공학박사학위논문

무선 센서 네트워크에서 에너지 절감을 위한 계층 토폴로지 제어

Energy-Aware Hierarchical Topology Control for Wireless Sensor Networks

2015 년 8 월

서울대학교 대학원

전기·컴퓨터공학부

윤 익 준

무선 센서 네트워크에서 에너지 절감을 위한 계층 토폴로지 제어

Energy-Aware Hierarchical Topology Control for Wireless Sensor Networks

지도교수 신 현 식

이 논문을 공학박사학위논문으로 제출함

2015 년 7 월

서울대학교 대학원

전기·컴퓨터공학부

윤 익 준

윤익준의 박사학위논문을 인준함

2015 년 7 월

위 원 장 _________________________ (인)

부위원장 _________________________ (인)

위 원 _________________________ (인)

위 원 _________________________ (인)

위 원 _________________________ (인)

Abstract

Energy-Aware Hierarchical Topology Control for Wireless Sensor Networks

Ikjune Yoon
Department of Electrical Engineering and Computer Science
Collage of Engineering
The Graduate School
Seoul National University

Simple wireless sensor networks (WSNs) usually have a flat topology and transmit data using a flooding scheme of which there are several variants. However, these can cause the broadcast storming problem, reducing the efficiency and reliability of the WSN. Due to these problems, most WSNs have a cluster or tree structure; but this causes an imbalance of residual energy between nodes, which gets worse over time as nodes become defunct and replacements are inserted. Moreover, a defunct cluster head leads to a sharp drop of network connectivity. Therefore, an efficient way to improve the energy imbalance and the network connectivity is needed.

In this thesis, we propose a hierarchical topology control scheme, in which each node periodically selects its own layer accommodating itself with different levels of residual energy and the amount of data to transfer, in order to balance the energy level and to increase the network connectivity. Simulations show that this scheme can balance node energy levels, and thus extend network lifetime.

We also introduce a hierarchical topology control scheme for WSNs, which contains both energy-harvesting nodes and battery-powered nodes, in order to extend the lifetime of battery-powered nodes and to increase the network connectivity. In such a WSN, the energy harvesting nodes are also arranged in layers like the battery-powered nodes depending on their expected level of residual energy. This scheme is shown to increase the lifetime of battery-powered nodes preferentially by locating energy-harvesting nodes on the higher layers.

Keywords: wireless sensor network, topology control, hierarchy, layer, energy-aware, energy-harvesting

Student Number: 2006-21234

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Recent advances in wireless sensor networks (WSNs) have led to topology controls, which is one of the most important techniques to extend network lifetime and to preserve connectivity between sensor nodes [7]. In this chapter, we set out our research motivations and objectives, summarize the major contributions of our research, and outline the contents of the rest of the thesis.

1.1 Motivation

Wireless sensor networks (WSNs) are now widely used to gather environmental data. A WSN consists of many wireless sensor nodes, which communicate the data they acquire to a sink node. Because of wireless range limitations, this usually involves multi-hop transmission, passing data from one node to the next. Nodes also have a limited lifetime because they run on batteries, and when these are exhausted, transmissions cease. Other nodes, which used the dead node for relaying data, may no longer be able to communicate with the sink. Moreover, in many cases, WSNs is used in inaccessible area such as virgin forest, polar regions, or deep water [8–11]. For this reason, it makes difficult to manage WSNs maintained for a long time. Therefore, it is important to prolong lifetime of the sensor nodes and to sustain connectivity between nodes in WSNs. The noticeable problems of WSNs are as follows:

- Consideration of sparsely deployed network: In the WSN literature, it is often assumed that sensor nodes are densely deployed. But this may not be possible because of the installation environment or cost. Sparsely deployed nodes make a lot of previous work meaningless.
- Unreliable sensors: Since a WSN tends to be composed of a large number of off-the-shelf sensor nodes, fault tolerance is a significant attribute, especially in a harsh environment. For reliable communication, network connectivity should be preserved even when some of the sensor nodes fail or run out of power.
- Broadcast storming: Simple wireless sensor networks (WSNs) usually have a flat topology and transmit data using a flooding scheme, of which there are several variants. However, flooding can cause the broadcast storming problem [12], reducing the efficiency and reliability of the WSN.
- Energy imbalance: The nodes closest to the sink in a flat topology or cluster heads in a cluster topology consume more energy than the other nodes [13]. When they fail the rest of the network becomes isolated and thus useless.

Early WSNs had a flat topology and transferred data using a flooding scheme. This can cause a broadcast storm and the redundant transmissions that result quickly exhaust the nodes' batteries. This problem is exacerbated because the nodes closest to the sink use more energy than the others, and when they fail the rest of the network becomes isolated and thus useless.

Hierarchical topology control schemes [6, 14–24] are designed to overcome these problems. A popular way to create a hierarchical topology is clustering. Nodes can organize themselves into clusters, each of which has a head which aggregates data from the nodes in the cluster, and forwards it to the sink node. This scheme is better then the flat topology, but the cluster heads owe its weak links: these nodes consume more energy than the other nodes, and die early, isolating the cluster from the sink [25]. In previous schemes [6, 15–17], this problem has been overcome by getting nearby energy-rich nodes to take over the roles of defunct aggregation nodes; or new aggregation nodes must be deployed to extend the lifetime of the WSN.

In a WSN which uses hierarchical topology control and operates for a long period, the imbalance in residual energy between the nodes can become serious as the number of failed and insertions or removals replacement nodes recounts of [26]. Draconian changes to the network topology are also likely to be necessary over time.

The life of a network can be prolonged if energy is harvested from the environment: an energy-harvesting node potentially operate forever by recharging its battery with the environmental energy. However, even an energy-harvesting node may cease to operate it consumes more energy than it is able to harvest. Many researchers [27] have designed several topology control schemes to maximize the lifetime of a wireless sensor network of energy-harvesting nodes. However, prolonging the lifetime of WSNs with a mixed population of nodes, some of which can and some of which cannot harvest energy, has received much less attention.

1.2 Research Objectives and Contributions

In this section, we explain the main contributions of this thesis.

We designed a topology control scheme designed for WSNs which are to be maintained for a long period. It minimizes the variation in residual energy between nodes, and thus extends the network lifetime. This goal is achieved by replacing the usual 1- or 2-layered topology with multiple layers, which can accommodate a wide range of node energy levels more precisely. This scheme also gets nodes to change roles dynamically as the energy and traffic context changes. This is necessary because the energy level of each node and the network topology can both change radically in long-term WSNs. In our scheme, each node periodically determines its own layer in response to its energy status, with traffic and topology information. There has been a lot of research on hierarchical topologies for WSNs; but, to the best of our knowledge, this is the first context-aware multi-layer topology control scheme for long-term WSNs.

We propose a topology control scheme for preserving connectivity between nodes in long-term maintained WSNs, in which defunct nodes are manually replaced, and which contain both energy-harvesting nodes and battery-powered nodes. In such a WSN, the battery-powered sensor nodes can be expected to have diverse amounts of residual energy, which decline with length of deployment. In this scheme, the nodes are arranged in several virtual layers, and the nodes on the upper layer are given the job of aggregating sensory data received from the nodes on the lower layers, and sending it to a sink node. Like previous layering topology control schemes, this arrangement improves energy efficiency and connectivity; but our scheme also increases the lifetime of battery-powered nodes preferentially by locating energy-harvesting nodes on the higher layers.

1.3 Organization of This Thesis

The rest of the thesis is organized as follows:

In Chapter 2 gives information, which consists of the previous implementations of WSNs, and a summary of previous research on hierarchical topology control, It also provides a brief survey of previous research on energy-harvesting WSNs.

In Chapter 3 we propose a new topology control scheme designed specially for WSNs which are to be maintained for a long period. It minimizes the variation in residual energy between nodes, and thus extends the network lifetime. In this scheme, Each node determines its own layer in response to its energy status, with traffic and topology information. The nodes periodically change their roles so as to balance their residual energy. Simulation results show the effectiveness of this scheme in expending the life of WSNs.

Chapter 4 introduces a topology control scheme for preserving connectivity between nodes in long-term WSNs, in which defunct nodes are manually replaced, and which contain both energy-harvesting nodes and battery-powered nodes. In this scheme, like the scheme introduced in Chapter 4, the nodes are arranged in several virtual layers, and the nodes on the upper layer are given the job of aggregation data received from the nodes on the lower layers, and sending it to a sink node. The scheme balances the residual energy of the entire network and prolongs the network lifetime by using the harvested energy of energy-harvesting nodes.

Chapter 5 summarizes the results and concludes this thesis.

Chapter 2

Background and Related Work

2.1 Wireless Sensor Networks

Wireless Sensor Networks(WSNs) have recently come into prominence in various areas such as military, academia, environment and et cetera. WSNs are used as sensing ambient conditions, e.g., temperature, humidity, movement, sound, light, vibration and gases, detecting disaster, tactical surveillance, diagnosis in mechanical and structure monitoring, and recently it also used as patient monitoring. Specifically, WSNs can be easily applied to hard-to-access locations such as underwater, volcanoes or vast area because nodes in WSNs do not need wires. Figure 2.1 shows the overview of a WSN, which consists of wireless sensor nodes detecting an event and notifying it, sink nodes collecting the data sent from sensor nodes. Users can get the gathered data from the WSN by accessing the sink nodes. The design, implementation and operation of a WSN requires the confluence of many disciplines, including signal processing networking and protocols, embedded system, information manage-

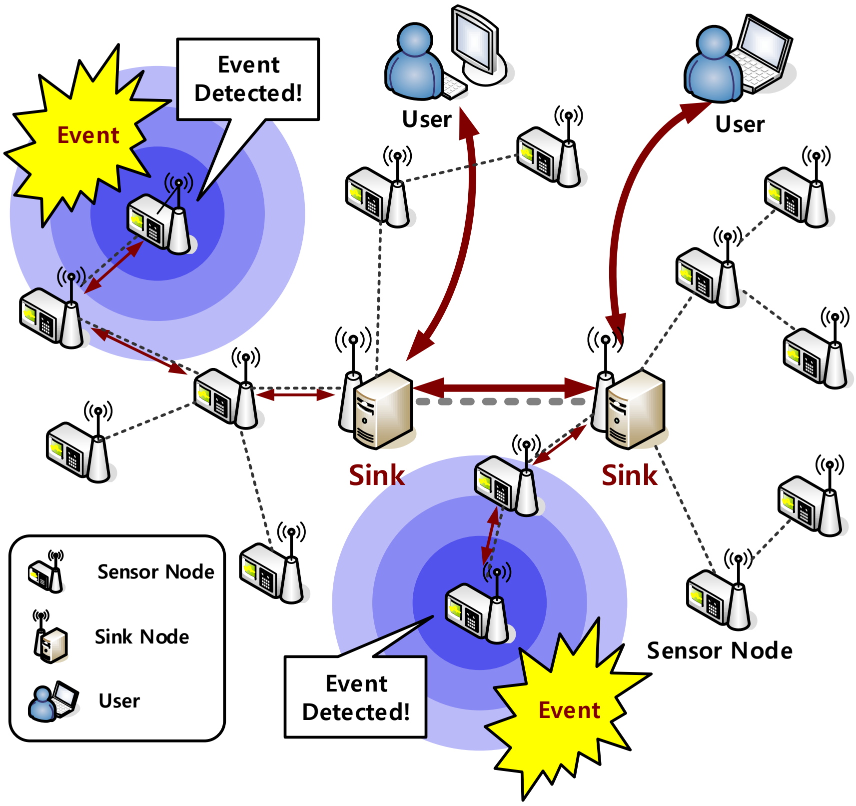


Figure 2.1 General architecture of a WSN.

ment, and distributed algorithms. This section introduces the field of WSN by synthesizing the diverse literature on key elements of WSN design.

2.1.1 Characteristics of WSNs

WSNs are very application-specific, thus they have a lot of challenging issues and engineering problems because of their flexibility. Accordingly, there is no single set of requirements that clearly classifies all WSNs, and there is also not a single technical solution that encompasses the entire design space. However, there are some of the required mechanisms that form typical pars of WSNs [28].

Multi-hop wireless communication Wireless communication is a core technique in a WSN, but a direct communication between nodes has some limitations. Particularly, communication over long distances makes nodes spend a lot of energy due to the high transmission power and it shortens their lifetime. In order to reduce the transmission distance, intermediate nodes have to be used as relay nodes. Therefore, for many forms of WSNs, so-called multi-hop communication will be a necessary ingredient.

Energy-efficient operation Most sensor nodes in WSN have a limited lifetime because they operate with batteries. Therefore, energy-efficient operation is one of the most important issues to support a long lifetime of a WSN.

Auto-configuration A WSN has to configure most of its operational parameters autonomously. For example, nodes should be able to determine their geographical locations only using other nodes of the network. Also, the network should be able to tolerate failing nodes or to integrate new nodes.

Collaboration and in-network processing In some applications, several sensor nodes have to collaborate to get accurate information whether an event occurs. Information is processed in the network itself to achieve this collaboration, as opposed to having every node transmit all data to the sink node and process it.

Data centric In WSNs, what is important are the answers and values of nodes, not which node has provided them. The data-centric approach is closely related to query concepts known from databases; it also combines well with collaboration, in-network processing, and aggregation.

Locality Nodes, which are very limited in resources should attempt to limit the state that they accumulate during protocol processing to only information about their direct neighbors. The hope is that this will allow the network to scale to large numbers of nodes without having to rely on powerful processing at each single node.

A WSN applies ad hoc manner to communication between nodes, similar to conventional Mobile Ad Hoc NETwork (MANET). However, there general problems are shared between MANETs and WSNs. Nonetheless, Table 2.1 shows some differences between the two concepts warranting a distinction between them and regarding separate research efforts for each one.

2.1.2 WSN Applications

Recently, the WSN applications have emerged as well as researches. The applications are classified into two types: an event detection and a periodic measurement.

Event detection One of the simplest WSN applications is *event detection*. In event detection applications, when a node senses a specific event it notifies the event to a sink node. Moreover, when the event occurs, some nodes can start the actuator of themselves in order to handle the event(e.g. when a fire breaks out the node detecting it turns on a sprinkler system to extinguish the fire.). There are some event detection applications in WSNs as follows:

Table 2.1 Principal differences between MANETs and WSNs

Attribute	MANET	WSN
Equipments	Fairly powerful with large batteries	Cheep and week devices
Environment interaction	Conventional applications with well understood characteristics	Very low data rates but burst traffic
Scale	Relatively small number of entities	Much larger number of entities
Energy	Limited	Limited but more important than MANETs
Self configurability	Required to self-configure into connected networks	Similar to MANETs
Dependability	Fairly reliable	Irrelevant
QoS	Similar to traditional applications	Entirely new QoS concepts are required
Data centric	Irrelevant	Redundant deployment will make data-centric protocols attractive
Mobility	Caused by nodes moving around	Sinks can be mobile as well

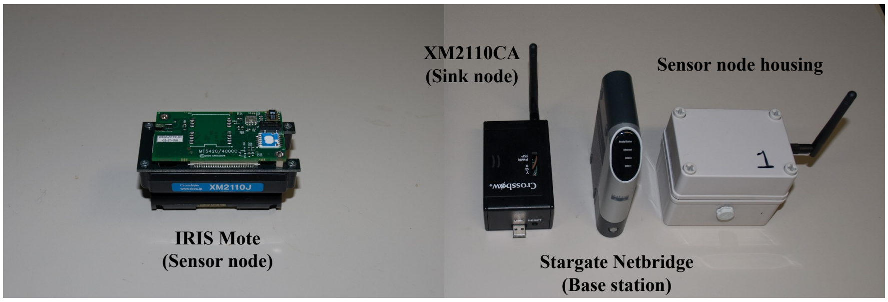


Figure 2.2 Components of a WSN for wildfire monitoring system.

- Disaster monitoring system: A WSN can be applied to wildfire monitoring system [8, 29–32] in which nodes measure temperature and humidity of environment in order to detect a wildfire, and notify it to a sink node when a fire break out, and volcanic eruption monitoring system [1, 33] in which nodes measure heat, gases and et cetera to detect the volcanic eruptions. Figure 2.2 and 2.3 show a wildfire monitoring system [29]. Wires cannot be used to develop the monitoring system in difficult-to-reach areas, so a WSN using no wires would be an efficient method.

  [1] designs a volcanic eruption monitoring system using one gateway and three sensor nodes which have no energy harvesting units. Figure 2.4 shows the volcanic eruption monitoring system using a WSN [1]. The system employs packet retransmission scheme to increase the transmission success rate. However, the transmission success rate still could be low depending on the environmental situation.

- Intruder surveillance system: The second type of the event detection application is intruder surveillance system [2, 34–43]. In the intruder surveillance system, sensor nodes sense infrared, sound or laser to detect motion or track objects. When the system have to move frequently

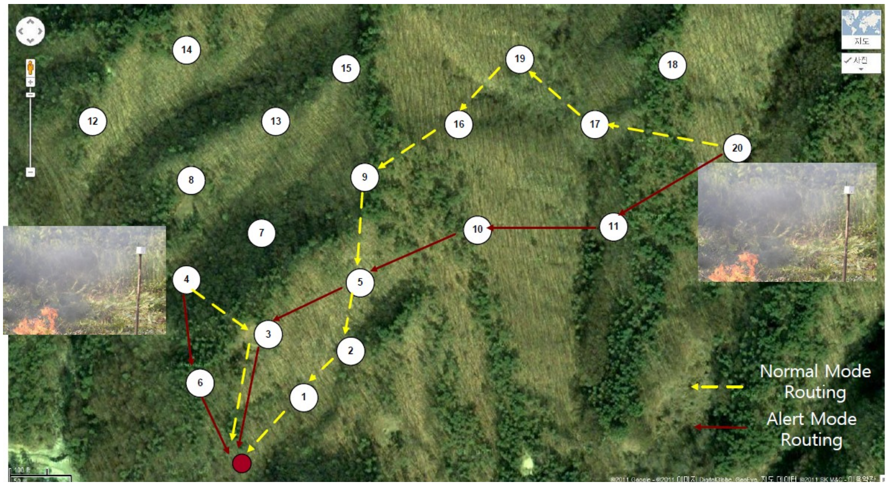


Figure 2.3 Node deployment of a WSN for wildfire monitoring system.

or deployed in a vast area, a WSN can effectively reduce the installation cost compared with traditional wired networks.

Last, He, et al. [2] describes one of the major efforts in WSNs to build an integrated system for surveillance missions. The VigilNet allows sensor devices to detect and track the positions of moving vehicles in an energy-efficient and stealthy manner. It uses 70 nodes to manage failures by rebuilding the route periodically. Figure 2.5 describes the overview of VigilNet [2].

Periodic measurements Wireless sensor nodes can periodically read sensory data, and send it to a sink node. It is mainly applied to monitor environmental information as follows:

- Environmental monitoring system: Environmental information such as temperature, humidity, light, vibration and sound can be periodically collected by sensor nodes in order to monitor habitats or agricultural

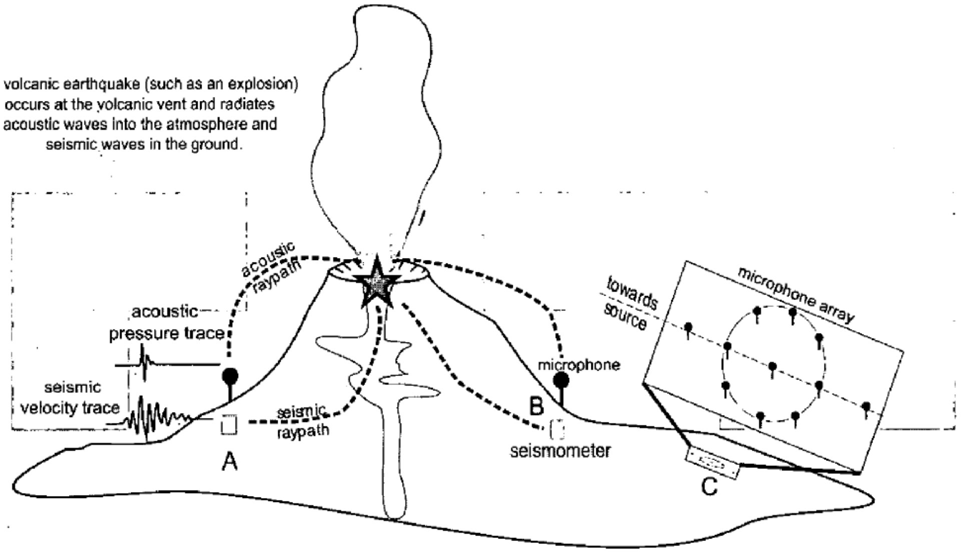


Figure 2.4 Overview of volcanic eruption monitoring system [1].

environments [2, 3, 9, 44–49]. A WSN is expected to drastically reduce the installation cost compared with wired networks because many sensor nodes should be deployed in a vast area for environmental monitoring.

A WSN which observes rare species of plants in Hawaii Volcanoes National Park is presented in [47]. In the WSN, battery powered sensor nodes are equipped with cameras and other sensors. They employ a multi-hop routing protocol to route packets adaptively and implement a synchronization scheme. Baggio [3] designs a system to monitor a crop field. Each node has temperature and humidity sensors. It consumes relatively small amount of energy because its application is based on low duty cycle. It employs multi-hop communication protocol to deliver data to a sink node. Some nodes which have no sensor module are deployed to relay packets between nodes. Figure 2.6 show the crop field monitoring system [3].

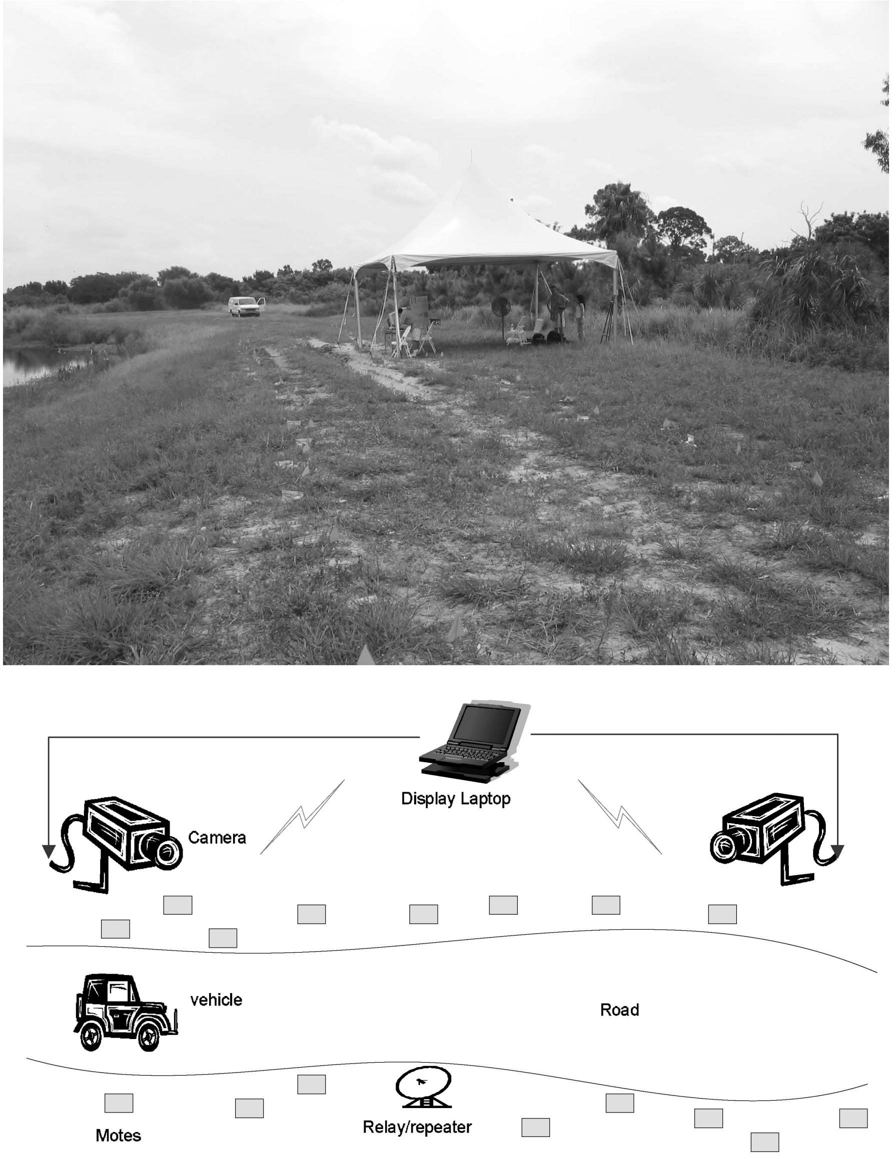


Figure 2.5 Overview of VigilNet: the integrated system tracking the position of moving vehicles [2].

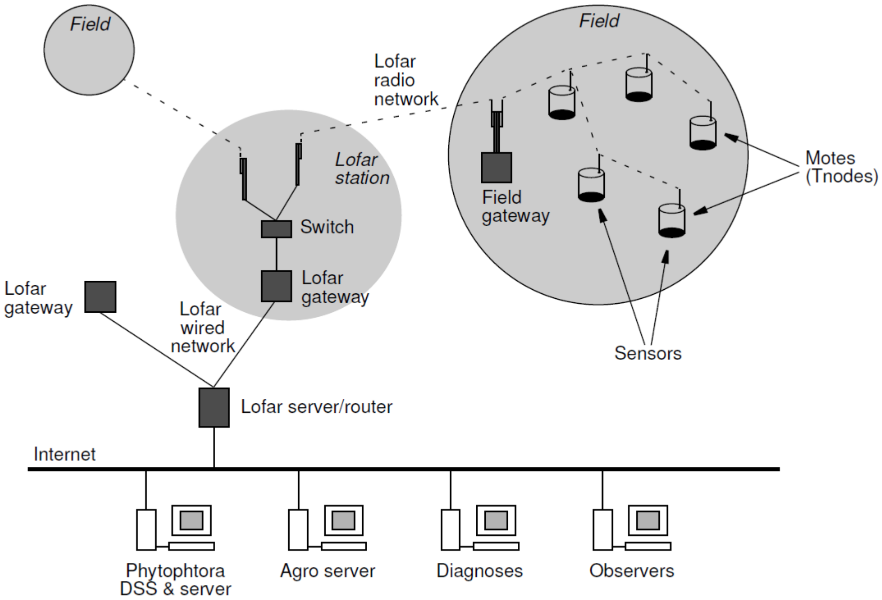


Figure 2.6 Overview of crop field monitoring system [3].

- Health care system: Some patients have to be monitored for health care. Wireless wearable sensor devices can periodically gather information such as heartbeat, blood pressure and body heat [4, 50–59]. Generally, people tend to hesitate to equip heavy electric devices on their body, so sensor nodes which use no wires would be useful. Figure 2.7 shows the health care tracking and monitoring system using a WSN [4].

- Structure monitoring system: We can utilize WSNs to monitor architectural structures by sensing corrosiveness, concussion and solidity in order to discover risks [5, 60–65]. Particularly, WSNs are useful when we cannot use wires to install sensor nodes on the structure. Figure 2.8 shows the underground structure monitoring system [5].

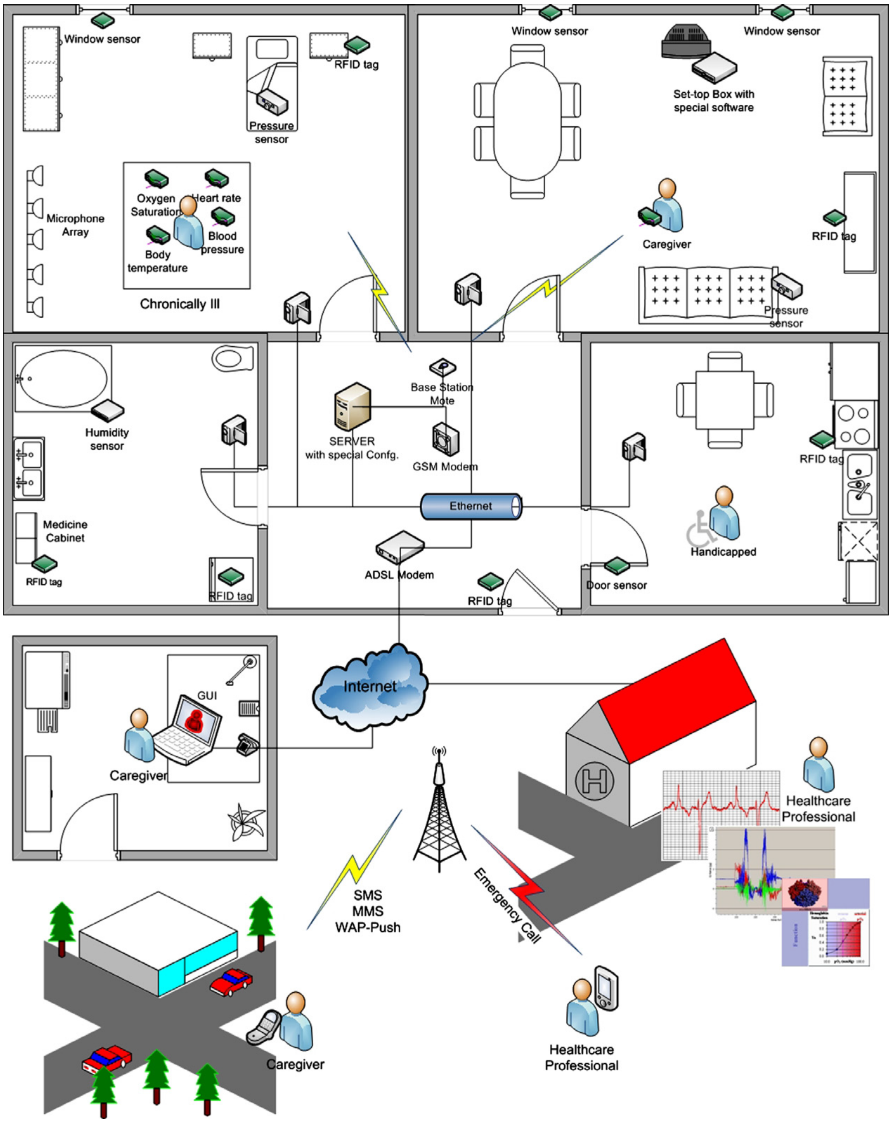


Figure 2.7 Overview of patient tracking and monitoring system for health care [4].

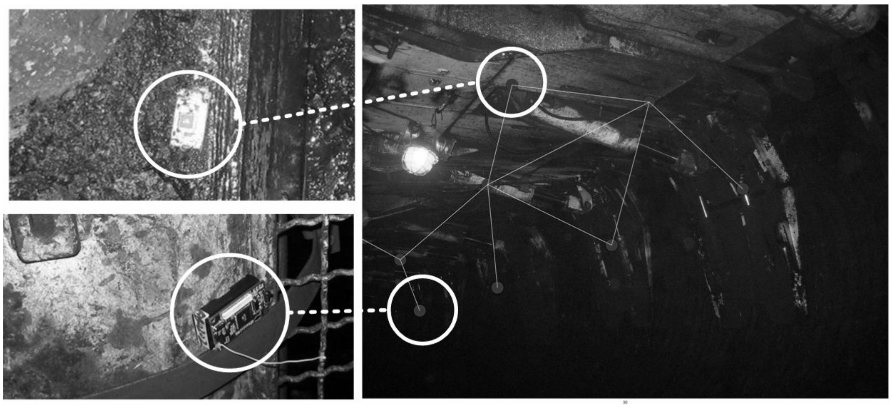


Figure 2.8 Overview of underground structure(coal mine) monitoring system [5].

2.1.3 Topology Control for WSNs

In WSNs, as described above, resources such as computation power, available energy or the accuracy of sensing unit have been limited. Therefore, the insufficient resources are the most principal issue in WSN. A sensor network has overcome the issue by large number of nodes and their coordination. The large number of nodes creates the redundancy and resource under utilization problem which reduces networks effective throughput. WSN solution provider tries to optimize this trade-off at every level of network design. Topology control is one of the ways through which researchers are trying to optimize this trade-off in WSNs [66].

Topology control is a mechanism that determines how nodes connect each other in order to reduce the size of network for better resource utilization during keeping link connectivity. Topology control mechanisms can be classified based on structure that topology control produces like, flat network and hierarchical network. Figure 2.9 shows the two kinds of topology control mechanisms

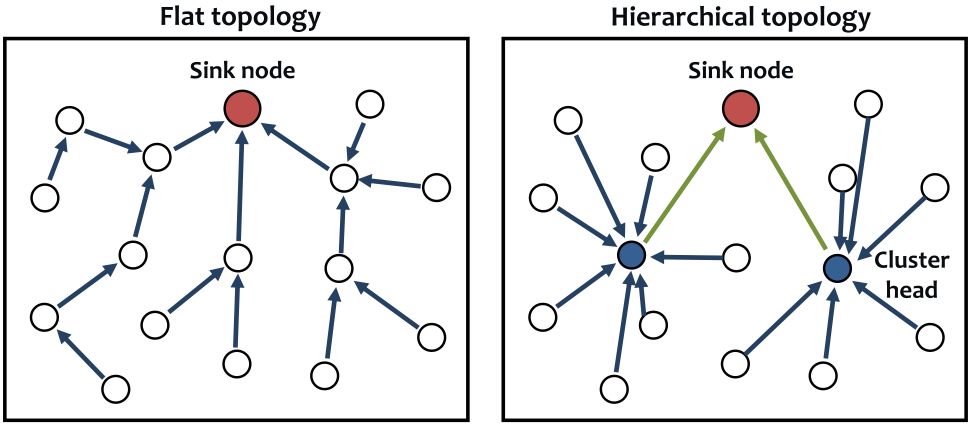


Figure 2.9 Flat and hierarchical topologies for WSNs.

for WSNs.

Flat topology control

In flat topology control all nodes are considered to be same nodes and performs the same functionalities specifically for data forwarding. Sensor nodes and communication patterns between nodes have been represented as graphs in flat network. There have been many graph model like Unit disk graph [67], Yao graph [68], Relative neighborhood graph [69, 70], Gabriel graph [71] and Further extension of this model is Encloser graph [72,73], Voronoi graph [74,75] and Delaunay graph [76].

Hierarchical topology control

In flat topology it has been considered that all nodes in network are of same type and performs same functionality for entire network life cycle. Generally, flat topology control schemes have *energy imbalance* problem according to the distance from the sink node as shown in Figure 2.10 and can not support

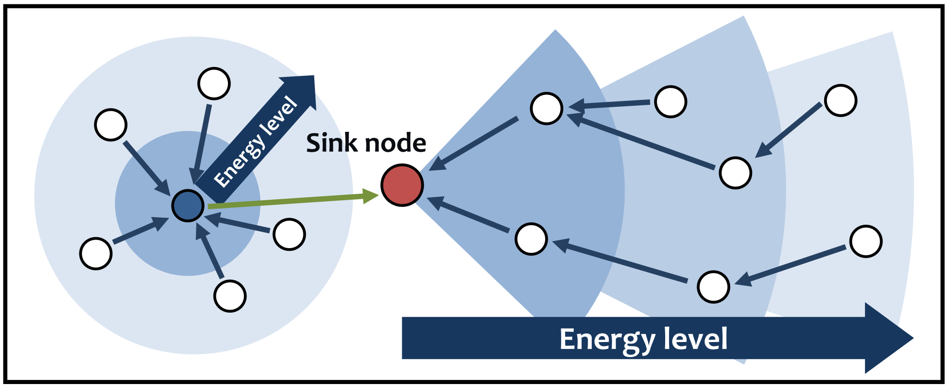


Figure 2.10 Energy imbalance of nodes according to their locations.

various heterogeneous nodes. In order to overcome these problems, hierarchical topology control schemes have emerged. In hierarchical topology control nodes are classified into aggregation nodes, ordinary nodes, redundant nodes and et cetera, and connects them hierarchically. Aggregation nodes gather sensory data from ordinary nodes and send the aggregated data to a sink node, then it effectively reduces the number of packet traversing the network. However, it may have energy imbalance problem because the aggregation nodes spend more energy than ordinary nodes. In order to solve this problem, nodes should change their roles periodically.

The most famous scheme of hierarchical topology is LEACH [6, 15]. In LEACH all nodes periodically organize the certain number of clusters. Each node determines that it will be a cluster head by probability. Nodes which are not selected as cluster head recently, has high probability of being a cluster head. Figure 2.11 depicts the overview of LEACH [15]. Another probabilistic hierarchical topology control scheme is HEED [17], which uses two level of cluster head selection parameters. The first level parameters are used to declare

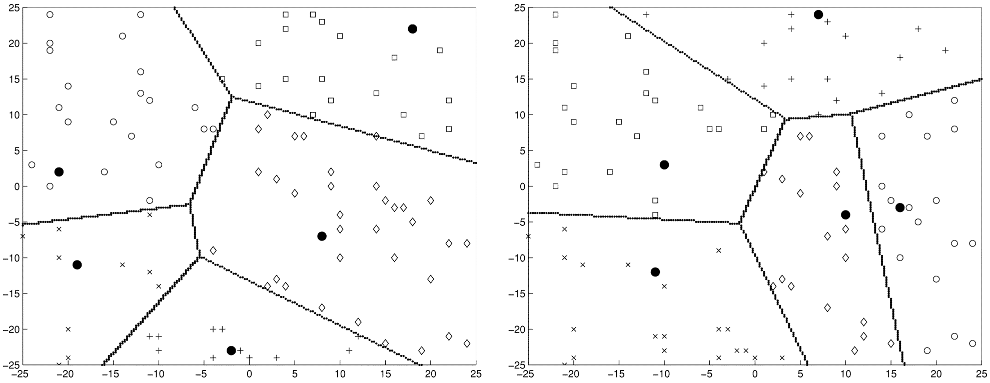


Figure 2.11 Dynamic cluster formation during two different rounds of LEACH [6].

cluster heads and the second level parameters are used to break tie between cluster heads. There have been several hierarchical topology control schemes, which are GAF [14], PEGASIS [18], SPAN [16] and et cetera.

Also, there have been multi-layer topology control schemes. Fan et al. [19] proposed a three-layer topology for large-scale WSNs in which each layer has its own role. They provide an algorithm that finds all the bottleneck nodes and eliminates them. Intanagonwiwat et al. [20] create a tree which is limited to a single sink at its root. Paths from sensor nodes nearer to the sink are added earlier gradually. Other authors [21–23] have also utilized a three-layer topology as the basis for control algorithms for static WSNs. Sharma et al. [21] create high-performance clusters and zonal sink nodes in order to increase the lifetime of sensor nodes. Duan et al. [22] used several fusion nodes and a control node to reduce the energy consumption of mobile nodes. Ming et al. [23] introduced a logical three-tiered TC model which consist of relay nodes, application nodes and sensor nodes. Their cluster-heads organize themselves into a near-uniform distribution across the network.

2.1.4 WSN using multiple sink nodes

In a general WSN, a single sink node gathers data, but if several sink nodes are used in a network, they can distribute the heavy load from all nodes efficiently. As a result, they can make network lifetime increases.

Oyman and Ersoy [26] investigated the choice of multiple sinks. They use a cluster scheme in which the sinks act as cluster-heads, and each node only reports to one cluster-head. Das and Dutta [77] proposed a similar method of sink selection, designed to minimize overall energy consumption. Buratti et al. [78–80] considered the reachability of multiple sinks in tree-based topologies with fixed node and sink densities. Kim and Lee [81], and Fan et al. [19] used a spanning tree to control a topology in multi-sink WSNs. Kim and Lee's scheme reconfigures itself automatically when nodes fail, increasing network lifetime. Ciciriello et al. [82] introduced a scheme in which the search for a multi-sink topology is mapped to a multi-commodity network design problem. This scheme periodically adapts message paths to reduce the number of network links that are used, increasing the efficiency of data transmission.

2.2 Energy-harvesting WSNs

In general, WSNs have a limited lifetime because they consist of battery-powered wireless sensor nodes. There have been many proposals [27] to use energy-harvesting nodes to extend the lifetime of WSNs. The environmental sources from which it is practical for nodes to harvest energy are the sun [83–88] and the wind [88, 89]. Solar-powered wireless sensor nodes have been preferred [90] because of the high power density of solar power (about 15 mW/cm^3).

Many schemes have been proposed to maximize the utilization of solar energy, with the aim of extending the life of a WSN indefinitely. Kansal et al. [91] proposed an energy model for solar-powered nodes, which estimates the levels of energy consumption and harvesting which will allow a node to survive indefinitely. Noh et al. [92] designed an energy allocation scheme which estimates the energy that will be harvested by a solar-cell every hour, and determines future consumption levels which will allow the node to survive; a complementary routing algorithm has also been presented [93].

The efficient use of solar energy requires accurate prediction, which is relatively easy because of its obvious relation to the diurnal cycle. Nevertheless, the availability of sunlight is of course affected by weather. Kansal et al. [83], Piorno et al. [94], Moser et al. [95] and Cammarano et al. [96] have all proposed models for estimating the availability of solar energy, using respectively an exponentially weighted moving-average (EWMA), the so-called weather-conditioned moving average (WCMA), a weighted sum of historical data, and Pro-Energy model.

2.2.1 Hierarchical Topology Control for WSNs with Energy-Harvesting Nodes

Topology control, which determines how nodes connect to each other, is an important issue which affects WSN performance. Only relatively recently has it been applied to WSNs which use energy-harvesting nodes.

Voigt et al. [97] introduced the solar-aware LEACH algorithm (sLEACH) which applies LEACH [6] to WSNs consisting of both solar-powered and battery-powered nodes. The sLEACH algorithm strongly favors the use of solar-powered nodes as cluster heads, which have to do more work than others. LEACH is effective in equalizing energy usage across (battery-powered) nodes, but sLEACH cannot be applied to large WSNs because all the cluster heads must connect directly to the sink. In addition, sLEACH can allow the harvested energy to be under-utilized, or run out, because sLEACH does not consider the rate at which each of solar cell acquires energy, and the energy level of each node.

Zhang et al. [98, 99] proposed a scheme which overcomes the need for each cluster head to connect directly to the sink, by using further energy-harvesting nodes to relay transmissions from the cluster head to the sink. This can also save energy by reducing the range over which cluster heads are required to transmit.

Gou et al. [100] also addressed the drawbacks of sLEACH by partitioning a WSN and taking the energy remaining in each node into account.

Chapter 3

Multi-layer Topology Control for Long-term Wireless Sensor Networks

3.1 Layer-Based Topology Control

We now introduce the architecture of our scheme and explain how the nodes decide which layer they should belong to. The network we will discuss is composed of battery-based sensor nodes, each of which periodically transmits the data it has acquired to a sink node.

3.1.1 Proposed Scheme

Each node selects a layer to join from its context, which is primarily its residual energy and the amount of traffic it has to handle, and the nodes in each layer treat the nodes in higher layers as sinks, as they would in a multi-sink network. Figure 4.2 shows a simple three-layer topology that might be constructed at a particular time by our scheme. If a node can send its data to an upper-layer

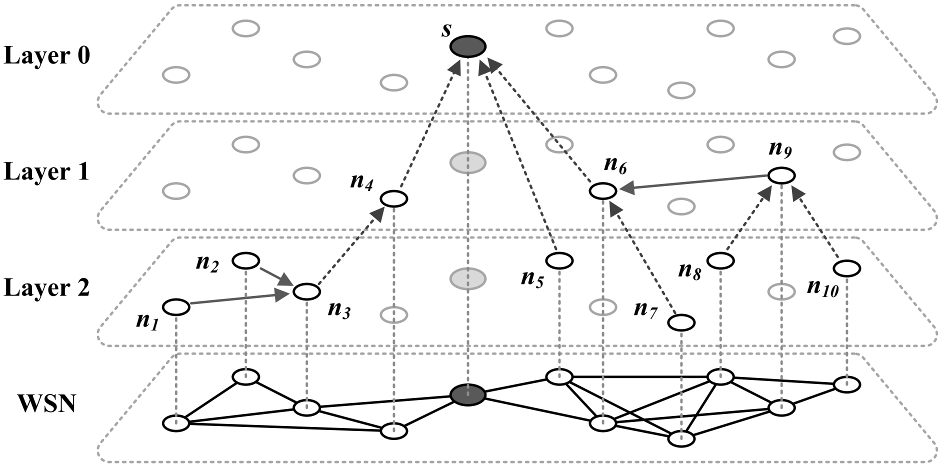


Figure 3.1 Overview of the layer-based topology.

node directly, it does so; otherwise it sends it to a neighbor on the same layer for relaying. In Figure 4.2, for example, nodes n_3, n_5, n_7, n_8 and n_{10} in Layer 2 send data to the upper-layer nodes. Node n_5, for example, can send data to a higher-layer node directly; but nodes n_1 and n_2 have to transmit data through neighboring nodes.

Since our scheme uses same-layer nodes for relaying, fewer void nodes are caused by the absence of cluster heads, as shown in Figure 3.2. We will explain this in more detail in Section 3.1.2.

3.1.2 The Layering Algorithm Design

We consider each node n_i to have a set N_i of neighbor nodes and each node sends data towards the sink node s at the end of every period P_G. The sink node s broadcasts topology control information (TCI) messages at the start of every setup period P_S by using data flooding. This data determines the layer of each node.

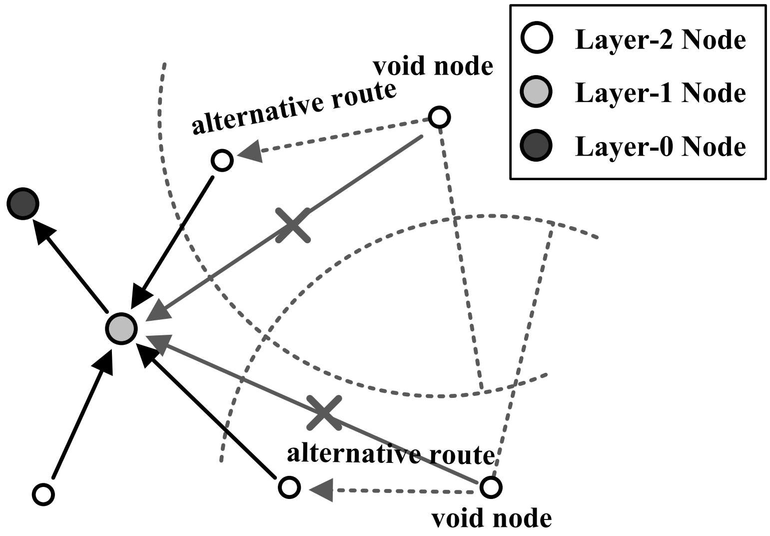


Figure 3.2 Prevention of void nodes.

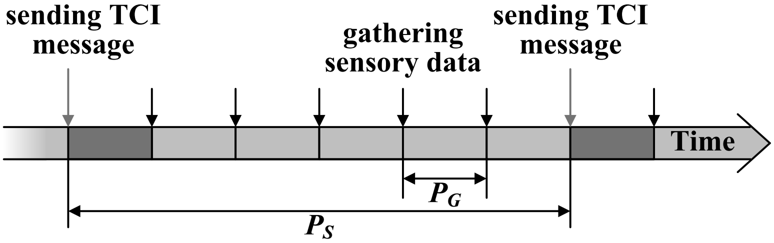


Figure 3.3 Stages in the topology control process.

Figure 3.3 shows the stages in our algorithm. First, the sink node s^0 (in Layer 0) broadcasts a *TCI* message for the current round, of length P_S. This message is composed by the sink node, using information gathered from all the nodes in the WSN before the setup period. We assume that P_S is even greater than P_G because our target is long-term WSN in which layer selection is rarely occurred. Thus the overhead of flooding *TCI* message is little enough to be negligible.

A node n_i can determine its layer j from this information, and can be written n_i^j. If a node n_i^j receives another *TCI* message from a node in a layer higher than j, it re-determines its layer from the new *TCI* message; but this layer must not be higher than the layer of the sender of the *TCI* message. Otherwise, the node would become void, as shown in Figure 3.4, and the data sent to it from lower-layer nodes would be lost. After it has determined its layer, a node adds that information to the *TCI* message it received, and broadcasts the revised message to its neighbors. Repetition of this process makes it possible for all the nodes to determine their own layer. Algorithm 1 is a formal statement of this procedure.

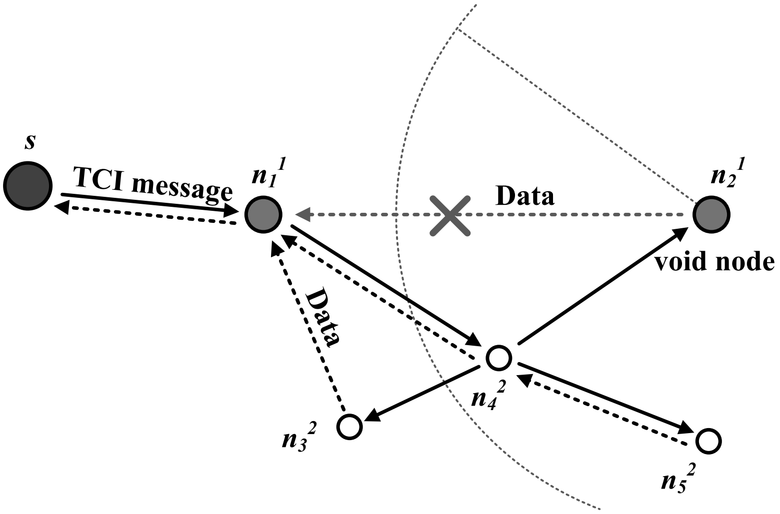


Figure 3.4 Void node created by the incorrect delivery of *TCI* messages.

3.2 Layer Determination

Each node determines its layer by comparing its expected lifetime with the average lifetime of all the nodes. We now explain this layer-determination algorithm in detail.

Algorithm 1 Layer-Based Topology Control Algorithm

```
Require: Sink node $s$ broadcasts the $TCI$ message.
Require: The layer $j$ of each node is MAX_LAYER
Require: $s$ calculates $\bar{R}$, the average expected lifetime of all nodes.
Ensure: Determining the layer $j$ for $n_i^j$
if $n_i^j$ receives the $TCI$ message from $n_k^a$ then
    if $j > a$ then
        $l \leftarrow j$
        $j \leftarrow a$
        while $j < l$ do
            calculating the estimated lifetime $R_i^j$ of $n_i^j$        ▷ Equation (3.6)
            if $R_i^j > \bar{R}$ then
                broadcasting the $TCI$ message
                                        ▷ $j$ is selected as the new layer for $n_i^j$
                return
            else
                $j \leftarrow j + 1$
            end if
        end while
    end if
end if
```

3.2.1 *TCI* message

Table 3.1 Node Information Required in Preparing a *TCI* message

Notation	Descriptions
j	Number of layers in which n_i^j is present
R_i^j	Expected lifetime of n_i^j (in rounds)
$L_{i\,relay}^j$	Amount of data that node n_i^j should relay on layer j during one round

All the nodes in the WSN send the information described in Table 3.1 to the sink node s, so that it can prepare the *TCI* message. From this information the sink node calculates C^j, the number of nodes currently in each layer, the average lifetime of all nodes, R, and the expected amount of data that each layer should relay, $\bar{L}_{relay}^j$. This information is put into the *TCI* message; Table 3.2 describes all the information contained in a *TCI* message.

Table 3.2 Information in a *TCI* Message

Notation	Descriptions
j	Number of the layer in which the sender resides
C^j	Total number of nodes in each layer
$\bar{R}$	Expected lifetime of the entire WSN (in rounds)
$\bar{L}_{relay}^j$	Amount of data that each layer should relay during one round
M	Number of layers in the WSN

3.2.2 How a node selects its layer

A node receiving a *TCI* message determines its layer by comparing $\bar{R}$ with its expected lifetime. This is calculated from the following energy model [101]:

$$e_{consume} = e_{trans} + e_{receive} + e_{elec}, \tag{3.1}$$

where e_{trans} and $e_{receive}$ respectively are the amount of energy consumed when the node sends and receives packets, and e_{elec} is the energy consumed by the electronics. If packets are transmitted for P seconds and the transmission range is tr, the amount of energy consumed is

$$\int_P e_{consume} = L_{trans}\beta tr^{\alpha} + \int_P (e_{receive} + e_{elec}), \tag{3.2}$$

where L_{trans} is the number of bits in packets transmitted for P seconds, α is the path loss ($2 \leq \alpha \leq 5$), and β is the energy used by the power amplifier for transmitting 1 bit over a distance of 1m. Thus we can calculate the energy consumed in one round by a node n_i belonging to layer j, as follows:

$$e_{round}{}_i^j = \left\lfloor \frac{P}{P_G} \right\rfloor L^j_{trans}\beta tr_i^{\alpha} + \int_{P_S} (e_{receive} + e_{elec}) \, . \tag{3.3}$$

L^j_{trans} is the size of a packet, which is the sum of the packet header size L_{head}, the amount of data c_i^j received from a lower layer, and the amount of data L^j_{relay} to be relayed from nodes on the same layer, as follows:

$$L^j_{trans} = L_{head} + (L_{data}c_i^j) + L^j_{relay}, \tag{3.4}$$

where c_i^j is the number of lower-layer nodes that send data to n_i^j. However, we cannot measure the value of c_i^j accurately because it can change with the topology. Therefore we estimate c_i^j from h_i^j, the number of nodes in each layer that sent data to n_i^j during a previous round, and N^j, the total number of nodes in each layer, as follows:

$$c_i^j = \frac{\sum\limits_{k=j+1, h_i^k=-1}^{M} N^k}{\sum\limits_{k=0}^{j} N^j + 1} + \sum_{k=j+1, h_i^k \neq -1}^{M} h_i^k, \quad \text{initial} \quad h_i^k = -1 \tag{3.5}$$

If use replace L_{relay}^j in Equation 3.4 with $\bar{L}_{relay}^j$, and use an average path loss calculated during a previous round as α because we can not get an accurate value, then we use the modified equation to determine the energy consumed by node n_i^j in layer j during the current round. The lifetime of n_i^j can be calculated from $e_{round_i^j}$ as follows:

$$R_i^j = \left\lfloor \frac{e_{remaini}}{e_{round_i^j}} \right\rfloor. \tag{3.6}$$

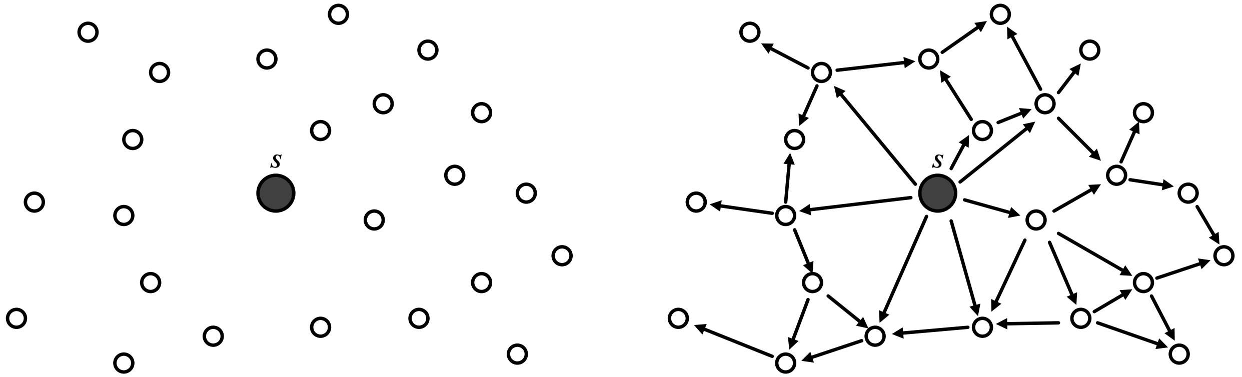
This computation is repeated for decreasing values of j: the first value of j which satisfies $R_j^j > \bar{R}$ determines the layer of the node. If this inequality is never satisfied, the node enters the lowest layer M. Getting nodes to choose their layers dynamically in this way moves nodes with a lot of energy into higher layers, where they will work harder, while nodes that are nearly exhausted go to lower layers, where they can conserve energy.

3.3 Experimental Results

We wrote a simulation in C++ to evaluate the performance of the proposed scheme.

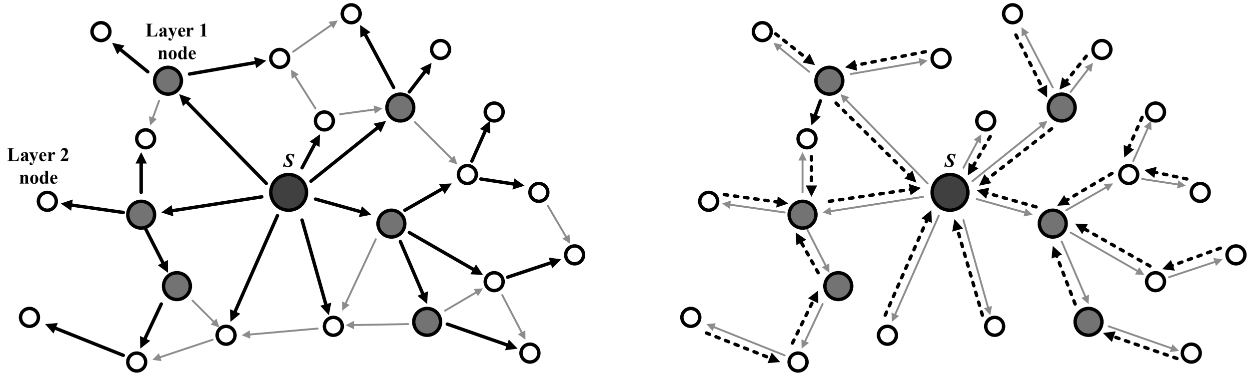
3.3.1 Simulation Environment

From the simulation results, we computed the average lifetime of sensor nodes, and its standard deviation, the average number of data packets transmitted, and the number of dead nodes. We considered a flat network topology, and two, three, and four layers. We simulated a network of 1000 nodes at a density of 0.02 to 0.1nodes/m^2, and each test set was run for 6000 rounds and repeated 10 times; then the results use averaged. At the beginning of each round, all nodes were classified into layers (except for the flat topology) using Algorithm 1. For example, Figure 3.6 shows the distribution of nodes in each of three layers. The nodes transmitted data to a sink node using the AODV algorithm [102] at an interval of P_G. Each sink node broadcasts a TCI message (similar to a Routing REQuest message in AODV) and all nodes send their data along the reverse of the path that TCI message pass through (similar to a Routing REPly message in AODV). Then the nodes in a higher layer aggregate the data received from the lower-layer nodes with their own data, and transmit all the data to the next node. Figure 3.5 depicts an example of our simulation process. Each node is powered by a battery of finite life, and dead nodes are immediately replaced with new nodes in the same position. Table 4.4 contains the important parameters used in our simulation.



(a) Nodes are deployed on random locations.

(b) Sink node *s* floods TCI messages



(c) Nodes select their own layers and reverse paths.

(d) Nodes send data to sink node *s* along the reverse paths

Figure 3.5 An example of our simulation process.

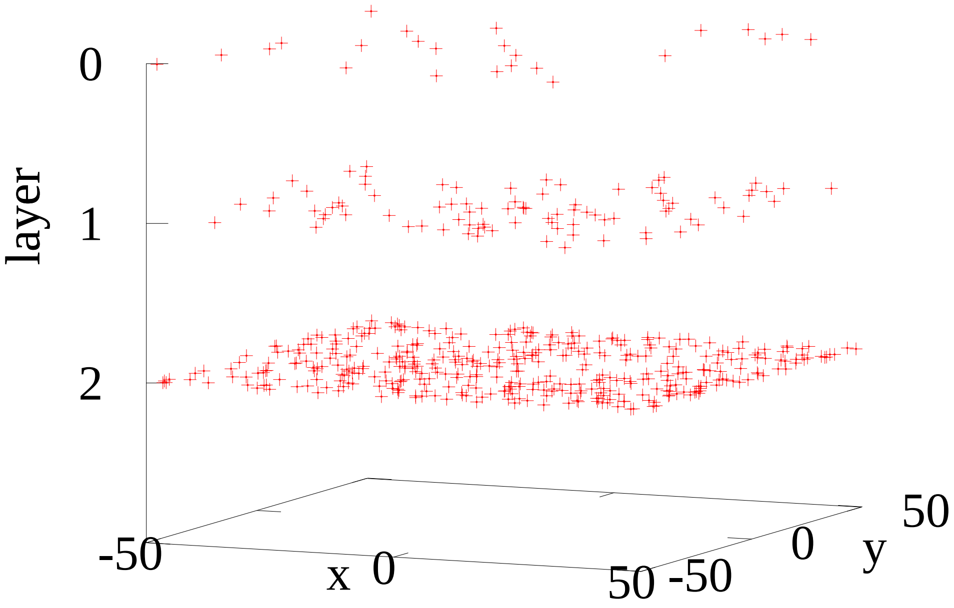


Figure 3.6 Distribution of nodes in each layer in a three-layer simulation.

3.3.2 Simulation Results

Figures 3.7 and 3.8 respectively show the number of packets in the network and the average lifetime of the nodes, measured from our simulation. Figure 3.8 shows that the layered topologies increased the average lifetime of a node by 2.1 to 22%, compared to the flat topology. In general more layers give a higher performance. Figure 3.7 shows that the number of packets transmitted decreased by between 30 and 50% as the density of the nodes increased. As node density decreases, the opportunity to receive a *TCI* message direct from

Table 3.3 Important Simulation Parameters

Parameter	Value
Number of nodes	1000
Node density	$0.02 \sim 0.1$
Node placement	Random
Amount of data of a packet	100bytes
Size of header of a packet	40bytes
Transmission range	$10 \sim 20$m
Maximum number of layers	$1 \sim 4$
α and β in Equation (4.3)	3 and 100pJ/bit/m^2
e_{elec} and $e_{receive}$ in Equation (4.3)	0.003J and 0.066J
P_S and P_G in Equation (3.3)	1000s and 100s

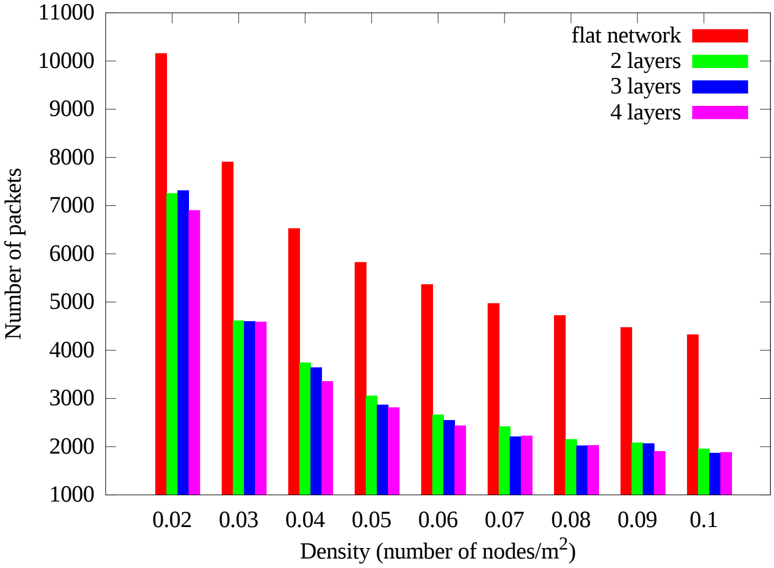


Figure 3.7 Number of packets transmitted.

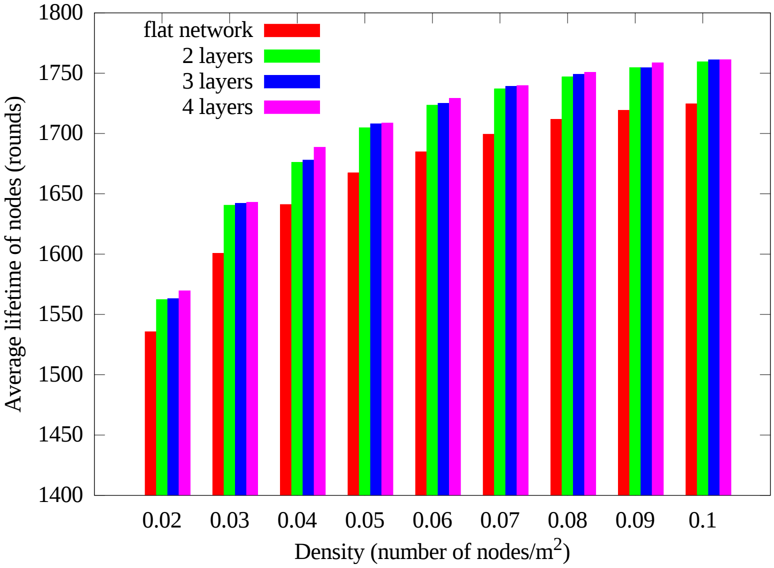


Figure 3.8 Average lifetime of nodes.

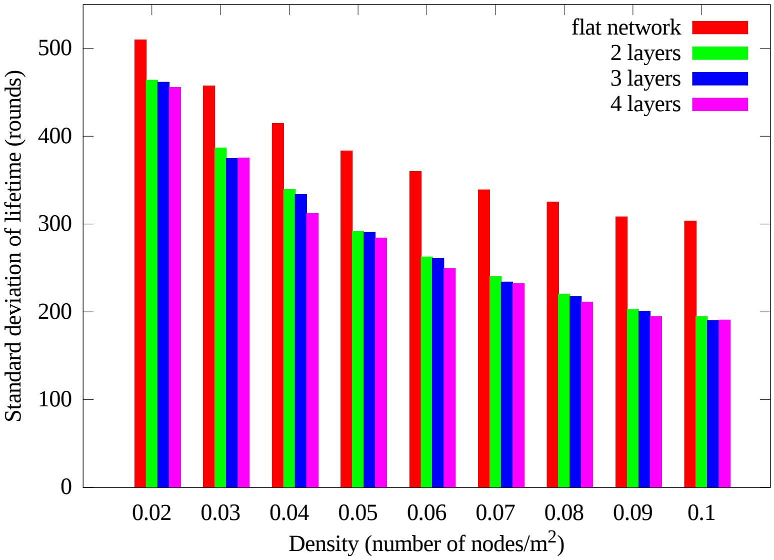


Figure 3.9 The effect of node density on the standard deviation of node lifetime.

a higher layer decreases. Thus there are fewer nodes in the higher layers, and there is less data aggregation. Conversely, as the density of nodes increases, the fewer packets are transmitted, and the lifetime of the WSN increases, as shown in Figure 3.7. Our simulation used a simple data aggregation strategy, in which a node which receives data from a lower layer removes the headers and forwards the incoming data with its own data. This strategy only eliminates the packet headers received from lower-layer nodes, and thus the average lifetime of the WSN did not increase very greatly. A more efficient aggregation scheme would have much more effect on the amount of data to be sent to upper-layer nodes and we could expect the lifetime of the WSN to improve significantly.

Figure 3.9 shows the standard deviation of the lifetime of nodes. This is quite high because the nodes closer to the sink node have a shorter lifetimes since they have to relay more data. Depending on the density of the nodes, the standard deviation is reduced by 9 to 36% by our scheme, compared to the flat topology.

Figure 3.10 shows the cumulative number of dead nodes. Using the flat topology, many nodes died early because of the high standard deviation of lifetime, whereas nodes separated into layers survived longer. This shows how our scheme can prolong the lifetime of a WSN.

Figure 3.11 shows the overhead of the topology control scheme, as the density of nodes was varied from 0.02 to 0.1. The overhead indicates the average consumed energy during a round for *TCI* messages of a node. In Figure 3.11, the overhead of the other schemes decreased as the density increased because the shorter hop made reduce the number of *TCI* messages. However the overhead of the topology control was heavier than one of the flat network because the *TCI* message has more information for topology control but one of flat topology has only routing information. 4-layer network consumed more energy than 2-layer network because the nodes in 4-layer network had more chances to move to other layers and retransmit *TCI* messages than nodes in 2-layer network.

Our scheme adjusts the number of layers adaptively, in response to the changing situations of the nodes, up to a specified maximum. Our experiments suggest that performance is usually improved by increasing the maximum number of layers, up to a certain number, after which there is no further improvement. For example, in Figure 3.8, when the density is 0.1, the average lifetime of a node is not increased by going from three layers to four. This suggests that the maximum number of layers should be determined by simulation before deployment, since it varies with the number of nodes, their density and distribution.

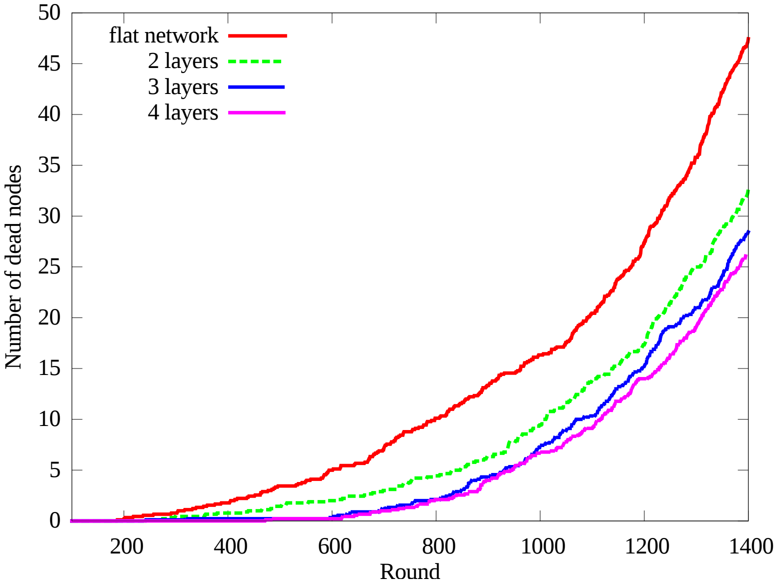


Figure 3.10 Cumulative number of dead nodes.

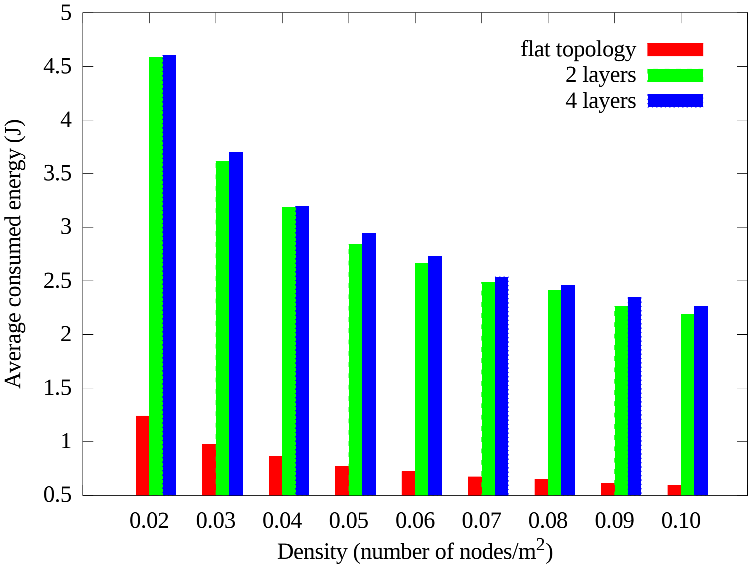


Figure 3.11 Overhead of each topology control scheme.

Chapter 4

Energy-aware Hierarchical Topology Control for Wireless Sensor Networks with Energy-Harvesting Nodes

4.1 Layer-based Topology Control with Energy-harvesting Sensor Nodes

We now introduce a new layer-based topology control scheme, and explain how to allocate nodes to layers, how nodes operate within the layers, and how the scheme applies to energy-harvesting nodes. We will assume that a WSN contains both battery-powered and energy-harvesting nodes, which periodically forward the data that they have gathered to the sink node. We also assume that defunct nodes will be replaced, because this is a long-term WSN.

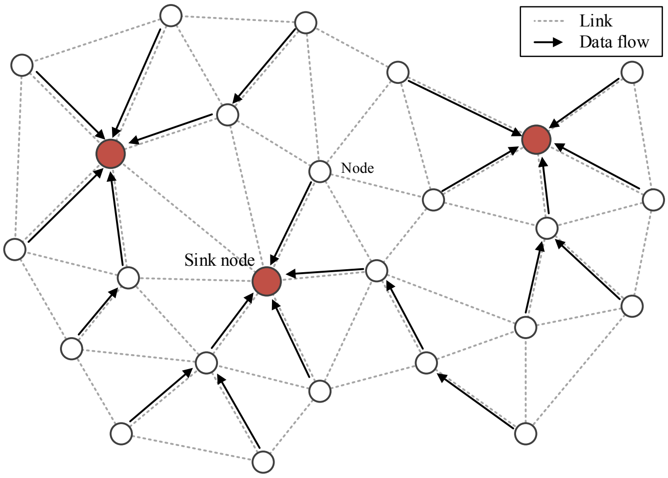


Figure 4.1 Operation of the multi-sink aware minimum-depth tree (m-MDT) algorithm.

4.1.1 Review of Layer-based Topology Control for Long-term WSNs with Battery-powered Nodes

In our previous layer-based topology control scheme [103], all the nodes gather data and periodically forward it to a sink node. Nodes are arranged in layers, and data travels hierarchically. This scheme has been shown to reduce the imbalance in available energy across the nodes.

Each node is periodically allocated to a layer on the basis of its estimated lifetime. The nodes in each layer affectively consider all higher-layer nodes as local sink nodes, relative to their own layer, and data is routed by the multi-sink-aware minimum-depth tree (m-MDT) algorithm [104]. Nodes in the upper layers forward the data received from lower-layer nodes, as well as the data they themselves have acquired, to nodes in yet higher layers. By repetition of these processes all the data arrives at the actual sink node. In order to prevent data being sent on void [105] or cyclic routes, nodes are not permitted to send data to nodes in lower layers. Figure 4.2 shows a topology with three layers.

Naturally, nodes in higher layers will have to transit more data, requiring more energy: This is why energy-rich nodes are placed in higher layers. Each node to determines its own layer, on the basis of the amount of energy that it has remaining, the amount of data which it is likely to have to transmit, and its estimated lifetime.

As we already mentioned, a drawback with this scheme is that the nodes' estimate of the amount of data which they will have to transmit during a subsequent data transmission phase is not accurate, because it is an average of the amount of data transmitted by all the nodes, whereas the actual amount depends on the location of a particular node. In addition, this scheme potentially allows all nodes to change their layers very frequently, making it difficult to determine which layer is actually best, because the amount of energy that a node forecasts it will consume in a layer is likely to be very different from its actual consumption when it move itself to that layer. Furthermore, our scheme does not consider energy-harvesting nodes, which is the rationale of this present paper. We address all these deficiencies in the new scheme which we now describe.

4.1.2 The Layer Determination Algorithm

We now introduce our improved layer-based topology control scheme and explain how each node determines its layer. Unfortunately, this requires a lot of notation, and so we summarize frequently used notation in Appendix **??**.

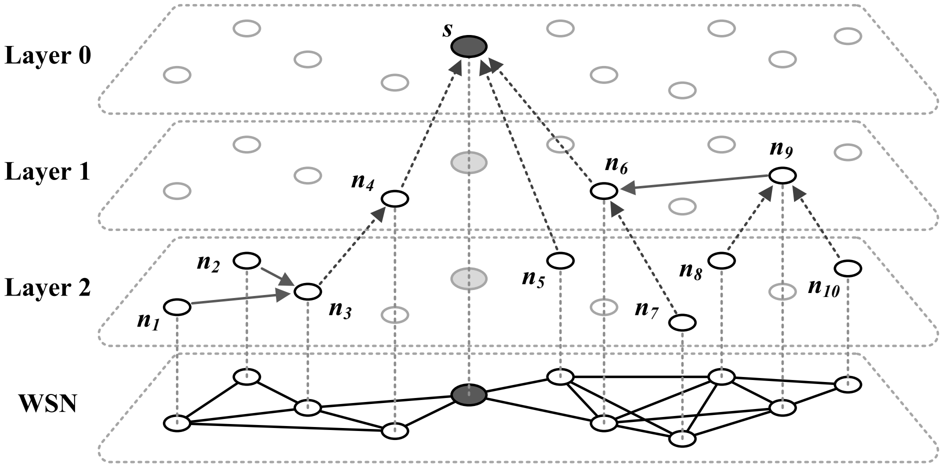


Figure 4.2 Overview of layer-based topology control.

Operation of a Node

When a node n_i is deployed in a network for the first time, it joins the lowest layer, and selects a target node among neighboring nodes in response to *Local topology control information (TCI)* messages from neighboring nodes. Node n_i then begins to gather information from the environment, and sends it to the sink node via its target node, during every gathering period p_{gather}. At the beginning of each round the sink node sends a *Global topology control information (TCI)* message to the entire network, using a flooding process. This occupies a setup period p_{setup}. Each node then begins its *Layer determination* process at the beginning of the subsequent layer determination period p_{layer_i}, or earlier, if it has received the *Global TCI* message from the sink node. This process determines a nodes' new layer. If a node n_i is moving to another layer, then it sends a *Layer Notification* message to its neighbors, and they may have to selects new target nodes as a result. The logic of node operation is shown in Figure 4.3, and Figure 4.4 shows an example timeline.

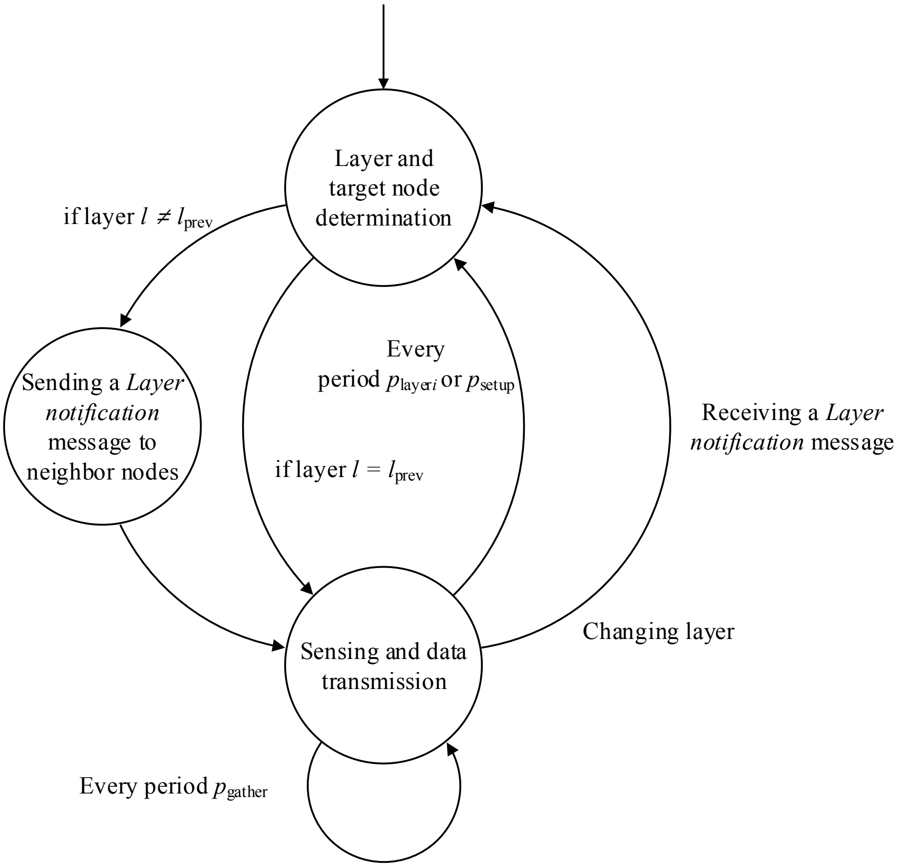


Figure 4.3 Node operation.

Topology Control Information for Layer and Target Determination

To determine the layer in which it should reside, a node needs information about its 1- and 2-hop neighbor nodes in the network. This information is delivered to nodes by *topology control information (TCI)* messages, which can be local or global, depending on their content and the range over which they are delivered.

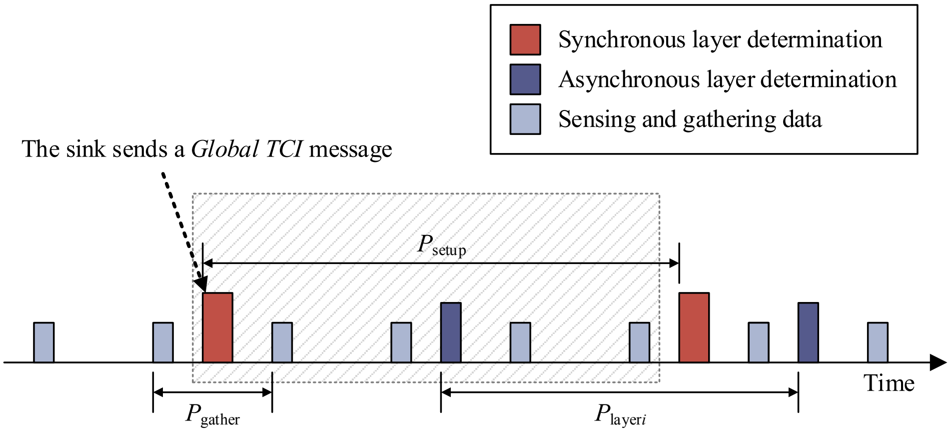


Figure 4.4 Example of node operation. The shaded box indicates a single round.

Global TCI At the beginning of each setup period p_{setup} a *Global TCI* message, containing information relating to topology control and routing, transmitted by the sink node to the entire network, using a flooding process. When a node which receives a *Global TCI* message, it executes the *Synchronous layer determination* process, presented as Algorithm 3, and may then change its layer. The sink node selects setup period p_{setup}. A longer setup period makes topology control more accurate, but the allowable topology control overhead is limited. Table 4.1 shows the contents of a *Global TCI* message.

Table 4.1 *Global TCI* message

Notation	Description
a	Index of this *TCI* message
i	Index of the sender node
j	The layer to which the sender belongs (for routing)

Local TCI A *Local TCI* message contains information about a node's 2-hop neighbors, the shortest hop-count from the sending node to an higher-layer node, and the nodes' expected lifetime, together with general neighborhood information commonly exchanged by nodes within WSNs. A node n_i^j periodically broadcasts to its 1-hop neighbors, following a generic WSN protocol at the beginning of its layer determination period p_{layer_i}. Tables 4.2 and 4.3 show the contents of a *Local TCI* message.

Table 4.2 *Local TCI* message

Notation	Description
i	Node index
j	The sender's layer
N	Set of neighbor nodes $n_k^l \in N$
h_i	Hop-count to the closest known node on a higher layer
L_i	Expected lifetime of the sender (in rounds)

Table 4.3 Information about each neighbor node in N

Notation	Description
s_k	Amount of data which this neighbor node transmitted during the previous gathering period p_{gather}
t_k	Neighbor's target node
l	Neighbor's layer
h_k	Shortest hop-count from this neighbor to a node on a higher layer
L_k	Expected lifetime of this neighbor (in rounds)

In our previous system [103], the average expected lifetime of all nodes was delivered to all the nodes in the WSN by means of a *Global TCI* message, and this information was used in determining each node's layer; but this procedure is not appropriate for WSNs in which nodes closer to a sink use more energy. In our new scheme, therefore, each node calculates a local average expected lifetime from information supplied by its 1- and 2-hop neighbors. This average is derived from the expected lifetime L_i of a node i, which is determined as follows:

$$L_i = \frac{E_{\text{remain}i}}{E_{\text{round}i}}, \tag{4.1}$$

where $E_{\text{remain}i}$ is the remaining energy in node i's battery, and $E_{\text{round}i}$ is the energy that node i consumed during the previous round.

Layer Determination

Each node n_i determines its layer by executing a *Layer determination* process at the beginning of its layer determination period p_{layer_i}, or when it receives a *Global TCI* message or a *Layer notification* message. There are two kinds of *Layer determination* process, as follows:

Asynchronous layer determination Node n_i^j performs this process at the start of every layer determination period p_{layer_i}.

It determines whether to move up one layer by computing L_i^{j+1}, which is the estimated lifetime of node i, if it moves up to layer $j+1$. The expression for L_i^{j+1} follows later, as Equation 4.9. Node n_i^j compares L_i^{j+1} with $\bar{L}_i$, the average local expected lifetime of its 1- and 2-hop neighbors. If L_i^{j+1} is greater than $\bar{L}_i$, then n_i^j ascends to the next higher layer, provided that it will have at least one node in its set of neighbor nodes N_i at the new level. Otherwise, n_i^j does not change layers, to avoid becoming a void node.

If node n_i^j moves from layer j to layer $j+1$ as a result of running the *Asynchronous layer determination* process, it broadcasts a *Layer notification* message to inform its neighbors of the change. A node which receives such a message redetermines its target node by running the *Target determination*

Algorithm 2 Asynchronous layer determination

```
Require: $p_{\text{layer}_i}$ of node $n_i^j$ expires
Ensure: Determine whether node $n_i^j$ moves to layer $j+1$
procedure LAYER DETERMINATION
    Calculate $L_i^{j+1}$ using Equation 4.9
    if $L_i^{j+1} > \bar{L} \wedge \exists n_k^l (l > j) \in N_i$ then
        $j \leftarrow j+1$
        TARGET DETERMINATION($j$, $N_i$)
        Broadcast Layer notification message
    end if
end procedure
procedure TARGET DETERMINATION($j$, $N_i$)
    if $S = \{n_k^l | l > j \wedge n_k^l \in N_i\} \neq \emptyset$ then
        $t_i \leftarrow n_k^l$(has minimum $h_k$) $\in S$
    else
        $t_i \leftarrow n_k^l$(has minimum $h_k$) $\in \{n_k^l | l = j, n_k^l \in N\}$
    end if
end procedure
```

process. This selects the node's neighbor $n_k^l (l > j)$ which has the minimum hop-count to an upper-layer node to be its new target node. The *Asynchronous layer determination* and *Target determination* processes are presented as Algorithm 2.

Synchronous layer determination This process determines whether a node will descend to the layer below its current layer. The sink node broadcasts a *Global TCI* message of the form described in Table 4.1, and every node which receives it runs the *Synchronous layer determination* process.

If node n_i^j receives a *Global TCI* message from node n_k^l, it calculates its expected lifetime L_i using Equation 4.1. If L_i is less than the local expected lifetime $\bar{L}_i$, it moves to the next lower layer $j-1$. Then it selects n_k^l as its target node, and sends a *Global TCI* message to its neighbors. However, if node n_i^j

subsequently receives a duplicate *Global TCI* messages from another node n_o^p, it runs the *Layer determination* process again. If this changes its choice of layer, it selects n_o^p as its target node t_i and broadcasts its *Global TCI* message again. Each node repeats this process until all the nodes have determined their layers and target nodes. This process is presented as Algorithm 3.

We now explain why there are two *Layer determination* processes. If all the nodes were to change their layers at the same time, their estimates of their own lifetimes $\bar{L}$ would be less accurate, because the method of determining $\bar{L}_i$ assumes that neighbor nodes will not change their layers until $\bar{L}_i$ is calculated again. Therefore the *Asynchronous layer determination* process is used to determine whether a node should move up to the next layer. However, when a node n_i^j moves down a layer, it is not certain that the nodes which have n_i^j as their target node will be able to find a new route to a higher-layer node. This is why the process which determines whether a node moves up a layer is executed by asynchronously at the beginning of each layer determination period p_{layer_i}, whereas all the nodes run the process which determines whether they will move down at the same time.

Estimated Lifetime of a Battery-powered Node

Our scheme requires each node to calculate its expected lifetime, and then to determine its layer by comparing its own expectation with the local expected lifetime $\bar{L}_i$. The expected lifetime of a node can be obtained by applying Equation (4.1) to the amount of energy which that node consumed during the previous round. However, the *Synchronous layer determination* process, presented as Algorithm 2, requires a node to estimate its lifetime on the assumption that it is elevated to layer $j + 1$. We will explain this move complicated process in

Algorithm 3 Synchronous layer determination

```
Require: Node $n_i^j$ receives Global TCI message from $n_k^l$
Ensure: Determine whether node $n_i^j$ moves to layer $j-1$
Calculate $L_i$ using Equation (4.1)
if $n_i^j$ has already received a TCI from another node then
    if $j_{next} > j \land l > j$ then
        $j \leftarrow \mathrm{MIN}(j_{jext}, l)$
        $t_i \leftarrow n_k^l$
        Broadcast Global TCI message
    end if
else
    if $L_i < \bar{L}$ then
        $j_{next} \leftarrow j - 1$
    else
        $j_{next} \leftarrow j$
    end if
    $j \leftarrow \mathrm{MIN}(j_{next}, l)$
    $t_i \leftarrow n_k^l$
    Broadcast Global TCI message
end if
```

due course.

Estimation of energy consumption An estimate of the energy consumed by a node can be made as follows [101]:

$$E_{\text{consume}} = E_{\text{trans}} + E_{\text{receive}} + E_{\text{elec}}, \tag{4.2}$$

where E_{trans} and E_{receive} are respectively the amount of energy consumed by the node's radio transceiver during the transmission and receipt of data; and E_{elec} is the energy consumed by the node's electric circuits. The energy which a node n_i^j consumes during the setup period p_{setup}, starting at time t, is

$$E_{\text{round}_i^j}(t) = \int_t^{t+p_{\text{setup}}} E_{\text{consume}_i^j}(\tau)d\tau \tag{4.3}$$

$$= \int_t^{t+p_{\text{setup}}} (E_{\text{trans}_i^j}(\tau) + E_{\text{receive}_i^j}(\tau) + E_{\text{elec}_i}(\tau))d\tau, \tag{4.4}$$

where $E_{\text{trans}_i^j}$ is the energy required by node n_i^j during period p_{setup}, which can be determined follows:

$$\int_t^{t+p_{\text{setup}}} E_{\text{trans}_i^j}(\tau)d\tau = s\beta d^\alpha \tag{4.5}$$

$$= \left\lfloor \frac{p_{\text{setup}}}{p_{\text{gather}}} \right\rfloor s_{\text{trans}_i^j} \beta d_i^\alpha. \tag{4.6}$$

In this expression s is the amount of data to be transmitted; $s_{\text{trans}_i^j}$ is the amount of data which n_i^j transmits during gathering period p_{gather}; α is the path loss $(2 \geq \alpha \geq 5)$; β is the energy used by the power amplifier in transmitting 1 bit over a distance of 1 meter; d is the distance between the nodes [m]. By substituting Equation (4.6) into Equation (4.4), $E_{\text{round}_i^j}(t)$ can be expressed as follows:

$$E_{\text{round}_i^j}(t) = \left\lfloor \frac{p_s}{p_g} \right\rfloor s_{\text{trans}_i^j} \beta d_i^\alpha + \int_t^{t+p_{\text{setup}}} (E_{\text{receive}_i^j}(\tau) + E_{\text{elec}_i}(\tau))d\tau. \tag{4.7}$$

Let us assume that the data from lower-layer nodes is aggregated into a single packet, which is the sent immediately. The amount of data to be transmitted $s_{\text{trans}_i^j}$, which varies from layer to layer, can be expressed as follows:

$$s_{\text{trans}_i^j} = s_{\text{head}} + s_{\text{sensor}} + s_{\text{lower}_i^j} + s_{\text{relay}_i^j}, \tag{4.8}$$

where s_{head} is the amount of data in a packet header; s_{sensor} is the amount of data gathered by the sensor in node n_i^j; ${s_{\text{lower}}}_i^j$ is the amount of data received by node n_i^j from nodes on lower layers, excluding packet headers; and ${s_{\text{relay}}}_i^j$ is the amount of data received from nodes on the same layer.

Estimating the amount of data that a node must transmit if it moves to a higher layer The amount of data which node n_i^j will have to transmit if it is elevated to layer $j+1$ can be determined by considering three cases, shown in Figure 4.5. In describing these cases, we will write the set of neighbor nodes of n_i^j, N_i: and the target node of n_k as $\forall n_k^l \in N_i$, as t_k.

Case 1: target node $t_k = n_m^j$.

The highest layer of any of the neighbor nodes n_k^l is j, because a node sends data to its highest-layer neighbors. However, if n_i^j is elevated to become n_i^{j+1}, it will be the highest-layer node among neighbor of n_k^l. Therefore, n_i^{j+1} becomes the new target node t_k, and the $s_{\text{data}k}$ bits which node n_k transfered during the previous round must be added to ${s_{\text{lower}}}_i^{j+1}$.

Case 2: $t_k = n_m^{j+1}$ and the hop-count of n_m^{j+1}, $h_m > h_i$.

The target node t_k will become n_i^{j+1}. However, if $l = j+1$, then $s_{\text{data}k} + s_{\text{head}}$ should be added to ${s_{\text{relay}}}_i^{j+1}$, because n_k^l will use n_i^{j+1} as a relay node; but if $l < j+1$, then $s_{\text{data}k}$ should be added to ${s_{\text{lower}}}_i^{j+1}$, because n_i^{j+1} will be used as an aggregation node.

Case 3: Otherwise.

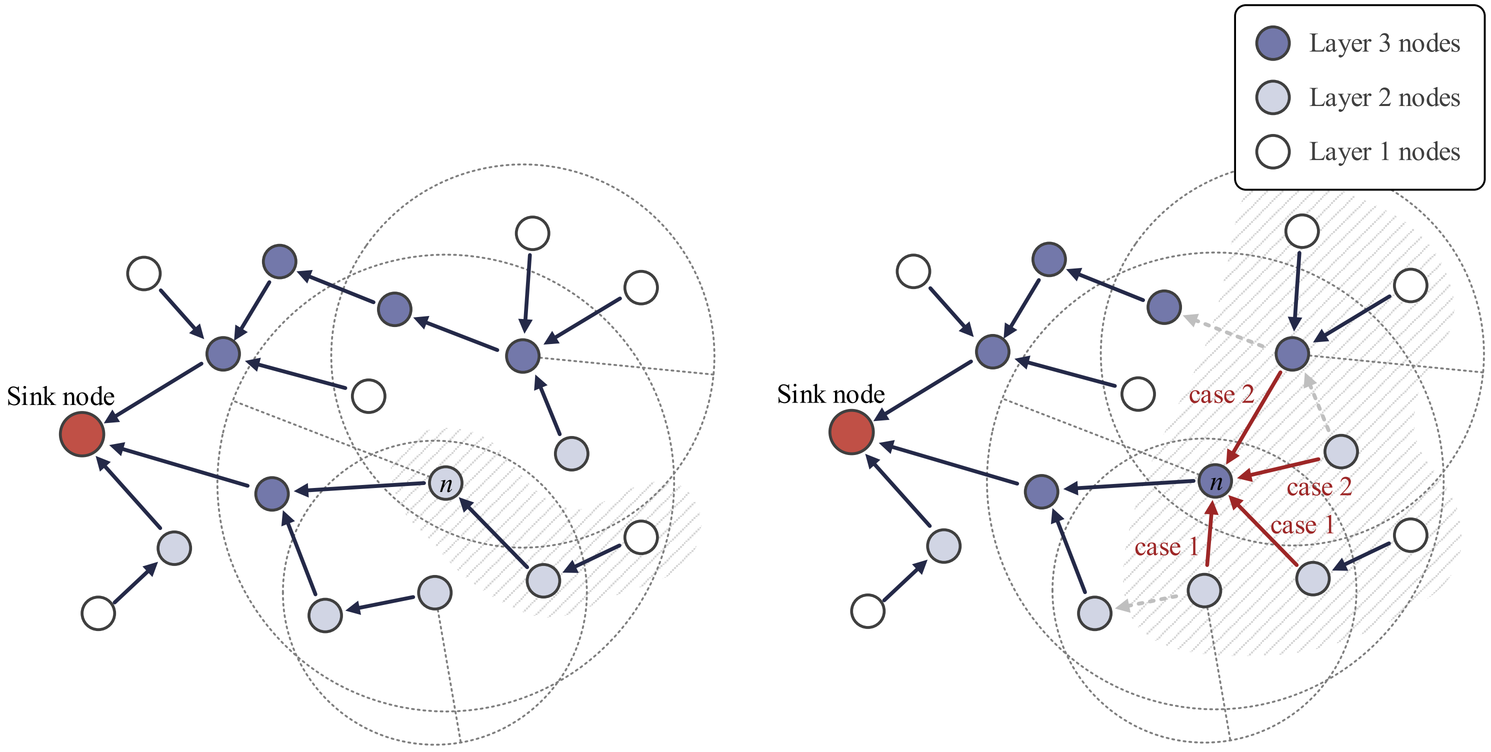
Other nodes already have better target nodes than n_i^{j+1}, and so they do not send it any data.

All the information which a node requires in order to estimate the amount of data that it will have to relay can be obtained by exchanging *Local TCI* messages with its neighbors.

The estimated lifetime L_i^{j+1}, which is the number of rounds that node n_i^{j+1} can be expected to survive, can now be calculated from $E_{\text{round}}{}_i^{j+1}(t)$, as follows:

$$L_i^{j+1} = \left\lfloor \frac{E_{\text{remain}\,i}}{E_{\text{round}}{}_i^{j+1}(t)} \right\rfloor . \tag{4.9}$$

L_i^{j+1} is used in Algorithm 2 to determine whether n_i^j moves up a layer.



(a) Network topology before *Asynchronous layer determination*

(b) Network topology after *Asynchronous layer determination*

Figure 4.5 Estimating the amount of data to be transmitted, as part of *Asynchronous layer determination*.

4.1.3 Introducing Energy-Harvesting Nodes to a Layered Topology

In a WSN that has both energy-harvesting and battery-powered nodes, the network lifetime can be extended by preferential deployment of energy-harvesting nodes on upper layers as aggregation nodes. However, a node will eventually die if it uses more energy than it harvests and this is a greater threat to upper-layer nodes, because they have to transmit a lot of data. It is better to deploy energy-harvesting nodes on layers with workloads that allow them to them survive indefinitely.

In the previous section, we showed how to determine the layer of a battery-powered node on the basis of a comparison between its estimated lifetime and the average expected lifetime of neighboring nodes. However, energy-harvesting nodes do not have well-defined lifetimes, and so they must be allocated to layers in a different way.

Energy Model for Energy-Harvesting Nodes

Although its long-term survival cannot be predicted, an energy-harvesting node n_i^j, so that it can determines its layer, needs to know whether its lifetime extends beyond the current round. Kansal et al. [91] introduced a model of energy-harvesting and consumption, and proposed a bound on the residual energy which should allow a node to survive indefinitely. Based on this model, the residual energy available to node n_i^j, after a setup period p_{setup} beginning at time t, can be expressed as follows:

$$\begin{aligned} E_{\text{remain}_i^j}(t + p_{\text{setup}}) &= E_{\text{remain}i}(t) - E_{\text{round}_i^j}(t) + E_{\text{charge}_i}(t, p_{\text{setup}}), \\ &0 < E_{\text{remain}i}(x) < E_{\text{max}i}, \end{aligned} \tag{4.10}$$

where $E_{\text{remain}_i^j}(t)$ is the residual energy available to node n_i^j at time t; $E_{\text{round}_i^j}(t)$ is the energy consumed during p_{setup}; $E_{\text{charge}_i}(t, p_{\text{setup}})$ is the energy harvested during p_{setup}; and E_{max} is the capacity of the node's battery.

The expected energy harvest, $E_{\text{charge}_i}(t, p_{\text{setup}})$ can be determined as follows:

$$E_{\text{charge}_i}(t, p_{\text{setup}}) = \int_t^{t+p_{\text{setup}}} \lambda_i(\tau) d\tau, \tag{4.11}$$

where $\lambda_i(\tau)$ is the charging rate of n_i at time τ. Some researchers have suggested ways of estimating λ. For instance, Kansan et al. [83] use an exponentially weighted moving-average (EWMA) algorithm to estimate the charging rate of a solar cell.

The expected energy consumption $E_{\text{round}_i^j}(t)$ can be calculated using Equation (4.1) or (4.7).

Determining the Layer of an Energy-Harvesting Node

Because of the impracticality of estimating the lifetime of an energy-harvesting node, we settle for estimating whether an energy-harvesting node n_i^j is likely to be alive after p_{setup} if it is allocated to a certain layer. If it is expected to survive on a higher layer, then it moves up to that layer; conversely, if it is not expected to survive on its current layer, it moves to a lower layer.

Determining whether a node should move to a higher layer

If the estimated energy conserved stored in the battery of an energy-harvesting node n_i^j, which is ${E_{\text{remain}}}_i^{j+1}(t+p_{\text{setup}})$ and can be found using Equation (4.10), satisfies the following condition, then the node can move to the next higher layer $j+1$:

$$ {E_{\text{remain}}}_i^{j+1}(t+p_{\text{setup}}) > E_{\text{min}i} + \epsilon_i, \tag{4.12} $$

where $E_{\text{min}i}$ is the minimum energy required to operate node n_i properly, which depends on the specification of each sensor node; ϵ_i is the estimation error of ${E_{\text{charge}}}_i(t, p_{\text{setup}})$, which can be estimated using a charging energy estimation algorithm [83]. If Equation 4.12 is satisfied, node n_i will not become defunct until next round at the earliest. We can now modify the *Asynchronous layer determination* process to apply to energy-harvesting nodes by modifying line 3 of Algorithm 2 to reflect Equation (4.12)

Determining whether a node should move to a lower layer

If the expected energy remaining in energy-harvesting node n_i^j, which is ${E_{\text{remain}}}_i^j(t+p_{\text{setup}})$, and can be found using Equation (4.10), satisfies the following condition, then that node should move to a lower layer, because it will die in its current layer:

4.2 Experimental Results

4.2.1 Simulation Environment

We wrote a simulation in C++ to evaluate the performance of the proposed scheme. In this simulation, we measured the average amount of dead nodes, and the average number of data arriving at the sink node, with a network topology consisting of up to four layers respectively: a single layer is a flat topology. We deployed 500 nodes and each test set was run 20 times for 5000 rounds to obtain the average values. For topologies with more than one layer, all nodes were assigned to layers at the beginning of each round, using the proposed scheme. Each node transfers the data that it obtains from its sensor to one of the nodes in the layer above by running the minimum-depth tree algorithm at intervals of p_{gather}. The nodes in the upper layers aggregate their own data with that received from the nodes below, and forward it all to their target node. Table 4.4 summarizes the important parameters used in this simulation.

4.2.2 Simulation Results

Figures 4.6 and 4.7 show the cumulative number of defunct nodes and the amount of data which the sink node received successfully. In this experiment, all nodes are battery-powered, and dead nodes is not replaced. We did not use any data aggregation scheme, to exclude its influence on the results. Figure 4.6 shows that the layered topology reduces the number of defunct nodes, especially at the beginning of the simulation, compared to the flat topology. This demonstrates how the nodes in a layered topology prolong their lifetimes by changing layer to manage their energy usage. By round 800, the number of dead nodes in a 4-layer network is respectively reduced by 15%, compared to a flat network. Figure 4.7 shows that this reduction in the number of dead nodes increases the amount of data reaching the sink.

Figure 4.8 traces the number of defunct nodes over time in the network, in a scenario in which dead nodes are replaced with new nodes at every 200

Table 4.4 Important parameters used in our simulation

Parameter	Value
Number of nodes	500
Node density	$0.03 \sim 0.07$ nodes/m^2
Node placement	Random
Amount of data in a packet	60 bytes
Amount of a packet header	40 bytes
Transmission range	$10 \sim 20$ m
Maximum number of layers	$1 \sim 4$
α and β in Equation (4.6)	4 and 100 pJ/bit/m^2
E_{elec} and E_{receive} in Equation (4.4)	0.000048 J and 0.048 J
p_{layer_i} and p_{setup}	6 hours
p_{gather}	60 s

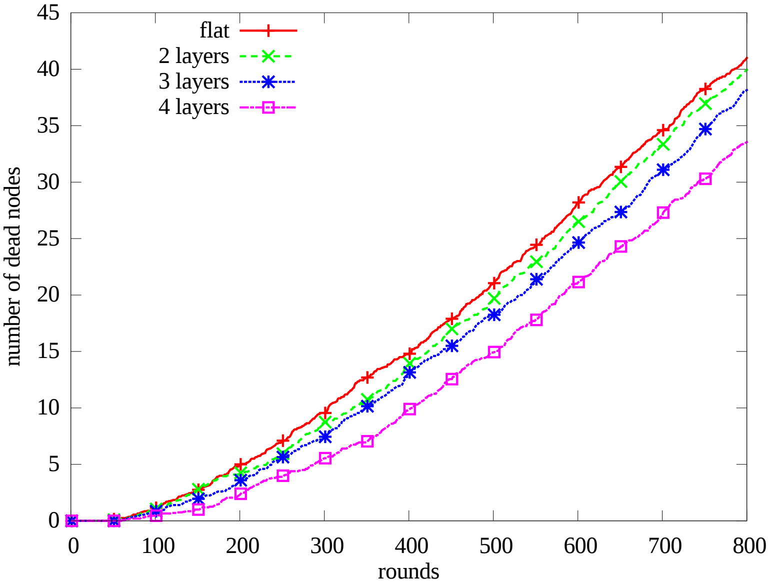


Figure 4.6 Change in the number of dead nodes without replacement.

rounds. This resulting sawtooth curve results the pattern of Figure 4.6 as the network is refreshed by new nodes with fully changed batteries. In Figures 4.9 and 4.10 respectively show the number of dead nodes and the amount of data that arrived at the sink node, as the replacement period was varied from 100 to 300. The flat topology causes more nodes to die and data transmission is less effective that it is with the layered topology. The effectiveness of transmission declines as the redeployment period increases, as we would expect, because dead nodes are waiting longer for replacement.

Figures 4.11 and 4.12 show the cumulative number of defunct nodes and the amount of data which the sink node received successfully. Figures 4.13 and 4.14 respectively show the number of dead nodes and the amount of data delivered to the sink, as the data size was varied from 75 bytes to 15 bytes. In this simulation, we assume that the network consists of only battery-powered nodes, and each node was a naive data aggregation scheme. Nodes aggregate the data received from the nodes in lower layers with their own data before sending it to their target node. The size of the resulting packet can be reduced omitting the headers of the packets received from lower-layer nodes; and the effectiveness of data aggregation is proportional to the size of a packet header.

Figure 4.11 shows that the layered topology reduces the number of defunct nodes, especially at the beginning of the simulation, compared to the flat topology and our previous hierarchical topology control scheme. This demonstrates how the nodes in a layered topology prolong their lifetimes by changing layer to manage their energy usage. By round 1000, the number of dead nodes in a 4-layer network is respectively reduced by 40% and 35%, compared to a flat network and a 4-layer network organized by the previous scheme. Figure 4.12

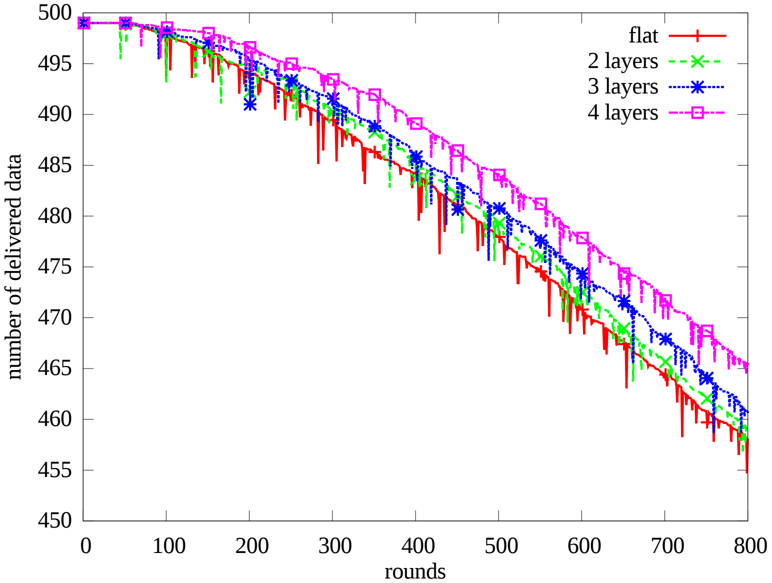


Figure 4.7 Change in the amount of data gathered at the sink node without replacement.

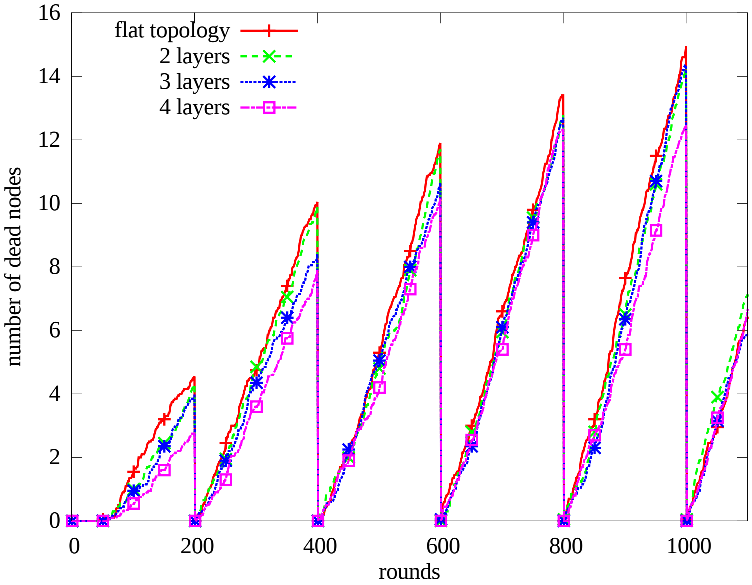


Figure 4.8 Variation in the number of dead nodes with periodic replacement.

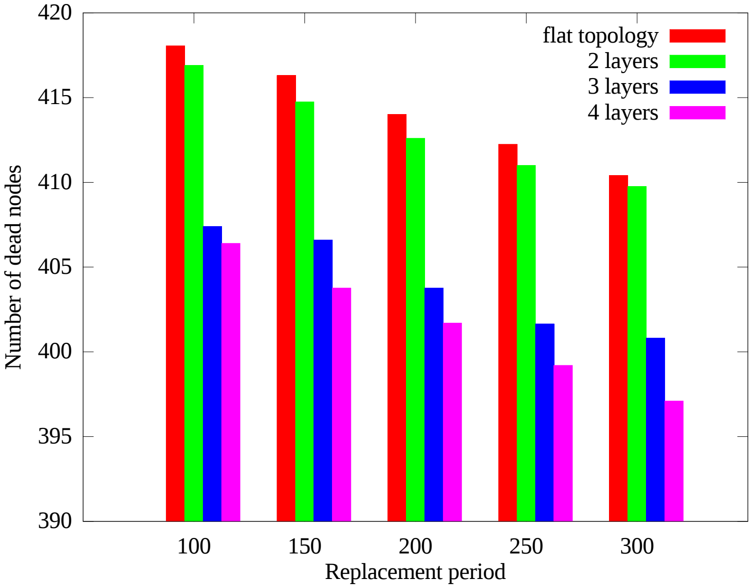


Figure 4.9 Variation in the number of dead nodes with the replacement period.

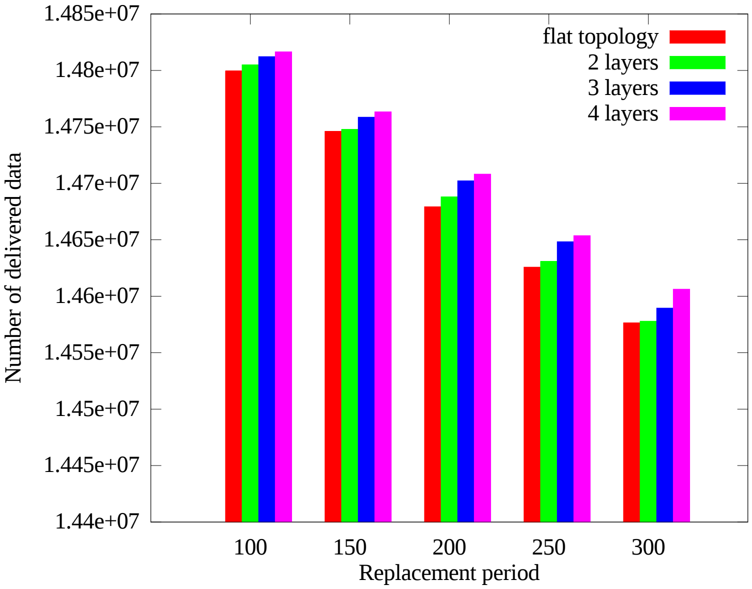


Figure 4.10 Variation in the amount of data arriving at the sink node with the replacement period.

shows that this reduction in the number of dead nodes increases the amount of data reaching the sink. As shown in Figures 4.13 and 4.14, our scheme shows reduces the number of dead nodes by about 10%, and delivers slightly more data than the flat topology.

Figure 4.15 shows the round when the number of defunct nodes becomes 1 to 20% of the total number of nodes. As shown in Figure 4.15, our scheme shows it prolongs the lifetime of network, compared to the flat topology and our previous scheme because it can balance the remaining energy of each node.

Figure 4.15 represents the loss rate of gathered data, as the duty cycle was varied from 0.0002 to 0.002. When the duty cycle was relatively large the sink node was able to receive almost every gathered data of all schemes. When the duty cycle was small, however, the sink node could barely receive some data because the packet sent from distant nodes was delayed by the intermediate nodes, as a result, nodes could not receive it within their wakeup time.

Figure 4.17 shows the overhead of each topology control scheme, as the density of nodes was varied from 0.02 to 0.1. The overhead indicates the average consumed energy during a round for *TCI* messages of a node. In this scheme, a node which has the more neighbor nodes occurs the larger control overhead because *Local TCI* message contains the information of 2-hop neighbor nodes. In Figure 4.17, therefore, the consumed energy for *TCI* message of our scheme increased as the density of nodes increased. However, the overhead of the other schemes decreased as the density increased because the shorter hop made reduce the number of *TCI* messages. Moreover 4-layer network consumed more energy than 2-layer network because the nodes in 4-layer network had more chance to move to other layers and retransmit *TCI* messages than

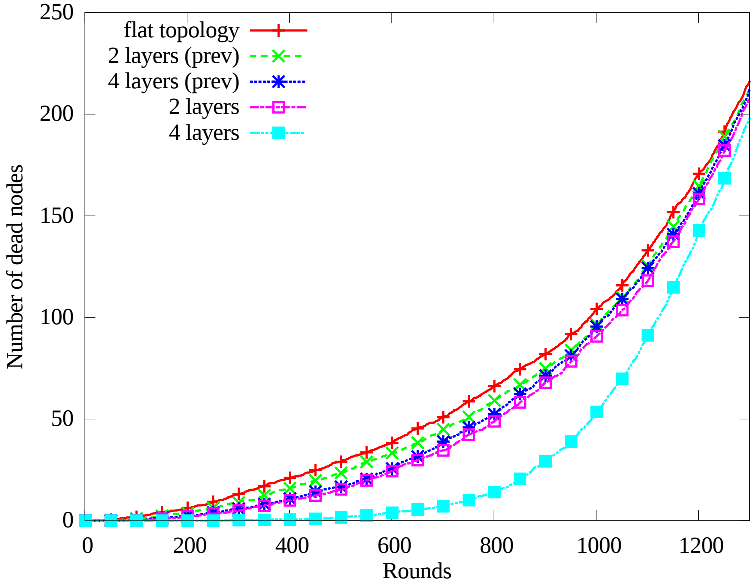


Figure 4.11 Change in the number of dead nodes with data aggregation.

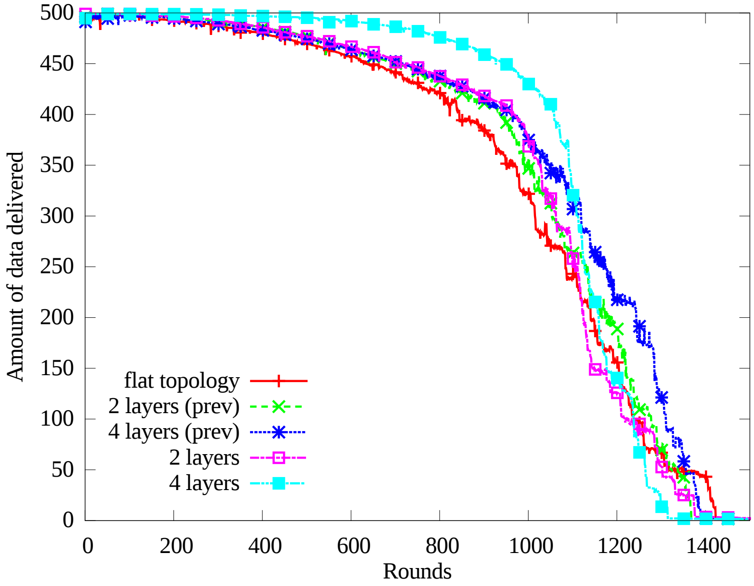


Figure 4.12 Change in the amount of data gathered at the sink node with the data aggregation.

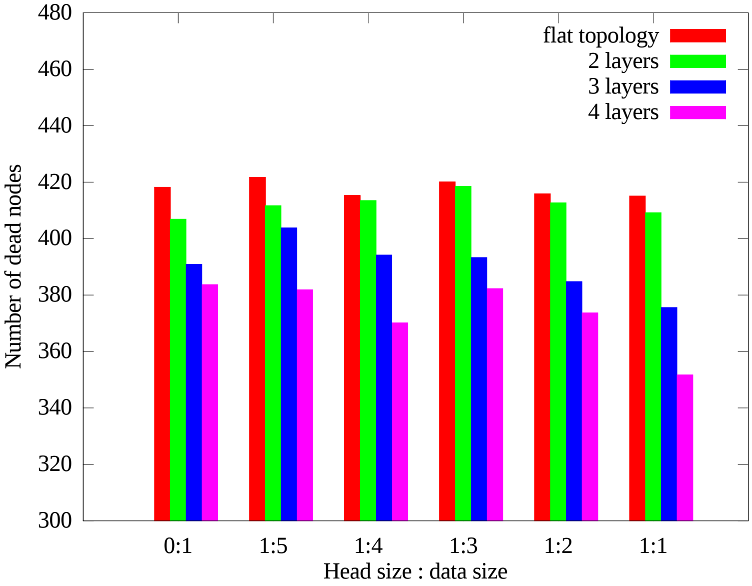


Figure 4.13 Number of dead nodes with data aggregation.

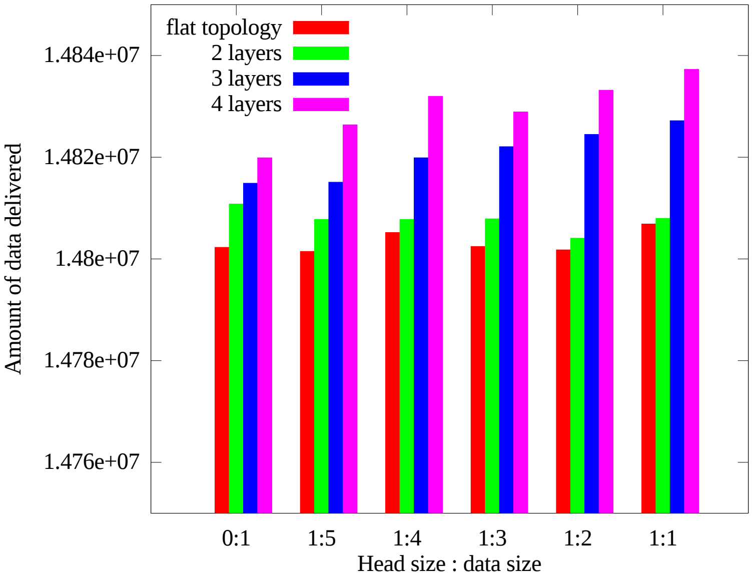


Figure 4.14 Amount of data arriving at the sink node with the data aggregation.

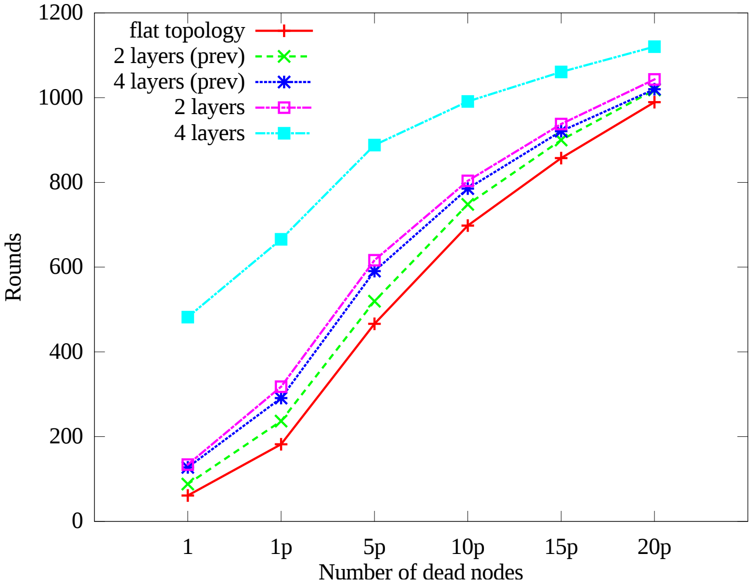


Figure 4.15 Lifetime of each network.

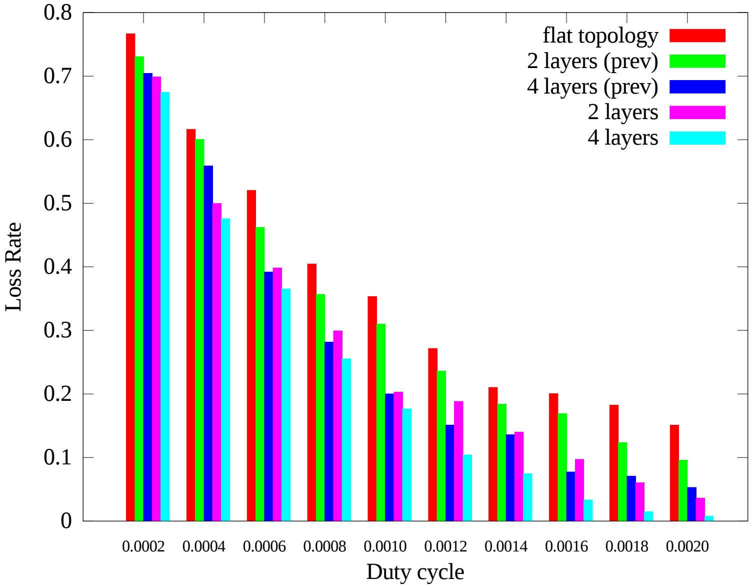


Figure 4.16 Loss rate of gathered data.

nodes in 2-layer network.

In order to analyze the effectiveness of our scheme in networks with energy-harvesting nodes, we measured the number of dead nodes and the amount of transmitted data in a network consisting of 80% battery-powered nodes and 20% of energy-harvesting nodes.

Figures 4.18 and 4.19 show the cumulative number of defunct nodes and the amount of data which the sink node received successfully. Figure 4.18 shows that our scheme reduces the number of defunct nodes, especially at the beginning of the simulation, compared to the flat topology and the scheme that uses the energy-harvesting nodes as cluster heads. This demonstrates how the nodes in a hierarchical topology prolong their lifetimes by changing layer to manage their energy usage. At 1000 round, nodes started being defunct in a 4-layer network. Meanwhile, the first node in a flat network and a clustered network respectively died earlier than 200 round and 400 round. That can be also shown in Figure 4.20. Figure 4.19 shows that this reduction in the number of dead nodes increases the amount of data reaching the sink.

As shown in Figure 4.21, the number of dead nodes is reduced by about 10% by our scheme, which is very similar to the result for the network with only battery-powered nodes. Figure 4.22 shows that our scheme also delivers more data. However, it also shows that the effectiveness of transmission is sometimes reduced by the presence of energy-harvesting nodes. We suggest that this is because the energy-harvesting nodes are not replaced, even when they have little energy; instead, they wait for energy to arrive and charge their batteries. Meanwhile transmissions are failing. We also found in Figure 4.23, which using energy-harvesting nodes as cluster nodes produces the least

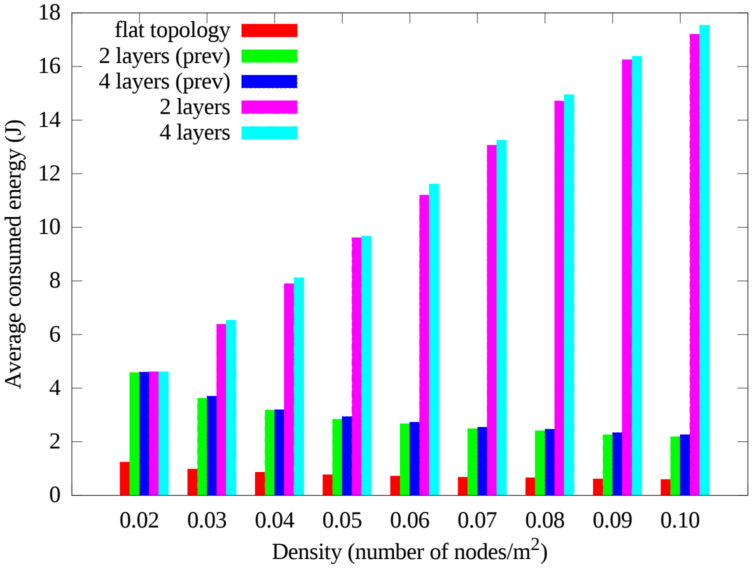


Figure 4.17 Overhead of each topology control scheme.

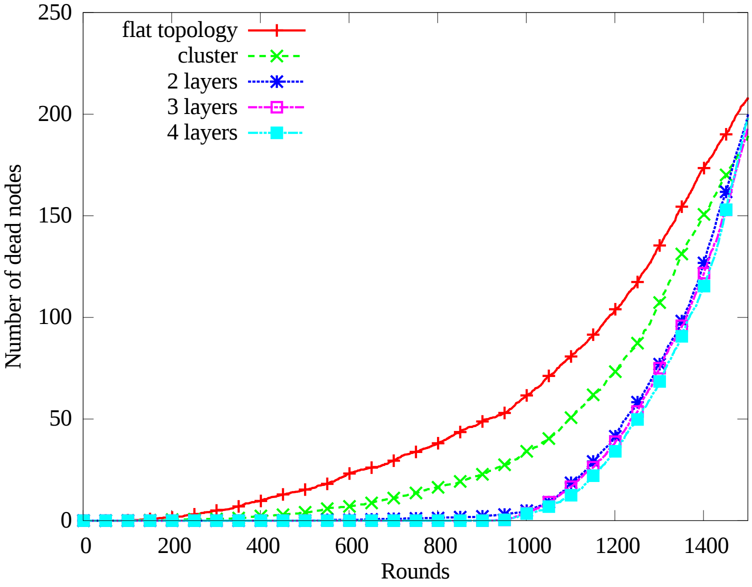


Figure 4.18 Change in the number of dead nodes.

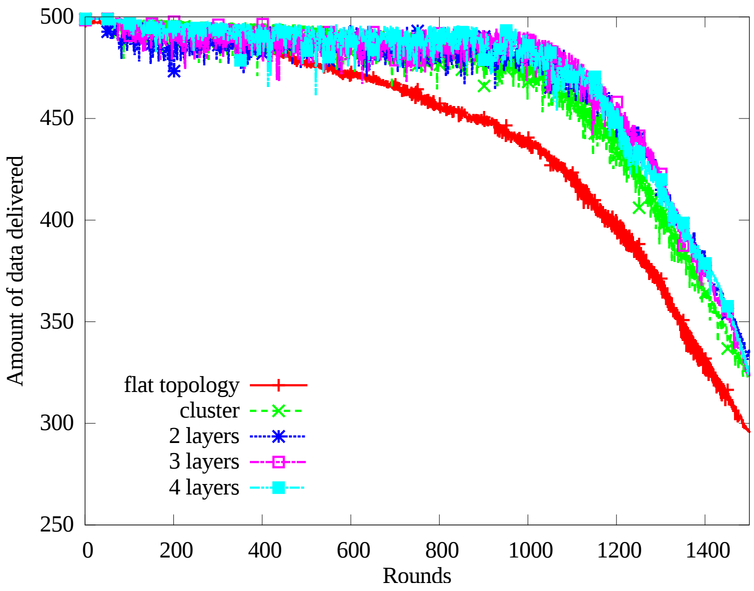


Figure 4.19 Change in the amount of data gathered at the sink node.

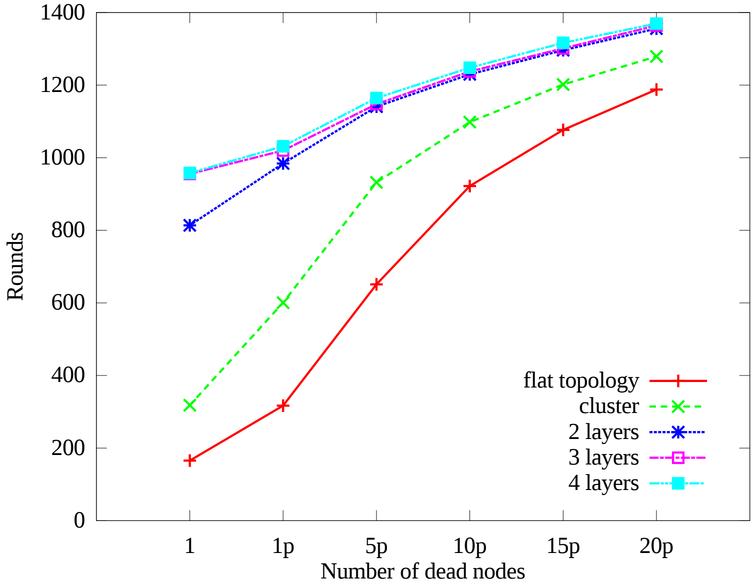


Figure 4.20 Lifetime of each network.

effective transmission of data. Again, we attribute this to the amount of energy acquired by these nodes, which is inadequate to transmit lots of data. This problem may be addressed by increasing the size of solar panels fitted to energy-harvesting nodes, or reducing the proportion of these nodes that are deployed.

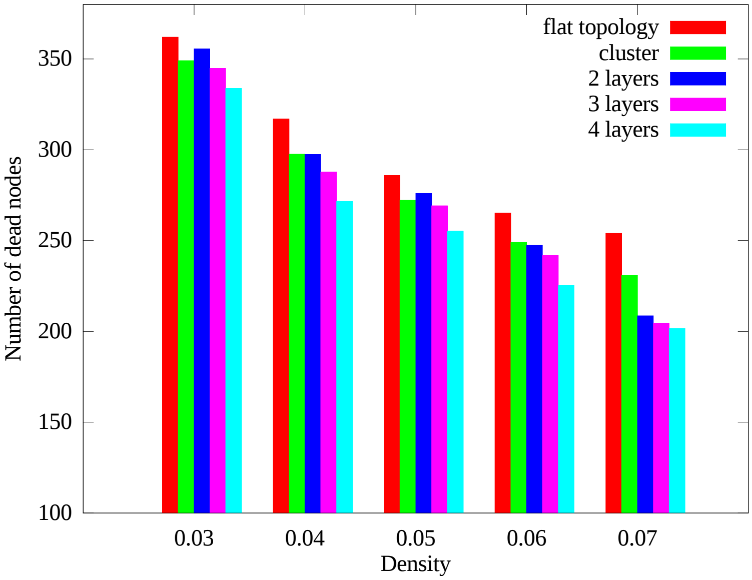


Figure 4.21 Variation in the number of dead nodes as the proportion of energy-harvesting nodes is increased.

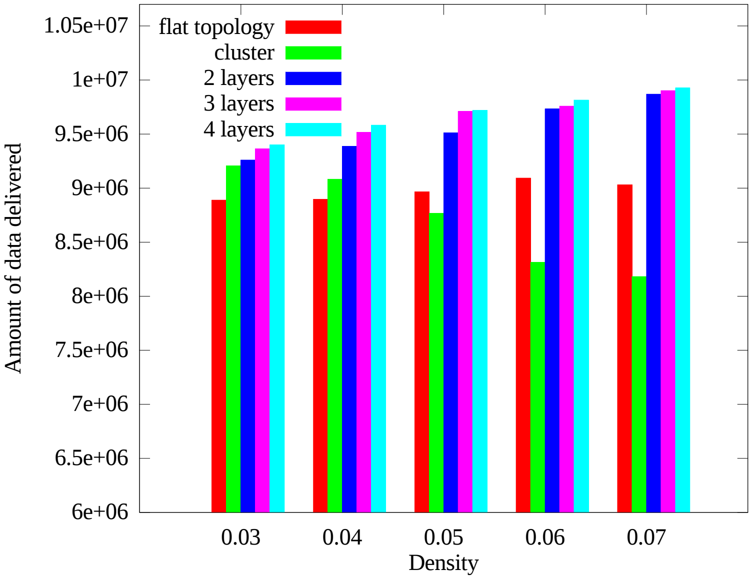


Figure 4.22 Variation in the amount of data arriving at the sink node as the proportion of energy-harvesting nodes is increased.

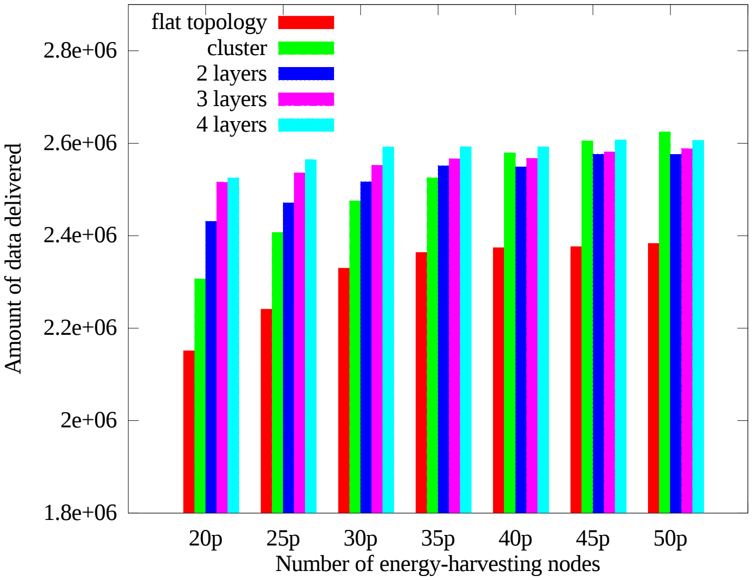


Figure 4.23 Lifetime of each network.

Chapter 5

Conclusion

5.1 Summary

Since nodes used in general WSNs are powered by batteries, they have limited lifetime. If intermediate nodes are defunct or links between nodes are not reliable, the success probability of gathering sensory data may be dropped drastically, because they employ multi-hop communication protocol to deliver data to a sink node. Some nodes that have no sensor modules may be deployed to relay packets between nodes.

We have described the implementation of a WSN designed for wildfire monitoring, focusing on the reliability issue. In our experiment, 20 sensor nodes were deployed sparsely to collect temperature and humidity data in order to detect wildfire. The nodes are designed to turn off their transceivers to save energy, and to synchronize their sleep and wake-up cycles to simplify communications. The end-to-end delay in data transmission to the sink node is reduced by using the modified PISA-I routing algorithm. By addressing some implementation issues such as irregular transmission ranges, Fresnel zone and unreliable links, we have made our system more stable and reliable.

Hierarchical topology such as clustering has been widely used in wireless sensor networks (WSNs) to address the limitations of a flat topology. In the cluster topology, however, a serious imbalance of residual energy between nodes can happen inevitably, which leads to a sharp drop of network connectivity. A lot of research has been put into topology control techniques to address this problem. We have proposed a multi-layer topology control scheme for long-term WSNs to improve network lifetime and balance residual energy. The WSN assumed in this scheme consists of battery-powered nodes, and if some nodes get defunct, new nodes replace the dead nodes periodically. In this scheme, each node periodically determines its own layer, which depends on its residual energy and the current network status when the sink node broadcasts a topology control information message to the whole network. This allows the WSN to balance itself by allocating energy-intensive roles to energy-rich nodes. As a result, more nodes can survive longer than the nodes in flat topology. Moreover, using more layers can balance the residual energy.

We also have proposed the topology control scheme for WSNs consisting of many nodes, some of which are battery-powered while others are energy-harvesting nodes. The battery-powered nodes decide their layers considering their residual energy and the average network lifetime. The energy-harvesting nodes also decide their layers, but they consider their charging rates in addition to the residual energy. The nodes in the higher layers aggregate data received from the lower layer nodes, and send it to the sink node with their own data. This preserves the connectivity of the WSN because the expected lifetime of nodes are balanced by assigning them into multiple layers, and the aggregation nodes live longer than nodes in a flat topology. Particularly, energy-harvesting nodes help reduce the energy consumption of battery-powered nodes due to their harvested energy. Therefore, the harvested energy can be used to extend the network lifetime and to preserve the connectivity of the network.

5.2 Future Research Directions

The routing algorithm described in this thesis causes a node to choose the route that reaches a higher-layer node with the minimum hop-count. This algorithm considers not residual energy in the nodes on those routes or their contribution to overall network connectivity but the length of routes. Therefore it does not explicitly contribute to the lifetime or connectivity of the network. We believe that it should be possible to design a routing scheme for the multilayer topology control which will enhance overall network efficiency.

The aggregation scheme proposed in this paper is a simple in that only data delivered from lower layers is aggregated. More efficient data aggregation or fusion schemes have been proposed by other researchers. We will incorporate such ideas in order to reduce the volume of data to be transmitted, and hence to increase the lifetime of the WSN.

Bibliography

[1] G. Werner-Allen, J. Johnson, M. Ruiz, J. Lees, and M. Welsh, "Monitoring volcanic eruptions with a wireless sensor network," *Proceedings of the Second European Workshop on Wireless Sensor Networks.* IEEE, 2005, pp. 108–120.

[2] T. He, S. Krishnamurthy, L. Luo, T. Yan, L. Gu, R. Stoleru, G. Zhou, Q. Cao, P. Vicaire, J. A. Stankovic *et al.*, "Vigilnet: An integrated sensor network system for energy-efficient surveillance," *ACM Transactions on Sensor Networks*, vol. 2, no. 1, pp. 1–38, 2006.

[3] A. Baggio, "Wireless sensor networks in precision agriculture," *ACM Workshop on Real-World Wireless Sensor Networks*, 2005.

[4] H. Alemdar and C. Ersoy, "Wireless sensor networks for healthcare: A survey," *Computer Networks*, vol. 54, no. 15, pp. 2688–2710, 2010.

[5] M. Li and Y. Liu, "Underground structure monitoring with wireless sensor networks," *Proceedings of the 6th International Conference on Information Processing in Sensor Networks.* ACM, 2007, pp. 69–78.

[6] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, “Energy-efficient communication protocol for wireless microsensor networks,” *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences.* IEEE, 2000, pp. 10–pp.

[7] J. Yick, B. Mukherjee, and D. Ghosal, “Wireless sensor network survey,” *Computer Networks*, vol. 52, no. 12, pp. 2292–2330, 2008.

[8] D. M. Doolin and N. Sitar, “Wireless sensors for wildfire monitoring,” *Proceedings of Smart Structures and Materials.* International Society for Optics and Photonics, 2005, pp. 477–484.

[9] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Anderson, “Wireless sensor networks for habitat monitoring,” *Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications.* ACM, 2002, pp. 88–97.

[10] P. Juang, H. Oki, Y. Wang, M. Martonosi, L. S. Peh, and D. Rubenstein, “Energy-efficient computing for wildlife tracking: Design tradeoffs and early experiences with zebranet,” *ACM Sigplan Notices*, vol. 37, no. 10. ACM, 2002, pp. 96–107.

[11] D. H. Roemmich, R. E. Davis, S. C. Riser, W. B. Owens, R. L. Molinari, S. L. Garzoli, and G. C. Johnson, “The argo project. global ocean observations for understanding and prediction of climate variability,” DTIC Document, Tech. Rep., 2003.

[12] Y. Tseng, S. Y. Ni, Y. S. Chen, and J. P. Sheu, “The broadcast storm problem in a mobile ad hoc network,” *Wireless Networks*, vol. 8, no. 2, pp. 153–167, 2002.

[13] M. Perillo, Z. Cheng, and W. Heinzelman, "On the problem of unbalanced load distribution in wireless sensor networks," *Proceedings of Global Telecommunications Conference Workshops.* IEEE, 2004, pp. 74–79.

[14] Y. Xu, J. Heidemann, and D. Estrin, "Geography-informed energy conservation for ad hoc routing," *Proceedings of the 7th Annual International Conference on Mobile Computing and Networking.* ACM, 2001, pp. 70–84.

[15] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Transactions on Wireless Communications*, vol. 1, no. 4, pp. 660–670, 2002.

[16] B. Chen, K. Jamieson, H. Balakrishnan, and R. Morris, "Span: an energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks," *Wireless Networks*, vol. 8, no. 5, pp. 481–494, 2002.

[17] O. Younis and S. Fahmy, "Heed: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks," *IEEE Transactions on Mobile Computing*, vol. 3, no. 4, pp. 366–379, 2004.

[18] S. Lindsey and C. Raghavendra, "Pegasis: Power-efficient gathering in sensor information systems," *Proceedings of Aerospace Conference Proceedings*, vol. 3. IEEE, 2002, pp. 1125–1130.

[19] Y. Fan, Q. Chen, and J. Yu, "Topology control algorithm based on bottleneck node for large-scale wsns," *Proceedings of International Con-*

ference on Computational Intelligence and Security, vol. 1. IEEE, 2009, pp. 592–597.

[20] C. Intanagonwiwat, D. Estrin, R. Govindan, and J. Heidemann, “Impact of network density on data aggregation in wireless sensor networks,” *Proceedings of the 22nd International Conference on Distributed Computing Systems.* IEEE, 2002, pp. 457–458.

[21] S. Sharma and M. Rani, “Three-layer architecture model (tlam) for energy conservation in wireless sensor networks,” *Proceedings of International Conference on Ultra Modern Telecommunications & Workshops.* IEEE, 2009, pp. 1–5.

[22] Z. Duan, F. Guo, M. Deng, and M. Yu, “Shortest path routing protocol for multi-layer mobile wireless sensor networks,” *Proceedings of International Conference on Networks Security, Wireless Communications and Trusted Computing*, vol. 2. IEEE, 2009, pp. 106–110.

[23] Z. Ming and X. Bugong, “Layer-based self-organizing topology control for sensor networks,” *Proceedings of 27th Chinese Control Conference.* IEEE, 2008, pp. 516–520.

[24] D. K. Noh and J. Hur, “Using a dynamic backbone for efficient data delivery in solar-powered wsns,” *Journal of Network and Computer Applications*, vol. 35, no. 4, pp. 1277–1284, 2012.

[25] J. Li and P. Mohapatra, “An analytical model for the energy hole problem in many-to-one sensor networks,” *Proceedings of IEEE Vehicular Technology Conference*, vol. 62, no. 4. IEEE, 2005, pp. 2721–2725.

[26] E. Oyman and C. Ersoy, “Multiple sink network design problem in large scale wireless sensor networks,” *Proceedings of IEEE International Conference on Communications*, vol. 6. IEEE, 2004, pp. 3663–3667.

[27] S. Sudevalayam and P. Kulkarni, “Energy harvesting sensor nodes: survey and implications,” *IEEE Communications Surveys & Tutorials*, vol. 13, no. 3, pp. 443–461, 2011.

[28] H. Karl and A. Willig, *Protocols and architectures for wireless sensor networks.* John Wiley & Sons, 2007.

[29] I. Yoon, D. K. Noh, D. Lee, R. Teguh, T. Honma, and H. Shin, “Reliable wildfire monitoring with sparsely deployed wireless sensor networks,” *Proceedings of the 26th International Conference on Advanced Information Networking and Applications.* IEEE, 2012, pp. 460–466.

[30] Y. Li, Z. Wang, and Y. Song, “Wireless sensor network design for wildfire monitoring,” *Proceedings of the 6th World Congress on Intelligent Control and Automation*, 2006.

[31] M. Hefeeda and M. Bagheri, “Wireless sensor networks for early detection of forest fires,” *Proceedings of IEEE International Conference on Mobile Adhoc and Sensor Systems.* IEEE, 2007, pp. 1–6.

[32] B. Son, Y.-s. Her, and J.-G. Kim, “A design and implementation of forest-fires surveillance system based on wireless sensor networks for south korea mountains,” *International Journal of Computer Science and Network Security*, vol. 6, no. 9, pp. 124–130, 2006.

[33] G. Werner-Allen, K. Lorincz, M. Ruiz, O. Marcillo, J. Johnson, J. Lees, and M. Welsh, "Deploying a wireless sensor network on an active volcano," *Internet Computing*, vol. 10, no. 2, pp. 18–25, 2006.

[34] T. He, S. Krishnamurthy, J. A. Stankovic, T. Abdelzaher, L. Luo, R. Stoleru, T. Yan, L. Gu, J. Hui, and B. Krogh, "Energy-efficient surveillance system using wireless sensor networks," *Proceedings of the 2nd International Conference on Mobile Systems, Applications, and Services.* ACM, 2004, pp. 270–283.

[35] T. Bokareva, W. Hu, S. Kanhere, B. Ristic, N. Gordon, T. Bessell, M. Rutten, and S. Jha, "Wireless sensor networks for battlefield surveillance," *Proceedings of the Land Warfare Conference*, 2006, pp. 1–8.

[36] E. Onur, C. Ersoy, H. Deliç, and L. Akarun, "Surveillance wireless sensor networks: deployment quality analysis," *Network*, vol. 21, no. 6, pp. 48–53, 2007.

[37] W.-P. Chen, J. C. Hou, and L. Sha, "Dynamic clustering for acoustic target tracking in wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 3, no. 3, pp. 258–271, 2004.

[38] S. Pattem, S. Poduri, and B. Krishnamachari, "Energy-quality tradeoffs for target tracking in wireless sensor networks," *Information Processing in Sensor Networks.* Springer, 2003, pp. 32–46.

[39] C.-Y. Lin, W.-C. Peng, and Y.-C. Tseng, "Efficient in-network moving object tracking in wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 5, no. 8, pp. 1044–1056, 2006.

[40] L. Song and D. Hatzinakos, "A cross-layer architecture of wireless sensor networks for target tracking," *IEEE/ACM Transactions on Networking*, vol. 15, no. 1, pp. 145–158, 2007.

[41] H.-W. Tsai, C.-P. Chu, and T.-S. Chen, "Mobile object tracking in wireless sensor networks," *Computer Communications*, vol. 30, no. 8, pp. 1811–1825, 2007.

[42] C. Gui and P. Mohapatra, "Power conservation and quality of surveillance in target tracking sensor networks," *Proceedings of the 10th Annual International Conference on Mobile Computing and Networking.* ACM, 2004, pp. 129–143.

[43] G. Mao, *Localization Algorithms and Strategies for Wireless Sensor Networks: Monitoring and Surveillance Techniques for Target Tracking: Monitoring and Surveillance Techniques for Target Tracking.* IGI Global, 2009.

[44] R. Szewczyk, E. Osterweil, J. Polastre, M. Hamilton, A. Mainwaring, and D. Estrin, "Habitat monitoring with sensor networks," *Communications of the ACM*, vol. 47, no. 6, pp. 34–40, 2004.

[45] J. R. Polastre, "Design and implementation of wireless sensor networks for habitat monitoring," Ph.D. dissertation, Citeseer, 2003.

[46] J. Polastre, R. Szewczyk, A. Mainwaring, D. Culler, and J. Anderson, "Analysis of wireless sensor networks for habitat monitoring," in *Wireless Sensor Networks.* Springer, 2004, pp. 399–423.

[47] E. S. Biagioni and K. Bridges, "The application of remote sensor technology to assist the recovery of rare and endangered species," *International Journal of High Performance Computing Applications*, vol. 16, no. 3, pp. 315–324, 2002.

[48] P. Zhang, C. M. Sadler, S. A. Lyon, and M. Martonosi, "Hardware design experiences in zebranet," *Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems.* ACM, 2004, pp. 227–238.

[49] W. Hu, N. Bulusu, C. T. Chou, S. Jha, A. Taylor, and V. N. Tran, "Design and evaluation of a hybrid sensor network for cane toad monitoring," *ACM Transactions on Sensor Networks*, vol. 5, no. 1, p. 4, 2009.

[50] J. Ko, C. Lu, M. B. Srivastava, J. Stankovic, A. Terzis, M. Welsh *et al.*, "Wireless sensor networks for healthcare," *Proceedings of the IEEE*, vol. 98, no. 11, pp. 1947–1960, 2010.

[51] S. Kim, S. Pakzad, D. Culler, J. Demmel, G. Fenves, S. Glaser, and M. Turon, "Health monitoring of civil infrastructures using wireless sensor networks," *Proceedings of the 6th International Symposium on Information Processing in Sensor Networks.* IEEE, 2007, pp. 254–263.

[52] A. Milenković, C. Otto, and E. Jovanov, "Wireless sensor networks for personal health monitoring: Issues and an implementation," *Computer Communications*, vol. 29, no. 13, pp. 2521–2533, 2006.

[53] J. Stankovic, Q. Cao, T. Doan, L. Fang, Z. He, R. Kiran, S. Lin, S. Son, R. Stoleru, and A. Wood, "Wireless sensor networks for in-home health-

care: Potential and challenges," *High Confidence Medical Device Software and Systems Workshop*, 2005, pp. 2–3.

[54] C. R. Baker, K. Armijo, S. Belka, M. Benhabib, V. Bhargava, N. Burkhart, A. D. Minassians, G. Dervisoglu, L. Gutnik, M. B. Haick *et al.*, "Wireless sensor networks for home health care," *Proceedings of the 21st International Conference on Advanced Information Networking and Applications Workshops*, vol. 2. IEEE, 2007, pp. 832–837.

[55] J. M. Corchado, J. Bajo, D. Tapia, A. Abraham *et al.*, "Using heterogeneous wireless sensor networks in a telemonitoring system for healthcare," *IEEE Transactions on Information Technology in Biomedicine*, vol. 14, no. 2, pp. 234–240, 2010.

[56] R. Jafari, A. Encarnacao, A. Zahoory, F. Dabiri, H. Noshadi, and M. Sarrafzadeh, "Wireless sensor networks for health monitoring," *Proceedings of the 2nd Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services.* IEEE, 2005, pp. 479–481.

[57] S. Rost and H. Balakrishnan, "Memento: A health monitoring system for wireless sensor networks," *Proceedings of the 3rd Annual IEEE Communications Society on Sensor and Ad Hoc Communications and Networks*, vol. 2. IEEE, 2006, pp. 575–584.

[58] Y. Hao and R. Foster, "Wireless body sensor networks for health-monitoring applications," *Physiological Measurement*, vol. 29, no. 11, p. R27, 2008.

[59] H. Huo, Y. Xu, H. Yan, S. Mubeen, and H. Zhang, "An elderly health care system using wireless sensor networks at home," *Proceedings of the*

3rd International Conference on Sensor Technologies and Applications. IEEE, 2009, pp. 158–163.

[60] N. Xu, S. Rangwala, K. K. Chintalapudi, D. Ganesan, A. Broad, R. Govindan, and D. Estrin, "A wireless sensor network for structural monitoring," *Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems.* ACM, 2004, pp. 13–24.

[61] S. Kim, S. Pakzad, D. Culler, J. Demmel, G. Fenves, S. Glaser, and M. Turon, "Wireless sensor networks for structural health monitoring," *Proceedings of the 4th International Conference on Embedded Networked Sensor Systems.* ACM, 2006, pp. 427–428.

[62] M. Li and Y. Liu, "Underground coal mine monitoring with wireless sensor networks," *ACM Transactions on Sensor Networks*, vol. 5, no. 2, p. 10, 2009.

[63] K. Chintalapudi, T. Fu, J. Paek, N. Kothari, S. Rangwala, J. Caffrey, R. Govindan, E. Johnson, and S. Masri, "Monitoring civil structures with a wireless sensor network," *Internet Computing*, vol. 10, no. 2, pp. 26–34, 2006.

[64] N. Kurata, B. F. Spencer, and M. Ruiz-Sandoval, "Risk monitoring of buildings with wireless sensor networks," *Structural Control and Health Monitoring*, vol. 12, no. 3-4, pp. 315–327, 2005.

[65] J. Paek, K. Chintalapudi, J. Caffrey, R. Govindan, and S. Masri, "A wireless sensor network for structural health monitoring: Performance and experience," *Center for Embedded Network Sensing*, 2005.

[66] S. Jardosh and P. Ranjan, “A survey: Topology control for wireless sensor networks,” *Proceedings of International Conference on Signal Processing, Communications and Networking.* IEEE, 2008, pp. 422–427.

[67] B. N. Clark, C. J. Colbourn, and D. S. Johnson, “Unit disk graphs,” *Discrete Mathematics*, vol. 86, no. 1, pp. 165–177, 1990.

[68] A. C.-C. Yao, “On constructing minimum spanning trees in k-dimensional spaces and related problems,” *SIAM Journal on Computing*, vol. 11, no. 4, pp. 721–736, 1982.

[69] J. W. Jaromczyk and G. T. Toussaint, “Relative neighborhood graphs and their relatives,” *Proceedings of the IEEE*, vol. 80, no. 9, pp. 1502–1517, 1992.

[70] K. J. Supowit, “The relative neighborhood graph, with an application to minimum spanning trees,” *Journal of the ACM*, vol. 30, no. 3, pp. 428–448, 1983.

[71] D. W. Matula and R. R. Sokal, “Properties of gabriel graphs relevant to geographic variation research and the clustering of points in the plane,” *Geographical Analysis*, vol. 12, no. 3, pp. 205–222, 1980.

[72] X.-Y. Li and P.-J. Wan, “Constructing minimum energy mobile wireless networks,” *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 5, no. 4, pp. 55–67, 2001.

[73] V. Rodoplu and T. H. Meng, “Minimum energy mobile wireless networks,” *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 8, pp. 1333–1344, 1999.

[74] M. Viera, L. Viera, L. B. Ruiz, A. A. Loureiro, A. O. Fernandes, and J. M. S. Nogueira, "Scheduling nodes in wireless sensor networks: A voronoi approach," *Proceedings of the 28th Annual IEEE International Conference on Local Computer Networks.* IEEE, 2003, pp. 423–429.

[75] C. Zhang, Y. Zhang, and Y. Fang, "Detecting coverage boundary nodes in wireless sensor networks," *Proceedings of IEEE International Conference on Networking, Sensing and Control.* IEEE, 2006, pp. 868–873.

[76] C.-H. Wu, K.-C. Lee, and Y.-C. Chung, "A delaunay triangulation based method for wireless sensor network deployment," *Computer Communications*, vol. 30, no. 14, pp. 2744–2752, 2007.

[77] A. Das and D. Dutta, "Data acquisition in multiple-sink sensor networks," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 9, no. 3, pp. 82–85, 2005.

[78] C. Buratti, J. Orriss, and R. Verdone, "On the design of tree-based topologies for multi-sink wireless sensor networks," *Proceedings of the NEWCOM-ACORN Workshop*, 2006, p. 6.

[79] C. Buratti and R. Verdone, "Tree-based topology design for multi-sink wireless sensor networks," *Proceedings of IEEE International Symposium on Personal, Indoor and Mobile Radio Communications.* IEEE, 2007, pp. 1–5.

[80] C. Buratti, F. Cuomo, S. Luna, U. Monaco, J. Orriss, and R. Verdone, "Optimum tree-based topologies for multi-sink wireless sensor networks using ieee 802.15. 4," *Proceedings of IEEE 65th Vehicular Technology Conference.* IEEE, 2007, pp. 130–134.

[81] J. Kim and S. Lee, “Spanning tree based topology configuration for multiple-sink wireless sensor networks,” *Proceedings of the 1st International Conference on Ubiquitous and Future Networks.* IEEE, 2009, pp. 122–125.

[82] P. Ciciriello, L. Mottola, and G. Picco, “Efficient routing from multiple sources to multiple sinks in wireless sensor networks,” *Proceedings of the 4th European Conference on Wireless Sensor Networks.* Springer-Verlag, 2007, pp. 34–50.

[83] A. Kansal, J. Hsu, S. Zahedi, and M. B. Srivastava, “Power management in energy harvesting sensor networks,” *ACM Transactions on Embedded Computing Systems*, vol. 6, no. 4, p. 32, 2007.

[84] X. Jiang, J. Polastre, and D. Culler, “Perpetual environmentally powered sensor networks,” *Proceedings of the 4th IEEE International Symposium on Information Processing in Sensor Networks.* IEEE, 2005, pp. 463–468.

[85] V. Raghunathan, A. Kansal, J. Hsu, J. Friedman, and M. Srivastava, “Design considerations for solar energy harvesting wireless embedded systems,” *Proceedings of the 4th IEEE International Symposium on Information Processing in Sensor Networks.* IEEE Press, 2005, p. 64.

[86] J. Hsu, S. Zahedi, A. Kansal, M. Srivastava, and V. Raghunathan, “Adaptive duty cycling for energy harvesting systems,” *Proceedings of International Symposium on Low Power Electronics and Design.* ACM, 2006, pp. 180–185.

[87] F. Simjee and P. H. Chou, “Everlast: long-life, supercapacitor-operated wireless sensor node,” *Proceedings of International Symposium on Low Power Electronics and Design.* IEEE, 2006, pp. 197–202.

[88] C. Park and P. H. Chou, “Ambimax: Autonomous energy harvesting platform for multi-supply wireless sensor nodes,” *Proceedings of the 3rd Annual IEEE Communications Society on Sensor and Ad Hoc Communications and Networks*, vol. 1. IEEE, 2006, pp. 168–177.

[89] M. A. Weimer, T. S. Paing, and R. A. Zane, “Remote area wind energy harvesting for low-power autonomous sensors,” *Proceedings of Power Electronics Specialists Conference.* IEEE, 2006, pp. 1–5.

[90] S. Roundy, D. Steingart, L. Frechette, P. Wright, and J. Rabaey, “Power sources for wireless sensor networks,” in *Wireless Sensor Networks.* Springer, 2004, pp. 1–17.

[91] A. Kansal, D. Potter, and M. B. Srivastava, “Performance aware tasking for environmentally powered sensor networks,” *ACM SIGMETRICS Performance Evaluation Review*, vol. 32, no. 1, pp. 223–234, 2004.

[92] D. K. Noh and K. Kang, “Balanced energy allocation scheme for a solar-powered sensor system and its effects on network-wide performance,” *Journal of Computer and System Sciences*, vol. 77, no. 5, pp. 917–932, 2011.

[93] D. Noh, J. Kim, J. Lee, D. Lee, H. Kwon, and H. Shin, “Priority-based routing for solar-powered wireless sensor networks,” *Proceedings of the 2nd International Symposium on Wireless Pervasive Computing.* IEEE, 2007.

[94] J. R. Piorno, C. Bergonzini, D. Atienza, and T. S. Rosing, “Prediction and management in energy harvested wireless sensor nodes,” *Proceedings of the 1st International Conference on Wireless Communication, Vehicular Technology, Information Theory and Aerospace & Electronic Systems Technology.* IEEE, 2009, pp. 6–10.

[95] C. Moser, L. Thiele, D. Brunelli, and L. Benini, “Adaptive power management in energy harvesting systems,” *Proceedings of the Conference on Design, Automation and Test in Europe.* EDA Consortium, 2007, pp. 773–778.

[96] A. Cammarano, C. Petrioli, and D. Spenza, “Pro-energy: A novel energy prediction model for solar and wind energy-harvesting wireless sensor networks,” *Proceedings of the 9th International Conference on Mobile Adhoc and Sensor Systems.* IEEE, 2012, pp. 75–83.

[97] T. Voigt, A. Dunkels, J. Alonso, H. Ritter, and J. Schiller, “Solar-aware clustering in wireless sensor networks,” *Proceedings of the 9th International Symposium on Computers and Communications*, vol. 1, June 2004, pp. 238–243 Vol.1.

[98] P. Zhang, G. Xiao, and H.-P. Tan, “A preliminary study on lifetime maximization in clustered wireless sensor networks with energy harvesting nodes,” *Proceedings of the 8th International Conference on Information, Communications and Signal Processing.* IEEE, 2011, pp. 1–5.

[99] P. Z̃hang, G. Xiao, and H.-P. Tan, “Clustering algorithms for maximizing the lifetime of wireless sensor networks with energy-harvesting sensors,” *Computer Networks*, vol. 57, no. 14, pp. 2689–2704, 2013.

[100] H. Gou, G. Li, and Y. Yoo, "A partition-based centralized leach algorithm for wireless sensor networks using solar energy," *Proceedings of International Conference on Hybrid Information Technology.* ACM, 2009, pp. 60–66.

[101] T. Melodia, D. Pompili, and I. F. Akyildiz, "Optimal local topology knowledge for energy efficient geographical routing in sensor networks," *Proceedings of the 23rd Annual Joint Conference of the IEEE Computer and Communications Societies*, vol. 3. IEEE, 2004, pp. 1705–1716.

[102] C. Perkins and E. Royer, "Ad-hoc on-demand distance vector routing," *Proceedings of IEEE Workshop on Mobile Computing Systems and Applications.* IEEE, 1999, pp. 90–100.

[103] I. Yoon, D. K. Noh, and H. Shin, "Multi-layer topology control for long-term wireless sensor networks," *EURASIP Journal on Wireless Communications and Networking*, pp. 1–9, 2012.

[104] H. Kim, Y. Seok, N. Choi, Y. Choi, and T. Kwon, "Optimal multi-sink positioning and energy-efficient routing in wireless sensor networks," in *Information Networking. Convergence in Broadband and Mobile Networking.* Springer, 2005, pp. 264–274.

[105] B. Karp, "Challenges in geographic routing: Sparse networks, obstacles, and traffic provisioning," *Presentation at the DIMACS Workshop on Pervasive Networking*, 2001.

요약

초기의 무선 센터 네트워크는 노드들이 평면 토폴로지를 구성하고 범람 방식으로 데이터를 전달했다. 이 방법은 한 번에 많은 데이터를 중복해서 전송함으로써, 전송 성공률을 하락시키고 에너지의 효율을 나쁘게 한다. 평면 토폴로지의 이러한 특성 때문에 많은 무선 센서 네트워크에서는 클러스터 방식과 같은 계층 토폴로지를 사용해왔다. 하지만, 클러스터 방식은 클러스터 헤드가 다른 노드에 비해 많은 에너지를 소모하기 때문에, 시간이 지날수록 에너지 불균형이 심해지고, 클러스터 헤드가 죽으면 전체 클러스터의 데이터를 싱크 노드에 전달할 수 없게 된다. 한편, 배터리를 사용하는 센서 노드는 한정된 수명을 가지기 때문에, 네트워크를 오래 유지하기 위해서는 죽은 노드를 새 노드로 교체해야 하고, 이로 인하여 에너지의 불균형 문제가 더욱 심각해질 수 있다. 따라서 이러한 에너지 불균형과 노드간의 연결성이 낮아지는 문제를 해결하는 방법의 필요성이 대두되었다.

이를 위해 본 논문에서는 다중 계층 토폴로지 제어 기법을 제안한다. 이 기법은 모든 노드를 여러 계층으로 나누고, 각 노드들은 그보다 상위 계층의 노드들을 그들의 싱크 노드로 간주하고, 다중 싱크 네트워크 기법으로 데이터를 상위 계층의 노드에게 전달한다. 각 노드는 주기적으로 자신이 전달할 데이터량과 남은 에너지량을 이용하여 자신의 수명을 예측하고, 이를 네트워크의 평균 수명과 비교하여 평균 수명 이상으로 살 수 있는 계층을 선택한다. 실험 결과, 이 기법을 적용시킨 네트워크는 전체 네트워크의 수명이 증가하고 노드간의 에너지 불균형이 감소한 것을 확인 할 수 있었다.

추가로 에너지 수집 노드와 배터리를 사용하는 노드가 섞여있는 네트워크에서 수집된 에너지를 활용해서 네트워크의 수명을 연장시키고 연결성을 유지하는 다중 계층 토폴로지 제어 기법을 제안한다. 에너지 수집 노드는 배터리를 사용하

는 노드와는 달리, 영원히 사는 것을 목표로 하기 때문에 노드의 수명을 예측할 수 없기 때문에 일반 노드가 네트워크의 수명을 이용하여 자신의 계층을 정하는 것과는 다른 방식으로 자신의 계층을 결정해야 한다. 따라서 에너지 수집 노드는 충전될 에너지와 소모될 에너지를 계산하여 다음 번 계층 선택 주기까지 살아남을 수 있는 계층을 자신의 계층으로 선택한다. 실험 결과는 이 기법은 수집한 에너지를 활용하여 전체 네트워크의 수명을 성공적으로 연장 시키고 노드간의 연결성을 유지하게 하는 것을 알 수 있었다.

주요어: 무선 센서 네트워크, 토폴로지 제어, 계층, 에너지 인지, 에너지 수집

학번: 2006-21234

저와 함께한 모든 분들께 감사드립니다.