A label based routing protocol for wireless sensor networks

Naoto Kimura

University of Nevada, Las Vegas

Follow this and additional works at: https://digitalscholarship.unlv.edu/rtds

Repository Citation

This Thesis is brought to you for free and open access by Digital Scholarship@UNLV. It has been accepted for inclusion in UNLV Retrospective Theses & Dissertations by an authorized administrator of Digital Scholarship@UNLV. For more information, please contact digitalscholarship@unlv.edu.
A LABEL BASED ROUTING PROTOCOL
FOR WIRELESS SENSOR NETWORKS

by

Naoto Kimura

Bachelor of Science in Chemical Engineering
University of Washington
1995

Bachelor of Science in Engineering
University of Nevada, Las Vegas
2003

A thesis submitted in partial fulfillment
of the requirements for the

Master of Science Degree in Engineering
Department of Electrical and Computer Engineering
Howard R. Hughes College of Engineering

Graduate College
University of Nevada, Las Vegas
August 2005
INFORMATION TO USERS

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleed-through, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

UMI Microform 1429709
Copyright 2006 by ProQuest Information and Learning Company. All rights reserved. This microform edition is protected against unauthorized copying under Title 17, United States Code.

ProQuest Information and Learning Company
300 North Zeeb Road
P.O. Box 1346
Ann Arbor, MI 48106-1346

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
Thesis Approval
The Graduate College
University of Nevada, Las Vegas

July 18, 2005

The Thesis prepared by

Naoto Kimura

Entitled

"A Label Based Routing Protocol for Wireless Sensor Networks"

is approved in partial fulfillment of the requirements for the degree of

Master of Science in Electrical Engineering

[Signatures]

Examination Committee Chair

Dean of the Graduate College

[Signatures]

Examination Committee Member

Examination Committee Member

Graduate College Faculty Representative
ABSTRACT

A Label Based Routing Protocol for Wireless Sensor Networks

by

Naoto Kimura

Dr. Shahram Latifi, Examination Committee Chair
Professor of Electrical and Computer Engineering
University of Nevada, Las Vegas

One of the challenging issues in wireless sensor networks is to acquire, process, and transmit information using the least amount of battery power.

In response to this problem, a novel routing scheme for wireless sensor networks, called "Information Dissemination via Label Forwarding (IDLF)", is presented. IDLF consists of three information exchanging stages. In the first stage, a label is disseminated by a source. After a sink receives the label, it replies a request to the source. By exchanging a label and request, a data path between a source and sink is formed. Finally, an actual data is transmitted through the data path. Transmitting labels and requests, instead of actual data, reduces the redundant transmissions of data packets, and thus achieves energy savings.

In addition to IDLF, four energy management schemes – directional forwarding, minimum transmission around the sink, battery threshold value, and differential coding – are proposed. We compare IDLF with flooding and another wireless routing protocol named SPIN. The simulation results show that the IDLF together with four energy
management schemes can save a significant amount of energy compared to the other two routings.
TABLE OF CONTENTS

ABSTRACT .................................................................................................................................. iii

LIST OF FIGURES .................................................................................................................... vii

ACKNOWLEDGMENTS ........................................................................................................ viii

CHAPTER 1  INTRODUCTION .......................................................................................... 1

CHAPTER 2  CONVENTIONAL ROUTING AND DATA COMPRESSION
SCHEMES IN SENSOR NETWORKS ....................................................... 8
  2.1 Flooding ......................................................................................................................... 8
  2.2 Sensor Protocols for Information via Negotiation (SPIN) ..................................... 10
  2.3 Directed Diffusion ..................................................................................................... 12
  2.4 Power Aware Routing Schemes ................................................................................ 13
    2.4.1 Minimum Total Transmission Power Routing ......................................... 14
    2.4.2 Min-Max Battery Cost Routing .................................................................. 16
    2.4.3 Conditional Max-Min Battery Capacity Routing ..................................... 17
    2.4.4 Modified Conditional Max-Min Battery Capacity Routing .................... 18
  2.5 Data Compression Schemes for Wireless Sensor Networks................................. 19
    2.5.1 Coding by Ordering ..................................................................................... 20
    2.5.2 Pipelined In-Network Compression ........................................................... 22

CHAPTER 3  INFORMATION DISSEMINATION via LABEL FORWARDING ... 24
  3.1 IDLF Routing Scheme .............................................................................................. 24
  3.2 Label Dissemination Method ................................................................................... 29
  3.3 Power Management in Sensor Networks ................................................................. 30
    3.3.1 Minimum Transmission around the Sink .................................................. 30
    3.3.2 Battery Threshold Value ............................................................................. 30
    3.3.3 Differential Coding ...................................................................................... 32

CHAPTER 4  SENSOR NETWORK PERFORMANCE EVALUATION ............... 34
  4.1 Network Simulation Model ....................................................................................... 34
  4.2 Routing Performance Evaluation ............................................................................. 36
    4.2.1 Energy Consumption over Time ................................................................37
    4.2.2 Data Transmission over Time ....................................................................40
    4.2.3 Energy Consumption by a Data Packet ....................................................43
  4.3 Power Management Analysis ................................................................................... 44
    4.3.1 Minimum Transmission around the Sink .................................................. 45
    4.3.2 Battery Threshold Value ............................................................................. 46

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
LIST OF FIGURES

Figure 1.1 Wireless Sensor Network Configuration ............................................................. 2
Figure 1.2 Sensor Node Components ...................................................................................... 3
Figure 2.1 Data Propagation Steps for SPIN ........................................................................ 11
Figure 2.2 Data Propagation in Directed Diffusion ............................................................ 12
Figure 2.3 Example of In-network Compression .................................................................. 23
Figure 3.1 Example of Label Information .......................................................................... 25
Figure 3.2 Label Transmission Process .............................................................................. 27
Figure 3.3 Label and Request Transmission ....................................................................... 28
Figure 3.4 Example of Directional Forwarding .................................................................. 29
Figure 3.1 Example of Nodes Allocation ........................................................................... 35
Figure 4.1 Energy Consumed by Entire Network over Time: (a) 5 nodes, (b) 10 nodes, (c) 15 nodes, (d) 20 nodes, (e) 25 nodes, (f) 30 nodes ............................................ 38
Figure 4.3 Number of Data Packets Delivered over Time: (a) 5 nodes, (b) 10 nodes, (c) 15 nodes, (d) 20 nodes, (e) 25 nodes, (f) 30 nodes ................................................... 41
Figure 4.4 Energy Consumption by Each Data Packet ....................................................... 44
Figure 4.5 Average Node Energy Consumption by Hop Count ........................................ 45
Figure 4.6 Number of Data Packets Delivered: (a) 600 unit energy, (b) 700 unit energy, (c) 800 unit energy, (d) 900 unit energy ...................................................... 48
Figure 4.7 Total Energy Consumption by a network: (a) 600 unit energy, (b) 700 unit energy, (c) 800 unit energy, (d) 900 unit energy .............................. 50
Figure 4.8 Percentage of Information Packets Dropped: (a) 600 unit energy, (b) 700 unit energy, (c) 800 unit energy, (d) 900 unit energy .......................... 52
Figure 4.9 Total Energy Consumption by Data Transmission ......................................... 55
Figure 4.10 Energy Saved for 100 Data Packets ............................................................... 56
ACKNOWLEDGMENTS

I would like to express my deepest gratitude and appreciation for my advisor Dr. Shahram Latifi for the extraordinary support, guidance, and encouragement that he has shown me. I would also like to thank Dr. Emma Regentova, Dr. Muthukumar Venkatesan, and Dr. Kazem Taghva for serving on my committee and providing valuable feedbacks. Especially, I would like to express my great appreciation to Dr. Taghva for becoming my committee member in such a short notice. I would like to thank Vasu Jolly. Without his help and encouragement, I could not finish this work. Finally, I would like to thank my family for everything they have done for me.
CHAPTER 1

INTRODUCTION

Increasingly smaller and faster semiconductor fabrication technology has fueled an information technology rumble. Fabricating cheaper and more powerful computing devices have boosted virtually every facet of our economy. One of the fields, which have been benefited from the technological advancement, is the field of wireless sensor networks. In recent years, a substantial research effort has been offered in the field of wireless sensor networks, in both academia and industry igniting the thrust to realize its unlimited applicability in various fields. Environmental and habitat monitoring is one of such wireless sensor applications [2]. Sensors can monitor temperature, humidity, and barometric pressure of certain areas. Development of wireless sensor networks for military purpose is also one of the most active areas [7]. Related studies include detecting radiation from nuclear attacks by terrorists [9], or detecting enemy movement in a battle field. Other applications for wireless sensor networks are structural monitoring, equipment diagnoses, disaster management, and traffic control [10]. However, the technology of wireless sensor networks is still relatively premature, and the massive research and implementation of wireless sensor networks are advancing at a rapid pace. Wired sensor networks have been deployed broadly for decades, with a range of estimations measuring temperature, pressure, humidity, seismic wave, and noise levels, etc. In a conventional set up, the sensor units are connected to centralized equipment,
which processes the collected data and controls behavior of the entire network. For example, if the pressure increases or the noise level changes, the event data is passed on to the centralized equipment, which in turn processes the data and determines the subsequent action. Therefore, the sensor units in wired sensor networks are often equipped with only sensing capability.

![Figure 1.1: Wireless Sensor Network Configuration](image)

Wireless sensor networks are expected to revolutionize information gathering, processing and dissemination in many diverse environments. The basic wireless sensor network configuration is shown in Figure 1.1 [1]. Sensor nodes are densely deployed over a desired application area, called a sensor field. The sensor nodes are connected by radio frequency, infrared, or other medium without any physical wire connection. On the contrary to wired sensor networks, wireless sensor networks are distributed networks. Upon detecting an event, a sensor node starts collecting and processing the event data and the event data collected by a sensor node traverses among other nodes in a wireless
medium. The data from sensor nodes is gathered by a sink node. There may be multiple sinks in one wireless sensor network. The sink usually is robust in terms of processing speed, memory size, and battery capacity compared to all other sensor nodes in the network. The sink can be connected to the outside world through Internet or satellite so that a user can access the collected data.

Wireless sensor networks are made of nodes that have a processor, memory, wireless transceivers, sensor(s), location finding unit, and an onboard battery. Figure 1.2 [1] shows the typical components of sensor node. Expected size of each sensor node is approximately $1 \text{ cm}^3$ in its volume and less than 100 g in its weight. Its memory size, combining data and program memory, will be several tens of Mbytes. On the contrary, the sensor node currently available on the shelf [3] has its volume of 70 cm$^3$ with 128 KB of instruction memory, 4 KB of data RAM, and 512 KB of flash memory. The processor is operated at 4 MHz.

![Figure1.2: Sensor Node Components](image)

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
To realize wireless sensor networks, wireless ad hoc network techniques have been widely used for data dissemination [8]. An ad hoc network is one form of wireless network systems. It is a peer-to-peer network allowing direct communication between two devices (nodes). Many nodes are connected in peer-to-peer style to form an ad hoc network. If two nodes cannot communicate directly, other nodes, located between those two nodes, will transmit a data packet from the source node to the destination node. This is called multi-hop routing. Because of their peer-to-peer communication style, no centralized point to control a network formation, like a base station for a cellular system, is required for the ad hoc network. Since no fixed infrastructure is necessary for the ad hoc network to connect sensor nodes, a network will be constructed inexpensively. Also, nodes can be added to and separated from the network easily. Therefore, they offer fast and easy network deployment and are ideally suited for situations where either no supporting structure is available or can be installed. Mobile nodes in mobile ad-hoc networks play an important role in establishing communication between various devices in the network. These mobile nodes are free to move and arrange themselves in a subjective fashion. Each user is free to roam about while in communication with others. The path between each pair of the mobile devices may have multiple links. This organization allows an association of various links to be a part of the same network.

Even though currently wireless ad hoc network systems are widely used for realizing wireless sensor networks, they were not specifically designed for wireless sensor networks. Therefore, there exist decisive differences between wireless ad hoc networks and wireless sensor networks. Ad hoc network systems have been developed for wireless mobile hosts, such as laptop computer, to form a network temporally without any
centralized point, so these hosts are usually equipped with a more powerful CPU, larger memory, more battery power and broader communication bandwidth than sensor nodes. These equipment superiorities of ad hoc hosts make them possible to utilize relatively complicated network systems. Furthermore, since in a sensor field sensor nodes are densely deployed, the network size of sensor networks is much larger than that of ad hoc networks. The communication types of two networks are also different. An ad hoc network usually uses peer-to-peer communication style. On the other hand, in a sensor network broadcast communication style is mainly employed. Because of these differences, wireless ad hoc network systems are not suitable for wireless sensor networks and it is necessarily to design own networking system. As it was mentioned, the size of each sensor node is expected to be small. To achieve its requirement, the size of every sensor components, such as the power source and processing and data storing memory in Figure 1.1, also has to be small. In addition, a large number of sensor nodes are scattering over the sensor field, and they will be often deployed to the location, where it is hard to access. It is not practical to perform maintenance operations, such as changing batteries, on deployed sensor nodes. Because of above reasons, wireless sensor networks, or more specifically each sensor node, are resource constrained. They have limited power supply, bandwidth for communication, processing speed, and memory space. Therefore, many of studies have been conducted so far focusing on how to achieve the maximum utilization of limited sensor resource.

Among many concerns about design of sensor networks are those of growing bandwidth demands, speed of information retrieval and transporting bytes over the backbone networks to provide a quality service for the diverse requirements of the users,
such as signal processing or multimedia applications. In terms of power consumption, operation of a wireless sensor node can be divided into three parts: sensing, processing, and transmission. Among those three operations, it is known that the most power consuming task is data transmission. Approximately, 80% of power consumed in each sensor node is used for data transmission.

One field of resource utilization studies for sensor networks is routing protocols. Since a sensor network has limited bandwidth, it is necessary to minimize communication between sensor nodes. Although traditional routing protocols ignore power management issues, for sensor networks, energy efficient routing is a pivotal issue, since nodes need to be almost inconspicuous and thus cannot afford to have more than a small power pack. Another important technique for resource utilization in wireless sensor networks is data compression. Data compression reduces the power consumption due to processing and transmitting data in each node, and thus, extends the life time of an entire sensor network. Also, by reducing data size, less bandwidth is required for sending and receiving an information packet. Therefore, applying data compression along with an optimal routing protocol, which will minimize communication among sensor nodes, in a wireless sensor network is one effective method to utilize limited resources.

The core objective of this research is to design and implement effective energy-aware routing schemes. It is to be noted at this time that the primary objective of this work is to commune the data packets from a source node to a sink efficiently both in terms of energy and time. Chapter 1 discusses the background of wireless sensor networks, its architecture, applications and the potential. Chapter 2 explains the existing wireless sensor routing protocols and data compression schemes. Chapter 3 discusses our
proposed routing method. The performance analysis of our method and comparison with existing routing methods are conducted in Chapter 4. Finally, in Chapter 5 summary of our energy-aware routing scheme and future work are presented.
CHAPTER 2

CONVENTIONAL ROUTING AND DATA COMPRESSION SCHEMES IN SENSOR NETWORKS

In comparison with the wired networks, wireless sensor networks are relatively resource constraint. Because of this fact, an effective use of wireless sensor networks depends on how to manage those limited resources. Therefore, a routing scheme and data compression method for wireless sensor networks has to be simple so that it does not require much computation power and memory space, and minimize communication between nodes to save its power. Also there is no centralized node to organize an entire network, so a routing scheme for wireless sensor networks has to be self-organizing. In this chapter, we will survey and examine in detail existing routing schemes and data compression methods for wireless sensor networks.

2.1 Flooding

Flooding is a straightforward and long-standing routing scheme. In flooding, when a node receives a data packet, the node stores the data and broadcasts the data to its neighboring nodes. This process will be repeated until the data reaches all connected nodes in an entire network. To execute flooding, sensor nodes do not have to have any knowledge of network configuration. All they have to do while receiving or transmitting a data packet is to distinguish each data packet. This will save the limited memory space
of each node. Since flooding does not require any complicated routing algorithms, it can be easily implemented for sensor networks. However flooding has a few deficiencies, which will waste the limited resources of sensor nodes [4]. One of deficiencies in flooding is called implosion. Implosion is caused by a data receiving node broadcasting the data packet to its all neighboring nodes, whether a neighboring node already has the same data or not. Due to indiscriminate transmission of data, sensor nodes in flooding consume scarce transmission power and transmission bandwidth. A node will receive the same data packet repeatedly from different neighboring nodes. Other deficiency is called overlap. When multiple nodes observe the same sensor region, they will produce the overlapping data. Then, neighboring nodes receive the multiple data, which contains the same information. Similar to implosion, overlap dissipates transmission power and bandwidth. These deficiencies will shorten the battery life of each sensor node and therefore shorten the entire network life span.

Because of its simplicity, flooding has been studied intensively to reduce the deficiencies, mentioned previously, and several derivatives have been introduced. In [11] and [12], each sensor node only needs to know a small portion of entire network configuration, which is the location information of its neighboring nodes, instead of the information of entire network topology. By utilizing the location information, the implosion problem of classic flooding is avoided. Gossiping is another derivative of flooding [5]. In gossiping, when a node receives the data packet, it randomly selects a subset of its all neighboring nodes and transmits the data packet to the subset, instead of all neighboring nodes. By changing the number of neighboring nodes in the subset, energy consumed by transmission is controlled. Therefore, if the number of nodes is
reduced, transmission energy is also reduced. Furthermore, since indiscriminate transmission of data to all neighbors is not happening in gossiping, implosion can be avoided. However, the propagation delay of gossiping is long. In a densely deployed network, performing simple flooding cost a large overhead of recurring information; as many nodes in the vicinity will repeat the message, even though many other nodes have done so.

2.2 Sensor Protocols for Information via Negotiation (SPIN)

SPIN [4] is an improved version of classical flooding. It is designed to overcome the deficiencies of flooding. SPIN protocol exchanged three different types of messages (ADV, REQ, and DATA) between sensor nodes. Flooding wastes communication and power resources sending needless information throughout the network. SPIN uses data negotiation and resource-adaptive algorithms to efficiently disseminate the data by assigning a name to their data, called meta-data. Before any data is transmitted, meta-data negotiations take place. This assures that there is no redundant data sent throughout the network. Upon receiving the data packet, a receiving node transmits a small advertisement packet (ADV) to its all neighboring nodes except one, from which the node receives the data packet. The ADV contains the information of actual data. When receiving the ADV, a neighboring node checks its cache whether the node already has the same data or not. If the neighboring node already has the data, the ADV is ignored. If the neighboring node does not have the data, the node sends a request message (REQ) to the receiving node. Then, the receiving node transmits the data packet (DATA) to the neighboring nodes, which request the data by sending the REQ message. This whole
process is shown in Figure 2.1. Test results in [4] show that SPIN is more energy-efficient than flooding or gossiping while distributing data at the same rate or faster than these protocols.

![Data Propagation Steps for SPIN](image)

**Step 1:** Receiver acquiring DATA
**Step 2:** Receiver sending ADV
**Step 3:** Neighbors sending REQ
**Step 4:** Receiver sending DATA

**Figure 2.1: Data Propagation Steps for SPIN [4]**

SPIN and any other flooding based routing protocols are designed for disseminating a data packet to an entire network efficiently. All nodes in the network possess the same data packet. In some application, this is advantageous – for instance a wireless sensor network is employed for detecting and warning fire in a building. In that case, when one sensor node senses a fire, the information has to be disseminated to the entire network so that all people inside the building are noticed. On the other hand, for many applications, point-to-point data dissemination is more preferable. When measuring temperature, relative humidity, or barometric pressure, the data usually is collected at one place. In that case, disseminating data packets to the entire network wastes limited resources of sensor nodes and is not desirable.
2.3 Directed Diffusion

In [12], a data-centric routing protocol, called the directed diffusion, is introduced. In the directed diffusion, first a source broadcasts an interest, which contains various attribute values, to an entire network. Attribute values specify the sensing task, include type of sensing event, sensing area, duration of sensing task, and event transmission frequency. The interest is disseminated throughout the network in a hop-by-hop manner. Upon receiving an interest, a node stores the interest and also sets up a gradient toward the node, from which the interest is transmitted. Then, if a node has the data, which matches the received interest, the node starts sending back the data packet to the sink in multiple paths according to the gradients. When the sink receives the data packet, the best path is selected by the sink. The best path can be selected based on any criterion – i.e. shortest path or minimum energy consuming path, which will be suitable for each sensor network application. The selected path is reinforced by the sink sending a new interest to the path. Figure 2.2 shows an example of directed diffusion path set-up.

![Directed Diffusion Diagram](image)

**Figure 2.2: Data Propagation in Directed Diffusion**
One aspect, which distinguishes directed diffusion from flooding and SPIN, is data collection method. In flooding and SPIN, data collection scheme is initiated by source nodes. In other words, source nodes start transmitting data whenever data is available. On the contrary, in directed diffusion data collection is initiated by sink nodes. Because of sink initiated data collection, directed diffusion can limit data flow. By doing so, it will reduce unnecessary data transmissions and thus energy consumption of sensor nodes will be reduced.

However, there are disadvantages for employing directed diffusion in sensor networks. One of disadvantages is the possibility of transmission overhead created by interests. When a sink broadcasts an interest, the sink does not know whether the data, which will match the interest, is available or not. If the data is not available at a moment, the sink can not collect any data at all. Therefore, the interest becomes transmission overhead and the energy consumed for transmitting the interest is wasted. In addition to transmission overhead, this on-demand type data collection scheme may not be suitable for some sensor applications. For instance, directed diffusion is not applicable data dissemination scheme for surveillance purpose because sensor nodes have to transmit data as soon as they detect abnormality. Another example is temperature monitoring. In case of temperature monitoring, data is collected continuously, so on-demand data collection is not suitable.

2.4 Power Aware Routing Schemes

In previous three sections, well known routing protocols for wireless sensor networks – flooding, SPIN, and directed diffusion – are discussed. These three routing
schemes are designed so that collected data is disseminated efficiently in wireless sensor networks. However, there is another consideration to design a routing scheme for wireless sensor networks: power management. As it was explained, energy is scarce for sensor nodes. It is important to select a path, which consumes less amount of energy. Moreover, there is no centralized unit to monitor power status of all sensor nodes in a network. It is necessary to design a routing scheme, which will select sensor nodes with enough battery power left. Otherwise, important data packets will be dropped on a way to sink nodes. In this section, we will examine a few routing schemes based on power management.

2.4.1 Minimum Total Transmission Power Routing

The Minimum Total Transmission Power Routing (MTTPR) [14] protocol is an on-demand, reactive routing scheme which seeks an optimal path from a source to a destination node in mobile ad hoc networks only when such a path is needed. The objective of MTTPR development was to design an algorithm for finding a minimum transmission power consumption path from a source to a destination in a power-constrained network. The basic idea is that if a shortest path between two nodes is employed to transmit a data packet, the power consumed by the transmission will be minimized, because radio transmission power is proportional to the distance. More specifically, the power consumed is directly proportional to $d^n$, where $d$ is the distance between the two nodes and the value of $n$ depends on $d$; namely $n=2$ for short distances and $n=4$ for long distance [15]. Since data packets in ad hoc networks are transmitted in a multi hopping manner, the total power required in transmitting between a source and a destination is the sum of the transmission power consumed by each hop between two
nodes necessary for a packet to reach the destination node. Therefore, the total transmission power $P_t$ can be expressed as follows:

$$P_t = \sum_{i=0}^{D-1} P(n_i, n_{i+1})$$

where $P(n_i, n_{i+1})$ is the transmission power required between two nodes $n_i$ and $n_{i+1}$ the route, $D$ is the total number of nodes in the route excluding the source node, $n_0$, and $n_D$ is the destination node [15]. An optimal route is determined by minimizing the total transmission power $P_t$ over all possible routes between a source and destination node. This can be achieved by applying a shortest path algorithm, such as Dijkstra's algorithm. Because the value of $n$ in $d_n$ is determined by the distance between the two nodes, MTTPR protocol tends to select routes, which have more nodes, but with shorter distances for each hop. By selecting a path between a source and destination node with many short-distance hops, the total transmission power efficiency will be optimal. However, another consideration of the MTTPR is propagation delay. Because of the MTTPR route selecting method and the proportionality of transmission power to the distance, more nodes are usually involved in delivering data packets. Since each node requires some processing time, each node contributes to the propagation delay. Therefore, the more nodes in the route, the longer the propagation delay. Furthermore, each node consumes power in processing data packets. To address this problem, the receiving power of a node was introduced in addition to the transmission power [14]. By considering both power consumption factors, the number of nodes included in an optimal path can be reduced, as well as the propagation delay. Other consideration of the MTTPR protocol is the energy state of each node. Once an optimal path is selected, it can be used to transmit data packets as long as the route remains connected. Since some nodes can consume all
of their energy while other nodes consume very little, paths can become disconnected and the network can become fragmented.

2.4.2 Min-Max Battery Cost Routing

In order to avoid the early fragmentation of a network caused by over-usage of a particular node, the Min-Max Battery Cost Routing (MMBCR) protocol [14] can be used. This protocol is also an on-demand reactive routing scheme. The MMBCR scheme selects an optimal data path based on the power remaining in each node. To measure how much a node is willing to transmit a data packet at any given time “t”, one proposed equation is

\[ f_i^t = \frac{1}{C_i^t}, \]

where \( C_i^t \) is battery capacity of node \( i \) at time \( t \). As the residual battery capacity decreases, a node is less willing to participate in transmitting data packets. This phenomenon is expressed by increasing the f-value. The battery cost of a route \( j \), \( R_j \), is defined as the maximum f value among nodes in the j-route.

\[ R_j = \max f_i(C_i) \]

Hence, an optimal route in the MMBCR protocol is determined by finding a route having a minimum \( R_j \) value over the set \( A \) of all the possible routes \( j \in A \) between two nodes.

\[ R_{(optimal)} = \min \{ R_j \mid j \in A \} \]

The MMBCR protocol is guaranteed to select a path, whose minimum power capacity node is a maximum. However, unlike the MTPR protocol, MMBCR does not take into account the total transmission energy consumed by each data packet transmission. Therefore, the path selected by MMBCR is not necessarily the most energy efficient path.
2.4.3 Conditional Max-Min Battery Capacity Routing

The Conditional Max-Min Battery Capacity Routing (CMMBCR) protocol [15] is a routing scheme, which combines MTPR and MMBCR in an effort to maximize network power efficiency. CMMCR considers the best possible routing in terms of total transmission power and power consumption fairness over all routes in a network [24]. In CMMBCR, the battery capacities of a node are divided into two states according to a threshold capacity value. There are three possible scenarios: all nodes have capacities above the threshold (ii) all nodes' have capacities below the threshold, and (iii) some capacities are above and some are below the threshold. If the battery capacities of all nodes are above the threshold value, MTPR is used and CMMBCR selects a route with minimum total transmission power consumed per packet. Consequently, the power consumption of the whole network is minimized. On the other hand, if the battery capacities of all nodes are less than the threshold value, MMBCR is used, so that the lifetime of nodes with low capacity can be extended. In the third case, if there exists a route, between a source and a destination for which all nodes have capacities above the threshold value; the optimal route is selected by applying MTPR. If all possible routes from a source to a destination contain only nodes with capacities below the threshold value, a route is selected by applying MMBCR.

One disadvantage of CMMBCR is that it does not allocate energy utilization evenly throughout all nodes, as was expected [24]. Since the CMMBCR scheme is also a reactive routing scheme, a routing process is activated only when a route is needed for transmitting data packets. The power status of each node is not monitored continuously unlike proactive routing schemes which maintain routes periodically. Thus, after an
optimal route is selected and as long as it is used for transmitting data packets, the power status of all nodes on the route is not monitored. This means that even if the power capacity of a node on a route is below the threshold level, it has to keep transmitting data packets as long as the route is active.

2.4.4 Modified Conditional Max-Min Battery Capacity Routing

To incorporate the battery power awareness into CMMBCR, [24] introduced the Modified- CMMBCR scheme. In this scheme, two threshold values – selective-victim-search-zone (SVSZ) and forced-victim-search-zone (FVSZ) – are used in addition to the threshold value, γ, used by the conventional CMMBCR. The general idea of Modified-CMMBCR is as follows. The two constant values, SVSZ and FVSZ are applied to all nodes in a network, where SVSZ > FVSZ. On the other hand, γ is determined by a source node, so if a source applies a low γ value for one route, the route can be used despite having a low node capacity. A source node can change the threshold value depending on the data type transmitted. Also, each route can have a different γ value. Then, if the remaining power of node on a route becomes less than γ, a new route will be sought. Unless the remaining power of a node becomes less than both γ and SVSZ, all nodes continue transmitting data packets. In case the remaining power is less than SVSZ and greater than FVSZ, a source node receives a signal from a low power node to seek a new route, while the low power node continues to transmit data packets. Finally, if the remaining power of node is less than FVSZ, it sends a signal to a source node to seek a new route, and stops transmitting data packets. At this point, a node transmits data packets only when it is a source node. One advantage of this scheme over CMMBCR is that it reflects the power status of all nodes on a route during the data transmission state,
so more power-aware routing can be achieved. In addition, since each source can
determine the $\gamma$-value, a route can be selected according to the priority of data packets.
For instance, if a source has a low $\gamma$ value, more nodes participate in a selected route
than if the source has a high $\gamma$-value, so a better route, which will be a shorter and have
smaller propagation delay, can be selected. On the contrary, one disadvantage of this
modified CMMBCR is the overhead created by transmitting control signals. When the
remaining power of a node reaches SVSZ, FVSZ, or $\gamma$, it has to transmit a control signal
to its source node to select another route. This will cause more control signals throughout
the network as compared to MTPR, MMBCR, and CMBCR.

Small size of sensor nodes and the economic handling of computational power play a
substantial role in defining wireless sensor networks – a step ahead of the traditional ad
hoc networks. Furthermore, specific applications necessitate redefining some of the basic
paradigms with which energy aware protocols are engineered. Persistent effort is set to
sustain and maintain an optimal energy supply for the desired applications in wireless
sensor networks various routing schemes have been addressed in this Chapter. Each
scheme has some advantages over other. We are interested in finding an optimal blend of
the routing management scheme, which should be purely decentralized in order to cope
with scalability issues, and should consume enough power for an efficient application.

2.5 Data Compression Schemes for Wireless Sensor Networks

In [19], an extensive research has been conducted on power consumption by data
compression and transmission in wireless communication. In their experiments, the
Compaq Personal Server, which is research version of the Compaq iPAQ, was used for
data collection instead of wireless sensor nodes, so their experimental results may not be totally applicable for wireless sensor networks. However, it still gives significant insights of power consumption in data processing and transmission. The studies concluded that sending data is more power consuming than computation, and thus minimize data size before transmitting in wireless medium is effective to reduce total power consumption.

However, many existing data compression schemes – such as LZO, bzip2, or PPMd – are not suitable for wireless sensor networks because of required memory size and processing speed. Thus, it is necessary to design a low-complexity and small size data compression algorithm for sensor networks. In this section, some of data compression schemes for wireless sensor networks are introduced.

2.5.1 Coding by Ordering

The Coding by Ordering data compression scheme is introduced in [20] as part of Data Funneling Routing. The compression scheme is works as follows. First, a data pass from sensor nodes in the region of interest to a collector node is set up. In Data Funneling Routing, some of sensor nodes work as a data aggregation node. At an aggregation node, sensed data collected by other nodes is combined, and the aggregated data is sent to its parent node.

In the algorithm, when data is combined at an aggregation node, some data is dropped. To include the information of dropped data in the aggregated data, the order of data packet should be preserved. For example, four nodes (N1, N2, N3, and N4) send the data to an aggregation node (Na). The data value of each node can be any integer ranging from 0 to 5. If we decide to drop the data from N4 and express the data from N4 by ordering packets from other 3 nodes (N1, N2, and N3), there are 3! = 6 possible permutations.
Therefore, by using permutations of three packets, the data value of N4 can be included in an aggregated packet without actually including the packet of N4. The possible combination of permutation and data value is presented in Table 2.1.

<table>
<thead>
<tr>
<th>Packet Permutation</th>
<th>Integer Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>N1,N2,N3</td>
<td>0</td>
</tr>
<tr>
<td>N1,N3,N2</td>
<td>1</td>
</tr>
<tr>
<td>N2,N1,N3</td>
<td>2</td>
</tr>
<tr>
<td>N2,N3,N1</td>
<td>3</td>
</tr>
<tr>
<td>N3,N1,N2</td>
<td>4</td>
</tr>
<tr>
<td>N3,N2,N1</td>
<td>5</td>
</tr>
</tbody>
</table>

Table 2.1: Permutations and Represented Integer Values

For a general case, let’s assume that \( n \) is the total number of sensor nodes – each node has a different node ID, \( m \) is the number of nodes sending a packet to an aggregation node, \( k \) is the possible range of data value, and \( l \) is the number of sensor nodes dropped at the aggregation node. Then, the number of possible combination of IDs, which dropped nodes have, can be expressed as \( \binom{n-m+l}{l} \). Since each of \( l \) nodes can take any value among possible \( k \) data values, there are \( k^l \) possible data value combinations. When combining possible IDs and data values, there is total of \( \binom{n-m+l}{l} k^l \) possible values. This combination of values needs to be expressed by \((m-l)!\) permutations. Therefore, the following inequality has to be satisfied.

\[
(m-l) \geq \binom{n-m+l}{l} k^l
\]
Theoretically, when n = 2^7, k = 2^4, and m = 100, approximately 44% of packets can be dropped at the aggregation node by applying Coding by Ordering. Since this method has good compression ratio and simple algorithm, it may be possible to use for wireless sensor networks. One difficulty of utilizing this scheme is that since there is no efficient algorithm mapping permutation to data value, it requires a mapping table. As the number of sensor nodes aggregated increases, the size of table increases exponentially.

2.5.2 Pipelined In-Network Compression

The pipelined in-network compression scheme is discussed in [21]. The basic idea is trading high data transmission latency for low transmission energy consumption. Collected sensor data is stored in an aggregation node’s buffer for some duration of time. During that time, data packets are combined into one packet, and redundancies in data packets, will be removed to minimize data transmission.

For example, each data packet has the following form: ⟨measured value, node ID, timestamp⟩. Then, the compressed data packet has the following form: ⟨shared prefix, suffix list, node ID list, timestamp list⟩. The "shared prefix" is the most significant bits, which all measured values in combined data packets have in common. The length of shared prefix can be changed by a user based on the knowledge of data similarity. If the measured values are expected be close to each other, the length of prefix value can be set to relatively long. The "suffix list" is the list of measured values excluding the shared prefix part. The "node ID list" is the list of node identifiers and the "timestamp list" is the list of timestamp. The compression scheme is illustrated in Figure 2.3. In the figure, three nodes send the data packets to the compression node. At the compression node, three data packets are compressed into one packet. In this example, the length of shared
prefix is set to 3. In this example, total number of bits is reduced from 33 to 27.

One advantage of this simple compression scheme is that the shared prefix system can be used for node IDs and timestamps. By doing so, more data compression can be achieved. The efficiency of data compression depends on the length of shared prefix. If we could set a long shared prefix and measured values have commonality, the compression ratio increases. However, there is no similarity in measured sensor values. Even if we could set a long shared prefix, the efficiency of Pipelined In-Network Compression will decrease. In addition, if we are combining a large amount of data packets, than a large data buffer is required to temporary store those packets. Since a sensor node has only a limited size of memory space, enough buffer space will not be available.
CHAPTER 3

INFORMATION DISSEMINATION via LABEL FORWARDING

The aim of our work is to design and evaluate our novel routing scheme for wireless sensor networks. The main concern in designing our routing scheme is power consumption. In other words, we would like to create a new routing scheme, which transmits information packets from sources to destinations in low latency while conserving the power of each sensor node. To achieve that goal, we propose a new routing protocol, “Information Dissemination via Label Forwarding (IDLF)”. To achieve efficient data transmission, a routing scheme of IDLF is divided into three stages: label transmission, request transmission, and data transmission stage. In each stage, a different type of information is exchanged among sensor nodes.

This section presents characteristics of IDLF and how information packets are disseminated in a wireless sensor network by utilizing our routing scheme. Also, the energy management methods, which can be used with IDLF, are proposed and discussed.

3.1 IDLF Routing Scheme

Wireless sensor networks have numerous possible applications. For each application, there is an appropriate data dissemination scheme. For instance, flooding and SPIN work effectively for applications with disseminating data to an entire network. Data routing scheme is always initiated by source nodes for these two routing schemes. On the other
hand, directed diffusion is more suitable for applications with point-to-point data transmission. In directed diffusion, a sink node broadcasts an interest to initiate data collection. IDLF is designed for point-to-point data transmission, and routing scheme is initiated by source nodes. IDLF is a reactive and on-demand routing scheme, which seeks a routing path only when it is needed. Every time a sensor node detects an event, a new data path is constructed. The data dissemination scheme of IDLF is executed as follows.

When an event is detected by a source node, first the source forms a small information packet, called a “label”. We do not specify the structure and contents of the label here because the format of label heavily depends on application. However, the label should contain the minimum information about the event so that it can be distinguished from other events. For instance, if a wireless sensor network collects homogeneous event, each label should include the information about event generated location, event generation time, the source node’s ID, and sender node’s ID as shown in Figure 3.1. Usually, sensor nodes do not have worldwide unique IDs, such as IP addresses, so node IDs have to be assigned before or after deployment [16]. The essential design aspect of label is that the size of label is significantly smaller than the size of data packet. Otherwise the advantage of utilizing label for power efficient wireless sensor routing scheme will diminish.

<table>
<thead>
<tr>
<th>Source ID</th>
<th>Event Location</th>
<th>Time Stamp</th>
<th>Sender ID</th>
</tr>
</thead>
</table>

Figure 3.1: Example of Label Information
Then, the source starts broadcasting the label to other sensor nodes. Conventionally, to transmit data through a network efficiently, one requires either a centralized server, which controls the routing of entire network or a huge routing table in each server. Since wireless sensor networks are decentralized and distributed, they do not have a centralized node. Also, these networks are resource constraint, so each node does not have enough memory space to store a routing table. Instead, sensor nodes utilize partial information of entire sensor network. Each sensor node stores the information, such as relative location and ID, of neighboring nodes. The neighboring nodes are nodes, which locate within the node's radio transmission range. In other words, neighboring nodes are the nodes located one transmission hop away from a node.

The source broadcasts the label to all neighboring nodes. Upon receiving the label, a receiving node checks its label cache, where all received labels are stored. If the received label already exists in the label cache, the node ignores the received label. If the node receives an entirely new label, the receiving node stores the label in the cache and retransmits the label to its neighbors. At this point, it is noticed that if the label contains the information about a sending node, the receiving node can avoid retransmitting the same label to the sending node. This mechanism reduces unnecessary communication between nodes. This label transmission process is repeated until the label reaches the sink or there is no more neighboring node, which does not have the label in its label cache yet. This label transmission scheme is depicted in Figure 3.2. First, the label is transmitted from node A to its neighbor nodes, B and C. Then, node B takes turn to transmit the label. Since B receives the label from A, B tries to transmit the label to C, D, and E. Even though C has already received the label from A, B does not know the fact that C already
has the label. When node D and E receive the label from B, they store the label in their label caches. On the other hand, node C already has the label, so node C discards the label from node B.

Once the sink receives the label, the sink replies back by transmitting a request packet toward the source. Similar to a label, this request packet should be small in size compared to the actual data packet in order to minimize communication burden between sensor nodes. The request packet follows the trace, on which the label traversed from the source to the sink, back to the source as shown in Figure 3.3. In Figure 3.3, S is the source node and Z is the sink. First, the label is disseminated from the sensor network. The label reaches Z by taking the path (S – B – A – Z). Then, the request packet is transmitted back to the source node from the sink by taking the path (Z – A – B – S).
When the request packet reaches the source, the source finally starts transmitting the actual data packet to the sink by using the same path the label traversed. As in the example of Figure 3.2, the data packet is delivered by taking the path (S – B – A – Z).

![Diagram showing Label Transmission Path and Request Transmission Path]

**Figure 3.3: Label and Request Transmission**

The advantage of this routing scheme is that each node only needs to know small portion of the entire network topology – sensor nodes located one-hop away. Each node stores minimum routing information. By doing so, the node saves the limited memory space in each sensor node, and reduces the processing time for routing. In addition to that point, the size of a label is smaller than that of actual data, so storing labels requires relatively smaller memory space than storing data packets. Also, since an actual data packet is transmitted after a data path has been formed, redundant data packet transmissions can be avoided.
3.2 Label Dissemination Method

To disseminate a label to a sensor network, there are several possible methods. One choice is just broadcasting a received label to all the neighbor nodes without any restriction, which is merely applying a classic flooding. This method is simple, robust, and effective if sensor nodes have no knowledge of sink’s location. Another possible choice is employing a directional forwarding. This method is based on the assumption that all nodes in the network know the approximate location of the sink node. If all nodes know the relative location of the sink node, it can minimize the number of neighboring nodes, to which a label has to be disseminated. Therefore, this method results in energy saving. For instance, node A in Figure 3.4 has six neighbors (node 1 – 6). The location of sink is south-west of node A. Then, we can limit the label transmission from node A to its neighboring nodes, which locate at south, west, or southwest of node A. In Figure 3.4, only nodes 4, 5, and 6 are qualified for receiving the label among total of six neighboring nodes. By limiting the number of nodes information disseminated, energy consumed for exchanging information will be reduced.

![Figure 3.4 Example of Directional Forwarding](image)

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
3.3 Power Management in Sensor Networks

Along with our new routing scheme, we implement three energy management schemes for wireless sensor networks: minimum transmission around the sink, threshold energy value, and differential coding.

3.3.1 Minimum Transmission around the Sink

If the sink is at a fixed location, information packets are gathered from entire network to one fixed sink. Then, the information traffic among sensor nodes located around the sink is heavier than one among nodes located away from the sink. This indicates that those nodes located close to the sink will consume more energy in faster pace for node communication than nodes away from the sink. If the nodes around the sink run out of their energy, the sink is isolated from the entire sensor network, since no sensor node can reach the sink. This makes data collection impossible and renders the entire sensor network ineffective. In order to avoid isolation of sink node from entire network, it is necessary to adapt an energy conservation heuristic on nodes located around the sink. One approach is to restrict transmission between the sink and its one-hop neighbors. In our proposed routing scheme, each sensor node has the knowledge of its neighboring nodes, so the nodes around the sink know that they are one-hop away from the sink. Then, when those nodes transmit a label, they can transmit the label only to the sink instead of broadcasting. By doing so, energy consumed for transmission by sink’s neighboring nodes will be reduced.

3.3.2 Battery Threshold Value

For any kind of sensor networks, it is crucial that the collected data is safely delivered to a desired destination. For conventional wired sensor networks, the flow of data
packets and conditions of sensor nodes are usually monitored and controlled by centralized units. On the contrarily, wireless sensor networks are not equipped with centralized controlling unit for monitoring the entire network. Therefore, in wireless sensor networks it is more probable that the information packets be dropped on the way to the destinations than in wired sensor networks.

There are several causes that the data packets are not delivered to the destination nodes properly. Limited battery power of each sensor node is one of the essential causes of data packets transmission failure. In Wireless sensor networks, information packets are disseminated hop-by-hop among sensor nodes and usually each sensor node has the minimum amount of information about one-hop away nodes. This is true since exchanging information frequently will consume node’s limited resources – battery power and bandwidth – and also increases the transmission overhead. In many cases a sensor node does not know the battery status of the next hop node. Therefore, it may occur that while a node is transmitting an information packet to its one-hop neighbor, the neighbor node runs out of battery, or the information sending node, itself, runs out of battery. In either case, the information packet is lost.

One way of avoiding information loss is to set a threshold energy value on sensor nodes. For instance, in [24] the Modified Conditional Max-Min Battery Capacity Routing (Modified-CMMBR) is introduced. The Modified-CMMBR employs three different battery threshold values, and the source node selects a different routing scheme depending on remaining power of sensor nodes so that all nodes participating in data propagation have enough remaining energy. Whenever a node reaches a certain threshold value, the node sends a signal to the source node. According to the signal, the source
selects a new route based on a different routing scheme depending on the available battery thresholds for the sensor network nodes. However, this scheme creates some transmission overhead by requiring for control signals.

Instead of using three different battery threshold values, we use only one battery threshold value. The threshold value is determined based on the total energy required to receive and broadcast a label, receive and transmit a request, and receive and transmit a data. However, unlike Modified-CMMBR, there is no control signal even if a node becomes below threshold value. While nodes are above threshold value, a normal routing scheme, which is explained in the previous section, is performed. When a node’s battery level drops below the threshold value, it means that the node cannot complete the entire routing scheme, so there is a possibility that an information packet is dropped. In that case, the node still keeps receiving all labels. However, the node does not participate in the rest of routing stages. Therefore, intuitively by applying a battery threshold value the probability of information packets being dropped will significantly decrease.

3.3.3 Differential Coding

One requirement for data compression schemes aimed at wireless sensor networks is simplicity. Such schemes must be simple enough not to require a super fast processor or a huge memory space, yet be reasonably effective in reducing the data size. To arrive at a simple yet effective compression scheme, one must take into account the data characteristics managed by the sensor network. There are various applications for wireless sensor networks. In many security systems, wireless sensor networks monitor abrupt changes in an area. For instance, a wireless fire alarm system observes sudden changes in temperature and concentration of CO in the monitored area. On the other hand,
many applications of environmental monitoring are collecting the information, which does not change rapidly from one sample to the next sample, such as ambient temperature, humidity or pressure. In such cases, subsequent measurements of sensor nodes are temporally correlated.

A very natural candidate for temporally correlated data is differential coding. Compared to other data compression schemes, differential coding is relatively simple. The simplest form of differential coding is to make the prediction based on the difference in consecutive samples. Since the difference between one sample to the next is small, fewer information bits need to be transmitted when sending the difference as compared to sending the sample itself. This will in turn result in reduction in energy consumption in the sensor network when sending the data.
SENSOR NETWORK PERFORMANCE EVALUATION

4.1 Network Simulation Model

To simulate various routing schemes, including IDLF, and compare their performance, we developed a sensor network simulator in C++. In our simulator, a specified number of sensor nodes, which is ranging from 2 to 80 nodes including one sink node, are randomly placed in a 10 x10 unit sensor simulation grid. Figure 4.1 shows an example of test sensor network configuration with 30 nodes. We assumed that all sensor nodes in the sensor area are completely connected in lossless network.

We assumed that a sink node is a special node. There is only one sink in the simulation area. The sink node is always placed at a fixed location. In Figure 4.1, “S” indicates the sink node, which is at (0, 0). The sink node only collects sensed data from other sensor nodes, but does not sense the event. Also, the sink is not resource constrained. It is equipped with enough memory space, battery power, and processing speed so we exclude the power consumed by the sink from the total power consumed by an entire sensor network during simulations.

Each sensor node can directly communicate with other nodes (neighbor nodes), which is located within one unit distance from the node. For instance, Node 11 in Figure 4.1 can directly communicate with Node 6, 16, 17, and 23. Based on node configuration and communication range of nodes in our simulation, each node can have a maximum of
eight neighbors. In our simulator, for any routing protocol, sensor nodes have to know their one-hop neighbors to disseminate information. The amount of energy consumed for exchanging information during a neighbor discovering stage is the same for any routing protocol for the same network topology. Therefore, we do not consider energy consumed during neighbor discovery in our energy analysis. During a simulation, there will be only one source node in the sensor field at a time. After a data packet reaches the sink, a new source will be selected randomly. Then, the new source starts propagating a label.

When directional forwarding is applied for a routing scheme, a node transmits a packet only to neighboring nodes, which are located closer to the sink than the sender. For instance, node 24 disseminates an information packet only to nodes 15, 19, and 22 in directional forwarding.

![Figure 4.1 Example of Nodes Allocation](image-url)
We did not set any specific values for packet size, transmission power, and radio speed. However, based on various sensor simulation characteristic models shown in [4], [17], and [18], we assumed that the size of data packet is 31 times greater than the size of the label and request packet. Then if we assume that transmitting a label or request packet between two neighboring nodes takes one unit time, transmitting a data packet will take 31 unit times. Also, we assumed that transmitting information consumes 3 times more energy per unit time than receiving, so transmission and receiving takes up 3-unit energy per unit time and 1-unit energy per unit time respectively. Table 3.1 summarizes the network characteristics.

<table>
<thead>
<tr>
<th>Simulation Area</th>
<th>10 ×10 unit area</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Nodes</td>
<td>2 - 80</td>
</tr>
<tr>
<td>Number of Sinks</td>
<td>1</td>
</tr>
<tr>
<td>Radio Range</td>
<td>3 ×3 unit area</td>
</tr>
<tr>
<td>Data Size</td>
<td>31 unit</td>
</tr>
<tr>
<td>Request Size</td>
<td>1 unit</td>
</tr>
<tr>
<td>Label Size</td>
<td>1 unit</td>
</tr>
<tr>
<td>Data Propagation Time</td>
<td>31 unit time</td>
</tr>
<tr>
<td>Request Propagation Time</td>
<td>1 unit time</td>
</tr>
<tr>
<td>Label Propagation Time</td>
<td>1 unit time</td>
</tr>
<tr>
<td>Transmission Power</td>
<td>3 unit energy/unit time</td>
</tr>
<tr>
<td>Receiving Power</td>
<td>1 unit energy/unit time</td>
</tr>
</tbody>
</table>

Table 4.1 Sensor Network Characteristics

4.2 Routing Performance Evaluation

To measure the effectiveness of IDLF compared to other routing algorithms, we simulated two additional routing schemes: Flooding and SPIN. In addition to these 3...
routing algorithms, we applied directional forwarding for each routing algorithm, so six routing algorithms were simulated. For each routing algorithm, we ran 50 simulations, and the averages of simulation results were plotted on graphs. In this set of simulations, the energy supply of each sensor node was set to be unlimited.

4.2.1 Energy Consumption over Time

Figure 4.2 shows the amount of energy consumed by an entire network over time with different number of nodes in the network by flooding (Flood), flooding with directional forwarding (Flood-D), IDLF, IDLF with directional forwarding (IDLF-D), SPIN, and SPIN with directional forwarding (SPIN-D).

As expected, the results show that at any number of nodes in a network, by applying directional forwarding each routing scheme achieves significant energy savings. While the number of nodes in a network is small, disseminating data packet to an entire network is not costly. Thus, the difference in the amount of consumed energy by each routing scheme is small. When the number of nodes is 5, flooding, which consumed the most energy, dissipated approximately 4.3% more energy than IDLF-D, which consumed least energy, after 3,000 simulation time. However, as the number of nodes in a network increases, the difference in energy consumption among routing schemes becomes quite obvious. At 30 nodes in a network after 3,000 simulation time, Flood, Flood-D, IDLF, SPIN, and SPIN-D consumed 112, 21, 29, 83, and 15% more energy than IDLF-D, respectively. Also, it is noticed that flooding and SPIN consume more energy than IDLF in both cases – with or without directional forwarding. This is because flooding and SPIN are designed for disseminating data through the entire network. On the other hand, IDLF is designed for point-to-point data transmission. Since SPIN is designed to prevent
implosion and overlap in flooding, SPIN performs better than flooding in terms of energy consumption.

Figure 4.2: Energy Consumed by Entire Network over Time
(a) 5 nodes, (b) 10 nodes
Figure 4.2: Energy Consumed by Entire Network over Time
(c) 15 nodes, (d) 20 nodes, (e) 25 nodes

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
4.2.2 Data Transmission over Time

To study data dissemination efficiency for various routing schemes, we collected the number of data packets arrived at a sink for a given simulation interval as shown in Figure 4.3. Each graph shows the simulation results of different number of nodes in a network. In general, as the number of nodes in a network increases, the number of data packets delivered to a sink in a given time diminishes. This is because the average distance from a source to a sink increases, so it takes more time to deliver a data packet. When comparing a routing with directional forwarding to without it in terms of number of data packets delivered, the routing with directional forwarding always surpasses the routing without directional forwarding. Flooding with directional forwarding conveys 53 to 85% more data packets than normal flooding. SPIN improved its performance between 40 to 52% by adopting directional forwarding. Likewise, IDLF improved between 24 to 37%. It is noticed that the data transfer efficiency is most by improved in flooding. Since applying directional forwarding limits information transmission, this result indicates that
flooding exchanges more information between nodes than other two protocols. When comparing three routing schemes without directional forwarding, IDLF performed better than SPIN and flooding. At 30 sensor nodes in a network, IDLF delivered 101 and 72% more data packets than flooding and SPIN respectively. Similarly, IDLF with directionality delivered more data packets to a sink in a given simulation time interval than other two routing schemes with directionality. At 30 sensor nodes, IDLF-D delivered 80 and 68% more data packets than Flood-D and SPIN-D respectively. It is worth to mention that the number of data packets delivered by Flood-D is very close to that delivered by SPIN-D. By applying directional forwarding, inefficiency caused by implosion and overlap in flooding is minimized. Thus, flooding could perform as effective by as SPIN in this scenario. Furthermore, SPIN has to exchange ADV and REQ packets before transmitting actual data. In some cases, these two packets become overhead compared to just transmitting data only. Therefore, in this simulation Flood-D could deliver as many packets as SPIN-D.

![Figure 4.3: Number of Data Packets Delivered over Time (a) 5 nodes](image)

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
Figure 4.3: Number of Data Packets Delivered over Time
(b) 10 nodes, (c) 15 nodes, (d) 20 nodes

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
4.2.3 Energy Consumption by a Data Packet

Figure 4.4 shows the average energy consumed by delivering each data packet from a source to a sink. At any number of nodes in a network, IDLF-D outperformed other five routing schemes. As the number of nodes in a network increases, the superiority of IDLF-D becomes more obvious. At five nodes in a network, IDLF-D consumed only 34, 8, 17, 37, and 8% less energy for each data packet than flooding, Flood-D, IDLF, SPIN, and
SPIN-D respectively. However at 30 nodes in a network, the number increases to 447, 99, 67, 325, and 82%. Obviously this trend indicates that IDLF-D is suitable for wireless sensor networks because the scalability is one of the main concerns in wireless sensor networks. Another significant point that needs to be mentioned about the analysis is that IDLF requires less energy to deliver a data packet than Flood-D and SPIN-D. This indicates that even featuring directional forwarding is not able to overcome the disadvantage of flooding and SPIN. Therefore, IDLF is greatly more suitable for disseminating information point-to-point in wireless sensor networks than flooding and SPIN.

![Figure 4.4: Energy Consumption by Each Data Packet](image)

4.3 Power Management Analysis

In this section, we display and examine the data collected to measure our propose power saving and management schemes for wireless sensor networks. First, the simulation results for measuring the effect of minimum transmission around a sink node...
on nodes’ energy consumption are studied. Then, we discuss the experimental results of employing a battery threshold value on sensor nodes.

4.3.1 Minimum Transmission around the Sink

Based on the idea proposed in the previous section, we simulated two scenarios – with and without transmission control around a sink node – and collected energy consumption data of nodes. 50 simulations for each scenario were performed over IDLF with 30 sensor nodes for 3,000 unit times. Then, the average of energy consumed by sets of sensor nodes, which are located in the same number of hops from the source node, was calculated and graphed. The result is shown in Figure 4.5.

The simulation results show that nodes located one hop away from the sink consumed an average of 1046 unit energy in unrestricted broadcast scenario. By applying the restricted transmission scheme, the average energy consumed by nodes was reduced to 791 unit energy, which is 24% improvement. In the mean time, this does not affect the energy consumption of other nodes. Therefore, by applying the restricted transmission...
scheme, total energy consumption of a sensor network can be also reduced. As it was expected, because of denser information traffic around the sink node, sensor nodes located close to the sink consume more energy than those farther from the sink. For instance, according to our simulation results, nodes located one hop from the sink in unrestricted broadcast scenario consumed 7.5 times more energy than those locates nine hops from the sink.

4.3.2 Battery Threshold Value

To measure the effect of battery threshold value, a threshold value for a sensor node was set according to our sensor network characteristics. The main purpose of applying a battery threshold value on a routing scheme is to prevent dropping data packets. Thus, we assumed that a node has a maximum number of neighboring nodes, which is eight so that as long as a node has a battery power over the threshold value, the node will never drop a data packet. The calculation result is shown in Table 4.2.

<table>
<thead>
<tr>
<th>Operation Description</th>
<th>Energy Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Receiving Label (8 neighbors)</td>
<td>8 unit energy</td>
</tr>
<tr>
<td>Transmitting Label (8 neighbors)</td>
<td>24 unit energy</td>
</tr>
<tr>
<td>Receiving Req</td>
<td>1 unit energy</td>
</tr>
<tr>
<td>Transmitting Req</td>
<td>3 unit energy</td>
</tr>
<tr>
<td>Receiving Data</td>
<td>31 unit energy</td>
</tr>
<tr>
<td>Transmitting Data</td>
<td>94 unit energy</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>160 unit energy</strong></td>
</tr>
</tbody>
</table>

Table 4.2: Battery Threshold Value

Along with this threshold value, we performed 50 simulations for different number of nodes in a sensor network – 25, 30, 35, 40, and 45 nodes – with four different initial
nodes energy - 600, 700, 800, and 900 unit energy. Simulation time was set to 3,000 unit time. Figure 4.6 illustrates the number of data packets arriving at the sink in a given simulation time and initial battery energy with and without a threshold value (No TH and TH respectively in the legend of Figure 4.6).

When comparing four different initial energy graphs, it is noticed that as the number of nodes in a network increases, the number of data packets delivered to a sink falls in all four initial energy simulations. When the number of nodes increases in the network, the average distance from a source to the sink also increases. Then, the time consumed for each data packet delivered to the sink is prolonged. Therefore, the number of data packets delivered is reduced.

Another point to be mentioned is that in all four graphs, it is noticed that as the number of nodes increases, the difference in number of data packets delivered between the cases of with and without threshold value is minimized. Since a network with a large number of nodes propagates less number of data packets, as explained above, each node consumes less energy. Also, in our simulation a source node is randomly selected and the data path is selected each time. If the number of nodes in the network is large, the probability of each node selected as a part of data path is small. It reduces power consumption of each node. Because of these two reasons, power failure of sensor nodes in a network with larger number of nodes is less likely to happen than that with smaller number of nodes. If that was the case, there is no difference between a routing with and without battery threshold value. Thus, at larger number of nodes in a network, the number of data packets delivered by a routing with threshold value becomes close to that delivered by a routing without threshold value. In addition to that, the effect of applying a
threshold value on a routing is more significant at low initial battery energy than high initial battery energy. This is due to the same reason explained above. At low initial battery energy, sensor nodes fail more quickly than those at high initial battery energy. Because of that, a routing with a threshold value can deliver significantly more data packets.

![Graphs showing number of data packets delivered at different initial energy levels](image)

Figure 4.6: Number of Data Packets Delivered
(a) 600 unit energy, (b) 700 unit energy, (c) 800 unit energy, (d) 900 unit energy

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
Figure 4.7 shows energy consumed by an entire network with different initial battery energy. When initial battery energy is more limited, the difference between with and without the battery threshold value is more significant, similar to Figure 4.6. At initial energy of 600 unit energy, a network with a threshold value obviously consumed more energy than that without a threshold value. This difference can be explained simply by the number of data packets delivered. Figure 4.6 (a) indicates that a network with a threshold value delivers more data packets than that without a threshold. Since delivering more data packets consumes more energy, a network with a threshold value consumed more energy than that without. Then, differences in the number of delivered data packets are reduced between with and without a threshold value as initial battery energy increases, shown in Figure 4.6. Therefore, differences in energy consumption between with and without a threshold value are reduced in Figure 4.7 as initial battery energy increases.

Another interesting point of Figure 4.7 is the change in power consumption at given number of nodes with respect to initial battery energy. Figure 4.6 shows that the number of data packets delivered does not change abruptly for a given number of nodes when changing initial battery power (except at 25 nodes with initial energy of 600). However, total energy consumption by a network for a given number of nodes steadily increases as initial battery energy increases. Since the number of nodes in a network, the number of data packets delivered, and the simulation time are the same, there should be another factor, which affects energy consumption. The power consumption in IDLF simulation is caused by exchanging 1) labels, 2) request, and 3) data. If the number of data packets delivered is same, the amount of energy consumed by 2) and 3) is the same as the number of nodes and simulation times. Thus, the only possibility is due to label exchanges. When
increasing initial battery energy, the number of nodes running out battery in a given simulation time will be reduced. Then, more nodes participate in exchanging labels. Therefore, at the higher initial battery energy, more total energy is consumed by a network.

Figure 4.7: Total Energy Consumption by a network
(a) 600 unit energy, (b) 700 unit energy, (c) 800 unit energy, (d) 900 unit energy

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
Figure 4.8 shows the percentage of information packets dropped out of all sensed events. There are three different types of information packets dropping – 1) dropping label, 2) dropping request and 3) dropping data. In IDLF, a data or request packet is transmitted only once. Thus, when 2) and 3) happen in our routing scheme, a data packet never reaches a sink. However, a label packet is broadcast, so it is possible that many copies of the label packet traverse a network at the same time. Dropping one label does not mean losing all copies of label. Therefore, when a label never reached a sink, we consider it as “dropped label”. In case of simulation results with a threshold value, only 1) can happen. Because of a threshold value, if there is a possibility of 2) or 3) occurring, a node does not participate in information transmission at all.

When the number of nodes is small at small initial node energy, the percentage of information packets dropped for two cases is approximately the same. This can be explained by the availability of alternate paths. When the number of nodes is small, there is very limited number of paths between any two nodes. Then, when nodes start failing, it is more likely that labels will not reach a sink. Dropping labels happen more frequently than other two types of information packets droppings. Therefore, there are no significant percentage differences between with and without threshold value. On the other hand, as the number of nodes increases, the percentage differences between two schemes become significant because of the availability of alternate paths.

It is obvious that at any point IDLF with a threshold value performs better than IDLF without a threshold value. According to Figure 4.4, IDLF with a threshold value always delivers more or equal amount of data packets, so IDLF with a threshold value drops less number of information packets than IDLF without a threshold value.
The simulation results show that applying a threshold value on sensor nodes increased the number of data packets delivered, and, in the means time, prevented a sensor network from dropping information packets. Since our proposed method does not require any information exchanges about their battery status between sensor nodes, there is no transmission overhead. Each sensor node has to keep track of own battery level and compare the battery level with a threshold value every once in a while. On the other hand,
because they have no knowledge about their neighbors' battery status, they transmit a label to neighbors indiscriminately. When a receiving node is below a threshold value, the transmission energy is totally wasted since those neighbors do not participate in the routing scheme at all.

4.3.3 Differential Coding

In our proposed routing scheme, the actual data packet is transmitted after a path from a source to a sink is constructed. Then, the energy consumed by each bit transmitted in a multi-hop sensor network, whereby all sensor nodes of a data path are connected in a straight line, is given by the following equation [22]:

\[ E_{\text{linear}} = \left[ n(e_{\text{TE}} + e_{\text{RE}}) - e_{\text{RE}} + \frac{e_{\text{TA}}D^\beta}{n^{\beta-1}} \right] \]

where \( n \) is the number of hops between the source to the sink, \( e_{\text{TE}} \) is the energy used by transmitter, \( e_{\text{RE}} \) is the energy used by a receiver, \( e_{\text{TA}} \) is the energy used by an amplifier, \( D \) is the total distance between a source and a sink, and \( \beta \) is the path loss exponent of the channel. In this equation, the distance between any two neighboring nodes of a data path is assumed to be equal to \((D/n)\). The energy used by transmitter can be further divided into [23]

\[ e_{\text{TA}} = \frac{(S/N)_{\text{i}}(NF_{\text{Rx}})(N_0)(BW)(4\pi/\lambda)^\beta}{(G_{\text{ant}})(\eta_{\text{amp}})(R_{\text{bit}})} \]

where \((S/N)_{\text{i}}\) is the signal to noise ratio at the receiver, \(NF_{\text{Rx}}\) is the receiver noise figure, \(N_0\) is the thermal noise floor in a 1 Hz bandwidth, \(BW\) is the channel noise bandwidth, \(\lambda\) is the wavelength, \(G_{\text{ant}}\) is the antenna gain, \(\eta_{\text{amp}}\) is the transmitter amplifier efficiency, and \(R_{\text{bit}}\) is the raw bit rate.
Along with Equations (1) and (2), the following parameters, shown in Table 4.3 [23], are used for calculating the energy consumption in a multi-hop data transmission.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\frac{S}{N}_i$</td>
<td>11 dB</td>
</tr>
<tr>
<td>$N_{F_{Rx}}$</td>
<td>10 dB</td>
</tr>
<tr>
<td>$N_0$</td>
<td>4.17E-21 J</td>
</tr>
<tr>
<td>$BW$</td>
<td>2.50E+05 Hz</td>
</tr>
<tr>
<td>$\lambda$</td>
<td>0.125 m</td>
</tr>
<tr>
<td>$\beta$</td>
<td>2</td>
</tr>
<tr>
<td>$G_{ant}$</td>
<td>-20 dB</td>
</tr>
<tr>
<td>$\eta_{amp}$</td>
<td>0.2</td>
</tr>
<tr>
<td>$R_{bit}$</td>
<td>2.50E+05 bps</td>
</tr>
<tr>
<td>$c_{TE}$</td>
<td>1.45E-08 J</td>
</tr>
<tr>
<td>$c_{RE}$</td>
<td>4.45E-08 J</td>
</tr>
<tr>
<td>$D$</td>
<td>1.00E+02 m</td>
</tr>
</tbody>
</table>

Table 4.3: Calculation Parameters for Multi-Hop Network

Figure 4.9 shows the amount of energy consumed by a multi-hop data transmission with different number of hops. In this experiment, we assumed that subsequent measurements of sensor nodes are temporally correlated, and the size of original data packet (8-bit) is twice the size of the data packet, which is constructed by applying a simple differential coding to the original data packet (4-bit). Since the energy consumed by each data bit is the same for both cases as defined in Equation (1), if the size of the original data packet is twice the size of the compressed data packet, the total energy consumed by disseminating the original data packet is twice that by disseminating the compressed data packet. As shown in Figure 4.9, a sample without differential coding consumes twice as much energy as a sample with differential coding.
To minimize the effect of corrupted data packets, it is necessary to transmit an original data packet once in a while. Figure 4.10 illustrates the amount of energy saved by transmitting different number of compressed packets per each original data packet in various number of hops compared with the case when all original data packets are transmitted. We assumed a total of 100 data packets were transmitted. The graph indicates that as the number of hops increases, the amount of energy saved increases almost linearly with the same number of compressed packets per an original packet. This result could be expected from Figure 1. Also, as the number of compressed data packets per original data increases, the amount of energy saved increases. The amount of energy saved can be expressed by the following equation:

\[ E_{\text{save}} = (E_{\text{linear}})(x)(B_0 - B) \]  

where \( E_{\text{save}} \) is the total amount of energy saved, \( x \) is the total number of compressed data packets, \( B_0 \) is the size of the original data packet in bit, and \( B \) is the size of the compressed data packet in bits.
Figure 4.10: Energy Saved for 100 Data Packets

Since a differential coding is simple, there is a minimal computational overhead. Also, this coding does not require any information exchange among sensor nodes, so there is no transmission overhead. By applying the differential coding, the amount of energy consumed for transmission is reduced largely. One draw-back of this method is spending effort on storing the previous values. When decoding a compressed data packet in the differential coding at a sink node, the sink has to know the previous data value in addition to the difference. Then, the sink has to store the all previous values to be decoded. Generally, this is not a problem as sink nodes are typically equipped with adequate processing and storage hardware.
CHAPTER 5

CONCLUSIONS AND FUTURE WORK

In this work, we have presented a new routing algorithm, called Information Dissemination via Label Forwarding (IDLF), for wireless sensor networks. IDLF disseminates a label through an entire network, followed by exchanging a request, before transmitting actual sensed data. By exchanging a label and request, a data path is formed between a source and sink node. The path ensures that a data packet is transmitted to the sink node without wasting energy on transmitting a data packet to redundant nodes.

Along with our new routing algorithm, we have also introduced four power management schemes for wireless sensor networks: directional forwarding, minimum transmission around the sink, battery threshold value, and differential coding. In directional forwarding, sensor nodes narrow the range of broadcasting data packets based on location information about a sink to reduce transmission energy. By applying minimum transmission around the sink, sensor nodes utilize neighboring nodes information. Nodes, locate one-hop away from the sink, transmit information packet only to the sink. Sensor nodes below a battery threshold value do not participate in the data dissemination process to prevent dropping important data packets. By applying differential coding on temporally correlated data of wireless sensor networks, the size of data transmitted is reduced. Thus, energy consumed of transmitting data is also reduced.
We simulated flooding, SPIN, and IDLF on our C++ wireless sensor network simulator. After examining the simulation results, we reached the following conclusions. First of all, IDLF outperformed flooding and SPIN in terms of energy. To deliver one data packet to a sink, IDLF consumes approximately 30% and 39% as much energy as flooding and SPIN respectively. According to this result, IDLF is greatly more suitable for disseminating information point-to-point in wireless sensor networks than flooding and SPIN. By applying directional forwarding, the average energy consumed by transmitting one data packet from a source to a sink is halved in all three routing protocols. The simulation results of other three energy management schemes also show significant improvement in total energy consumed by transmitting data. In addition to less energy consumption, when the battery threshold value scheme is applied on sensor nodes, the sensor network drops fewer number of data packets than the network without the threshold value. Therefore, more data packets are securely delivered to a sink in a given time interval.

Even though the experimental results show that IDLF with proposed energy management schemes has significant energy efficiency over conventional routing schemes, there are still a lot of studies has to be conducted. For instance, in our simulation we assumed that there is only one source node in a sensor network at one time. However, the network could have multiple sources at the same time. We would like to examine the effect of multiple sources on energy consumption and data dissemination. Another significant aspect to be considered is fault tolerance. Since sensor nodes are prone to fail due to environmental interferences or small battery capacity, it is important to examine how the IDLF sensor network behaves when number of sensor nodes fails. One way of
improving fault tolerance is forming multiple data paths from a source to a sink. It is obvious that maintaining multiple paths increase total power consumption. However, it will increase the probability of data reaching a destination node. Therefore, we would like to study the relationships of power consumption, number of data packets delivered, and number of data paths to examine an optimal number of paths between a source to a sink.
BIBLIOGRAPHY


[18] W. J. Kaiser, WINS NG 1.0 Transceiver Power Dissipation Specifications, Sensoria Corp.


VITA

Graduate College
University of Nevada, Las Vegas

Naoto Kimura

Local Address:
10710 Destination Ln #202
Las Vegas, NV 89144

Home Address:
1-5-17-703 Katahira Aoba
Sendai, Miyagi 980-0812
Japan

Degrees:
Bachelor of Science, Chemical Engineering, 1995
University of Washington

Bachelor of Science, Engineering, 2003
University of Nevada, Las Vegas

Thesis Title: A Label Dissemination Routing Protocol for Wireless Sensor Networks

Thesis Examination Committee:
Chairperson, Dr. Shahram Latifi, Ph.D., P.E.
Committee Member, Dr. Emma Regentova, Ph.D.
Committee Member, Dr. Muthukumar Venkatesan, Ph.D.
Graduate Faculty Representative, Dr. Kazem Taghva, Ph.D.