Multipath Routing Algorithm for Wireless Sensor Networks

by

Ye Ming Lu

B. Eng., École Ploytechnique de Montréal, 1999

A THESIS SUBMITTED IN PARTIAL FULFILMENT OF
THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF APPLIED SCIENCE

in

THE FACULTY OF GRADUATE STUDIES

(Electrical and Computer Engineering)

THE UNIVERSITY OF BRITISH COLUMBIA

December 5, 2005

Abstract

Unlike conventional wireless cellular networks, the energy efficiency is a critical design issue for wireless sensor networks (WSNs). Many works have been done to design routing protocols that allow the sensors to distribute data efficiently with limited energy supply.

In this thesis, we propose a novel routing algorithm to disseminate information via multiple paths in static and energy-constrained WSNs. The algorithm consists of a distributed multipath search protocol and a load balancing algorithm. The multipath search protocol discovers multiple node-disjoint paths that connect a pair of sink and source nodes. The load balancing algorithm helps the sink node to allocate traffic over multiple paths found based on their cost, which depends on the energy levels and the hop distances of nodes along each path. We consider it as a key to improve the energy efficiency in our protocol.

The results based on the use of ns-2 simulator show that our algorithm can prolong the network lifetime by 9% to 18% and reduce the node energy consumption by a maximum of 34% over comparable schemes, including the energy-aware routing, the directed diffusion, and the directed transmission. The results also indicate that the multipath routing has low control message overhead and incurs a small data packet transfer delay.

Contents

List of Tables

List of Figures

List of Acronyms

ADV	Advertisement
AODV	Ad hoc On-demand Distance Vector
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
DSR	Dynamic Source Routing
GAF	Geographic Adaptive Routing
GEAR	Geographic and Energy Aware Routing
GPS	Global Positioning System
IEEE	Institute of Electrical and Electronic Engineers
LEACH	Low Energy Adaptive Clustering Hierarchy
MAC	Media Access Control
M-MPR	Mashed Multipath Routing
QoS	Quality of Service
REQ	Request
SAR	Sequential Assignment Routing
SPIN	Sensor Protocols for Information via Negotiation
WSN	Wireless Sensor Network

Acknowledgements

I would like to express my sincere thanks to my supervisor, Professor Vincent W. S. Wong, for his guidance, advice, and support throughout the course of this thesis. I also wish to express my gratitude to my colleagues, particularly Marc Lee and Amir Hamed Mohsenian Rad, for their help and comments on my simulation work and optimization methods.

Many thanks go to my wife and parents, for their support and encouragement throughout this long but rewarding process.

Chapter 1

Introduction

Wireless sensor networks (WSNs) consist of densely deployed sensor nodes, which have limited computational capabilities, power supply, and communication bandwidth. These small, smart and inexpensive sensing and computing devices open up new vistas for scientists and engineers to observe and monitor physical phenomenon. The potential applications of sensor networks widely span both civil and military domains. For military applications, wireless sensor networks can be used for surveillance in battlefields. For civil applications, the sensor networks can be used to monitor light, temperature, humidity and other environmental factors that affect the habitat of endangered species. Other applications of wireless sensor networks can be found in [2].

There are still many technological hurdles to overcome before wireless sensor networks can be widely deployed. The individual sensor nodes are resource constrained. They have limited battery resources, processing capabilities, and communication bandwidth. The ability to conserve power will determine their lifetime. An energy-efficient and scalable routing protocol plays an essential role to facilitate data dissemination from the source nodes to the sinks. The scalability assures that the size of sensor networks will not impact their functionality, as the number of nodes in the network varies from several hundreds to thousands. It also helps nodes to adapt various topological and geographical conditions, since nodes are usually deployed randomly. The energy efficiency, on the other hand, allows sensor networks to prolong their lifetime, as sensor nodes can only carry limited energy supply. In summary, the characteristics of wireless sensor networks require unconventional and unique networking techniques to address these challenges.

Depending on the network structure adopted, the routing protocols for wireless sensor networks can be classified into *flat network routing*, *hierarchical network routing*, and *location-based routing* [12]. In *flat network routing*, all nodes have equal functionality and they cooperate to perform the sensing tasks. The Sensor Protocols for Information via Negotiation (SPIN) [24][25] and directed diffusion [7] fall into this category. The *hierarchical network routing* divides the network into clusters or grids in order to achieve scalability and energy efficiency. The Low Energy Adaptive Clustering Hierarchy (LEACH) [26] is an example of hierarchical network routing protocol. The *location-based routing* relies on the node positions, which can be obtained from Global Positioning System (GPS) device attached to the sensor to handle the data routing. The Geographic Adaptive Routing (GAF) [27] and Geographic and Energy Aware Routing (GEAR) [4] are two examples of the location-based routing protocol.

Given the adopted network structure, the routing protocols for WSNs can operate in different ways. That can be divided into *negotiation-based*, *query-based*, *QoS-based*, and *multipath-based*. The *negotiation-based* protocols have the objective to eliminate redundant data by including high level data descriptors in the message exchanged. The sensor node can make communication decisions based on the data descriptors and the energy level of its battery. The SPIN [24][25] protocol is an example of this type of protocol. For *query-based* protocols, such as the directed diffusion [7], the communication is initiated by the sink node that broadcasts query for data over the network. The source node sends the data back to the sink node if it has data that matches the query. The *QoS-based* protocols allow sensor nodes to balance between the energy consumption and certain pre-determined QoS metrics before they deliver the data to the sink node. The Sequential Assignment Routing (SAR) [33] is one of the *QoS-based* protocols. Finally, the *multipath-based* routing protocols, such as the schemes proposed in [5] and [16], tend to enhance the reliability through the use of multiple paths. The alternative path takes over the responsibility of relaying data when the primary path ceases functioning due to node failures or topology changes. However, the energy efficiency is sacrificed as extra energy is spent to maintain the alternative paths.

In this thesis, we focus on the design of a multipath routing protocol for WSNs, in which the data is transmitted through the available multipaths simultaneously. The traffic load handled by each individual path is optimized in order to extend the network lifetime. Such a protocol is able to increase the energy efficiency with traffic being balanced over the network, while preserving the advantage of high reliability of conventional multipath routing protocols.

1.1 Motivations and Objectives

Most of the existing routing protocols for WSNs use a single path to transmit data from the source nodes to the sink. The optimal path is selected based on the metrics, such as the gradient of information [7], the distance to the destination, or the node residual energy level [27]. Some other routing protocols that use multiple paths [5][14][16] choose the network reliability as their design priority. The data transmission relies mostly on the optimal path. The alternative path is used only when the nodes on the primary route fail. Although the existing single-path approach is flexible, simple and scalable, it has several drawbacks that cannot be ignored.

First, the path selection mechanism is usually performed in a hop-to-hop basis. Each node selects the next hop based on some empirical measurements of its neighbors, such as the hop distance and the residual node energy level, until the sink node is reached. The sink node does not have sufficient knowledge on the overall route conditions to make optimal decisions. Second, the usage of a single route will unnecessarily stress the nodes along the path. They deplete their energy supply much faster than other nodes in the network because of the limited power supply carried by each sensor node. Consequently, the wide disparity in the nodes' residual residual energy levels may result in a network being disconnected in an early stage. Third, the existing multipath protocols increase the network reliability at the expense of using extra control messages to discover the alternative routes and to maintain them.

The goal of our work is to design a multipath routing protocol for WSNs. The objectives are as follows:

1. Develop a distributed and scalable multipath search algorithm to discover multiple node-disjoint paths between the sink and source nodes.

2. Develop a load balancing algorithm to distribute the traffic over the multiple paths discovered.

3. Evaluate the performance improvements in terms of network lifetime and node energy efficiency under different traffic patterns and topologies by comparing with the existing routing schemes, such as the directed diffusion [7], the directed transmission [10] and the energy-aware routing [5].

1.2 Contributions of the Thesis

The main contributions of this thesis are as follows:

1. **Multipath Search and Routing Protocol:** To discover alternative node-disjoint paths that connect the sink and source nodes, we propose a multipath search protocol, which is distributed and scalable. In order to maintain a high energy efficiency and a low data transfer delay, the path selection is based on the evaluation of the node residual energy level and its neighbors' distance to the destination. The multipath routing protocol helps relaying data packets from the source to the sink over the newly discovered paths. It also allows the sink node to monitor the path conditions in order to make adjustments of traffic distribution in real-time.

2. **Load Balancing Algorithm:** We introduce the term "path cost" to reflect the cost of transmitting data with a unit rate through a path. It is obtained from the empirical measurements of the path, such as the residual and initial energy levels of nodes along the path and their hop distance to the destination. The load balancing algorithm is applied at the sink node to distribute the traffic over multiple paths based on their "path cost". The algorithm solves the optimization problem of traffic allocation to extend the network lifetime and maintain a reasonable packet delay.

1.3 Outline of the Thesis

The remainder of the thesis is organized as follows: In Chapter 2, we introduce some previous work on routing protocol for WSNs, especially the directed diffusion [7], the directed transmission [10], and the energy-aware routing [5]. We present our multipath routing protocol in Chapter 3, which includes the control messages and algorithms used for the search of multiple paths, the optimization method adopted to distribute the traffic over multiple paths, and the technique used to solve the optimization problem. In Chapter 4, we describe our experimental methodologies and the metrics used to evaluate the performance of our multipath protocol. We also compare our proposed scheme with other routing protocols. Numerical results obtained through extensive simulations are also presented in this chapter. Finally, we conclude the thesis in Chapter 5 and provide suggestions for future work.

Chapter 2

Related Work

In this chapter, we begin by reviewing several routing and data dissemination schemes for WSNs. We then describe some other work on multipath routing for mobile ad hoc and wireless sensor networks and discuss the design choices that are specific for multipath routing.

2.1 Routing and Data Dissemination Algorithms

In general, the routing techniques for ad-hoc networks and WSNs can be classified as *proactive*, *reactive*, and *hybrid*. The proactive routing maintains the route information to all destinations by periodic updating, which is expensive as network resources such as node energy and communication bandwidth are being consumed periodically. The reactive routing, on the other hand, retrieves routing information only when necessary. It saves considerably the network resources in comparison with proactive routing, by trading off the time required for route discovery. The hybrid routing uses the combination of the two previous categories to balance between the route discovery time and the conservation of network resources. The location-based routing constructs the routing entries based on the location of nodes. The coordinates of nodes may be obtained directly from the GPS (Global Positioning System) device attached or by exchanging messages with their neighbors. The distance information to the neighboring nodes is derived from the estimation of incoming signal strengths. In the following sub-sections, we describe some of the data dissemination schemes for WSNs proposed in the literature.

2.1.1 Directed Diffusion

The directed diffusion [7] is a data-centric routing scheme that is able to establish an energy-efficient data dissemination path between the source and the sink. It uses the in-network aggregation, which combines the data coming from different sources for the same target in order to save energy and prolong the network lifetime by eliminating the redundancy. The scheme incorporates localized algorithms to enable flexible path construction and recovery in case of node failures. Figure 2.1 gives a simplified version for the operation of the directed diffusion. The initial phase of the scheme is started with flooding of the network with interest messages from the sink node (Figure 2.1(a)). The interest message contains low-level abstraction to describe the application-specific information requested. As the interest is diffused hop-by-hop through the network, gradients are set up within the network to connect the source node with the sink (Figure 2.1(b)). Each gradient gives an attribute value to categorize the information and the direction to propagate the event data received as well. Each node in the network maintains an interest cache, which associates the gradient with distinct interests. The source node

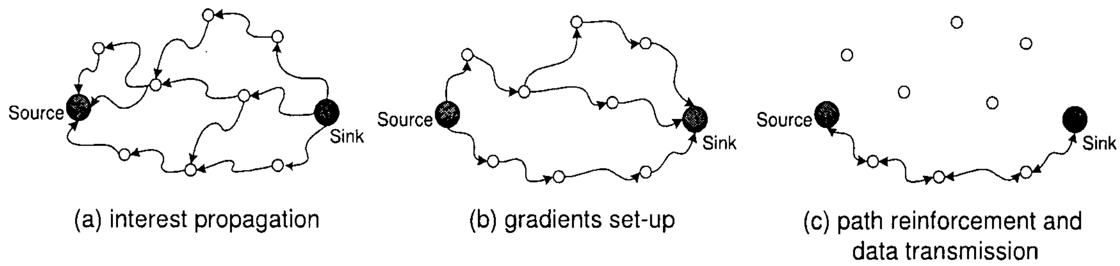

Figure 2.1: A simplified illustration of directed diffusion

injects periodically the exploratory data to the network at a low data rate. The hop-by-hop routing decision of the exploratory data is made based on the interest cache of each node. When the attribute of the exploratory data received matches the gradient, the exploratory data message is being forwarded. The sink, by collecting exploratory data arrived via different paths, sends a reinforcement message through one particular path to inform the source node to send data at a higher rate (Figure 2.1(c)). After the reinforced path is established, the sink still sends periodically its interest to the network to ensure that the interest cache is refreshed at each node. The path is maintained by the reinforcement message, which is injected by the sink node periodically as well. When the sink detects path degradation, it can send negative reinforcement message to the source in order to stop using it. As the sink does not have a complete view on the existing paths, this path switching mechanism may not be efficient in terms of resource utilization as the decision is made with empirical knowledge on neighboring nodes.

2.1.2 Directed Transmission

The directed transmission [10] is a probabilistic routing technique that is designed to be simple, localized and robust under the situation of node failure. The protocol requires an initialization phase, which allows each node in the network to be aware of its hop distances to different destinations. The retransmission probability function is the core of the protocol. When node i receives an event data from a source node, it calculates the retransmission probability P_i as follows:

$$P_i = e^{k[d(S,D)-d(i,D)-s]} \tag{2.1}$$

where $d(S, D)$ is the distance between the source S and the destination D, $d(i, D)$ is the distance between node i and the destination D, s is the number of steps that the data packet has traveled so far, and k is a variable used to tune the desired QoS level. If the retransmission probability is above a pre-defined threshold, the event data received is being forwarded. The rationale behind this function is that the nodes on the shortest path should have the highest probability to retransmit the data. The farther away a node locates from the shortest path, the smaller its retransmission probability should be.

2.1.3 Geographic Adaptive Fidelity (GAF)

GAF [27] is a location-based energy-aware routing protocol initially designed for mobile ad-hoc networks, and it can be applied to WSNs as well. The scheme divides the network into grids with equal size. The size of the grid is fixed according to the power and the communication direction of radio transmitter carried by sensor nodes. Each node in the network associates itself to a grid based on its coordinates in the initialization phase. The election process followed elects one node in each grid to stay awake for a certain period of time. This node takes the responsibility of monitoring the communication activities and relaying data received to its destination if necessary. In the meantime, other nodes in the grid go to the sleep state to conserve energy. The nodes that are turned off are scheduled to be awake in rotation to take over the monitoring and communication responsibilities of the current elected node. The philosophy of the GAF protocol behind the node scheduling is that all nodes in the same grid are considered equivalent in terms of the cost of data routing. Turning off unused nodes in the same grid will reduce considerably the energy consumption.

Similar to other location-based routing protocols, GAF requires each node to obtain its location information. The protocol maintains one active node in each grid to save energy. However, the data aggregation is not allowed during the relay of data from one grid to another.

2.1.4 Sensor Protocol for Information via Negotiation (SPIN)

SPIN [24][25] is a negotiation-based protocol. It tries to resolve the implosion problem of flooding technique, which leads to the propagation of duplicate data copies and wastes network energy and bandwidth. Another measure adopted by the protocol is to send only messages containing descriptive information before releasing the complete event data.

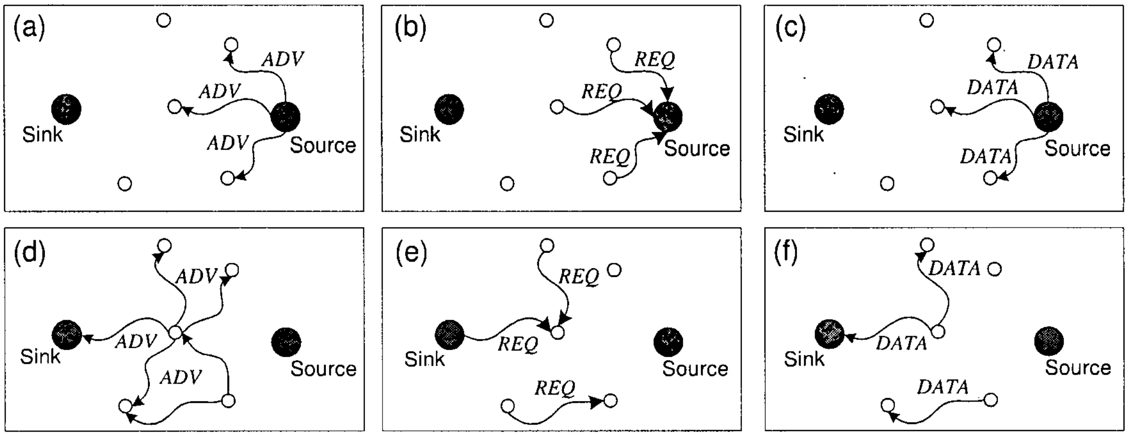

Figure 2.2: An example of a sensor network using SPIN protocol

The SPIN is a three-stage protocol as three types of messages are used, as illustrated by Figure 2.2. When the source node detects a stimulus and has event data for the sink, it first broadcasts to its neighbors with an ADV message (Figure 2.2(a)), which contains the descriptive information of the event data to be sent. The neighbors that are interested in the data will respond with an REQ message (Figure 2.2(b)). The actual event data is then sent back to them by the source through DATA message (Figure 2.2(c)). This process is repeated until the event data is received by the sink node (Figure 2.2(d)-(f)).

The SPIN-2 protocol is an extension of the original SPIN protocol described above. It includes a heuristic resource-awareness function, which reduces the level of participation when the energy supply of the node is below a fixed threshold. The node only participates when it is able to complete the data transmission process without having its energy reserve being dropped below the threshold.

The SPIN protocol and its extension are simple and localized, as each node only needs to have knowledge of its immediate neighbors. The topology change has little impact on the data transmission. However, the event data can be dropped if intermediate nodes do not have interest on the event data advertised. Another drawback of the protocol is that it cannot eliminate completely the redundant data copies in the network. As illustrated in Figure 2.2(d) - (f), multiple copies of the same event data are sent to different nodes along the way to the sink.

2.1.5 Energy-Aware Routing

The energy-aware routing proposed in [5] is a reactive routing protocol that has an objective to extend the lifetime of WSNs. The protocol starts with a setup phase, where the sink node injects the request message to the network. The localized flooding of request message helps to find all possible routes between the source and the sink and the energy costs associated to these paths. Each node makes a decision of forwarding the request message based on the following parameters: its distance and its neighbors' distance to the destination, and the energy metrics. Only neighbors that are closer to the source and farther away from the sink than the node itself are selected to forward the request message. To evaluate the energy metrics of possible paths, the node first computes the cost metric for links between itself and each of its neighbors. For link between node i and j, the cost metric C_{ij} is calculated as follows:

$$C_{ij} = e_{ij}^{\alpha} R_i^{\beta} \tag{2.2}$$

where e_{ij} is the transmission energy required for link between nodes i and j, R_i is the normalized residual energy level of node i, α and β are weighting factors to tune the cost metric. By including the cost metric with the accumulated path cost carried by the request message received, node N_i is able to evaluate $C_{N_iN_j}$, the cost of possible path through the neighbor N_j as follows:

$$C_{N_iN_j} = \text{Cost}(N_j) + C_{ij} \tag{2.3}$$

Only the neighbors on the low-cost paths are selected as the next hop and an entry for node N_j is added to the routing table of node N_i with the criteria below:

$$FT_i = \{j | C_{N_iN_j} \leq \alpha \min_k (C_{N_iN_k})\} \tag{2.4}$$

where α is the weighting factor and k is the number of neighbors. The route entry is assigned with a retransmission probability $P_{N_iN_j}$ to node N_j, which is inversely proportional to the cost and is computed as below:

$$P_{N_iN_j} = \frac{1/C_{N_iN_j}}{\sum_{k \in FT_i} 1/C_{N_iN_k}} \tag{2.5}$$

The accumulated path cost $Cost(N_i)$ at node N_i is then updated as follows:

$$Cost(N_i) = \sum_{j \in FT_i} P_{N_iN_j} C_{N_iN_j} \tag{2.6}$$

The request message is updated with the new path cost calculated and forwarded to the neighbor selected. In the data communication phase, the intermediate node generates a random probability. It retransmits the data to the neighbor, with its retransmission probability in the routing table equals to the probability obtained randomly.

The energy-aware routing maintains multiple paths and uses only one of them at a time, in order to avoid stressing a particular path and extend the network lifetime. A dead-lock situation may occur with data packet being transmitted back and forth between two neighboring nodes if they select each other as the next hop continuously. Such situation is costly in terms of energy consumption for the network. In addition, as the residual energy level of nodes varies along the time, we need to rebroadcast the request message through the network periodically in order to update the path costs, the routing table, and the retransmission probability associated with each entry in the table.

2.2 Multipath Routing

The multipath routing technique was initially used in wired networks [21][22][23] for its reliability and its ability to balance traffic load over the network. In recent years, such technique is extended to wireless ad hoc and sensor networks with objectives to achieve better energy efficiency and network robustness in case of node failures. In this section, we describe several algorithms in this area. We also discuss the specific design choices for multipath routing.

2.2.1 Multipath Routing Algorithm for Mobile Ad-Hoc and Wireless Sensor Networks

In [15] and [20], multipath extensions of Dynamic Source Routing (DSR) and Ad hoc On-demand Distance Vector (AODV) are proposed to improve the energy efficiency of ad hoc networks by reducing the frequency of route discovery. The protocols explore multiple link-disjoint routes with a single flooded query. When the primary route fails during the data communication phase, data traffic is switched to the shortest remaining alternate route. The new route discovery is re-initiated only when all the previous discovered paths are failed. An on-demand multipath routing protocol is proposed in [14], which is capable to find node-disjoint paths. This is achieved by comparing the node cache with the route information carried by the query message received. The query message is then forwarded selectively. The energy saving is achieved mainly by reducing the frequency of route discoveries as in [15] and [20]. The Meshed Multipath Routing (M-MPR) [28] aims to increase the traffic throughput of the WSNs. The multipath discovery process takes advantage of the geographical information known by the sensor nodes. In M-MPR, the sensor node restricts the number of routing queries to propagate and forwards query only to a limited number of downstream neighbors in order to avoid looping. The load balancing is achieved by selectively forwarding data to one of the downstream neighbors. The drawback of this scheme is that the hop-by-hop decision is not able to select optimal route because only partial network information is maintained by each node. The load balancing technique of the scheme does not actually forward data simultaneously over multiple disjoint paths.

In [16], a multipath routing approach is proposed to improve the directed diffusion [7] with an objective of increasing the resilience to node failures. Their work explores the possibility of rapidly finding alternate paths connecting source and sink nodes when isolated node failure occurs. They proposed localized algorithms for the construction of disjoint and braided multipaths. The braided paths share some common nodes among the paths and they are geographically close to the shortest path between the source and the sink. The advantage of using multipaths is to avoid periodic flooding of low-rate reinforcement messages which are necessary to maintain alternate paths in the original directed diffusion [7]. They conclude that the braided multipaths perform better in terms of maintenance overhead, especially for nodes in low densities. They demonstrate that the multipath approach can improve the robustness of the network without sacrificing the energy efficiency.

Our approach differs from the work described previously in two aspects. First, we focus on proposing a localized algorithm that discovers alternate disjoint paths and collects empirical measurements on path conditions at the same time. We aim to reduce the control message overhead by reducing the frequency of route discovery in order to minimize the energy consumption of our protocol. Second, we consider a mechanism that allows sources to transmit simultaneously event data generated through multiple paths discovered. The deliveries of event data through disjoint paths can help to make the energy consumption more uniform across the network and to increase the network lifetime.

2.2.2 Design Choices for Multipath Routing

As discussed in [14] and [16], an important design issue for multipath routing is whether to use disjoint paths or not. Each option has its advantages and drawbacks. The protocol adopts the disjoint paths do not have any bottleneck, as data is transmitted along different paths. But the hop distances of paths found may not necessary be close to the shortest path. An upper bound of path length is required for the route selection. Also, the node density has an impact on the number of disjoint paths that can be found in the network. The node connectivity level has to be high enough to allow multiple disjoint paths to exist. The braided paths, on the other hand, are not influenced by the node density. The route search algorithm is simpler, as the sharing of one or more common nodes is allowed. However, the drawback is the possible traffic congestion at these common nodes. A scheduling scheme is required to handle the traffic at the crossing point of different paths, which increases the complexity of the routing scheme and the packet delay.

The work in [13] shows that disjoint routes can make the load balancing more effective. It is noticed that when the node density is low, the probability of having multiple disjoint paths drops. But in the case of WSNs, nodes are normally deployed in large numbers in order to acquire the ability to monitor the sensor field efficiently and to provide redundancy. It is therefore likely to discover multiple disjoint paths connecting a pair of sink and source nodes in WSNs. Several protocols have been proposed to find disjoint paths in mobile ad hoc networks and wireless sensor networks. In [15], a multipath version of DSR is proposed. It has an advantage of early detection of routing loop. But the routing information accumulated in the query message requires extra transmission resources when the network size increases. In [16], multiple reinforcements are used to discover disjoint paths. But the route discovery process is time consuming since the disjoint routes are reinforced one by one after the shortest path is discovered.

In our multipath routing protocol, we adopt the approach of using disjoint paths as node density in WSNs is high enough. The choice also makes the protocol simpler since the traffic scheduling algorithm is not necessary. Furthermore, the disjoint paths allow us to include a load balancing algorithm in our algorithm. To overcome the drawback of the variation of path lengths mentioned earlier, we propose the use of timers during the route discovery phase to eliminate the paths with large delays.

2.3 Discussion and Summary

In this chapter, we described some of the previous work of routing protocols for WSNs, particularly the directed diffusion [7], the directed transmission [10], and the energy-aware routing [5]. We also described the advantages and drawbacks of these routing schemes. We summarized their characteristics in Table 2.1. Our multipath routing protocol differs from the existing data dissemination schemes by the fact that not only multiple paths are discovered in the setup phase, but they are also used simultaneously to relay data packets from the source to the sink.

Name	*Description*	*Remark*	*Network structure*	*Location based*
Directed Diffusion [7]	Data-centric routing with gradients established with periodic flooding of interests, and the data is transmitted through reinforced paths.	Application-oriented with data named by attribute value, possible in-network data aggregation.	Flat	No
Directed Transmission [10]	Retransmission probability from hop-to-hop is based on the distance to the destination. Nodes that are on the shortest path have a higher retransmission probability.	Initialization phase is required to allow each node to acknowledge its distance to the destination.	Flat	No
GAF [27]	Network is divided into grids with only one node elected to be awake to support the data transmission in a period of time. Other nodes awake in rotation to take the responsibility.	The scheduling algorithm is in place to keep one node active in the grid.	Hierarchical	Yes

Name	*Description*	*Remark*	*Network structure*	*Location based*
SPIN [24][25]	Node with sensing data broadcasts an advertisement message. Nodes which require data will then send a request message to establish the path.	Highly localized and insensitive to topology change.	Flat	No
Energy-Aware routing [5]	The propagation of interest message helps to find all routes and their energy costs. The retransmission decision is made based on the energy costs calculated earlier.	Multiple paths are maintained from the source to the destination with only one of them is being used.	Flat	Yes

Table 2.1: A comparison of some data dissemination schemes for WSNs

In the second part of this chapter, we described the work on multipath routing for mobile ad-hoc and wireless sensor networks, such as [14], [16], [15], and [20]. These schemes mainly focus on improving the network reliability with multipaths. Our work, on the other hand, explores the possibility of data transmission with multiple routes discovered. We also discussed the specific design choice for multipath routing schemes, i.e., whether or not to use disjoint paths. We decided to use disjoint paths in our protocol, for it eliminates the necessity of having a traffic scheduling algorithm. Such a choice also allows us to take full advantage of the load balancing algorithm.

Chapter 3

Multipath Routing for Wireless Sensor Networks

In this chapter, we present our proposed multipath routing algorithm for wireless sensor networks. In Section 3.1, we introduce the assumptions we made, the related definitions, and the system model. We describe the details of our multiple search protocol in Section 3.2. In Section 3.3, we present our method to balance the traffic among multiple paths discovered.

3.1 Assumptions, Definitions, and System Model

We consider that M identical wireless sensor nodes are distributed randomly in a field. Each sensor node carries a radio transmitter, which has a fixed transmission range of $\Re$. We assume that the network is connected and dense. That is, given an arbitrary pair of nodes, data can be sent from one to another in a multi-hop manner. There exists multiple paths between a pair of nodes. We further assume that each sensor node is stationary and contains an internal battery to support its sensing and communication activities.

This battery can neither be replaced nor recharged. Furthermore, the transmitter power of the node is fixed for both data transmission and reception. At any time, a sensor node m, $m \in 1, 2, \ldots, M$, is able to acquire the residual energy level $\grave{e}_{m,residual}$ of its battery.

When a stimulus is detected (or an event occurs), the surrounding nodes first exchange the information and select one of them to be the source node. The source node has the responsibility to aggregate data from the neighboring nodes and to transmit the aggregated data to the sink node. When different events occur in different regions within the coverage area, data from different source nodes are not being aggregated along the path to the sink node.

We define a *path*, which consists of K nodes, where $K < M$, as a group of nodes that relay the data generated from the source node x to the sink node y. Since we assume that the network is dense, it is possible to have multiple routes between the source node x and the sink node y. In this case, it is possible to use multipath routing instead of single path routing. We assume that the multiple paths used are disjoint. That is, the path A, which consists of K nodes, and the path B, which consists of L nodes, are two groups mutually exclusive except for the source node x and the sink node y. We define a *link* as an abstract representation of a radio connection established between two neighboring sensor nodes. A path A with K nodes therefore contains $(K - 1)$ links.

The link cost function is used by the node to select the next hop during the path search phase. Let N_a denote the neighbor set of node a, the sensor node a will choose the next hop by following the criteria defined below:

$$\text{Next hop } = \arg\min_{b \in N_a}\{(1 - e_{b,residual}/e_{b,init})^{[\beta(1-\frac{(\Delta d+1)}{d_{ay}})]}\} \tag{3.1}$$

where d_{ay} is the distance in hops between node a and the sink node y; d_{by} is the distance in hops between node b and the sink node y; Δd is the difference between d_{ay} and d_{by}; $e_{b,init}$ is the initial energy level of node b; $e_{b,residual}$ is the residual energy level of node b; and β is the weight factor and $\beta > 1$. Note that $(\Delta d + 1) \subset \{0, 1, 2\}$ and $(1 - e_{b,residual}/e_{b,init}) \in [0, 1]$. The link cost function takes both the node energy level and hop distance into account. Suppose $e_{b,residual}$ remains constant. In this case, the link cost increases when $(\Delta d + 1)$ increases. On the other hand, suppose $(\Delta d + 1)$ remains constant. In this case, the link cost increases as $e_{b,residual}$ decreases. In addition, the weight factor β adjusts the priority in the evaluation of link cost. A large β gives more weight to the node energy than to the hop distance. Figure 3.1 illustrates the impact on the evaluation of link price when β varies.

The link cost is used as one of the selection criteria for the selection of next hop in our multipath protocol which will be presented in Section 3.2. In the link cost function, we only consider the energy level of the receiver node b as it consumes energy for data reception and transmission if it is selected for forwarding. We do not take into account the energy level of node a, which is the sender in the equation (3.1). This is because no matter which node is selected as the next hop, node a still needs to spend the same amount of energy on data transmission.

For a path A, which consists of K nodes, the path cost P_A is the sum of individual

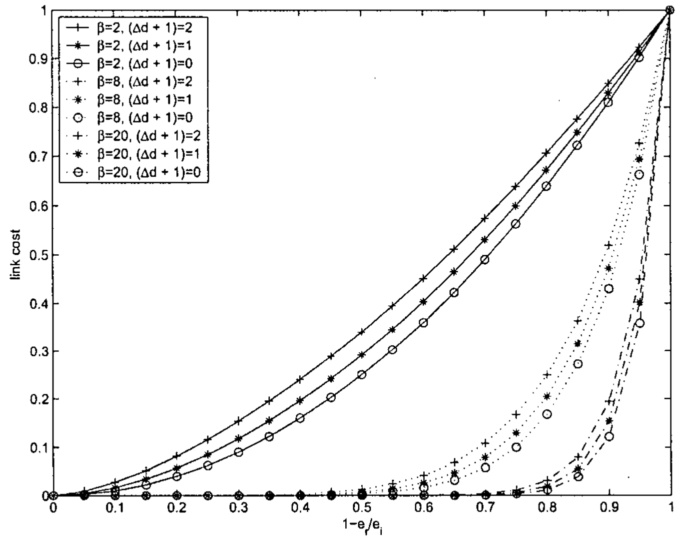

Figure 3.1: The evaluation of link price with $d_{ay} = 9$

link costs $l_{i(i+1)}$ along the path. That is:

$$P_A = l_{12} + l_{23} + \cdots + l_{K(K+1)} = \sum_{i=1}^{K-1} l_{i(i+1)} \tag{3.2}$$

The path cost P_A is used by our load balancing algorithm to allocate the data rate r_A through the path A. We will present the details on how it is integrated on our algorithm in Sections 3.2 and 3.3.

3.2 Multipath Routing Protocol

The multipath routing protocol is used to find multiple disjoint paths between a pair of sink and source nodes. It has three phases, the *initialization* phase, the *paths search* phase, and the *data transmission and maintenance* phase.

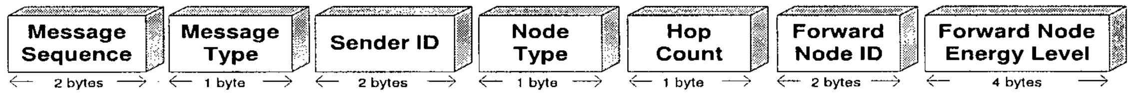

Figure 3.2: The format of a HELLO message

The initialization phase takes place after all sensor nodes are deployed in the target field. This phase has two objectives. First, the localized flooding of HELLO message allows all nodes to be aware of the status of their immediate neighbors. Second, the selective flooding of HELLO messages from sink nodes gives opportunities for each node to calculate its shortest distance to the sinks. Further details of the initialization phase is given in Section 3.2.1. The paths search phase follows next and it helps constructing multiple disjoint paths. We will introduce different control messages in Section 3.2.2. In Section 3.2.3, we describe how the data is transmitted and how the path failures are being handled by our protocol.

3.2.1 Initialization Phase

The HELLO message is one of the control messages exchanged between nodes in the initialization phase. Figure 3.2 shows different fields within a HELLO message. The first field *message sequence* is a number generated by the message originator. The number is incremented whenever a new message is created. It is reset to 1 whenever the maximum 65535 is reached, because the field size is 2 bytes. Combined with the node ID, it is possible to verify if the message has been received. The field *message type* carries information that it is a HELLO message. The field *sender ID* contains the node ID of the message originator. The *node type* field indicates whether the message originator is a sink, a source, or a regular sensor node. The *hop count* gives the hop distance of the message that has been passed from its originator. The *forward node ID* contains the ID of the upstream node, which forwarded the message at the last hop. Finally, the *forward node energy level* field gives the normalized node energy level of the node that forwarded the message at the last hop.

When the HELLO message arrives and if the message is received for the first time, each node will update its neighboring node table with the *forward node ID* and *forward node energy level.* Next, the node verifies if the *node type* is set to be *SINK*. In such case, the *sender ID* is compared with the sink list of the node. A new entry is created in the sink table if necessary, with the hop distance updated only when it is smaller than the value recorded. Finally, the HELLO message from the sink node is re-broadcast with the fields *hop count, forward node ID* and *forward node energy level* updated. Algorithm 1 gives the detail steps to process the HELLO message in a sensor node.

Algorithm 1 Algorithm to process the HELLO message

1: **Set** *tabH*: hash table of messages, *tabN*: table of neighbors,
tabS: table of sinks

2: **Input** seq : message sequence, sID : sender ID,
sT : sender type, h : hop count,
$fwdID$: forward node ID, $fwdE$: forward node energy

```
IF (seq, sID) exists in tabH
    RETURN
IF fwdID exists in tabN
    update the entry (fwdID, fwdE) in tabN
ELSE
    create new entry (fwdID, fwdE) in tabN
IF (sT == SINK)
  IF (sID exists in tabS)
    IF (h < tabS.sID.h)
      tabS.sID.h = h
    ELSE
      create new entry(sID, h) in tabS
  h = h + 1; fwdID = current node ID;
  fwdE = current node normalized energy level
  broadcast HELLO message to the neighbors
RETURN
```

Figure 3.3: Algorithm to process the HELLO message

As we can observe, the selective flooding of HELLO messages from sinks helps each node to acknowledge the existence of the sink nodes and to calculate the shortest hop distance to each sink node. At the end of the initialization phase, each node will have

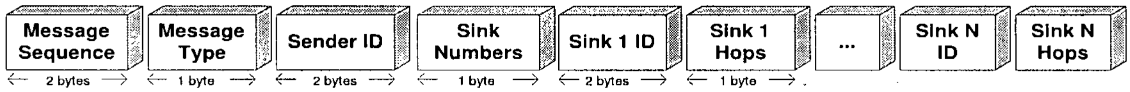

Figure 3.4: The format of a CONNECTIVITY message

the sink table and the neighboring node table updated. Each node then broadcasts a CONNECTIVITY message to its immediate neighbors. Figure 3.4 shows the structure of the CONNECTIVITY message.

In the message, except those same fields that we have already introduced for the HELLO message, the field *sink numbers* specifies the number of sinks that the sender is aware of. The subsequent fields give in order the sink IDs and the hop distance to each of them. The receiving node will update the corresponding entry in its neighboring node table.

3.2.2 Paths Search Phase

The paths search phase is initiated when a set of nodes detect the stimulus and the selected source node begins to send the aggregated data to the sink node. Since we need to explore multiple disjoint paths, the source node unicasts one REQUEST message to every neighboring node with a distinct route ID. As shown in Figure 3.5, not all REQUEST messages will arrive to the sink node. Some of them will be dropped by the intermediate nodes in order to avoid having paths that share common nodes.

Figure 3.6 shows the format of a REQUEST message. The fields *source ID* and *sink ID* indicate the node ID of the source and sink, respectively. The *route ID* is assigned by

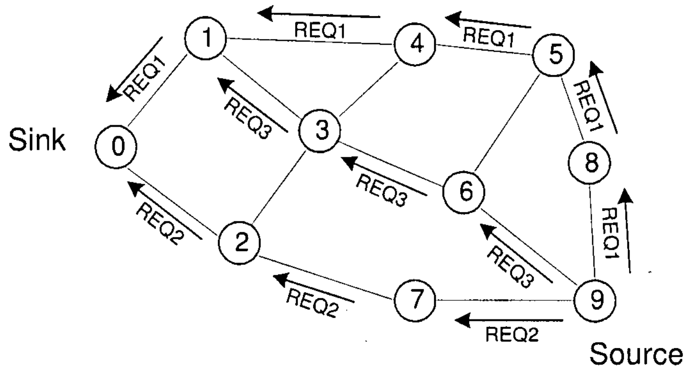

Figure 3.5: An example of path search

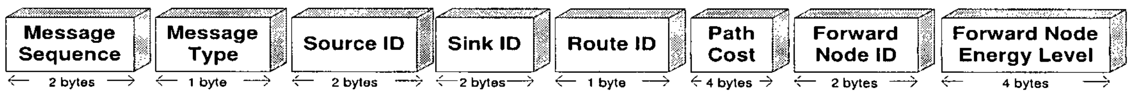

Figure 3.6: The format of a REQUEST message

the source node to distinguish between different routes that lead to the same sink node. The *path cost* field stores the accumulated path cost, starting from the source node. The rest of the fields carry the same information as in other control messages introduced previously.

Upon reception of the REQUEST message, a regular node (i.e., an intermediate node) examines its routing table with the values in fields *source ID* and *sink ID* and creates a new entry if necessary. If the sink node indicated by *sink ID* is in the neighboring node table, the routing table is updated and the REQUEST message is forwarded to the sink node directly with fields *forward ID* and *forward node energy level* updated. Otherwise, the node has to select one of the neighbors to forward the REQUEST message.

The selection is based on two criteria. First, the neighboring node should not have been selected for another path that connects the same pair of sink and source nodes. Second, the link cost to the selected neighbor has to be the lowest among all the available neighbors.

As illustrated in Figure 3.5, node 4 forwards the message REQ1 to node 1 rather than to node 3 as the link cost through node 1 is lower. The message REQ3 is dropped by node 1 as all its neighbors have already been selected by another path.

The link cost is defined in equation (3.1). The routing table will be updated if a neighbor is selected. The table of neighbors is updated at the same time. In future path search, the node will avoid to select the neighbor that has already been used for the path that connects the same pair of sink and source nodes. Finally, the node will update the fields *path cost*, *forward node ID* and *forward node energy level* before sending the REQUEST message to the neighbor selected. If none of the neighbors satisfies the conditions, the REQUEST message will simply be dropped. The Algorithm 2 gives the detailed steps to process the REQUEST message by a regular intermediate sensor node.

Algorithm 2 Algorithm to process the REQUEST message by a regular intermediate node

---

1: **Set** *tabN*: table of neighbors,
tabR: routing table, *nodeS*: neighboring node selected as the next hop,
minLP: the minimum link cost

```
Input seq : message sequence, srcID : source ID, skID : sink ID,
        p : path cost, rID : route ID,
        fwdID : forward node ID, fwdE : forward node energy
Initialize minLP = 0xFFFF, nodeS = NULL
IF (srcID, skID) does not exist in tabR
    create new entry (srcID, skID) in tabR
IF (skID exists in tabN)
    update tabR with (srcID, skID, rID)
    nodeS = skID
ELSE
    WHILE (not reach the end of tabN)
        IF (the neighbor node x has not been selected for the path that
          connects srcID and skID)
            LP = link cost to the neighbor node x
            IF (LP < minLP)
                nodeS = x
                minLP = LP
            ELSE
                point to the next neighboring node in tabN
IF (nodeS != NULL)
    fwdID = current node ID;
```

```
        fwdE = current node normalized energy level
        p = p + minLP
        send REQUEST message to nodeS
    update the entry (fwdID, fwdE) in tabN
    RETURN
```

Figure 3.7: Algorithm to process the REQUEST message by a regular intermediate node

Algorithm 3 Algorithm to process the REQUEST message at sink node

```
    Set tabR: routing table, tabS: table of source nodes
    Input seq : message sequence, srcID : source ID, skID : sink ID,
            p : path cost, rID : route ID, fwdID : forward node ID
    IF (srcID, skID) does not exist in tabS
        add srcID to tabS
    IF ((srcID, rID) does not exist in tabR)
        add entry (srcID, rID, fwdID, p) to tabR
    RETURN
```

Figure 3.8: Algorithm to process the REQUEST message at sink node

For the sink node, the received REQUEST message is processed differently. It first examines the *source ID* and creates a new entry in its source table if it is not known. It then updates the routing table with the information carried in the message. Algorithm 3

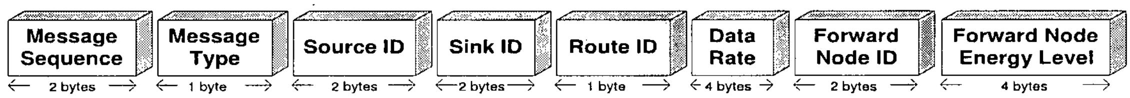

Figure 3.9: The format of an ASSIGN message

presents the pseudo-code of how the REQUEST message is being processed by the sink node.

The sink node starts a request timer when it receives the first REQUEST message from a source node. The REQUEST messages arrive after the timer expires will simply be dropped. Such measure allows the path exploration to be done within a reasonable period of time, as REQUEST messages that arrive late will include only paths with undesirable qualities (e.g., large delays and extra network resources). When the request timer expires, the sink node begins to allocate traffic to each of the path discovered. Different data rates are assigned to these paths depending on their path cost. We will present the algorithm used for traffic allocation in Section 3.3. The sink node then sends the ASSIGN messages to the source node via each of the selected multipath. Figure 3.9 shows the structure of the ASSIGN message.

In the ASSIGN message, the field *data rate* indicates the data transmission rate assigned for the path that is specified by *route ID*. When an intermediate node receives the ASSIGN message, it searches its routing table for the entry that matches *source ID*, *sink ID* and *route ID* values. It then forwards the message to the next hop after updating the fields *forward node ID* and *forward node energy level*. The source node behaves differently when it receives the ASSIGN message. It first finds the entry specified by *sink ID* and *route ID* from its routing table. The entry is then updated with the *data rate* carried in the ASSIGN message. Algorithm 4 describes the pseudo-code of how the ASSIGN message is being processed.

Algorithm 4 Algorithm to process the ASSIGN message

Set *tabR*: routing table, *tabN*: table of neighbors
Input seq : message sequence, $srcID$: source ID, $skID$: sink ID
        rID : route ID, $rate$: data rate, $fwdID$: forward node ID
update the entry ($fwdID$, $fwdE$) in *tabN*
IF (current node ID == $srcID$)
    search for entry ($srcID$, $skID$, rID) from *tabR*
    update the entry found with *rate*
ELSE
    $fwdID$ = current node ID;
    $fwdE$ = current node normalized energy level
    search for next hop from *tabR*
    unicast the message to the next hop
RETURN

Figure 3.10: Algorithm to process the ASSIGN message

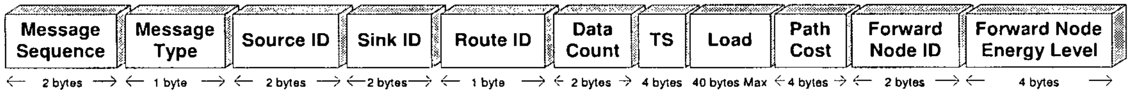

Figure 3.11: The format of a DATA message

3.2.3 Data Transmission and Paths Maintenance Phase

After multiple paths are discovered, the source node begins to transmit data packets with the assigned rates on each path. The DATA message carries the event data and has the fields as shown in Figure 3.11.

The DATA message has some specific fields. The field *data count* has the value of the data counter in the source node at the time when the DATA message that related to a stimulus detected is generated. The data counter increments continuously and it resets to 0 when the maximum is reached or the event data of a new stimulus is generated. The sink node can differentiate the event data from the same source node but related to distinct stimulus with the value of the field *data count.* The field *TS* carries the timestamp, which corresponds to the time when the DATA message is created at the source node. It allows the sink node to monitor the overall packet transfer delay. Finally, the field *Load* contains the actual event data from the source node and the field *path cost* gives the accumulated path cost.

At each hop, the node can determine the next step by searching its routing table with the information carried in the DATA message, such as *source ID*, *sink ID* and *route ID*. The fields *path cost*, *forward node ID* and *forward node energy level* in the DATA message are updated before the message is being forwarded.

At the sink node, it updates the path cost in its routing table each time a DATA message arrives. The updated values help the sink node to monitor closely the conditions of the multiple paths being used. The initial data rate assignments for the paths may not be optimal for the whole duration of the connection. Usually, the path with the lowest cost is more likely to be assigned with the highest data rate initially and the nodes on that path will dissipate energy at a faster rate. Its path cost will gradually lose "competivity" compared with other paths. The sink node has to redistribute the data rates over paths to optimize the usage of network resources. The re-distribution is triggered when the original route with the lowest cost has its path cost increased to a pre-determined threshold. The sink node will then adjust the traffic flows and notify the source node with the ASSIGN messages.

In order to detect the path failure, the sink also monitors the inter-arrival delay of data packets on each path. When the delay is above a pre-determined threshold, the sink presumes that the path is broken. If the number of current working paths is equal to or lower than two, the sink will send a RESET message to the source through the optimal path to re-initiate the paths search phase. Otherwise, the sink re-adjusts the data rate allocation over other functional routes. This mechanism can avoid having the path search phase being invoked frequently.

3.3 Load Balancing Algorithm

We assume that there exists N disjoint paths between a source node x and a sink node y. The requested data rate to be arrived at the sink node y via all these multipaths is R bits/sec. Let r_j be the data rate allocated to path j, we have

$$\sum_{j=1}^{N} r_j = R, \text{ where } r_j \geq 0 \tag{3.3}$$

For a path j, the product of the path cost p_j and the data rate allocated r_j gives the path cost rate c_j. The overall system cost C to transmit data with rate R between a sink node and a source node can be expressed:

$$C = \sum_{j=1}^{N} c_j = \sum_{j=1}^{N} r_j p_j \qquad \text{where } r_j \geq 0 \tag{3.4}$$

3.3.1 Optimize the Traffic for Multiple Paths

As we intend to improve the network energy efficiency through load balancing over multiple paths, we adopt the *Chebyshev Sum Equality* in our algorithm to measure how well the transmission cost is balanced. The *Chebyshev Sum Equality* is defined as follows:

For two sets of distribution $\bar{a}$ and $\bar{b}$, where $\bar{a} = (a_1, a_2, \ldots, a_n), \bar{b} = (b_1, b_2, \ldots, b_n)$,

if $a_1 \geq a_2 \geq \cdots \geq a_n$, and $b_1 \geq b_2 \geq \cdots \geq b_n$,

then

$$n \sum_{k=1}^{n} a_k b_k \geq (\sum_{k=1}^{n} a_k)(\sum_{k=1}^{n} b_k) \tag{3.5}$$

We use the following *load balance ratio* Φ to evaluate the level of load balancing over different multipaths:

$$\Phi(\bar{r}) = \frac{(\sum_{j=1}^{N} r_j p_j)^2}{N \sum_{j=1}^{N} (r_j p_j)^2} \tag{3.6}$$

where the vector $\bar{r}$ denotes the traffic rates allocated to all available routes and r_j is the traffic flow allocated to path j.

The *load balance ratio* presented by equation (3.6) reaches its global maximum of 1 under the condition that the traffic is perfectly balanced. This is a known property of the *Chebyshev Sum Equality*. The concavity of $\Phi(\bar{r})$ can be determined by proving its Hessian matrix is negative semidefinite. Such proof is a sufficient condition, but not a necessary condition. In all our experimental results, we reach the local maximum that is equal to the global maximum of 1. We can conclude safely that the equation (3.6) is "likely to be concave".

In summary, we can convert our traffic allocation problem to an optimization problem that is formulated as follows:

$$\begin{aligned} &\text{Minimize } f(\bar{r}) = -\Phi(\bar{r}) = -\frac{(\sum_{j=1}^{N} r_j p_j)^2}{N \sum_{j=1}^{N} (r_j p_j)^2} \\ &\text{subject to } h(\bar{r}) = R - \sum_{j=1}^{N} r_j = 0, \quad \text{where } r_j \geq 0 \end{aligned} \tag{3.7}$$

3.3.2 Solve the Optimization Problem

Our optimization problem, presented by equation (3.7), is an equality-constrained non-linear problem. The method of *reduced gradient* is one of classical methods to solve this type of optimization problem. Starting from a feasible solution, the method continuously improves the feasible direction $\bar{d}$ until $\bar{d} = 0$. In [18], it is proved that $\bar{d} = 0$ if and only if the solution satisfies the Karush-Kuhn-Tucker (KKT) conditions.

The Algorithm 5 gives the detailed steps of the *Reduced Gradient* method that is used to solve our optimization problem.

Algorithm 5 Algorithm of optimal rates search with Reduced Gradient Method

---

Set $\bar{p}$: the table of path costs, R: the total data rate,
    $\bar{r}$: the table of path rates, I: the table index of $\max\{\bar{r}\}$,
    $\bar{A}$: the table of coefficients of the constraint function $h(\bar{r})$,
    B: the constraint coefficient of basic vector,
    N: the table of constraint coefficients for non-basic vectors,
    $\bar{d}$: the table of search direction,
    d_B: the search direction for basic vector,
    d_N: the search direction for non-basic vectors,
    $d\bar{f}$: table of first partial derivatives of $\bar{r}$,
    df_B: the first partial derivative of r_I,
    $\bar{k}$: the table of reduced gradients, λ: the step size
Initialize $\bar{A} = 1$, $r[1] = R$ and $r[j] = 0$ for $j \neq I$, $\bar{d} = 1$
WHILE ($\bar{d}$!= 0)
    set I to be the index of $\max\{\bar{r}\}$
    set $B = A[I]$, $N = \bar{A}$ excludes B
    evaluate $d\bar{f}$, $df_B = d\bar{f}[I]$
    $\bar{k} = d\bar{f} - df_B B^{-1} A$

8: construct d_N:

IF (k[i] ≤ 0)

d_N[i] $= -k$[i]

ELSE

d_N[i] $= -r$[i]k[i]

9: $d_B = -B^{-1} N d_N$

10: $d = (d_B, d_N)$

11: construct table $range_\lambda$:

IF d[i] < 0

$range_\lambda$[i] = 0xFFFF

ELSE

$range_\lambda$[i] $= \frac{-r[i]}{d[i]}$

12: $\lambda_{max} = \min\{range_\lambda\}$

13: do line search to minimize $f(\bar{r} + \lambda\bar{d})$, with $0 \leq \lambda \leq \lambda_{max}$

14: $\bar{r} = \bar{r} + \lambda\bar{d}$

15: RETURN

Figure 3.12: Algorithm of optimal rates search with Reduced Gradient Method

The step 13 of our optimal rate search algorithm in Figure 3.12 requires a line search to find λ, which is the step size to increment at each iteration. We adopt the *golden section* method, which is an efficient search technique without using derivatives. The algorithm finds the extremum by reducing continuously the interval of uncertainty until it reaches a pre-determined level of tolerance. In Algorithm 6, we present the details on this line search algorithm.

Algorithm 6 Algorithm of step size search

Set a, b: the lower and upper bounds of uncertainty interval,
    μ, ν: the lower and upper intermediate points,
    δ: the stop criteria,
    f_μ, f_ν: the object function values at μ and ν,
Input λ_{max}: the upper bound of the search interval,
    $\bar{d}$: the table of search direction,
    $\bar{r}$: the table of path rates, $\bar{p}$: the table of path costs
Initialize $a = 0$, $b = \lambda_{max}$, $\delta = 0.001$
    $\mu = a + (1 - 0.618)(b - a)$,
    $\nu = a + 0.618(b - a)$
WHILE $(b - a) > \delta$
    calculate $f_\mu(\bar{r} + \mu\bar{d})$ and $f_\nu(\bar{r} + \nu\bar{d})$
    IF $f_\mu > f_\nu$
        $a = \mu$
        $\mu = \nu$
        $\nu = a + 0.618(b - a)$

```
    ELSE
        b = ν
        ν = μ
        μ = a + (1 − 0.618)(b − a)
RETURN a
```

Figure 3.13: Algorithm of step size search

3.4 Summary

In this chapter, we proposed a multipath routing algorithm for wireless sensor networks. We first presented our multipath search protocol, with the focus on different types of control messages and associated algorithms to process them. In the literature, many protocols work in a distributed manner. We consider it to be an important characteristic to make the system robust, which is essential for many mission-critical applications of wireless sensor networks. Our multipath search protocol follows the same philosophy. Each node makes decisions independently during the path search phase based on the link costs to each of its neighbors. In the second part of the chapter, we presented the load balancing algorithm. We first defined the *load balance ratio*, which is derived from the *Chebyshev Sum Equality*. We then demonstrated that the traffic distribution problem can be considered as an optimization problem. We adopted the *reduce gradient* method to solve our optimization problem, which is an equality-constrained non-linear problem.

Chapter 4

Performance Comparisons

In this chapter, we present the results for the performance of our proposed multipath routing protocol. We use a discrete-event data simulator to emulate the performance of the multipath routing protocol proposed under various traffic conditions and topology settings. We obtain the results including the energy consumption, network lifetime, data transfer delay, and the control message overhead. We also compare our proposed multipath routing protocol with the following protocols:

- Directed Diffusion [7]
- Energy-Aware Routing [5]
- Directed Transmission [10]
- Flooding

In Section 4.1, we describe the simulation methodologies and various performance metrics. In Section 4.2, we present the simulation results of different protocols for network lifetime and average node energy consumptions. We also present the results for the control message overhead, average data transfer delay, and node average energy level. A summary is given in Section 4.3.

4.1 Simulation Model

Our multipath routing protocol is implemented in the *ns-2* network simulator. In all our simulations, we consider a square sensor field of size L. Inside the field, M static sensor nodes are deployed randomly. The value of M is varied from 50 to 250. Each node has a fixed radio range of 40 meters. The node density is maintained at a constant level of $50/160^2$ nodes/m^2. The positions of the source and sink nodes are shown in Figure 4.1. Figure 4.1 also shows the multiple paths determined by the multipath routing protocol with 250 nodes for each topology setting. In these configurations, the sinks and sources are located far from each other. The minimum distance between any pair of sink and source is larger than $L/2$. Such settings facilitate our evaluation of the protocol where the routing path has to traverse a large area in the sensor field. We also assume that the source nodes detect different stimulus. Thus, their event data cannot be aggregated.

The data packet size is 64 bytes. We use an event-driven wireless sensor network in our experiments. After the route search phase, each source node generates data packets and sends them to the sinks through the network with a fixed rate. We experiment with different traffic conditions by altering the data transmission rate required by the sink nodes. The rates include: 1 packet per second, 2 packets per second, and 5 packets per second respectively. For the link cost evaluation in our simulations, we use $\beta = 20$ in equation (3.1).

We adopt the *ns-2* radio energy model and assign each node with the same initial energy level of 10 J at the beginning of each simulation in order to keep the simulation

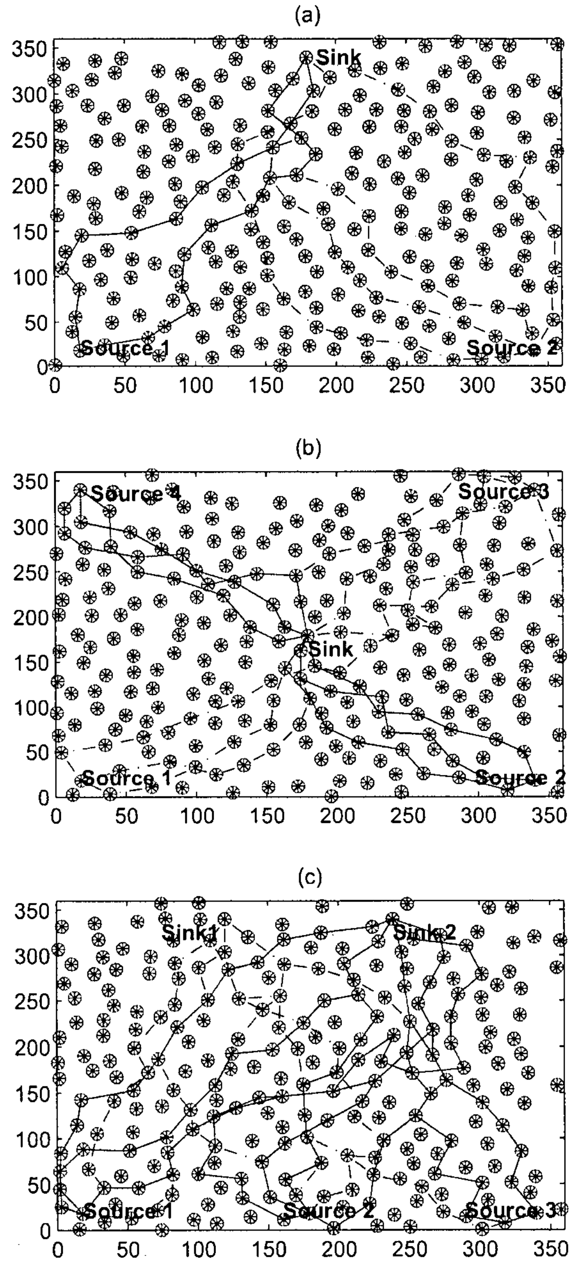

Figure 4.1: Configurations of sink and source nodes and examples of paths discovered in the simulations (a) Topology setting 1: 1 sink and 2 sources (b) Topology setting 2: 1 sink and 4 sources (3) Topology setting 3: 2 sinks and 3 sources.

Item	*Value*
Node density	$50/160^2$
Number of nodes	50, 100, 150, 200, 250
Data packet size	64 bytes
Control packet size	32 bytes
Idle power	35 mW
Receive power	395 mW
Transmit power	660 mW
Node initial energy	10 J
Node radio range	40 m
MAC protocol	IEEE 802.11 (CSMA/CA)
Bandwidth (802.11)	1.6 $Mbps$

Table 4.1: Simulation parameters

time within a reasonable time period. We further assume that each sensor node carries an omni antenna and the energy consumptions for idle time, transmission and reception are 35 mW, 660 mW, and 395 mW respectively. The energy dissipation for data processing in the node is neglected in our simulations. We adopt the IEEE 802.11 MAC layer provided in the *ns-2* with a bandwidth of 1.6 $Mbps$. Every 500 ms, we obtain the log of the energy level of each node. This allows us to trace the status of energy consumption of the network. In Table 4.1, we summarize the simulation parameters.

We use a number of metrics to evaluate the performance of our protocol and compare with other existing schemes. The *network lifetime* measures how long the network can sustain the data transmission from the source nodes to the sink nodes. It is obtained by calculating the average interval between the first and last data packet arrivals at each sink node. The *node energy consumption* measures the average energy dissipated by the node in order to transmit a data packet from the source to the sink. The same metric is used in the work on directed diffusion [7][8] to indicate the energy efficiency level of WSNs. It is calculated as follows:

$$node\ energy\ consumption \ = \frac{\sum_{i=1}^{M}(e_{i,init} - e_{i,res})}{M\sum_{j=1}^{S} dataN_j} \tag{4.1}$$

where M is the number of nodes in the network, $e_{i,init}$ and $e_{i,res}$ are respectively the initial and residual energy levels of node i, S is the number of sink nodes and $dataN_j$ is the number of data packets received by sink j. The *average delay* measures the average time spent to relay data packets from the source node to the sink node.

The *average node energy* measures the average energy level of all nodes in the network after the data transmission has been started for a certain amount of time. It gives an indication of the network state in terms of energy consumption. For the data rate of 2 or 5 packets per second, we measure the *average node energy* with a delay of 25 seconds. With the data rate of 1 packet per second, the delay is increased to 50 seconds to ensure that approximatively the same amount of event data has been handled by the network when taking the measurements. Finally, we compute the *control message overhead*, which counts the average amount of control messages received and transmitted by each node in bytes. It evaluates the extra workload required to sustain the data routing for various schemes.

4.2 Results and Discussions

4.2.1 Network Lifetime

We first determine the *network lifetime* of various routing schemes discussed in the previous section. For each network size, the results presented are averaged over 9 different topologies.

Figures 4.2, 4.3, and 4.4 for topology setting 1, show that the network lifetime has an decreasing trend as the network size becomes large. The data rate also has a significant impact on the network lifetime as a higher data rate adds more traffic to the network. Comparing with other schemes, the network lifetime with multipath routing has a substantial increase of 9% to 18% than energy-aware routing, in all three data rates used with 250 nodes.

Figures 4.5 - 4.7 and 4.8 - 4.10, show the results for topology settings 2 and 3, respectively. These graphs show that the multipath routing can achieve the highest network lifetime. We also observe that the network lifetime with multipath routing degrades more gracefully than other routing protocols when the network size increases. It demonstrates that our routing scheme is more stable with the variation of the network size.

We can observe, by comparing the results for the same topology setting under different

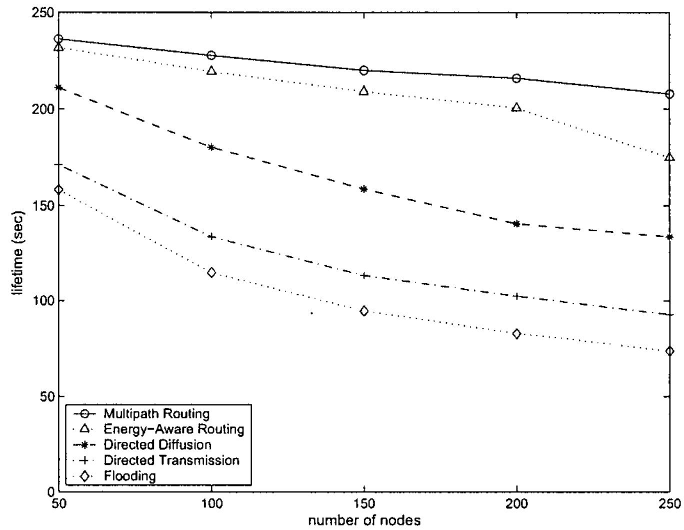

Figure 4.2: Average network lifetime with 1 sink and 2 sources at 1 packet/s.

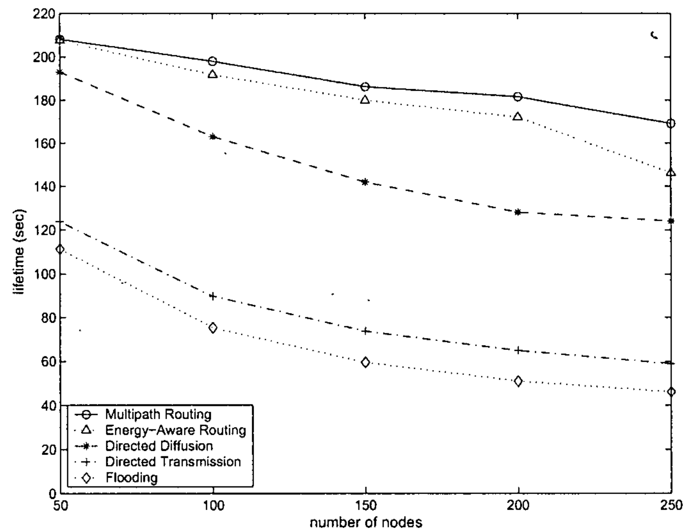

Figure 4.3: Average network lifetime with 1 sink and 2 sources at 2 packets/s.

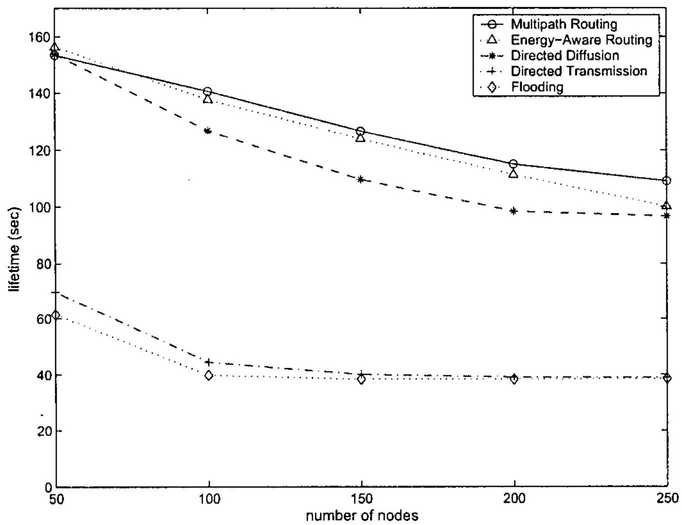

Figure 4.4: Average network lifetime with 1 sink and 2 sources at 5 packets/s.

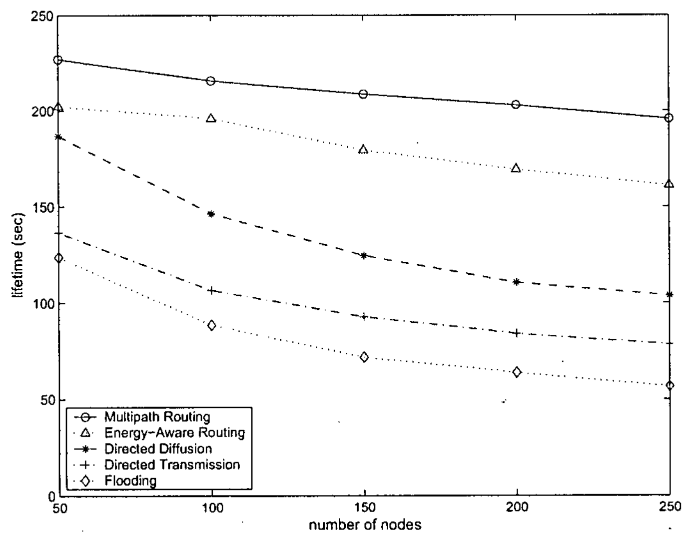

Figure 4.5: Average network lifetime with 1 sink and 4 sources at 1 packet/s.

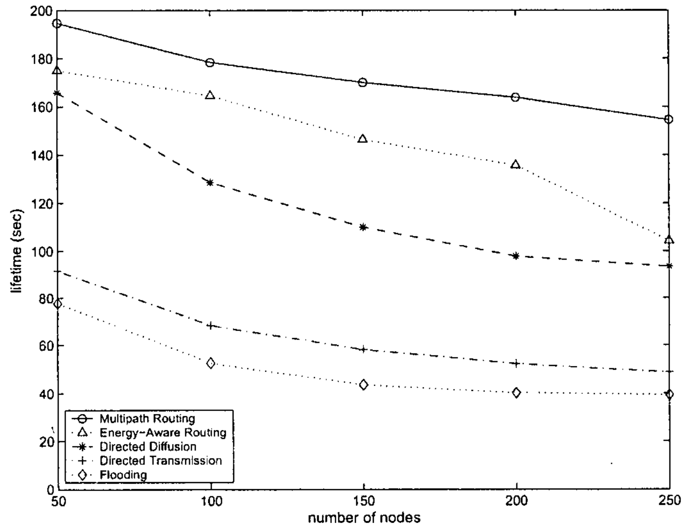

Figure 4.6: Average network lifetime with 1 sink and 4 sources at 2 packets/s.

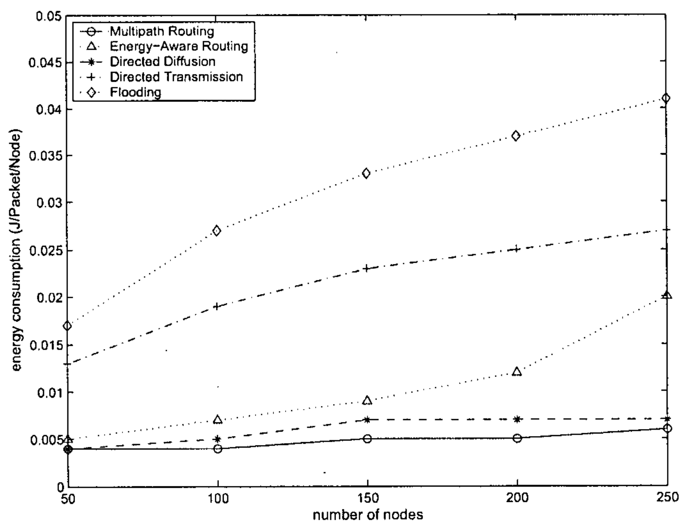

Figure 4.7: Average network lifetime with 1 sink and 4 sources at 5 packets/s.

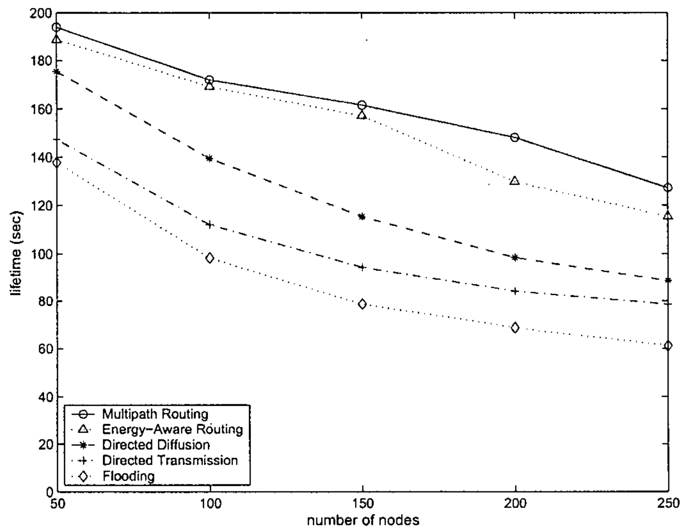

Figure 4.8: Average network lifetime with 2 sinks and 3 sources at 1 packet/s.

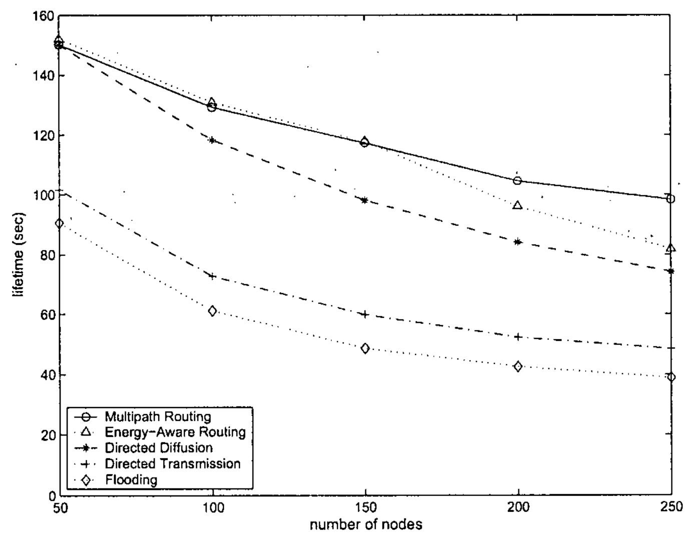

Figure 4.9: Average network lifetime with 2 sinks and 3 sources at 2 packets/s.

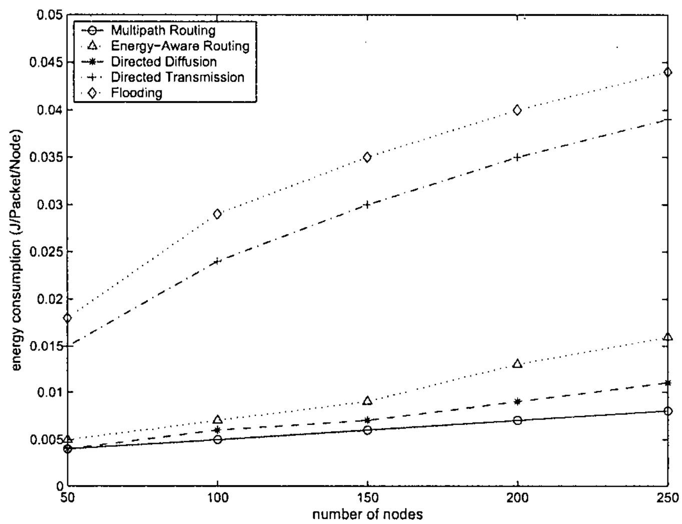

Figure 4.10: Average network lifetime with 2 sinks and 3 sources at 5 packets/s.

data rates, that the network lifetime does not increase proportionally as the data rate drops. This is simply because the duty cycle of the nodes is low even when the data rate doubles. Therefore, the energy spent on idle varies little and it prevents the network lifetime from increasing proportionally with the drop of the data rate.

We also notice from Figures 4.4, 4.7, and 4.10 that the curves of the network lifetime become flat with topologies of 100 nodes and more at 5 packets per second for flooding and directed transmission. It is due to the fact that both routing protocols do not have an efficient retransmission control mechanism. The large number of redundant data copies that are retransmitted between different sensor nodes deplete quickly the network resources, especially at a high data rate of 5 packets per second. Consequently, the network lifetime reaches the limit in such circumstance and we obtain a flat curve as shown in the figures.

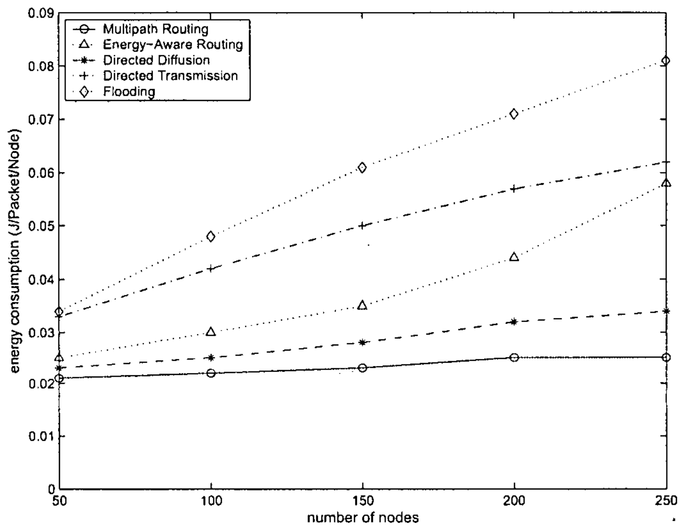

Figure 4.11: Average node energy consumption with 1 sink and 2 sources at 1 packet/s.

4.2.2 Node Energy Consumption

The next metric that we study is the *node energy consumption.* Figures 4.11 - 4.13, 4.14 - 4.16, and 4.17 - 4.19 show the simulation results of topology settings 1, 2 and 3, respectively. We can observe that there is a lower node energy consumption of our multipath routing over the other schemes. The flooding is the most costly protocol; by adding a simple mechanism of retransmission probability control on top of the flooding, the directed transmission improves the energy efficiency. The energy-aware routing obtains further improvement by calculating the retransmission probability as function of the node energy level. The multipath routing and directed diffusion perform better than other protocols we examined.

From these figures, we can observe that our multipath routing protocol can still

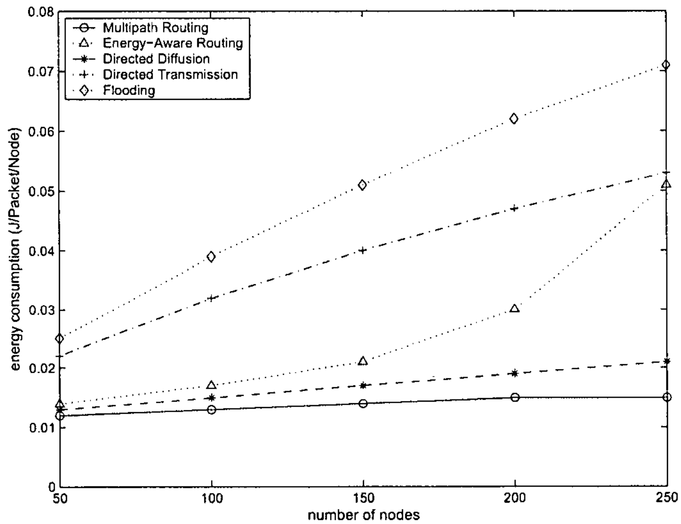

Figure 4.12: Average node energy consumption with 1 sink and 2 sources at 2 packets/s.

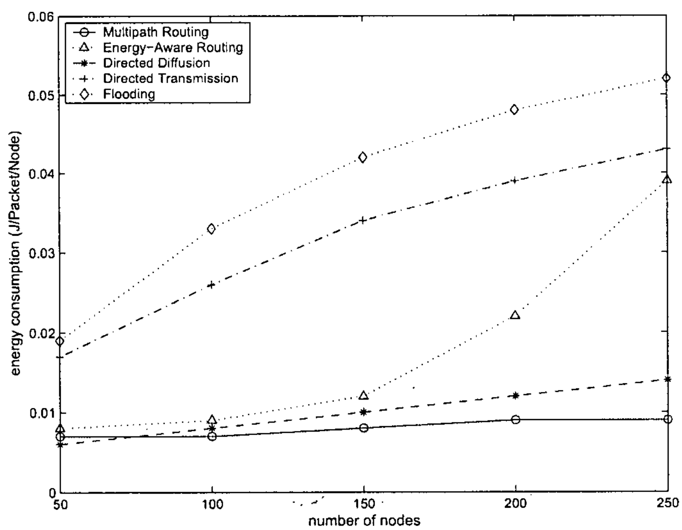

Figure 4.13: Average node energy consumption with 1 sink and 2 sources at 5 packets/s.

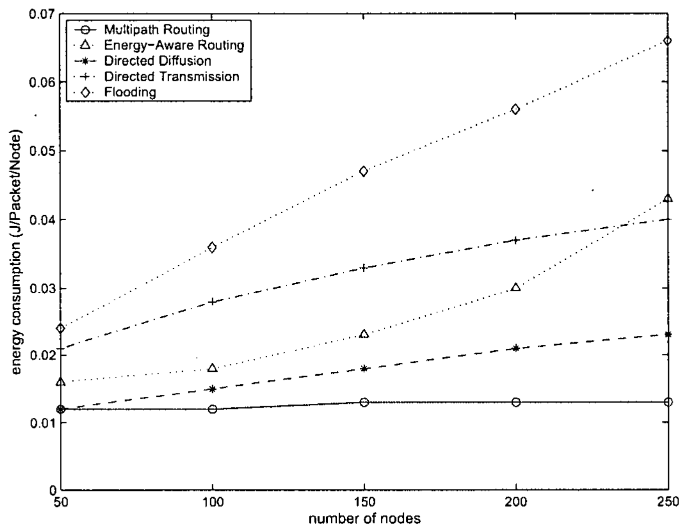

Figure 4.14: Average node energy consumption with 1 sink and 4 sources at 1 packet/s.

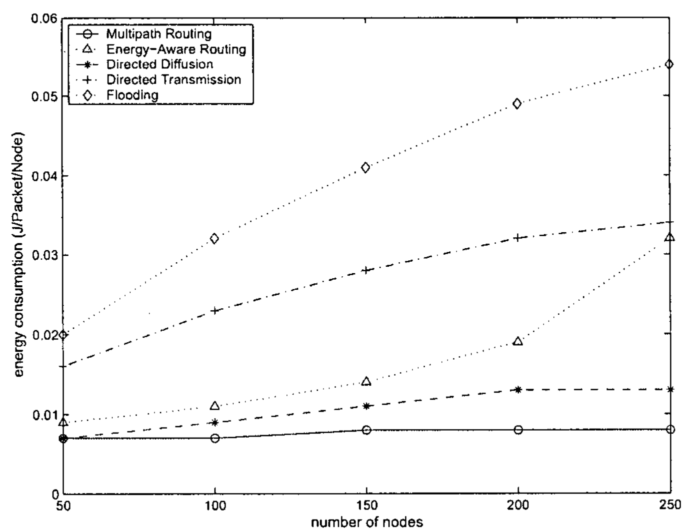

Figure 4.15: Average node energy consumption with 1 sink and 4 sources at 2 packets/s.

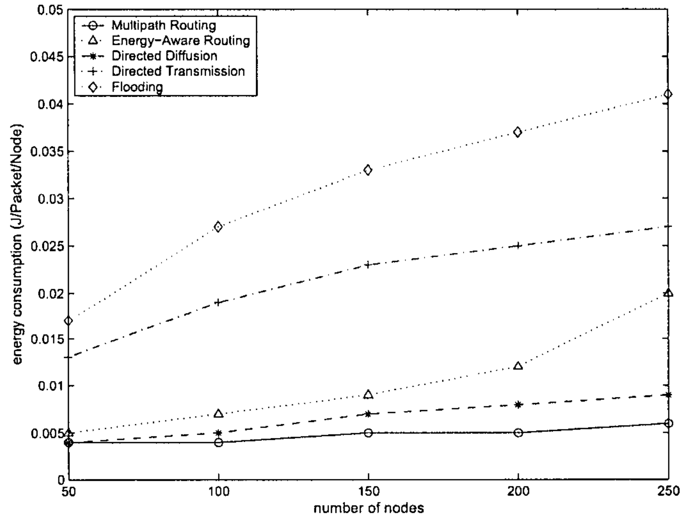

Figure 4.16: Average node energy consumption with 1 sink and 4 sources at 5 packets/s.

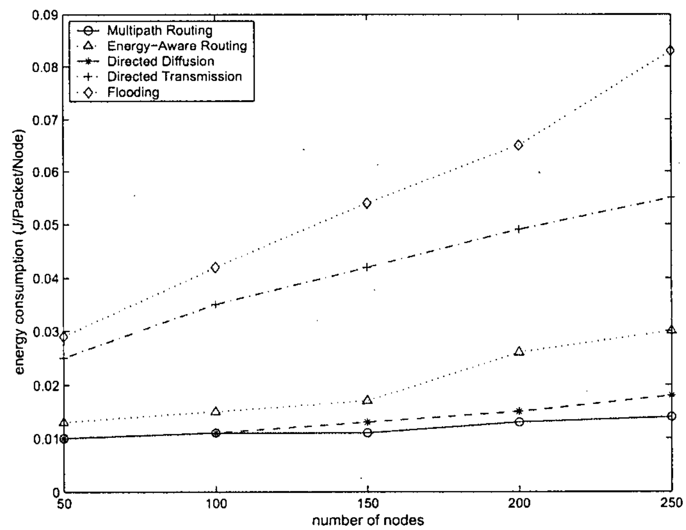

Figure 4.17: Average node energy consumption with 2 sinks and 3 sources at 1 packet/s.

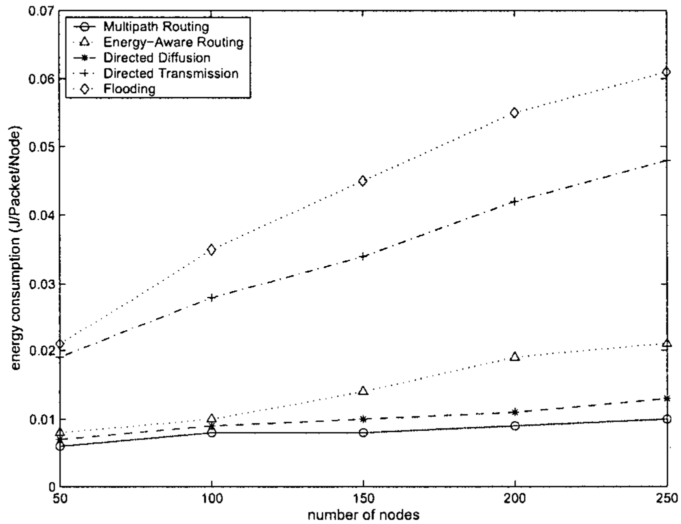

Figure 4.18: Average node energy consumption with 2 sinks and 3 sources at 2 packets/s.

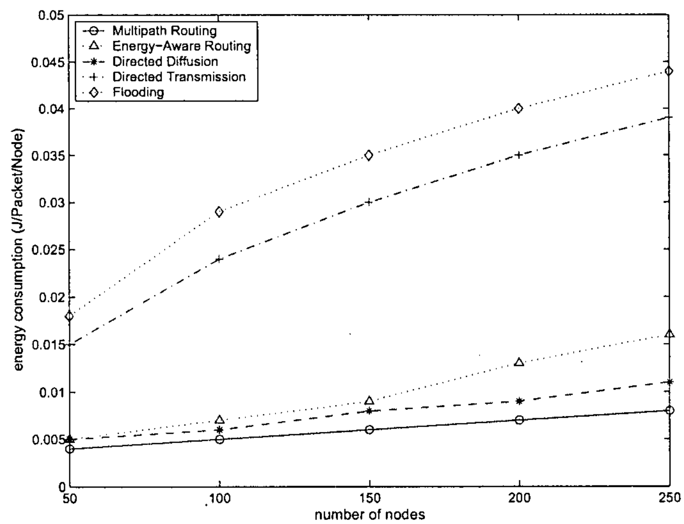

Figure 4.19: Average node energy consumption with 2 sinks and 3 sources at 5 packets/s.

maintain its node energy consumption at a low level even when the network size increases. For example, in Figure 4.12, compare to directed diffusion, the improvement of multipath routing is 1% to 34% when the network size increases from 50 nodes to 250 nodes, with 1 sink and 2 sources at the data rate of 2 packets per second. Such experimental results demonstrate that the energy efficiency of multipath routing is stable and has little impact by the increase of the network size, while the performance of other schemes degrades with larger network size.

We observe that with the same topology setting, the *average node energy consumption* decreases for the same routing scheme when the data rate goes higher. Such phenomenon is due to our definition of *node energy consumption*. As we defined in equation (4.1), we calculate the energy consumption based on the sum of initial energy level and the total number of data packets received by the sinks. When data rate increases, the number of data packets arrived to the sinks increases proportionally. With the same initial node energy level, we obtain better average node energy consumption when data rate increases.

With the same data rate, Figure 4.15 shows a better performance than Figure 4.12 for multipath routing. It is simply due to the difference of topology settings. With 1 sink in the center of the field and 4 sources at four corners in topology 1, the average path length is significantly smaller than that in the topology 2, where one sink and two sources are in opposite edges of the field. As a result, more energy is required to deliver data to the sink in the setting of one sink and two sources.

4.2.3 Average Node Energy Level

Figures 4.20 - 4.22 show the *average node energy level* in fixed intervals after the data transmission starts for three topology settings, with a data rate of 2 packets per second. We notice that the network size has an impact on the node energy level. The average node energy level decreases with larger network. It is more obvious with the flooding and the directed transmission, as they cannot eliminate completely the redundant data copies in the network. The lack of a data retransmission mechanism makes the average node energy level for flooding and directed transmission to degrade much faster than the other three protocols.

For other routing schemes, a larger network requires more exchange of control messages to discover and construct the routes; therefore more energy is consumed in the setup phase. Also, a larger network implies a larger distance that separates the sink and the source nodes. More intermediate nodes are required before a data packet can reach the sink nodes. As we expected, the multipath routing has the average node energy level about 5% to 9% higher than the energy-aware routing [5], which has the second best performance.

We observe from Figures 4.21 and 4.22 that the average node energy level for flooding with the topology of 1 sink and 4 sources is higher than that with the topology of 2 sinks and 3 sources. Such phenomenon can be explained by the lack of efficient routing mechanism for traffic between the sink and the source with flooding. Each intermediate node simply retransmits the data packet received if it is arrived for the first time. With

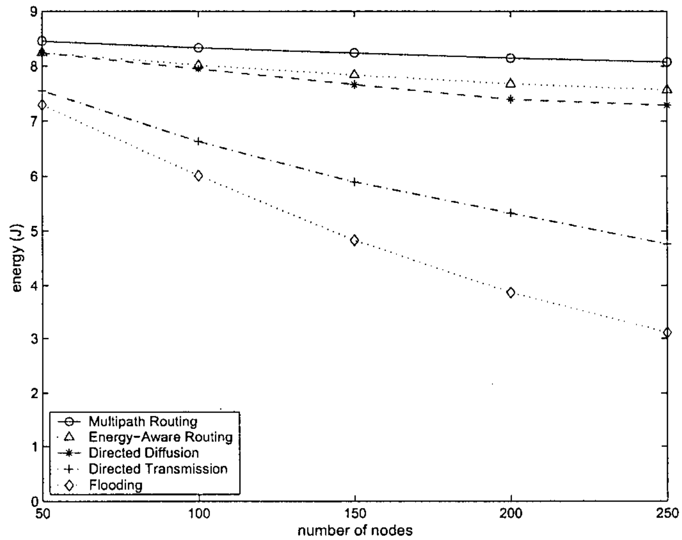

Figure 4.20: Average node energy level with 1 sink and 2 sources at 2 packets/s.

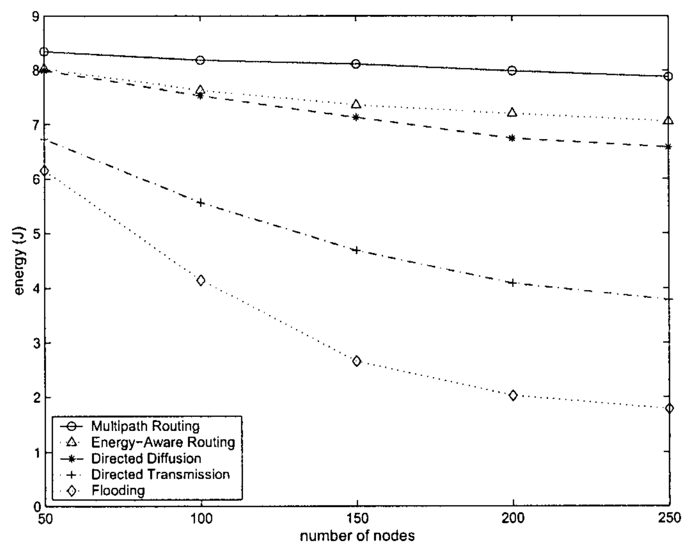

Figure 4.21: Average node energy level with 1 sink and 4 sources at 2 packets/s.

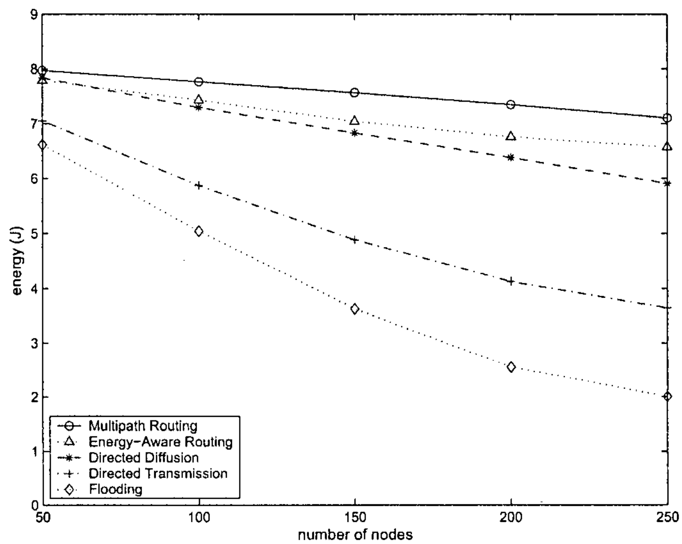

Figure 4.22: Average node energy level with 2 sinks and 3 sources at 2 packets/s.

more source nodes in the network, the average node energy level is reduced in a higher rate as more data copies are generated and transmitted between sensor nodes.

4.2.4 Control Message Overhead

Figures 4.23 - 4.25 show the *control message overhead* of different protocols. The directed diffusion spends much more energy on transmitting and receiving control messages than any other protocols, since it requires periodic interest broadcast and path reinforcement. The multipath routing has a much lower overhead for the control message, about 70% less than the energy-aware routing [5] with the topology setting of 1 sink and 4 sources.

We notice that the curve of directed diffusion decreases in an exponential manner as the network size increases. This may be due to the variation of the number of nodes that

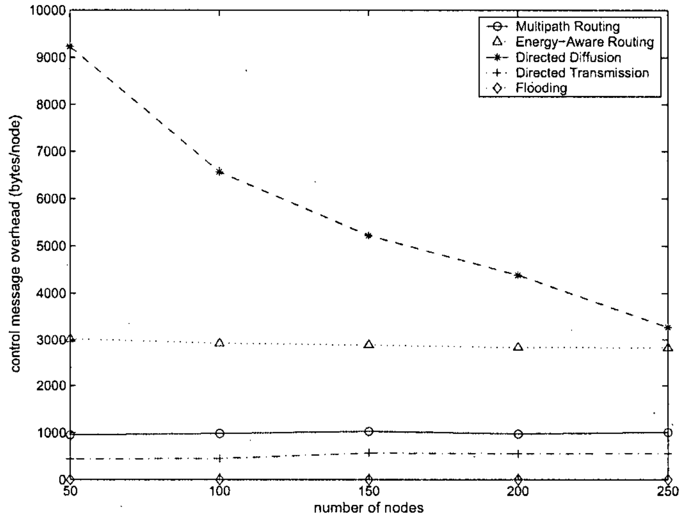

Figure 4.23: Control message overhead with 1 sink and 2 sources at 1 packet/s.

participate in the data transmission. The directed diffusion reinforces the nodes with the shortest delay to relay the data packets. When the network size increases from 50 nodes to 250 nodes, the number of nodes that participate the data relay may not increase significantly with the same proportion. As the *control message overhead* is obtained by averaging the total amount of control messages received by the number of nodes in the network, its value decreases significantly with larger network for the directed diffusion scheme.

4.2.5 Average Packet Delay

Figures 4.26 - 4.28 show the average packet delay for different topology settings with a data rate of 2 packets per second. The multipath routing has the shortest delay compared to other schemes. As we expected, the data packet is routed through different

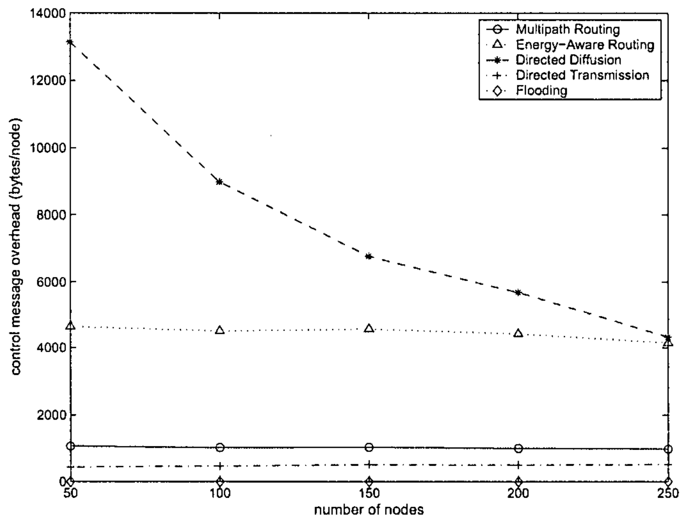

Figure 4.24: Control message overhead with 1 sink and 4 sources at 1 packet/s.

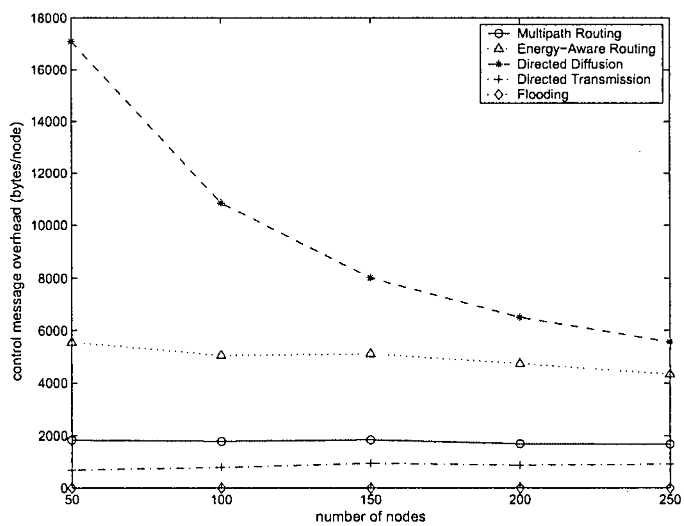

Figure 4.25: Control message overhead with 2 sinks and 3 sources at 1 packet/s.

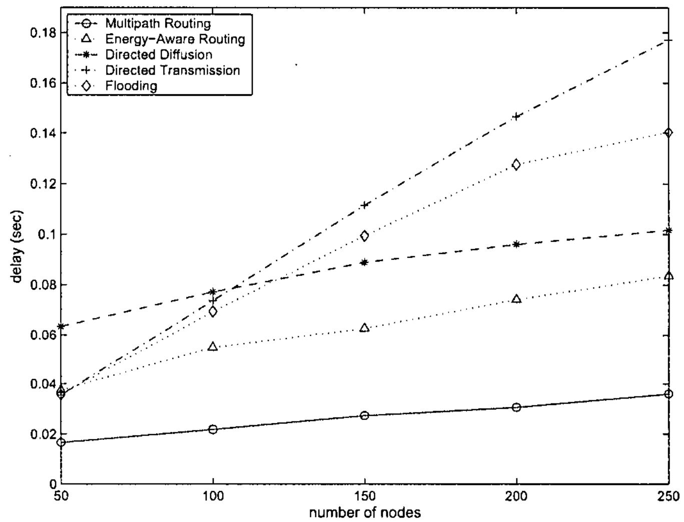

Figure 4.26: Average packet delay with 1 sink and 2 sources at 2 packets/s.

node-disjoint paths with the multipath routing. Hence, the network congestion and the transmission interferences are more likely to be avoided with our multipath routing scheme.

4.3 Summary

In this chapter, we presented the simulation results and compared our multipath routing protocol with other routing protocols including the directed diffusion [7], the directed transmission [10] and the energy-aware routing [5] under different topologies and traffic patterns. Results show that our multipath routing protocol is able to achieve higher network lifetime and better node energy efficiency. The multipath routing protocol also has a lower packet delay, with a maximum improvement of 48% over the directed diffusion

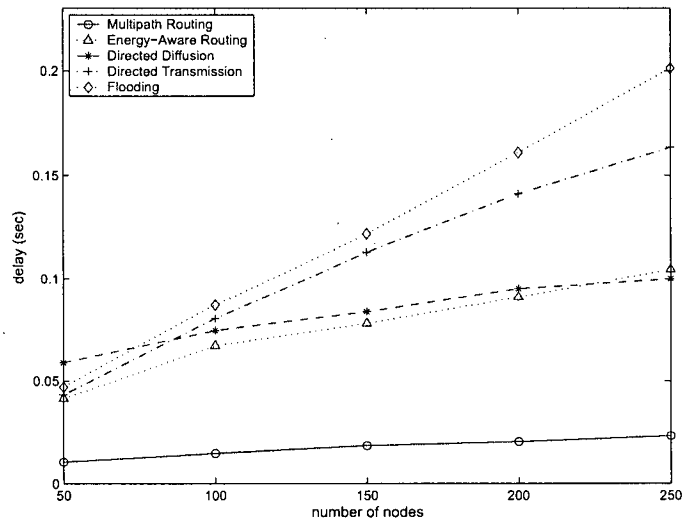

Figure 4.27: Average packet delay with 1 sink and 4 sources at 2 packets/s.

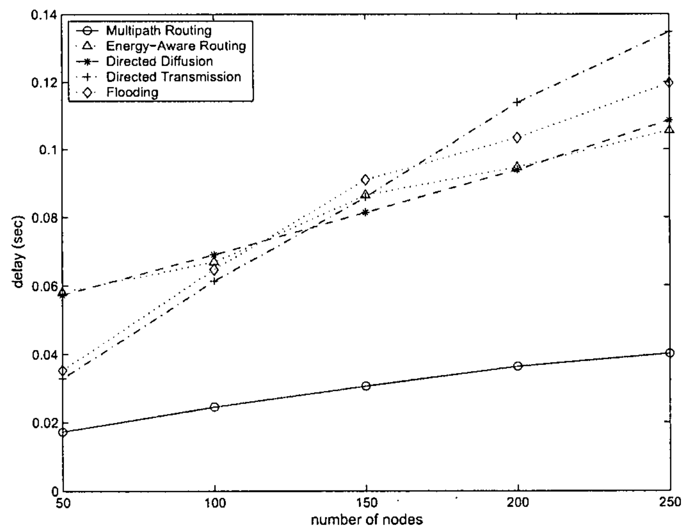

Figure 4.28: Average packet delay with 2 sinks and 3 sources at 2 packets/s.

scheme with the topology setting of 2 sinks, 3 sources and a data rate of 2 packets per second.

Chapter 5

Conclusion and Future Work

The capability of preserving energy is crucial for routing protocols in wireless sensor networks. We proposed in this thesis a novel multipath routing scheme with objectives of increasing the energy efficiency and extending the network lifetime. Our scheme consists of a multipath search protocol and a load balancing algorithm. The multipath search protocol is distributed, which discovers node-disjoint paths that connect the sink and the source node. The major difference between our protocol and the conventional multipath routing protocols is that the data traffic is handled through multiple paths simultaneously, instead of using single optimal path. This allows us to take full advantage of the energy spent on the search of node-disjoint multipaths. It also helps to avoid stressing one particular route and the premature partition of the network. The traffic rate at each route is allocated by the sink node via the load balancing algorithm, which performs the optimization based on the path conditions. In the following sections, we will conclude our work with our contributions and provide suggestions for future work.

5.1 Conclusion

We began our thesis with an investigation on previous work done for routing and data dissemination schemes in mobile ad-hoc and wireless sensor networks. We have demonstrated that:

- Most of the conventional routing schemes use a single path for data transmission between the sink and source nodes. A single node failure on the path will force the search of an alternate path, which is costly in terms of network resources. Another drawback of the single path routing is that it stresses a particular path and has a negative impact on the network lifetime.

- The multipath routing is able to improve the reliability of the wireless sensor networks, as alternate paths are made available in the initial phase. However, the majority of the existing multipath protocols still use only one primary path for data transmission and consider other alternative paths as backups. The energy saving is made by eliminating the route discovery when the primary path fails. The overall energy efficiency is not improved significantly compared with conventional single path routing protocols.

We have proposed our multipath routing scheme to overcome the drawbacks found in the existing multipath protocols. The major achievements of our work are as follows:

- We propose a distributed multipath routing protocol, which searches multiple node-disjoint paths. We introduce the "path cost" to reflect the cost of transmitting data with a unit rate through a path. It is updated constantly to allow the sink node to monitor and adjust the traffic distribution accordingly.

- The load balancing algorithm allocates the traffic rate to each path. It has the objective to extend the network lifetime and improve the energy efficiency by optimizing the *load balance ratio*.

We have evaluated the performance of our multipath routing protocol with the *ns-2* simulator. We used different topologies and traffic patterns in our simulations and compared with other routing protocols, such as the energy-aware routing [5], the directed diffusion [7], and the directed transmission [10]. We demonstrated that our proposed protocol had a higher network lifetime with an average increase of 9% to 18% than the energy-aware routing. We also noticed that the multipath routing had better node energy consumption when the network size increases.

5.2 Topics for Future Investigations

In this thesis, we proposed a multipath routing protocol for wireless sensor networks. Further research work is required to enhance the performance of the protocol. They include:

- **Data Aggregation:** Our multipath routing protocol does not include data aggregation. The future enhancement on data aggregation will make the protocol to be data-centric and application-aware. It will also allow further energy savings if the source nodes are close to each other and transmit the information collected for the same stimulus. The readings come from different source nodes will also be refined by data aggregation to make the data arrived at the sink node to be more accurate.

- **Mobility Support:** The multipath routing protocol we proposed applies for static sensor nodes. It will be useful to enhance the protocol to support nodes with limited mobility, as they are able to better adapt to the environment. A location update mechanism is required to allow each node to be aware of its own and its neighbors' positions constantly. It is a challenge to balance between the node energy consumption and the additional maintenance efforts that keep the node coordinates updated.

- **Cross-Layer Optimization:** The communication between wireless sensor nodes is influenced heavily by the physical medium, as the quality of radio channels varies over time. The cross-layer design is beneficial for the improvement of QoS and network efficiency. By interacting our multipath with the IEEE 802.11 MAC layer, which provides various information about the state of radio connections, the path selection and maintenance will be more accurate. The protocol can select route with better channel quality and avoid using path with unstable conditions. The load balancing algorithm will also be able to take the channel conditions into consideration, in order to further increase energy efficiency and network lifetime.

Bibliography

[1] J. M. Kahn, R. H. Katz, and K. S. J. Pister, "Next Century Challenges: Mobile Networking for "Smart Dust," in *Proc. of ACM MobiCom'99*, Seattle, WA, pp. 271-278, Aug. 1999.

[2] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless Sensor Networks: A Survey," *Computer Networks*, vol. 38, pp. 393-422, Mar. 2002.

[3] E. J. Duarte-Melo, M. Liu, and A. Misra, "A Modeling Framework for Computing Lifetime and Information Capacity in Wireless Sensor Networks," in *Proc. of 2nd WiOpt: Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks*, Cambridge, UK, Mar. 2004.

[4] Y. Xu, R. Govindan, and D. Estrin, "Geographical and Energy Aware Routing: A Recursive Data Dissemination Protocol for Wireless Sensor Networks," UCLA Computer Science Dept., Tech. Rep. UCLA/CSD-TR-01-0023, May 2001.

[5] R. C. Shah and J. M. Rabaey, "Energy Aware Routing for Low Energy Ad Hoc Sensor Networks," in *Proc. of IEEE Wireless Communications and Networking Conference (WCNC'02)*, Orlando, FL, pp. 350-355, Mar. 2002.

[6] U. Cetintemel, A. Flinders, and Y. Sun, "Power-efficient Data Dissemination in Wireless Sensor Networks," in *Proc. of Third International ACM Workshop on Data Engineering for Wireless and Mobile Access (MobiDE'03)*, San Diego, CA, Sept. 2003.

[7] C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks," in *Proc. of ACM MobiCom'00*, Boston, MA, pp. 56-67, Aug. 2000.

[8] F. Silva, J. Heidemann, R. Govindan, and D. Estrin, "A Overview of Directed Diffusion," USC/Information Sciences Institute, Tech. Rep. ISI-TR-2004-586, Jan. 2004.

[9] A. Rao, S. Ratnasamy, C. Papadimitriou, S. Shenker, and I. Stoica, "Geographic Routing without Location Information," in *Proc. of ACM MobiCom'03*, San Diego, CA, pp. 96-108, Sept. 2003.

[10] C. L. Barrett, S. J. Eidenbenz, L. Kroc, M. Marathe, and J. P. Smith, "Parametric Probabilistic Sensor Network Routing," in *Proc. of ACM International Workshop on Wireless Sensor Networks and Applications (WSNA'03)*, San Diego, CA, pp. 122-131, Sept. 2003.

[11] B. Deb, S. Bhatnagar, and B. Nath, "ReInForM: Reliable Information Forwarding Using Multiple Paths in Sensor Networks," in *Proc. of the 28th annual IEEE Inter-*

national Conference on Local Computer Networks, Bonn/Königswinter, Germany, pp. 406-415, Oct. 2003.

[12] J. N. Al-Karaki and A. E. Kamal, "Routing Techniques in Wireless Sensor Networks: A Survey," *IEEE Wireless Communications*, vol. 11, issue 6, pp. 6-28, Dec. 2004.

[13] M.R. Pearlman, Z.J. Haas, P. Sholander, and S.S. Tabrizi, "On the Impact of Alternate Path Routing for Load Balancing in Mobile Ad Hoc Networks," in *Proc. of the First ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc 2000)*, Boston, MA, Aug. 2000.

[14] K. Wu and J. Harms, "On-demand Multipath Routing for Ad Hoc Networks," in *Proc. of European Personal and Mobile Communications Conference (EPMCC)*, Vienna, Austria, Feb. 2001.

[15] A. Nasipuri and S.R. Das, "On-demand Multipath Routing for Mobile Ad Hoc Networks," in *Proc. of the 8th International Conference on Computer Communications and Networks (IC3N)*, Boston, MA, Oct. 1999.

[16] D. Ganesan, R. Govindan, S. Shenker, and D. Estrin, "Highly-Resilient, Energy-Efficient Multipath Routing in Wireless Sensor Networks," in *Proc. of the Second ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc 2001)*, Long Beach, CA, Oct. 2001

[17] H. Dai and R. Han, "A Node-Centric Load Balancing Algorithm for Wireless Sensor Networks," in *Proc. of IEEE Global Telecommunications Conference (GLOBECOM'03)*, San Francisco, CA, pp. 548-552, Dec. 2003.

[18] M.S. Bazaraa, H.D. Sherali, and C.M. Shetty, *Nonlinear Programming, Theory and Algorithms*, second edition, John Wiley & Sons Inc, 1993.

[19] P. H. Hsiao, A. Hwang, H. T. Kung, and D. Vlah, "Load-Balancing Routing for Wireless Access Networks," in *Proc. of IEEE INFOCOM'01*, Anchorage, AL, pp. 986-995, April 2001.

[20] M.K. Marina and S.R. Das, "On-demand Multipath Distance Vector Routing in Ad Hoc Networks," in *Proc. of the Ninth International Conference for Network Protocols (ICNP)*, Riverside, CA, Nov. 2001.

[21] H. Suzuki and F.A. Tobagi, "Fast Bandwidth Reservation Scheme with Multi-link & Multi-path routing in ATM Networks," in *Proc. of IEEE INFOCOM'92*, Florence, Italy, May 1992.

[22] P. Georgatsos and D. Griffin, "A Management System for Load Balancing Through Adaptive Routing in Multiservice ATM Networks," in *Proc. of IEEE INFOCOM'96*, San Francisco, CA, Mar. 1996.

[23] N.F. Maxemchuk, "Dispersity Routing in High-speed Networks," *Computer Networks and ISDN System*, vol. 25, issue 6, pp. 645-661, 1993.

[24] W. Heinzelman, J. Kulik, and H. Balakrishnan, "Adaptive Protocols for Information Dissemination in Wireless Sensor Networks," in *Proc. of the 5th ACM/IEEE MobiCom'99*, Seattle, WA, pp. 174-185, Aug. 1999.

[25] J. Kulik, W. R. Heinzelman, and H. Balakrishnan, "Negotiation-based Protocols for Disseminating Information in Wireless Sensor Networks," *IEEE Network*, vol. 8, issue 2, pp. 169-185, Mar. 2002.

[26] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-Efficient Communication Protocol for Wireless Microsensor Networks," in *Proc. of the 33rd International Conference on System Science (HICSS'00)*, Hawaii, Jan. 2000.

[27] Y. Xu, J. Heidemann, and D. Estrin, "Geography-informed Energy Conservation for Ad-hoc Routing," in *Proc. of the 7th ACM/IEEE MobiCom'01*, Rome, Italy, pp. 70-84, Jul. 2001.

[28] S. De, C. Qiao, and H. Wu, "Meshed Multipath Routing: An Efficient Strategy in Sensor Network," *Computer Networks, Special Issue on Wireless Sensor Networks*, vol. 43, issue 4, pp. 481-497, Nov. 2003.

[29] D. Niculescu and B. Nath, "Localized Positioning in Ad Hoc Networks," in *Proc. of IEEE International Workshop on Sensor Network Protocols and Applications (SNPA'03)*, Anchorage, AK, pp. 42-50, May 2003.

[30] J. Carle and D. Simplot-Ryl, "Energy-efficient Area Monitoring for Sensor Networks," *IEEE Computer Magazine*, vol. 37, no. 2, pp. 40-46, Feb. 2004.

[31] O. Younis and S. Fahmy, "HEED: A Hybrid, Energy-efficient, Distributed Clustering Approach for Ad Hoc Sensor Networks," *IEEE Trans. Mobile Computing*, vol. 3, no. 4, pp. 366-379, Oct. 2004.

[32] A. Cerpa and D. Estrin, "ASCENT: Adaptive Self-configuring Sensor Networks Topologies," in *Proc. of IEEE INFOCOM'02*, New York, NY, pp. 1278-1287, June 2002.

[33] K. Sohrabi and J. Pottie, "Protocols for Self-organization of Wireless Sensor Network," *IEEE Personal Communication*, vol. 7, no. 5, pp. 16-27, Oct. 2000.

[34] C. Schurgers, V. Tsiatsis, S. Ganeriwal, and M. B. Srivastava, "Topology Management for Sensor Networks: Exploiting Latency and Density," in *Proc. of the Third ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc 2002)*, Lausanne, Switzerland, pp. 135-145, June 2002.

[35] E. J. Duarte-Melo and M. Liu, "Analysis of Energy Consumption and Lifetime of Heterogeneous Wireless Sensor Networks," in *Proc. of IEEE Global Telecommunications Conference GLOBECOM'02*, Taipei, Taiwan, vol. 1, pp. 21-25, Nov. 2002.

[36] A. Sankar and Z. Liu, "Maximum Lifetime Routing in Wireless Ad-Hoc Networks," in *Proc. of IEEE INFOCOM'04*, Hong Kong, China, Mar. 2004.