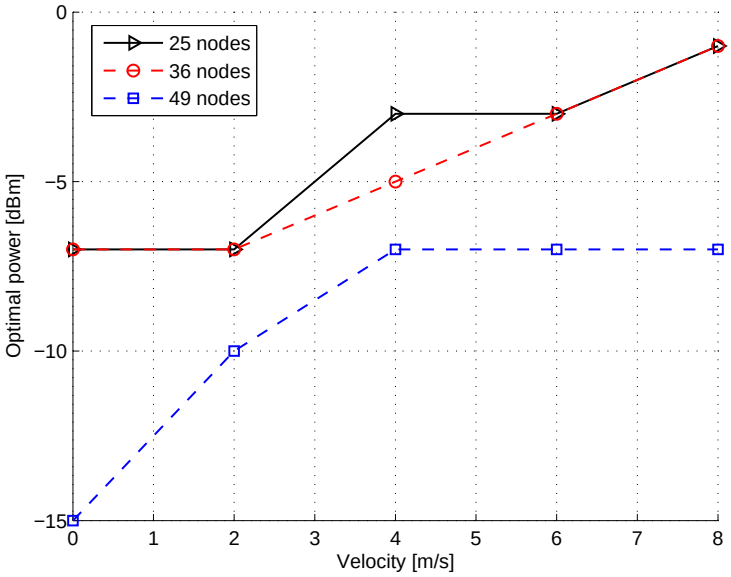


Figure 2.16: Optimal power with different velocity

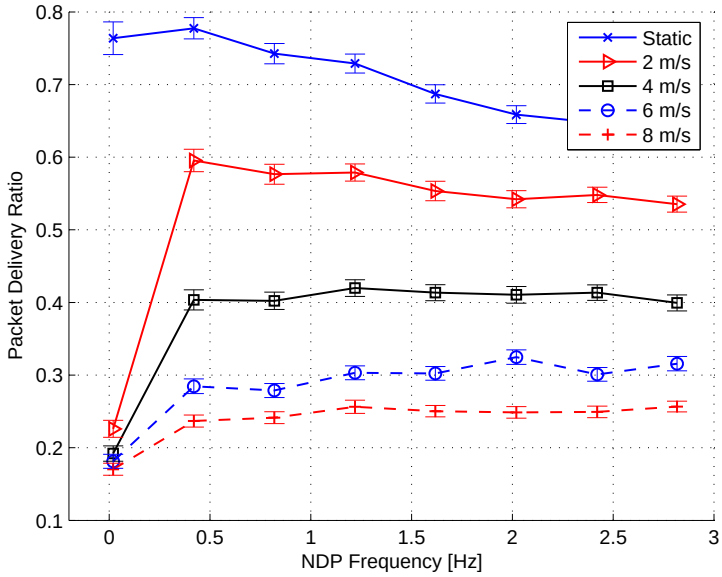


Figure 2.17: Packet delivery ratio vs. NDP (Most advance, 25 nodes)

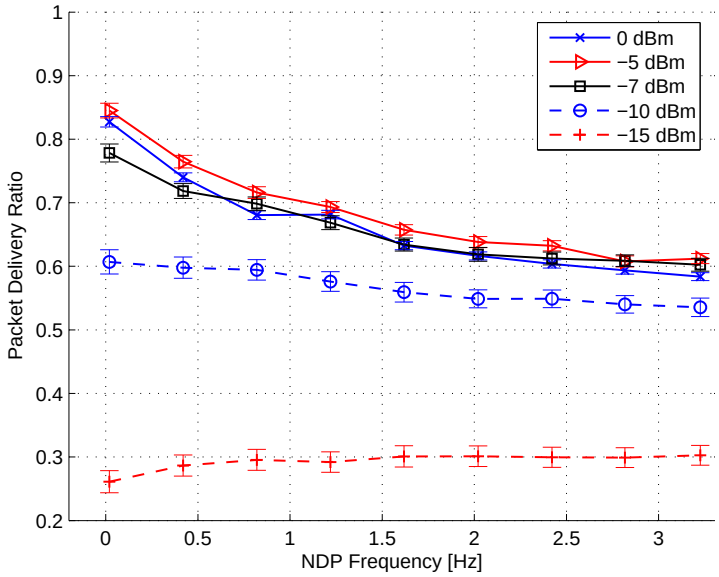


Figure 2.18: Packet delivery ratio vs. NDP Frequency for static nodes

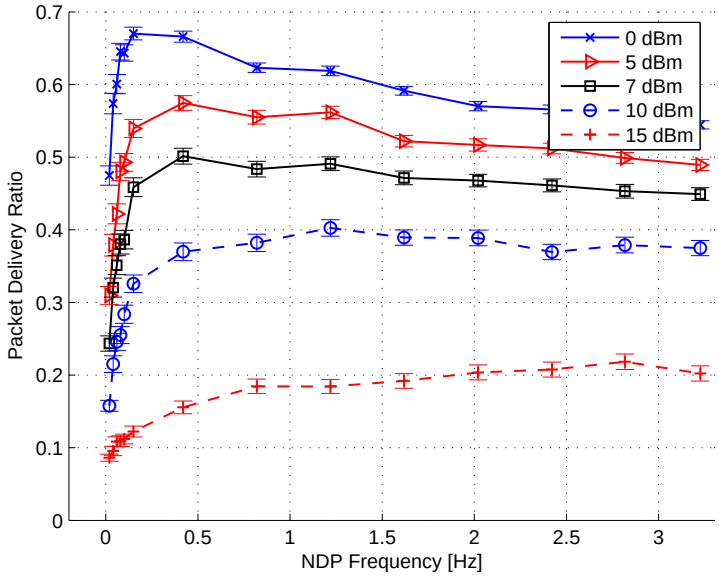


Figure 2.19: Packet delivery ratio vs. NDP Frequency for mobile nodes 2m/s

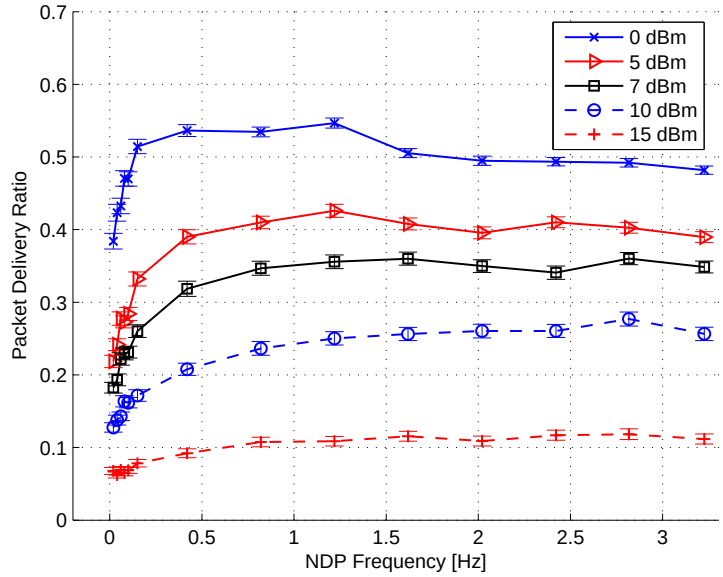


Figure 2.20: Packet delivery ratio vs. NDP Frequency for mobile nodes 4m/s

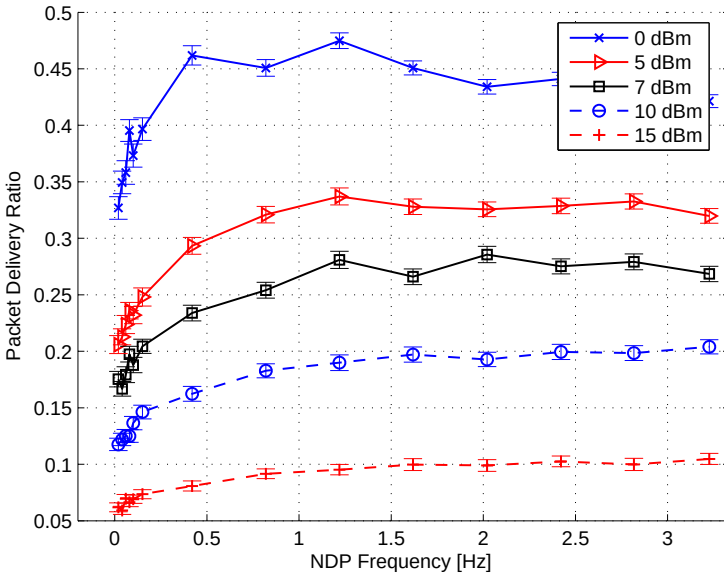


Figure 2.21: Packet delivery ratio vs. NDP Frequency for mobile nodes 6m/s

more frequently. Figure 2.17 shows that overall the packet delivery ratio decreases when NDP update frequency increases while the packet delivery ratio decreases as the velocity increases. Moreover, Fig. 2.19, Fig. 2.20, and Fig. 2.21show packet delivery ratio with specific power - (0571015) dBm. Nodes with high power show higher packet delivery ratio, reducing routing failure illustrated in the Fig. 2.6. The figures also show that if the NDP frequency is too low, there will be a decrease in the packet delivery as the information about the nodes is not updated.

Finally, we answer the question "How much information is required from the neighboring sensor nodes?". Different routing schemes need different information for routing packets and require some packet field to collect it. Most Advance, Compass, and PTKF only need position information, which accounts to 17 bytes, including header size of CC2420. Energy aware requires additional bytes for information on available energy and Tx-Power required to reach destination node, which makes the packet size to be 22 bytes. Also, Channel Aware needs 2 bytes in addition for sending bit error rate calculated using RSSI, thereby making the total packet size as 19 bytes. The more the information we use, the more efficient routing can be done, but this increases the overhead and thereby decreases the available bandwidth. However, simulations results show that whichever be the routing scheme there will not be a drastic change in the performance as compared to the dependence on NDP power and NDP frequency.

In summary, Fig. 2.22 shows relation between optimal power vs. optimal NDP frequency with mobility. When nodes move faster from 0m/s to 8m/s transmission power and frequency should be adjusted properly. Nodes should increase not only the transmission power increase NDP frequency if the nodes move faster in order to have higher packet delivery ratio. Furthermore, in slow mobility, nodes should decrease transmission power and NDP frequency.

2.4.2 Hardware Implementation

To substantiate our results in simulation we did hardware implementation in TelosB (TPR2420) motes built on IEEE 802.15.4 compliant CC2420 radio (2.4 GHz), having data rate of 250 kbps. Experiments were done on a grid size of 40 x 40 m^2 with a

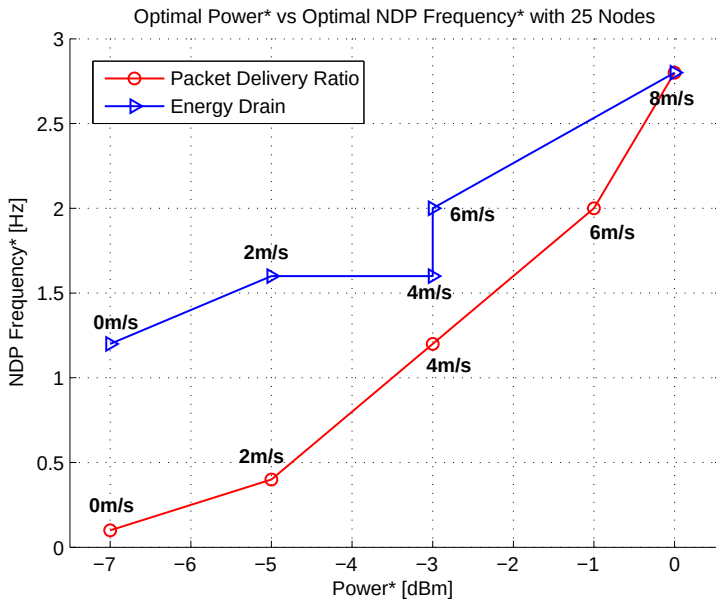


Figure 2.22: Optimal Packet delivery ratio vs. Optimal NDP Frequency for mobile nodes

total of 9 motes spaced approximately 20 m from each other. For assessing packet delivery ratio and energy drain with respect to NDP power, we considered those power values supported by TelosB $(0, -1, -3, -5, -7, -10, -15, -25)$ dBm at NDP frequency of 1 Hz. While for performance metrics against NDP frequency we took values of $(0.02, 0.42, 0.82, 1.22, 1.62, 2.02, 2.42, 2.82, 3.22)$ Hz at -5 dBm NDP power. The data frequency chosen for the experiment was 2 Hz. The figures from the experiments are as follows. Figures 2.23 and 2.25 show that there is an optimal power at which NDP packets should be sent, which confers with our results in simulation. Figures 2.24 and 2.26 also show that there is an optimal NDP frequency at which NDP packets need to be sent.

2.5 Conclusions

We analyzed how neighborhood discovery power and neighborhood discovery frequency can affect packet delivery ratio and energy consumption, thereby stressing on the impact of the neighborhood discovery protocol on routing schemes. We also verified how parameters such as power and frequency transmission of neighborhood discovery packets

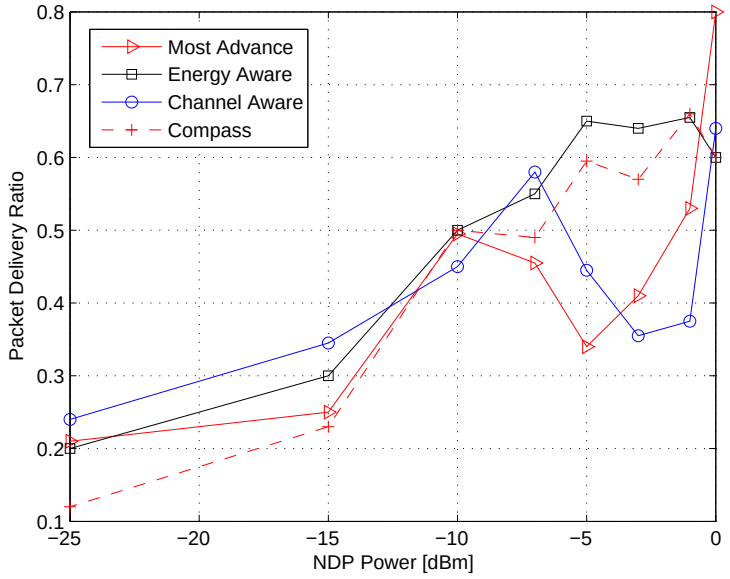


Figure 2.23: PDR vs. Power (TelosB)

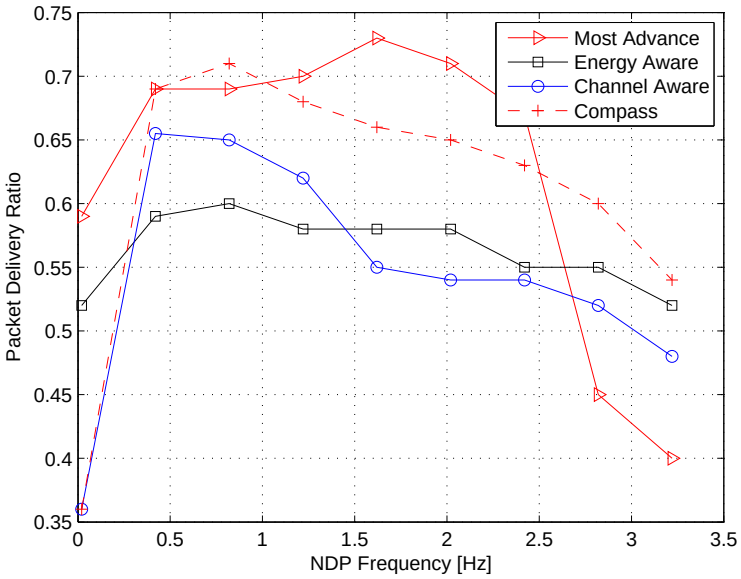


Figure 2.24: PDR vs. NDP (TelosB)

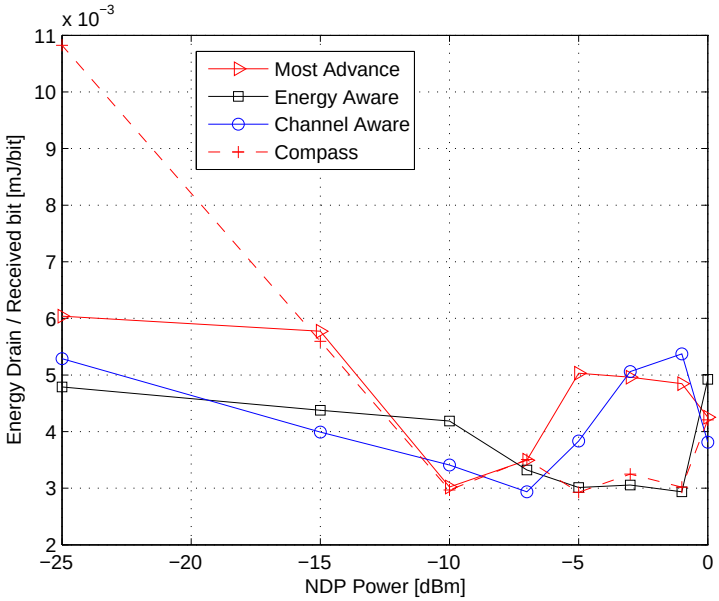


Figure 2.25: Energy drain vs. Power (TelosB)

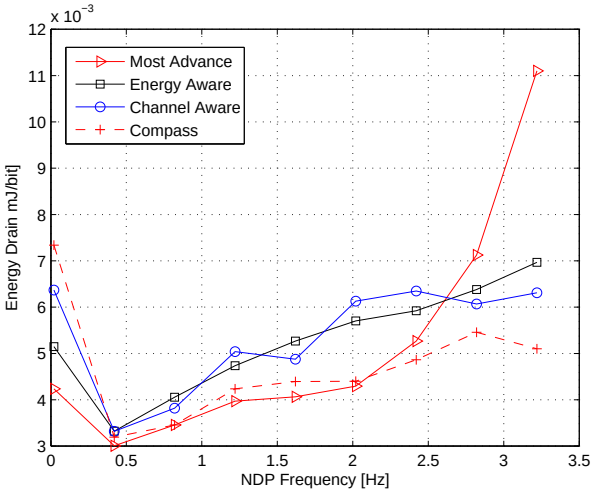


Figure 2.26: Energy drain vs. NDP (TelosB)

affect the communication process in both static and mobile environments. We conclude by saying that there is an optimal power and optimal frequency for neighborhood discovery in attaining energy efficiency rather using highest power or higher frequency.

Chapter 3

Adaptive Sampling in Wireless Sensor Networks Measuring Multiple Phenomenal Manifestations

3.1 Introduction

Wireless Sensor Networks (WSNs) are composed of autonomous devices that co-operatively monitor physical or environmental conditions with various manifestations such as temperature, radiation, vibration, light, pressure and sound in Fig. 3.1 at different points in space and time. The features that distinguish WSNs from classical wireless networks are strict limitations on energy consumption, high density of nodes and limited processing capability of the nodes. The reasons for dense deployment of sensor nodes can be stated as follows:

1. The constraint on output power of the battery-operated sensor nodes precludes them from using large transmission ranges [20].

2. The lack of apriori information about the terrain prevents pre-deployment optimization on the number of sensors mainly due to multiple manifestations of the phenomenon. For example, manifestations or functions of a nuclear explosion would be temperature, pressure and levels of toxic gases in the affected region.

3. Low cost of individual sensor nodes.

Due to the high density of sensor nodes, the degree of correlation among spatially proximal sensor observations increases with decrease in internode distance (Fig. 3.2) Also, the degree of correlation between consecutive sensor measurements may vary

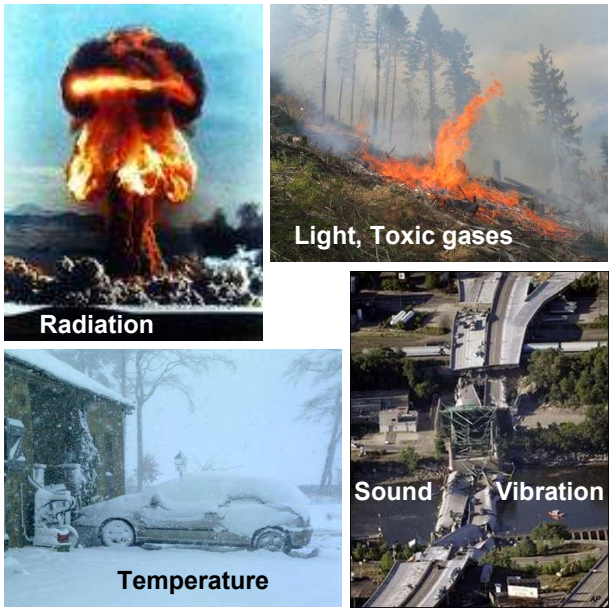


Figure 3.1: Various manifestations

according to the temporal variation characteristics of the phenomenon. The former is termed as *spatial correlation* and the latter is referred to as *temporal correlation* [21].

The main objective of monitoring environment using WSNs is to reliably estimate the phenomenon from the collective information provided by these sensor nodes [22]. Usually, higher densities of sensors may imply more measurements, higher resolution and better accuracy of estimation, but require more energy expenditure for communication and processing. However, energy-efficient estimation of the phenomenon can be done by leveraging the above mentioned spatial, temporal correlation and multiple measurement between the sensor data [23]. Thus, in order to increase the longevity of WSNs and/or to ensure effective utilization of resources, adaptive field estimation techniques have been developed.

Adaptive sampling is a method that employs the spatio-temporal correlation inherent in WSNs to estimate the phenomenon efficiently with a desired level of accuracy. Efficiency here refers to minimization of energy and/or delay by using only a subset of the deployed sensor nodes. The field estimate obtained using adaptive sampling is useful in developing energy-efficient MAC and routing protocols for sensor networks. Also,

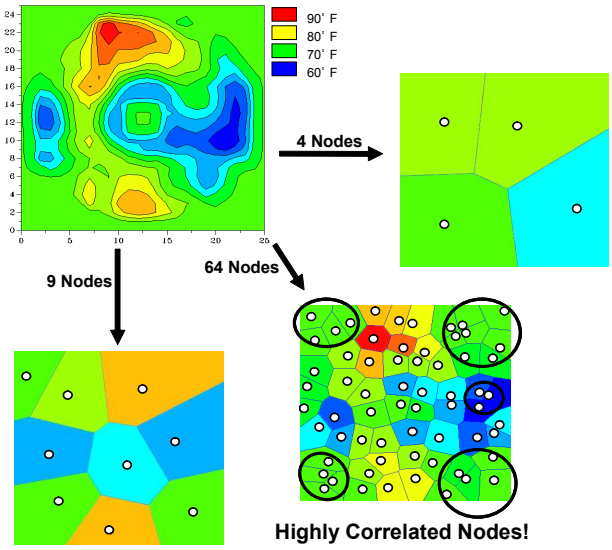


Figure 3.2: Voronoi diagram

they have been effectively employed in algorithms such as *sleep mode scheduling* and *data aggregation* [21]. Sleep mode scheduling ensures longevity of the WSNs by using different subset of nodes to observe the phenomenon in different time periods. The idea behind data aggregation is to combine the data coming from different sources so as to eliminate redundancy and minimize the number of transmissions. Thus, data aggregation limits the communication to the local domain and transmits only the necessary information to the whole network.

In addition, one sensor has a capability of sensing multiple manifestations using different types of sensors. This can help sensors to capture a phenomenon more accurately than using a single sensor because they are interrelated to each other. For instance, when fire occurs, temperature increases while humidity decreases. If only a temperature measurement is used as a manifestation to sense fire, other high-temperature but different phenomena would be misrepresented as fire (i.e., monsoon humid weather or hardware overheating). Increasing temperature by itself can not indicate fire occurrence. As for energy efficient issues, using multiple sensors does not significantly impact overall energy consumption because the energy required for activating sensing module is negligible when the node are in active status (Fig. 3.3). Thus, by simultaneously

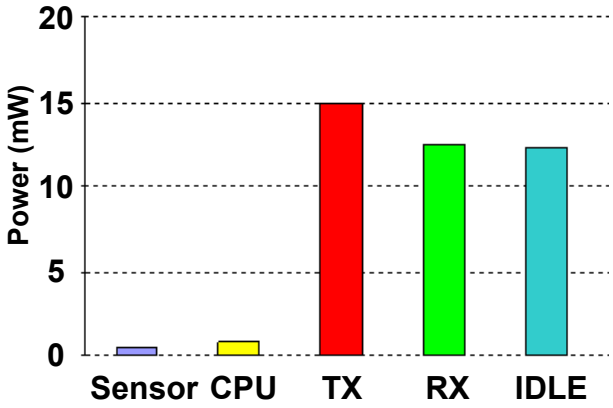


Figure 3.3: Energy consumption with different functionality of a node

using different sensors, accuracy of phenomena monitoring can be improved.

In this chapter, we propose a decentralized, hierarchial, cluster-based adaptive sampling framework for estimating a three-dimension scalar field with multiple interacting manifestations of the phenomenon that vary both in time and space. The sensor field is partitioned into clusters of nodes sensing similar values within a given threshold. Such clusters are aggregated at various levels based on similarity of sensed data and this results in a hierarchy of clusters. The spatial correlation characteristics of the phenomenon is leveraged to achieve this. The clusters are modified to ensure membership consistency as the sensor readings change over time. This is referred to as *cluster resizing* and the temporal correlation characteristics of the phenomenon is used to achieve this. This approach would minimize global communication and restrict data communication to the local domain (clusters), thus saving on communication cost as well as saving energy by applying sleep mode where the nodes are not used as a part of the network.

Our algorithm differs from most of the existing solutions [22] [24] [25] [23] in the following ways:

- It does not assume any apriori knowledge about the characteristics of the phenomenon and the topology of the WSN
- It ensures joint optimization by adaptively varying the sampling rates in both

space and time domains. Existing solution [22] performs adaptive sampling separately (spatial, temporal)

- It performs in-network clustering in real time. Cluster aggregation and separation are performed in real time
- It achieves accurate understanding of the phenomenon by analyzing cross-correlation among manifestations
- It supports adaptive sampling in the three-dimension environment.

This paper is organized as follows. Sect. 3.2 addresses related work and compares the existing adaptive sampling schemes for field estimation. In Sect. 3.3 we stress out the motivations and goals for our work. In Sect. 3.4 we describe our algorithm in detail. Sect. 3.5 shows the results of the simulations and evaluate the performance.

3.2 Related Work

Efforts to design efficient adaptive sampling policies fall under three different paradigms: *centralized*, *autonomous (distributed)* and *quasi autonomous*.

3.2.1 Centralized

The centralized paradigm is characterized by the presence of a central server/sink that has a global view of the network. All the sensor nodes report their data to a sink. Based on the collected data the sink decides on the appropriate spatial sampling rate (subset of sensor nodes to use) and the temporal sampling rates (the rates at which the nodes collect data).

The backcasting adaptive sampling method [23] [26] operates by first activating only a small subset of the wireless sensor nodes that communicate their information to a base-station. This provides an initial estimate of the sensed environment and guides the allocation of additional network resources. The base station then selectively activates additional sensor nodes in order to achieve a desired level of accuracy (based

upon this information). In [27], an algorithm for the selection of active sensors in a WSN, whose application is to reconstruct the data image of spatially bandlimited physical phenomenon, is proposed. The proposed selection method, which can be either centralized or distributed, creates a sampling pattern based on blue noise masking (spectral content with only high frequencies) and guarantees a near minimal number of activated sensors for a given signal-to-noise ratio. The main limitation of the algorithm is that it works under the assumption of single-hop communication between sensor nodes and the sink.

The above mentioned algorithms [26] [23] [27] do not address the issue of adaptive sampling in the time domain. Moreover, even though the centralized framework promises accurate field estimates, the cost of communication incurred will be very high as the number of nodes in the WSN increases. Also the robustness of the system entirely rests on the reliable communication between the sensor nodes and the sink.

Thus, the centralized approach has issues associated with scalability, robustness and the dynamic nature of WSNs. This served as the motivation for the evolution of autonomous and quasi autonomous approaches.

3.2.2 Autonomous

The autonomous framework aims to improve robustness of the system by eliminating the dependence on global communication (between the sensors and the sink) in the network.

Researchers have recently proposed a decentralized control mechanism for adaptive sampling called Utility based Sensing And Communication (USAC) [25] [28]. The USAC mechanism consists of two components namely, the *sensing protocol* and the *communication protocol*. The sensing algorithm uses a linear regression method that predicts the next observed data with some bounded error (termed as confidence interval, CI). If the next observed data falls outside this CI, the node sets its sampling rate to the maximum rate in order to incorporate this phase change. However, if data falls within the CI, it implies that the node is allowed to reduce its sampling rate for energy efficiency due to the lack of any new information. The main drawback of this paper is

that it fails to take into account the spatial correlation present in the phenomenon.

In [29] and [30], a distributed sensor selection strategy is proposed to choose a subset of sensors (spatial sampling) to achieve a desired fidelity in observing the physical phenomenon in the field. Intuitively, if the noise level is low, a small number of sensors is sufficient to achieve the desired fidelity; however, if the noise condition is severe, more sensors should be activated for accurate estimation. They propose a sampling and estimation framework based on *linear minimum-variance-unbiased-estimator* that exploits a strategy called *innovation diffusion*. Innovation refers to the new information that a sensor measurement contributes to the reduction of the estimation error, and diffusion refers to the process by which the innovation is communicated across the network. Moreover this method does not vary sampling rate based on the temporal correlation present in the sensor data. The decentralized flavor of blue noise spatial sampling algorithm [27] performs on par with its centralized counterpart.

3.2.3 Quasi Autonomous

The design of an efficient autonomous framework for adaptive sampling and estimation is difficult as it introduces additional control issues related to interactions between networks of interconnected nodes in the absence of a central co-ordinating server. The quasi autonomous paradigm offers a balance in sharing the tasks between the server and the individual sensor nodes.

Self Organizing Resource Allocation (SORA) is an approach for determining efficient node resource allocations in WSN by using a market-based approach [31]. Rather than manually tuning node resource usage, SORA defines a virtual market in which nodes sell goods (such as data sampling, data relaying, data listening, and data aggregation) in response to global price information that is established by the end-user. With SORA, nodes independently determine their ideal behaviors by taking actions to maximize their own utilities, subject to energy constraints (temporal sampling). However, prices are determined and set by an external coordinator agent (sink) to induce a desired network's global behavior (spatial sampling).

In [32] and [33] the authors propose a new prediction-based environment monitoring.

A spatio-temporal prediction model from historical sensor data is used to estimate the current reading at each sensor node. Only when the actual reading differs from the given prediction model above a specified threshold, it is transmitted over to the sink thus effectively reducing the energy spent on communication. The disadvantage lies in the inability to acquire accurate models for highly varying physical phenomenon.

In [24] a quasi autonomous Kalman Filter (KF) based adaptive sampling technique is proposed. KF estimation error is used to adaptively adjust the temporal sampling rate within a given range. If the desired sampling rate exceeds the allowable range then the new sampling rate is requested from the sink. This technique has not yet been validated for multi-hop sensor networks and in real life scenarios. This method addresses adaptive sampling only in the time domain and not in space.

Finally, analytical results obtained in [34] characterize the tradeoffs among energy usage, delay, temporal sampling rates and spatial sampling rates in WSNs. The authors have derived a lower bound on the delay incurred in gathering one packet of samples at a given spatial sampling rate and also the energy consumed in gathering data at a given spatial sampling rate.

The autonomous or distributed approach, characterized by the absence of central coordinating authority, promises scalability, robustness and support for a very high degree of dynamism in WSNs. This motivates us to explore this design paradigm further, to devise a novel, distributed, adaptive sampling scheme to estimate the phenomenon with a desired level of accuracy, while ensuring efficient energy and resource utilization.

3.3 Problem Formulation

As we discussed in Sect. 3.1, adaptive sampling plays a crucial role in the dense deployment of the sensors for energy efficiency. From this point of view, motivations of this paper are summarized hereafter in more detail.

- Most of the existing algorithms for adaptive field estimation exploit either the spatial or temporal correlation inherent in the sensor field. Very few ensure joint optimization by adaptively modifying both the spatial and temporal sampling

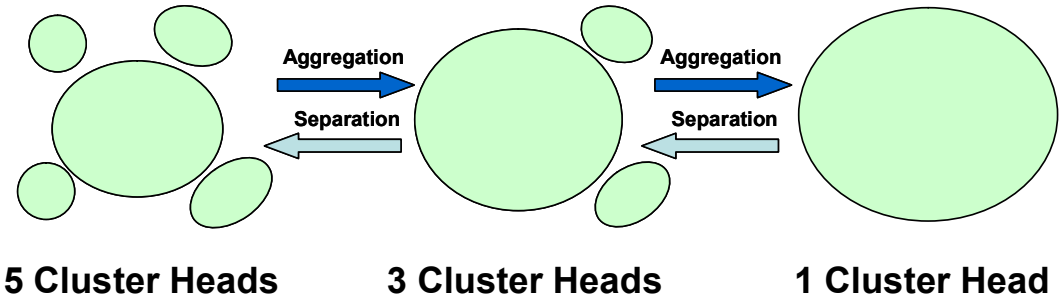


Figure 3.4: Dynamic joint optimization

rates [33] [32] [31];

- Energy consumed for data communication is much higher than the energy consumed for sensing and listening on the control channel combined. Hence, the key to energy efficiency is reduction of global communication of sensor data. A hierarchical clustering based approach to partition the sensor field would minimize global communication and restrict data communication to the local domain (clusters);

- Noise generated from multi hop communication due to the small radio range of sensors creates packet loss. Minimizing global communication by clustering reduces packet loss as well.

- The distributed adaptive sampling schemes unlike its centralized counterparts are scalable to dense and highly dynamic sensor networks.

- Existing algorithms set up sampling rate manually, but in our algorithm, cluster aggregation and separations are performed in real time

- More accurate understanding of the phenomenon could be achieved by analyzing cross-correlation among manifestations

- Further applications require three-dimension sensor deployment with plenty of data (i.e., data center monitoring and oil field monitoring)

Our goal is to develop a decentralized, hierarchial, cluster-based adaptive sampling framework, which will ensure joint optimization in both space and time domains in three-dimension environment, using multiple interacting manifestations. Fig. 3.4 shows dynamic optimization in space. Data in time change are leveraged and algorithm forms Cluster head by aggregating data. It will be specifically explained in Sect. 3.4. It is a generalized solution in the sense that it does not assume any apriori information about the field being estimated.

The given sensor field is partitioned into clusters of nodes sensing similar values within a given threshold. Such clusters are aggregated at various levels based on similarity of sensed data and this results in a hierarchy of clusters. The spatial correlation characteristics of the phenomenon is leveraged to achieve this. The temporal correlation characteristics of the phenomenon are used for cluster resizing, which ensures membership consistency as the sensor readings change over time. The *mean of error*, described in Sect. 3.4 helps in forming the hierarchy of clusters and the simple first order statistics. To achieve these aforementioned goals, we make the following basic assumptions about the sensor network.

- *Position:* Each sensor node i is aware of its position $\mathbf{p}_i$ in the sensor field and is also capable of deriving its relative position with any other node j (whose position is $\mathbf{p}_j$) in the field. This is necessary to spatially reconstruct the data at the sink.

- *Synchronization:* The computation of spatial correlation requires that the data obtained from all the sensors be from a particular time instant. Also, to find the temporal correlation we need sensor measurements from two different time instants. Hence, we assume that a mechanism to ensure time synchronization between the sensor nodes exists.

- *Multi-hop communication:* We assume that multi-hop communication between any two sensor nodes in the field is possible through the use of an appropriate routing protocol.

While in the following sections, we focus on two-dimensional fields and sensor networks to illustrate our ideas, extension of our theory and methods to three-dimensional fields is straight forward. The next section presents the background and notations that will be used in the rest of the paper.

3.4 Proposed Solution

The key concepts of adaptive sampling are data aggregation and energy conservation for an entire network. In order to aggregate data for adaptive sampling, criteria for data aggregation is introduced in Sect. 3.4.1. Brief notations for our algorithm will be explained in Sect. 3.4.2. Our algorithms for clustering and cluster resizing, and joint optimization will be explained in Sect. 3.4.3. Then, we expand the scope of criteria in Sect. 3.4.1 to discrete time domain with multiple manifestations in three-dimension space in Sect. 3.4.4. Finally, Illustrative example clarify our algorithms in Sect. 3.4.5.

3.4.1 Criteria for Adaptive Sampling

Adaptive sampling method achieves energy saving based on data aggregation. In this section, we introduce criteria for data aggregation in deciding where data from a node to be aggregated/clustered. Error (e) is a measure that helps in forming clusters, and based on this error (e) between sensor data, CHs compute them and decide data aggregation. CHs decide whether the data from a adjacent node is correlated or not based on the criteria denoted as error term (e). In other words, degree of error (e) affects degree of correlation, so that a node data which has high degree of error(e) will not be associated CH because high error value represents low correlation between data. This is spatial correlation.

Let $d_{ij} = \|\mathbf{p_i} - \mathbf{p_j}\|$ denote the Euclidean distance between the sensor nodes at locations $\mathbf{p_i}$ and $\mathbf{p_j}$ and $n(h)$ represent the number of sensors within the radius h, where $\{h : d_{ij} \in h \ \forall \ i\}$ from any node j. Also, ψ_i represents the data from manifestations

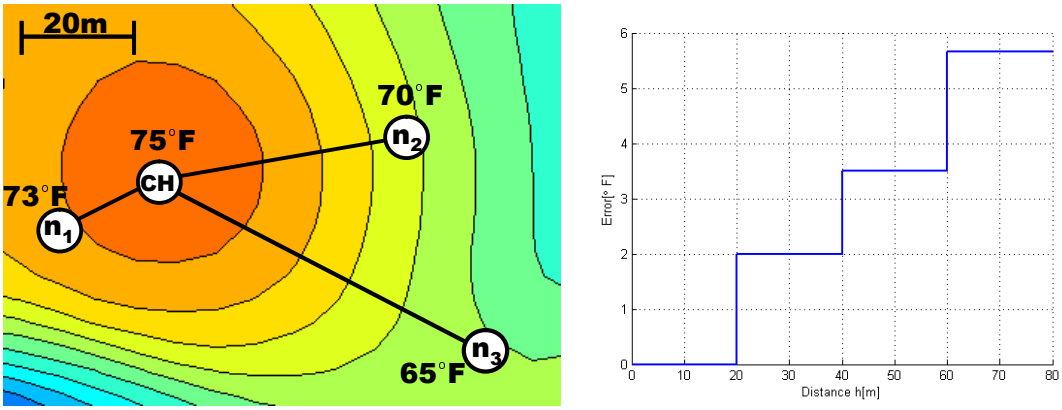


Figure 3.5: Mean of error calculation for a manifestation (temperature) on a contour temperature map

from node i. In spatial statistics, the error within h is defined as,

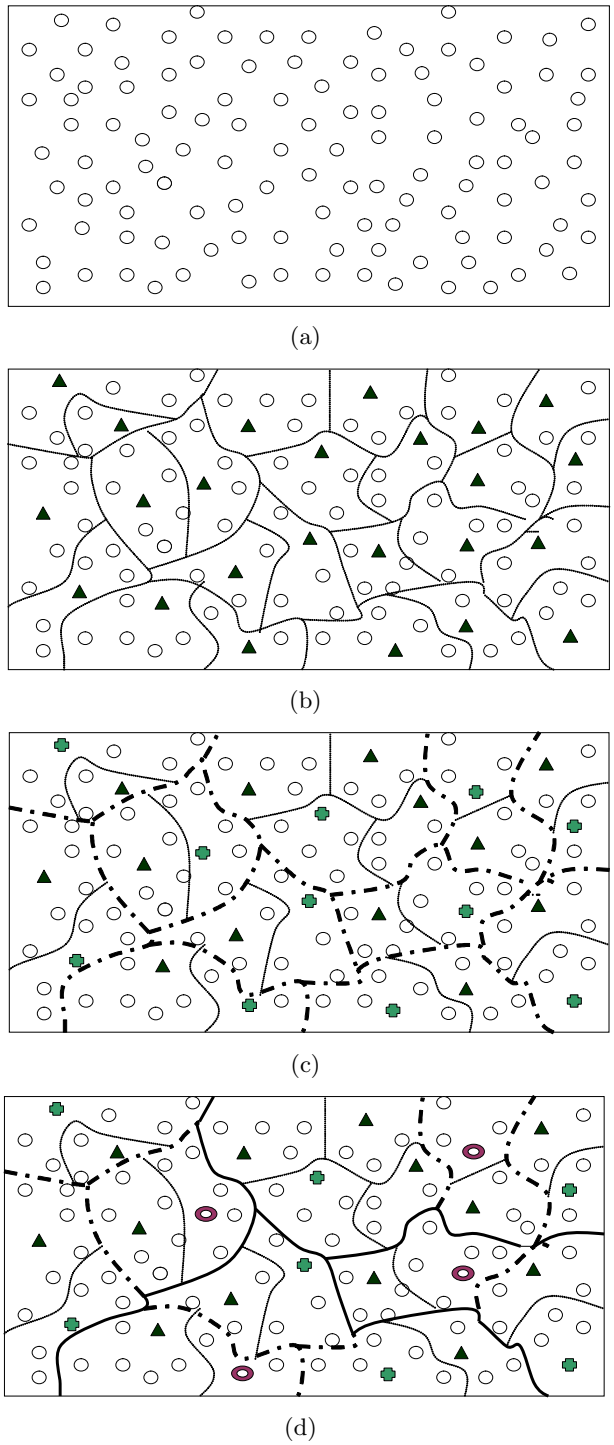
$$e(h) = \sum_{i=1, i \neq j}^{n(h)} \frac{|\psi_i - \psi_j|}{n(h)} \tag{3.1}$$

The error *e(h)* states the degree of spatial correlation among data from sensor nodes distributed in radius space(h). Fig. 3.5 shows how clusterhead node (CH) spatially associates with other adjacent nodes in contour temperature field and shows calculated error ($e(h)$) at the right. All the nodes having data within error *e(h)+ margin* will associate with CH.

For temporal correlation, change in the sensor readings over time indicates randomness in the system. Cluster resizing needs to be done to keep membership consistency among the nodes in a cluster. Let x_i denote the $i-th$ sample when temporal correlation algorithm starts and N represent the number of sample data in the data sequence.

$$e(i) = \sum_{i=1}^{N} \frac{|x_i - x_{i-1}|}{i} \tag{3.2}$$

If the change e_i from its previous values falls within a particular threshold, then the cluster need not be resized. The threshold is determined by the manifestations of the phenomenon being observed. The criteria described here help in estimating the amount of correlation present in the phenomenon in space and time domain.



(a)

(b)

(c)

(d)

Figure 3.6: (a) Sensor field with L_0 CHs. (b) Sensor field with L_1 partitioning (dotted lines) and L_1 CHs (solid triangle). (c) Sensor field with L_2 partitioning (dashed lines) and L_2 CHs (solid plus). (d) Sensor field with L_3 partitioning (solid lines) and L_3 CHs (rings)

3.4.2 Notations

In the course of the algorithm, sensor nodes will be organized into clusters at many levels and will have designated Cluster Heads (CHs). A *Level k* cluster is represented as L_k and it is composed entirely of L_{k-1} CHs. Levels L_{k-2} and lesser are completely abstracted from the view of L_k CHs, but every L_k CH knows all its higher level CHs from L_{k+1} to L_K, where K is the maximum number of cluster levels. A L_k CH's CH_LIST contains the details about all its L_{k+1} to L_K CHs. All the individual nodes are considered L_0 CHs for the sake of generality.

An example of a 2-dimensional sensor field is shown in Fig. 3.6(a). The L_0 CHs are represented by circles. A group of L_0 CHs within the dotted line boundary form the L_1 and its CH is represented by a solid triangle as shown in Fig. 3.6(b). L_1 CHs within a dashed line boundary form the L_2 with its CH represented by a solid plus sign as shown in Fig. 3.6(c). Although the L_0 CHs within this region form part of the L_2, the L_2 CH is only aware of the L_1 CHs and lower level CHs are completely abstracted from its view. L_3 is the region bounded by solid lines and L_3 CHs are represented by rings as shown in Fig. 3.6(d).

3.4.3 Algorithms for Clustering

The algorithm we propose consists of two separate processes: *clustering* and *cluster resizing*. Clustering is implemented to form a cluster with nodes which have highly correlated data in space. Cluster resizing runs to keep the consistency of membership of this cluster in varying phenomenon in time.

We used four-way handshake control protocol in order to jointly optimize spatial and temporal sampling rate. Fig. 3.7 shows the timeline diagram of the four-way handshake protocol. The algorithm begins with the election of L_1 CHs for clustering a sensor field. Each L_0 CH simultaneously picks a random number using the probability function RAND_GEN (*count*), where *count* represents the level for which the CHs are being chosen. If the number falls within a particular threshold (that depends on *count*) then the L_0 CH elevates itself to be a L_1 CH. Each of those newly elected L_1

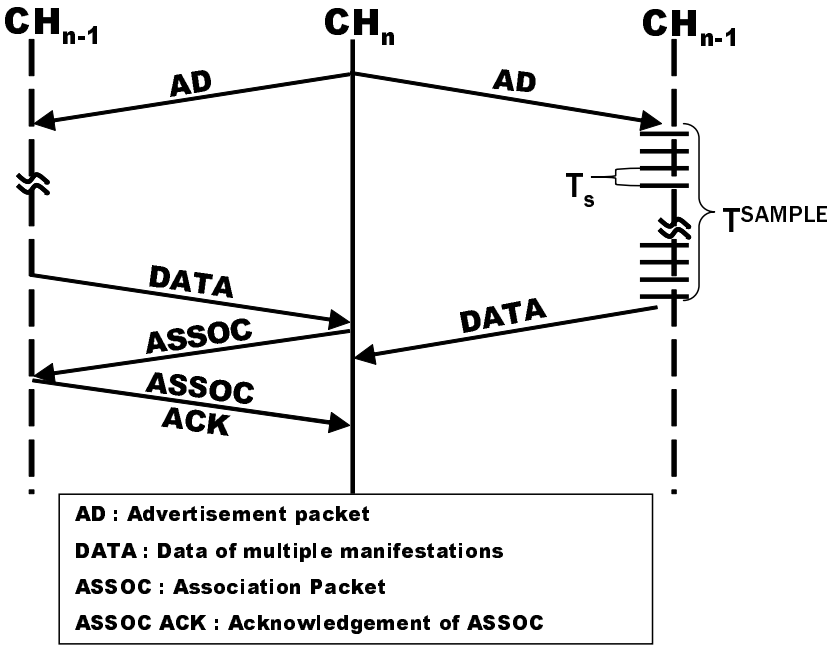


Figure 3.7: Timeline diagram for spatio-temporal adaptive sampling

CHs advertises its election to the adjacent nodes within certain distances which vary according to the level of CH using TX_ AD_ PKT($L_k - 1$) as shown in Algorithm 1. The sensor nodes reply to the advertisements with their position, sensed values from number of manifestations and the corresponding time stamp. After particular time T^{SAMPLE}, the L_1 CH computes the mean of error of the data from nodes at different distances using the function COMPUTE_ MEAN_OF_ERROR (L_1). Hence, Spatial adaptive sampling is achieved. Also, a number of samples from multiple manifestations are collected based on the sampling period (T^{SAMPLE}) with sampling rate of each node determined by the L_1 CH in the Fig.3.7. L_1 CH decides sampling period (T^{SAMPLE}) based on the sampled data gathered from the previous cycle of the protocol. Initial sampling rate is $2 \cdot F_{MAX} + 1$ because sufficient condition for exact constructability from samples is f ¿ 2B where B is a bandwidth based on Nyquist-Shannon sampling theorem. CALC_ REF() calculate sampling rate based on the data collected from each node and decide sampling rate. In this respect, temporal adaptive sampling is achieved.

If the data from a node is eligible for association in space and time domain, in turn,

if all the nodes within a distance d_{th} and within a time period T^{SAMPLE} that satisfy the threshold of error calculation (provided in Sect. 3.4.1) set by the CH, a node is considered as one of the members in particular L_1 CH. If the data is highly correlated each other, the L_1 CH send association packet (ASSOC) to the node (left-side node in Fig. 3.7) which received AD packet. Conversely, the node (right-side node in Fig. 3.7) which is not highly correlated with the data of CH cannot get be associated with CH.

In summary, L_k CHs are elected from the pool of L_{k-1} CHs formed in the previous iteration. The L_k CH then advertises for data from the L_{k-1} CHs. After a sampling period (T^{SAMPLE}), which depends on the previous sampling rate, they compute MEAN_OF_ERROR of the data from L_{k-1} CHs at different distances and time. Association packet is then sent to the L_{k-1} CHs that fall within the error threshold. On the reception of an associativity packet, each node updates its ch_list which contains associated nodes. If a L_{k-1} CH does not receive an advertisement for data or association details then it promotes itself to be a L_k CH by incrementing the level counter *count*. This process repeats itself to form a hierarchy of clusters until the error threshold is not satisfied.

Algorithm 1 Clustering Algorithm - Clusters nodes by computing MEAN_OF_ERROR

INIT_ CLUSTER_ ALGO:

1: $T^{SAMPLE} = 2 \cdot F_{MAX} + 1$
2: $t_o =$ TIME_ OUT(L_1)

CLUSTER_ ALGO:

1: **while** (1) **do**
2: **if** (RUN_ CLUSTER_ ALGO) **then**
3: TX_ AD_ PKT$(L_k - 1)$
4: *wait* T^{SAMPLE}
5: $e_h =$ COMPUTE_ MEAN_OF_ERROR$(L_k)+ margin$
6: **if** e_h **then**
7: TX_ ASSOC_ PKT$(L_k - 1)$
8: **end if**
9: *wait* t_o
10: **if** $L_k = 1$ **then**
11: $T^{SAMPLE} =$ CALC_ REF()
12: START_ RESIZE_ ALGO $= 1$
13: **end if**
14: RUN_ CLUSTER_ ALGO $= 0$
15: **end if**
16: **end while**

The algorithm for *resizing* the clusters based on the temporal correlations among

nodes runs as a separate background process. Cluster resizing algorithm plays an important role in maintaining memberships in cluster. The L_1 CHs periodically request for measured data from its members and compute the mean of error of the data proposed in Sect. 3.4.1. If the mean error reflect an anomalous variation in sensor data with respect to the space and time, then it propagates this information to all levels of hierarchy in the sensor network to achieve effective cluster resizing.

The mean of error (3.2) is calculated in time during every cycle of four-way handshake protocol during T^{SAMPLE} in Fig. 3.7. Each L_0 CHs which get AD packet from L_1 CHs periodically samples the field with *sampling period* t_s in Fig. 3.7. Once this calculation (3.2) faced on abnormal variation in sensor data, it propagate cluster resizing message.

In summary, each L_k CH (where k varies from 2 to K) requests its L_{k+1} CH to direct all its L_k members to recompute and decide on their new cluster members. The request to L_{k+1} CH is made by broadcasting a RESIZE message through TX_RESIZE_PKT(L_k+1). The L_{k+1} send resize message to the L_{k-1} member nodes on its ch_list. Also it directs all its L_{k-1} members to recompute the mean of error by sending a RESIZE message through TX_RESIZE_PKT($L_k - 1$). On reception of this message, the L_{k-1} CHs set the RUN_CLUSTER_ALGO flag that starts off the CLUSTER_ALGO. If the L_{k+1} CH does not exist, then the L_k directly requests all its counterparts to recompute mean of error by sending the RESIZE message through TX_RESIZE_PKT(L_k) that sets the RUN_CLUSTER_ALGO flag resulting in recomputation of mean of error.

3.4.4 Spatio-temporal Adaptive Sampling with Multiple Manifestations

As we introduced in Sect. 3.1, we expect to have a more accurate reconstruction of the phenomenon by using multiple manifestations than by using single manifestation. Based on the criteria (3.1) (3.2), spatio-temporal adaptive sampling with multiple manifestation is stated as following:

$$\psi_{m,s\in[1,S]} = [\psi^1_{m,s}; \psi^2_{m,s}; \psi^3_{m,s}; \ldots \psi^i_{m,s} \ldots; \psi^{N_s}_{m,s}] \tag{3.3}$$

$$e_s^{n,m} = \frac{\sum_{i=1}^{N_s} |\psi_{n,s}^i - \psi_{m,s}^i|}{N_s} \tag{3.4}$$

$$e_s^n(h) = \frac{\sum_{m=1,m\neq n}^{N(h)} \sum_{i=1}^{N_s} |\psi_{n,s}^i - \psi_{m,s}^i|}{N(h) \cdot N_s} \quad where \quad d_{n\forall m} \leq h \tag{3.5}$$

$$e^n(h) = \frac{\sum_{s=1}^{S} w_s \cdot e_s^n(h)}{\sum_{s=1}^{S} w_s} \tag{3.6}$$

- $N_s = \lfloor \frac{T^{SAMPLE}}{T_S^s} \rfloor$.
- $T^{SAMPLE} \gg T_s^S$.

Specifically,

- $\psi_{m,s}$ vector of manifestation(s) with discrete time of node m;
- $e_s^{n,m}$ error of different manifestations of a node(m) with a node (m) when time t_i;
- $e_s^n(h)$ error of different manifestations of nodes within a range(h) from m;
- $e^n(h)$ error of manifestations with weight value(W);
- m a generic node;
- n a clusterhead node;
- s a manifestation;
- $\psi_{m,s}^i$ s^{th} manifestation of node m of time t_i.
- S number of different manifestations of Phenomenon ψ;
- h advertisement range(distance);
- T^{SAMPLE} period of starting sampling;
- T_s^S sampling rate for manifestation(s);
- N_s number of sample during the period of T^{SAMPLE};
- w_s weigh on a manifestation(s);

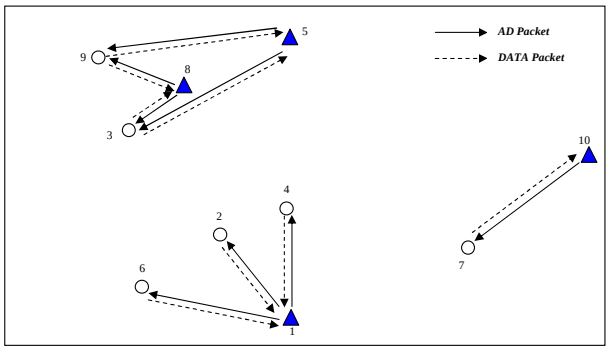
- d_{nm} distance between nodes n and m;

We consider a list of manifestations with discrete time(3.3) for a temporal correlation, and calculate the mean of errors of each manifestation by calculating difference between cluster head node(n) and each adjacent node(3.4). In the same way, mean of errors of manifestations of nodes within certain distance(h) can be calculated in (3.5), and the spatial correlation can be implemented. Eventually, equation (3.5) states overall spatial-temporal correlations of different manifestations using degree of error $e^n(h)$.

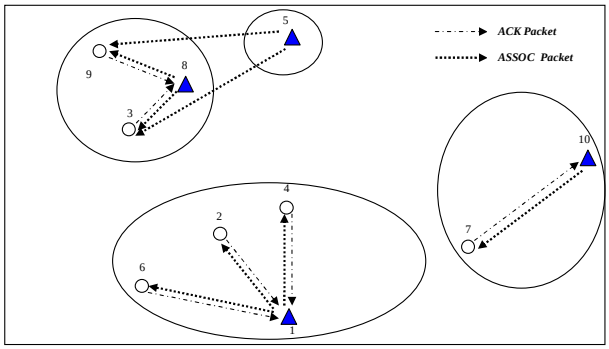
3.4.5 Illustrative Example

The following example would help understand the algorithms explained in the previous section. Let the number of sensors in the field be $N = 10$. Let the observed phenomenon be temperature. Fig. 3.8(a) shows the L_0 CH i in the field and the temperature value observed by it $\psi(\mathbf{p_i})$.

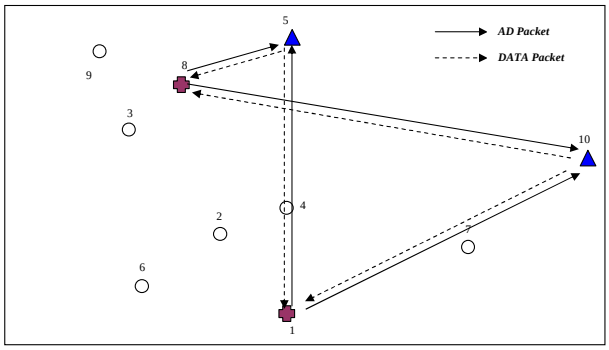
All the ten L_0 CHs simultaneously participate in the RAND_ GEN algorithm and only the nodes 1, 5, 8 and 10 get elected as L_1 CHs. The newly elected L_1 CHs send AD packets to the L_0 CHs within its transmission range as shown in Fig. 3.8(a). Here, node 1 sends AD packets to nodes 2, 4, 6 and similarly nodes 5 and 8 send to both 3 and 9 and node 10 sends to node 7. The L_0 CHs that receive the AD, then respond to the corresponding advertiser with its DATA. The L_1 CHs wait for a time t_o that depends on the worst case communication delay between L_1 and L_0 CHs. The L_1 CHs then compute the mean of error(3.5) with the received data using equation 3.5 and send the ASSOC packet to nodes within distance d_{th}) that satisfy the error threshold as shown in Fig. 3.8(b). In our example, node 1 sends ASSOC packets to 2, 4, 6; nodes 5 and 8 send to both 3 and 9 and node 10 sends to node 7. Since L_0 CHs acknowledge only the first associativity request, nodes 3 and 9 accept the request from node 8 only. Nodes 1 and 10 receive ACK from all the nodes that they sent an ASSOC request to. At this stage the first level of clustering is complete in the sensor field (shown in Fig. 3.8(b)) and this triggers the RESIZE_ ALGO.



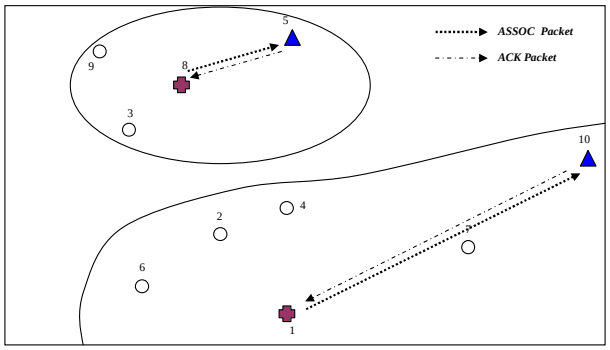
(a)



(b)



(c)



(d)

Figure 3.8: (a) Sensor field with L_1 CHs sending AD and L_0 CHs sending DATA. (b) Sensor field with L_1 CHs sending ASSOC and L_0 CHs sending ACK. (c) Sensor field with L_2 CHs sending AD and L_1 CHs sending DATA. (d) Sensor field with L_2 CHs sending ASSOC and L_1 CHs sending ACK.

Similar to the first stage, all the L_1 CHs participate in the RAND_ GEN algorithm and only the nodes 1 and 8 get elected as L_2 CHs. The newly elected L_2 CHs send AD packets to the L_1 CHs (shown in Fig. 3.8(c))made possible by multi-hop communication. Here, node 1 and 8 send AD packets to both 5 and 10. The L_1 CHs that receive the AD, then respond to the corresponding advertiser with its DATA. The L_2 CHs wait for a time t_o that depends on the worst case communication delay between L_2 and L_1 CHs. The L_2 CHs then compute the mean of error(3.5) with the received data using equation 3.5 and send the ASSOC packet to nodes within distance d_{th}) that satisfy the mean of error(3.5) threshold. In our example, node 1 sends ASSOC packet to 10 and node 8 sends to node 5 who acknowledge the associativity request as shown in Fig. 3.8(d). At this stage the second level of clustering is complete in the sensor field.

Nodes 1 and 8 participate in RAND_ GEN to select the L_3 CHs. One of the three following scenarios might happen: both the nodes get elected as L_3 CHs; one of the nodes get elected as a L_3 CH; neither of them get elected. In the first scenario, since no L_2 CHs are available to respond to AD, the spatial algorithm times out (i.e., converges) and the maximum level in the field is reset to L_2. In the second scenario, the mean of error(3.5) threshold will not be satisfied and this terminates the spatial algorithm resetting the maximum level in the field to L_2. In the third scenario, the L_2 CHs wait for the AD from L_3 CHs and time out resulting in the termination of spatial algorithm with L_2 as the maximum level of clusters.

The RESIZE_ ALGO runs as a separate background process triggered immediately after the L_1 CHs are formed. Node 1 periodically collects data every t_c seconds from its cluster members and computes the mean and variance of the data. The other L_1 CHs perform the same function. Whenever a L_1 CH observes abnormal variation between consecutive means and/or variances, it propagates the information to the various levels of clusters so that cluster resizing can be performed. Assume there is a sudden variation in temperature in node 10. Node 10, the corresponding L_1 CH detects the change in its cluster behavior and directs its L_0 CH members to participate in the RAND_ GEN algorithm to elect new L_1 CHs by sending an ALARM packet. Here 7 and 10 participate in the RAND_ GEN and both get elected as L_1 CHs. The newly formed L_1 CHs

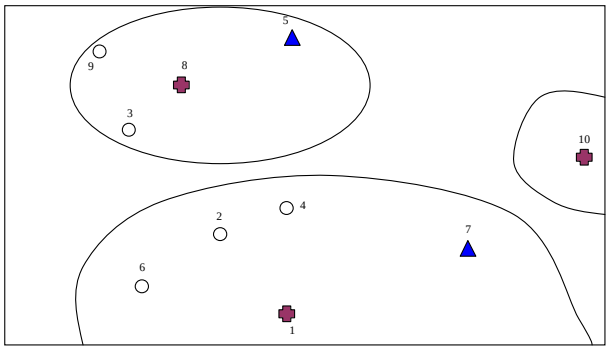


Figure 3.9: Sensor field after resizing

report to their corresponding L_2 CH by sending a RESIZE packet. Node 1, the L_2 CH in our example receives the resize packet and directs all its L_1 CH members to collect data recompute their mean of error(3.5). Since there is no L_3, the L_2 CH passes the RESIZE message on to its peers and directs them to recompute mean of error(3.5). During the recomputation at the second level, node 7 (now a L_1 CH) is accepted as a cluster member by node 1. Node 10, currently not part of any L_2 elevates itself to be a L_2 CH after a particular timeout value. The sensor field after resizing is shown in Fig. 3.9

Thus, the above example clearly illustrates both the clustering and resizing algorithms and how we are able to ensure joint optimization by leveraging the spatio-temporal characteristics of the phenomenon.

3.5 Performance Evaluation

We implemented the adaptive sampling for two and three dimensional space described in Sect. 3.4.3. Fig. 3.10 shows sensors deployed in 3-dimensional space with four types of different data. Simulations were done using TOSSIM 2.x, a TinyOS simulator. TOSSIM 2.x currently supports only MicaZ radio components (IEEE 802.15.4). Due to the constraint that we do not have a real data with multiple manifestations for a phenomenon, we did preliminary experiment based on the temperature. The radio propagation model described in [18] was used for our simulation. In addition to the

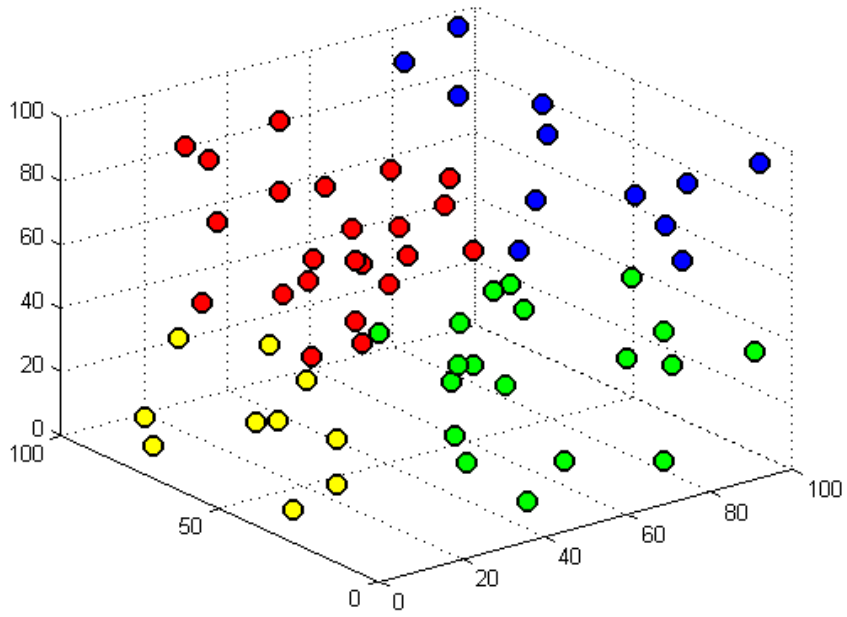


Figure 3.10: Sensor deployment in three dimensional area

radio propagation model, TOSSIM also simulates the RF noise and interference with other nodes. TOSSIM uses Closest Pattern Matching (CPM) algorithm [19], which takes a noise trace as input and generates a statistical model from it.

We are interested in scenario where a squared space is divided by four sections and a cube space divided by eight sections of each corner. Sensors in a section of the square or the cube have random temperatures which are different but highly correlated each other. Temperatures in different sections, however, are less correlated, more than the error *e(h)* stated in Sect. 3.4.4. In this case adaptive sampling algorithm should be converged as four or eight representative CHs after some time elapsed because they are spatially correlated. Converged CHs will represent their region. Fig. 3.11 shows how two dimensional voronoi place is converged in four sections as an example. When time elapse, the nodes are converged one by one until the CHs are converged as four. Also, Fig. 3.12 shows measures of convergence time based on CH random-election percentage[%]. This graph is not optimized but shows the tendency of relation between the convergence time and CH random-election percentage. This graph shows that the more CH elected in the network, the faster the network can be converged. However,

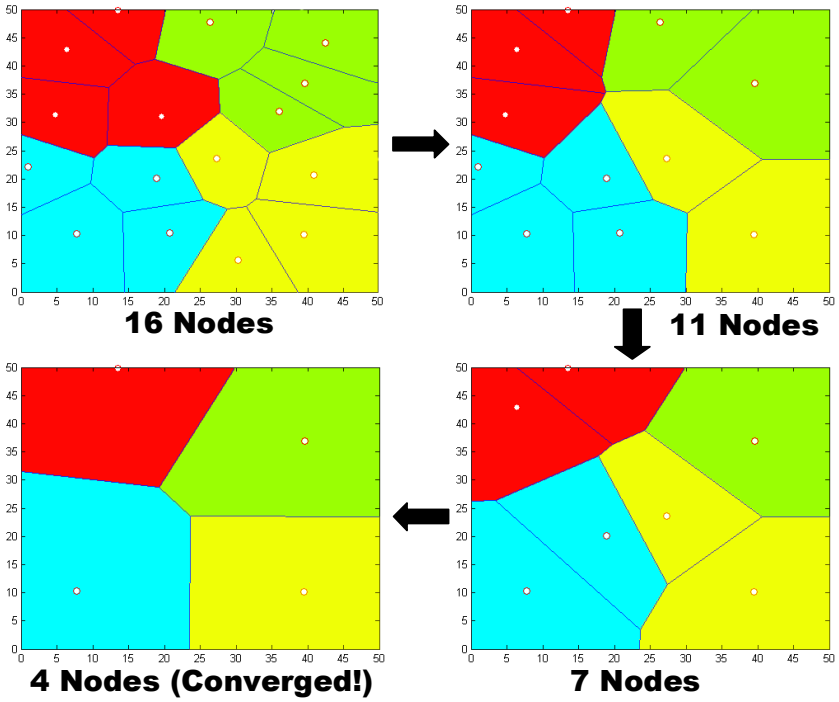


Figure 3.11: Steps for convergence

this infers that because of the communication cost to broadcast CH election, energy consumption will increase. This optimization issues can be solved as a future work.

Finally, a phenomenon can be reconstructed with around 75% ± 10% accuracy in voronoi region with less energy consumption in Fig. 3.3. We measured the area of reconstructed phenomenon in voronoi space and compared it with the real area of the phenomenon. For energy consumption in Fig. 3.3, stepwise graphs shows periodic energy drain of the network because of the data exchanges to form and reforming CH. Regular sampling method shows significant energy consumption in time because generally all the node report their data to the sink node, but for adaptive sampling, less energy is consumed because only CH send data to the sink. Communication cost is much less than regular sampling method for adaptive sampling.

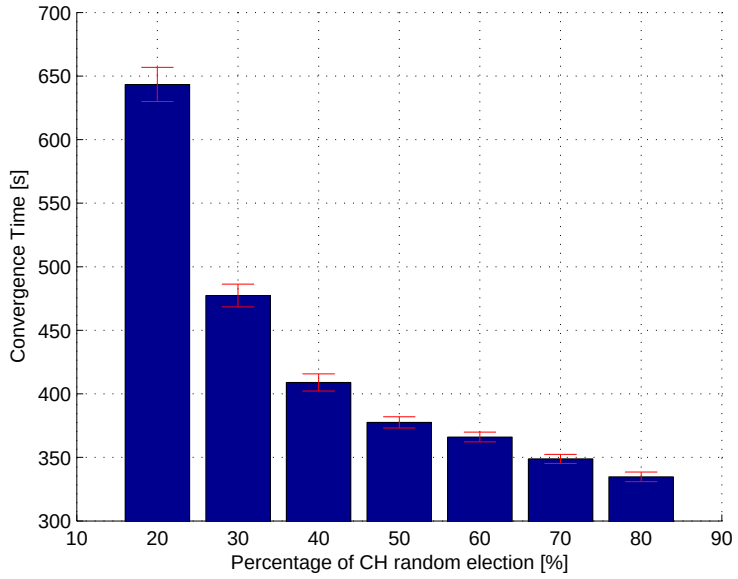


Figure 3.12: Convergence time measurement

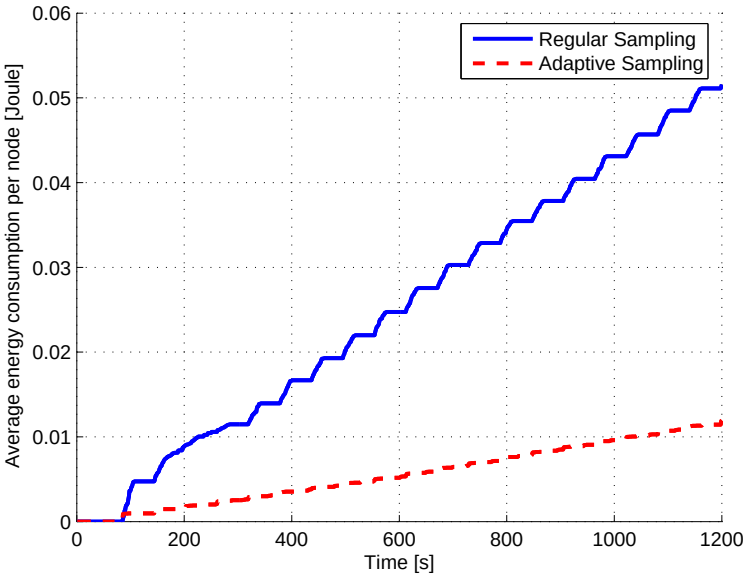


Figure 3.13: Energy Level of the network

3.6 Conclusions

Energy efficiency is mainly achieved in WSNs through minimization of communication cost by applying adaptive sampling and data aggregation with Clustering. Moreover, we proposed a phenomenon can be captured more reliably by using multiple manifestations than by using a single manifestation.

Chapter 4

Conclusion and Future Works

Location-based routing is a routing scheme that uses geographical positions in order to sense data and route packets to the destination. It is widely utilized in WSNs because i) sensor locations are needed to spatially reconstruct at the sink the phenomenon being sensed by the WSN and ii) it is highly scalable. Through the research in Chapter 2, we clarified that NDP heavily impact on the performance of the routing scheme itself, which in turn could affect end-to-end performance because of the overhead caused by the periodic broadcasts of NDP. In particular, through theoretical analysis, simulations, and test-bed experiments we answered the following three issues: "How far should a node know about its neighbors?", "How often should a node update the neighborhood information?", and "How much information should a node require?". Analytical and experimental study were conducted to determine how NDP parameters such as power, frequency, and packet size affect the communication process. Specifically, we investigated both static and mobile environments, as mobility of sensor nodes can drastically affect the routing decisions and hence overall performance. These scenarios are thoroughly tested on Tiny-OS SIMulator (TOSSIM), and real experiments will be performed on TelosB motes. Preliminary results showed that maximum NDP power, maximum frequency or more information for neighborhood discovery in WSNs do not guarantee optimal end-to-end performance. Conversely, there are optimal values for Tx-power, frequency, and packet size of NDP, which depend on node velocity and density of the network. We also proposed a protocol that aims at dynamically adjusting these parameters real time.

Based on the understanding of NDP and its impact, we conducted a research about adaptive sampling which dynamically aggregate data in space and time. Due to the

high density of sensor nodes, the degree of "similarity" among spatially proximal sensor observations increases as the inter-node distance decreases. Also, the degree of "similarity" between consecutive sensor measurements varies according to the temporal characteristics of the phenomenon's manifestation. The former is termed as spatial correlation while the latter is referred to as temporal correlation. In this research effort, adaptive distributed field estimation techniques were developed to prolong the lifetime of 3D WSNs and/or to ensure effective utilization of resources, whereas existing solutions perform spatial and temporal adaptive sampling only separately. By grouping sensors in clusters and electing cluster heads (CHs) that will report the data on behalf of the nodes in their clusters, communication cost can be minimized. This overhead reduction occurs because cluster heads do not need to collect data from all the nodes in the cluster as it is normally proposed in the literature when in-network processing is advocated. Obviously, efficient distributed clustering and resizing algorithms need to be developed so to assure data consistency as the phenomenon evolves in time and space. Furthermore, because nodes are usually provided with many sensors on their sensing board (e.g., light, temperature, humidity, accelerometer, etc.), in-network joint optimization was achieved by adaptively modifying both spatial and temporal sampling rates of these different sensors in such a way as to dynamically track the time and space evolution of the phenomenon's manifestations.

In future, performance evaluation of the proposed adaptive sampling with multiple manifestation using real data needs to be carried out. Also, threshold optimization for clustering can be determined based on application requirement. How to optimize weigh value in(3.5) could be one of the future work when multiple manifestations are measured. Moreover, CH election percentage(%) should be optimized according to the network characteristics.

References

[1] Jennifer Yick, Biswanath Mukherjee, and Dipak Ghosal. Wireless sensor network survey. *Computer Networks*, 52:2292–2330, April 2008.

[2] Tommaso Melodia, Dario Pompili, and F. Akyildiz. On the Interdependence of Distributed Topology Control and Geographical Routing in Ad Hoc and Sensor Networks. *IEEE Journal on Selected Areas in Commun.(JSAC)*, 23(3):520–532, March 2005.

[3] I. Stojmenovic. Position-based routing in ad hoc networks. *IEEE Commun. Mag.*, 40(7):128–134, July 2002.

[4] H. Takagi and L. Kleinrock. Optimal Transmission Ranges for Randomly Distributed Packet Radio Terminals. *IEEE Trans. Commun.*, 32(3):246–57, March 1984.

[5] Evangelos Kranakis, Harvinder Singh, and Jorge Urrutia. Compass Routing on Geometric Networks. In *Proc. of Canadian Conference on Computational Geometry (CCCG)*, Vancouver, BC, August 1999.

[6] Jennifer Yick, Biswanath Mukherjee, and Dipak Ghosal. Wireless sensor network survey. *Computer Networks*, 52(12):2292–2330, August 2008.

[7] Ting-Chao Hou and Victor Li. Transmission Range Control in Multihop Packet Radio Networks. *IEEE Trans. Commun.*, 34(1):38–44, January 1986.

[8] Gregory G Finn. Routing and Addressing Problems in Large Metropolitanscale Internetworks. In *ISI Res. Rep. ISU/RR-87-180*, March 1987.

[9] K. Seada, M. Zuniga, A. Helmy, and B. Krishnamachari. Energy-efficient Forwarding Strategies for Geographic Routing in Lossy Wireless Sensor Networks. In *Proc. of ACM Conference on Embedded Networked Sensor Systems (SenSys)*, Baltimore, MD, November 2004.

[10] Douglas S. J. DeCouto, Daniel Aguayo, John Bicket, and Robert Morris. A High-Throughput Path Metric for MultiHop Wireless Routing. In *Proc. of the ACM International Conference on Mobile Computing and Networking (MobiCom)*, San Diego, CA, September 2003.

[11] Alec Woo, Terence Tong, and David Culler. Taming the Underlying Challenges of Reliable Multihop Routing in Sensor Networks. In *Proc. of the International Conference on Embedded Networked Sensor Systems (SenSys)*, Los Angeles, CA, November 2003.

[12] M. Zuniga and B. Krishnamachari. Analyzing the transitional region in low power wireless links. In *Proc. of Sensor and Ad Hoc Communications and Networks(SECON)*, Santa Clara, CA, October 2004.

[13] Kannan Srinivasan and Philip Levis. RSSI is Under Appreciated. In *Proc. of the Third Workshop on Embedded Networked Sensors (EmNets)*, Cambridge, MA, May 2006.

[14] Jerry Zhao and Ramesh Govindan. Understanding packet delivery performance in dense wireless sensor networks. In *'03: Proceedings of the 1st international conference on Embedded networked sensor systems (SenSys)*, pages 1–13, New York, NY, USA, 2003. ACM.

[15] S. M. Woo, S. Singh, and C. S. Raghavendra. Power-aware Routing in Mobile Ad Hoc Networks. In *Proc. of ACM/IEEE International Conference on Mobile Computing and Networking(MobiCom)*, Dallas, TX, July 1998.

[16] M. Chiang. Balancing Transport and Physical Layers in Wireless Multihop Networks: Jointly Optimal Congestion Control and Power Control. *IEEE Journal on Selected Areas in Commun.(JSAC)*, 23(1):104–116, January 2005.

[17] Dario Pompili, Tommaso Melodia, and Ian F. Akyildiz. Routing Algorithms for Delay-insensitive and Delay-sensitive Applications in Underwater Sensor Networks. In *Proc. of International Conference on Mobile Computing and Networking (MobiCom)*, Los Angeles, CA, September 2006.

[18] Philip Levis, Nelson Lee, Matt Welsh, and David Culler. Tossim: accurate and scalable simulation of entire tinyos applications. In *Proc. of the 1st international conference on Embedded networked sensor systems (SenSys)*, pages 126–137, Los Angeles, CA, 2003.

[19] HyungJune Lee, Alberto Cerpa, and Philip Levis. Improving wireless simulation through noise modeling. In *Proc. of the 6th international conference on Information processing in sensor networks (IPSN)*, pages 21–30, Cambridge, MA, 2007.

[20] Apoorva Jindal and Konstantinos Psounis. Modeling Spatially Correlated Data in Sensor Networks. *ACM Transactions on Sensor Metworks (TOSN)*, 2(4):466–499, November 2006.

[21] Mehmet C. Vuran, Ozgur B. Akan, and Ian F. Akyildiz. Spatio-temporal Correlation: Theory and Applications for Wireless Sensor Networks. *Computer Networks (Elsevier Science)*, 45(3):245–259, June 2004.

[22] Ian F. Akyildizy, Mehmet C. Vuran, Ozgur B. Akan, and B. Akanz Weilian Su. Wireless Sensor Networks: A Survey Revisited. *Computer Networks (Elsevier Science)*, 2006.

[23] Rebecca Willett, Aline Martin, and Robert Nowak. Backcasting: Adaptive Sampling for Sensor Networks. In *Proc. of the International Conference on Information Processing in Sensor Networks (IPSN)*, Berkeley, CA, April 2004.

[24] Ankur Jain and Edward.Y.Chang. Adaptive Sampling for Sensor Networks. In *Proc. of the International Workshop on Data Management for Sensor Networks (DMSN)*, Toronto, Canada, August 2004.

[25] Paritosh Padhy, Rajdeep K. Dash, Kirk Martinez, and Nicholas R. Jennings. A Utility-based Sensing and Communication Model for a Glacial Sensor Network. In *Proc. of the International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, Hakodate, Japan, May 2006.

[26] Robert Nowak, Urbashi Mitra, and Rebecca Willett. Estimating Inhomogeneous Fields using Wireless Sensor Networks. *IEEE Journal on Selected Areas in Communications (JSAC)*, 25(7):999–1006, August 2004.

[27] Mark Perillo, Zeljko Ignjatovic, and Wendi Heinzelman. An Energy Conservation Method For Wireless Sensor Networks Employing a Blue Noise Spatial Sampling Technique. In *Proc. of the International Conference on Information Processing in Sensor Networks (IPSN)*, Berkeley, CA, April 2004.

[28] Johnsen Kho, Alex Rogers, and Nicholas R. Jennings. Decentralised Adaptive Sampling of Wireless Sensor Networks. In *Proc. of the International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, Honolulu, Hawaii, June 2007.

[29] Zhi Quan, William J. Kaiser, and Ali H. Sayed. A Spatial Sampling Scheme Based on Innovations Diffusion in Sensor Networks. In *Proc. of the International Conference on Information Processing in Sensor Networks (IPSN)*, Cambridge, MA, April 2007.

[30] Zhi Quan and Ali H. Sayed. Innovations-based Sampling over Spatially Correlated Sensors. In *Proc. of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Honolulu, Hawaii, April 2007.

[31] G. Mainland, D. C. Parkes, and M. Welsh. Decentralized, Adaptive Resource Allocation for Sensor Networks. In *Proc. of the USENIX/ACM Symposium on Networked Systems Design and Implementation (NSDI)*, Boston, MA, May 2005.

[32] S. Goel and T. Imielinski. Prediction-based Monitoring in Sensor Networks: Taking Lessons from MPEG. *ACM Computer Communication Review (CCR)*, 31(5):82–98, October 2001.

[33] Amol Deshpande, Carlos Guestrin, Samuel Madden, Joseph Hellerstein, and Wei Hong. Model-driven Data Acquisition in Sensor Networks. In *Proc. of the Very Large Data Base (VLDB) Conference*, Toronto, Canada, August 2004.

[34] Seema Bandyopadhyay, Qingjiang Tian, and Edward J. Coyle. Spatio-Temporal Sampling Rates and Energy Efficiency in Wireless Sensor Networks. *IEEE/ACM Transactions on Networking (TON)*, 13(6):1339–1351, December 2005.