Research on Effective Sensor Control Methods for Sparse Sensor Networks

January 2011

Kriengsak TREEPRAPIN

Research on Effective Sensor Control Methods for Sparse Sensor Networks

Submitted to
Graduate School of Information Science and Technology
Osaka University

January 2011

Kriengsak TREEPRAPIN

Lisf of Publications

1. Journal Paper

1. Treeprapin, K., Kanzaki, A., Hara, T., and Nishio, S.: An Effective Mobile Sensor Control Method for Sparse Sensor Networks, *Sensors*, Vol. 9, No. 1, pp. 327–354 (Jan. 2009).

2. Treeprapin, K., Kanzaki, A., Hara, T., and Nishio, S.: A Node Deployment Strategy for Efficient Sensing with Mobile Sensors in Sparse Sensor Networks, *IPSJ Journal*, Vol. 19, No. 3, (Mar. 2011, to appear).

2. International Conference Paper

1. Treeprapin, K., Kanzaki, A., Hara, T., and Nishio, S.: A Mobile Sensor Control Method for Sparse Sensor Networks, in *Proceedings of ACM Symposium on Applied Computing (SAC 2007)*, pp. 886–890 (Mar. 2007).

2. Treeprapin, K., Kanzaki, A., Hara, T., and Nishio, S.: On a Mobile Sensor Control Method for Uniform Sensing in Sparse Sensor Networks, in *Proceedings of International Workshop on Sensor Network Technologies for Information Explotion Era (SeNTIE 2008)*, pp. 125–132 (Apr. 2008).

3. Treeprapin, K., Kanzaki, A., Hara, T., and Nishio, S.: An Efficient Node Deployment Strategy in Sparse Sensor Networks, in *Proceedings of International Workshop on Data Management for Information Explosion in Wireless Networks (DMIEW 2009)*, pp. 1060–1065 (Mar. 2009).

3. Domestic Conference Paper (with peer-review)

1. Treeprapin, K., Kanzaki, A., Hara, T., and Nishio, S.: A Mobile Sensor Control Method for Sparse Sensor Networks, in *Proceedings of DICOMO 2006*, Vol. 2006, No. 6, pp. 173–176 (July 2006) (in Japanese).

2. Treeprapin, K., Kanzaki, A., Hara, T., and Nishio, S.: A Node Deployment Method for Efficient Sensing with Mobile Sensors in Sparse Sensor Networks, in *Proceedings of DPSWS*, Vol. 2008, No. 14, pp. 291–296 (Dec. 2008) (in Japanese).

4. Domestic Conference paper

1. Treeprapin, K., Kanzaki, A., Hara, T., and Nishio, S.: A Mobile Sensor Control Method in Wide-Area Sensor Networks, in *Proceedings of IEICE SIG Notes SN2006-47*, pp. 59–66 (Dec. 2006) (in Japanese).

Abstract

Recent advances in wireless communication technologies have led to an increasing interest in ad hoc networks constructed of only wireless terminals that play the role of a router. Especially, as an application of ad hoc networks, there has been a great deal of interest in wireless sensor networks. In a wireless sensor network, wireless nodes that equip several sensor devices (*sensor nodes*) construct an ad hoc network. The data acquired by each sensor node are transferred to the sink node that gathers and manages the sensor data using multi-hop wireless communications. Furthermore, with the development of robotics technologies in recent years, there have been many studies on sensors with a moving facility (*mobile sensors*). By introducing mobile sensors to a wireless sensor network, a large region can be monitored with a small number of sensor nodes. Until now, there have been several studies on data transfer in wireless sensor networks which fully or partially include mobile sensors (*mobile sensor networks*). These studies exploit the moving facility of mobile sensors for transferring sensor data to the sink node. However, in these studies, each mobile sensor has to move to the sink node every time it performs a sensing operation. Thus, the performances of sensing and data transfer become low due to the increase in the moving distance, especially in a large region.

In this thesis, we propose mobile sensor control methods in order to improve the performances of sensing and data transfer in sparse networks. Our methods uses two types of sensor nodes, *fixed node* and *mobile node*. First, to achieve efficient sensing, data acquired by nodes are accumulated on a nearby fixed node before being transferred to the sink node. By doing so, unlike conventional studies, mobile nodes do not need to move to the sink node every time they perform sensing operations. This decreases the moving distance of each mobile node for performing sensing operations, and enables each node to perform sensing operations more frequently than conventional methods. In addition, after accumulating data, each fixed node requests mobile nodes to construct a multi-hop communication route and transfers the accumulated data to the sink node. By doing so, efficient data transfer can be achieved even in a sparse environment.

This thesis consists of five chapters. First, we describe the research background of mobile sensor networks and discuss the problems of related work in Chapter 1. We also present the fundamental design of our methods for achieving efficient sensing and data transfer in a sparse environment in this chapter.

In Chapter 2, we propose a mobile sensor control method to transfer data efficiently in sparse networks. As described earlier, this method uses two types of sensor nodes. Moreover, in order to make it easy for fixed nodes to collect mobile nodes for transferring data, this method defines the moving strategy of mobile nodes. Specifically, each mobile node in this method travels around the nearby fixed nodes before moving to its sensing point. By doing so, fixed nodes can have many opportunities to connect with mobile nodes, and thus, can easily collect mobile nodes for transferring data. We also conduct simulation experiments to verify that our method can achieve efficient sensing and data transfer compared with conventional methods.

In Chapter 3, we extend the method proposed in Chapter 2 to further improve the performance focusing on the locations of fixed nodes. First, we analyze the effects of locations of fixed nodes on the performance of the method proposed in Chapter 2. Based on the result of the analysis, this extended method strategically determines the location of each fixed node. We also conduct simulation experiments to verify that this method further improves the performances of sensing and data transfer.

In Chapter 4, we propose another method which is an extension of the method proposed in Chapter 2 focusing on operations of mobile nodes. This method divides the target region into multiple areas, and statically deploys mobile nodes to each divided area. In addition, this method adjusts the number of mobile nodes deployed in each area based on the analysis of the performance. We also conduct simulation experiments to verify that this method further improves the performances of sensing and data transfer. Moreover, we discuss the integration of methods proposed in Chapters 3 and 4 to further improve the performance. Furthermore, we discuss the cost for deploying nodes based on the performance evaluation of the integrated method.

Finally, Chapter 5 summarizes this thesis and discusses our future work.

Contents

Chapter 1

Introduction

1.1 Background

Recent advances in wireless communication technologies have led to an increasing interest in ad hoc networks constructed of only wireless terminals that play the role of a router. In an ad hoc network, an arbitrary pair of wireless terminals communicate with each other without any infrastructure (e.g. access point in wireless LAN). Even when a pair of terminals cannot communicate directly, the communication can be conducted using a multi-hop communication route constructed of intermediate terminals. Since an ad hoc network allows devices with wireless interfaces to communicate with each other without any pre-installed infrastructure, many applications including disaster recovery and wide area surveillance are expected to be realized [19, 32, 44, 47]. Until now, there have been many studies on ad hoc networks such as media access controlling [10, 37], routing [9, 72, 73, 96], and data management [62, 68].

As an application of ad hoc networks, wireless sensor networks attract a lot of attention in recent years. In wireless sensor networks, a large-scale sensing system is constructed by cooperative behaviors of multiple sensor nodes. Thus, sensor networks are expected to be applied to many applications such as environmental monitoring [13, 22, 29, 46], investigation of ecological system [25, 27], object tracking [40, 93, 94], and building management [24, 28]. In wireless sensor networks, each sensor node senses physical phenomena in its sensing range. The sensor data are transferred to the sink node that gathers and manages data via multi-hop communication route. Generally, sensor nodes are equipped with poor resources such as battery, memory space, and communication range. Considering these limitations, there have been many studies on wireless sensor networks such as power saving [33,34], data compression [48,60], and routing [52,53].

Here, there are some environments where it is difficult to deploy a large number of sensor nodes such as disaster sites [45], planetary exploration [2,3,8,11, 20,39], pollute areas [50,86], and underwater [1,16,51,55,57,61]. For example, in a planetary or underwater exploration, it is necessary to deploy sensor nodes with high durability in order to operate the system in hazardous environments with several effects such as radiation and water pressure. This may lead the significant increase of the cost for developing and deploying sensor nodes. Thus, a large number of nodes cannot be deployed. Moreover, in a polluted plant, the target region for monitoring is too large to deploy an enough number of nodes to form a stable and fully connected multi-hop network. Although some studies assume applications where a large number of nodes are deployed from the air (e.g. from airplanes or helicopters) [14,15], such a deployment becomes impossible in a building or under the heap of ruins. Furthermore, long range radio waves cannot improve the connectivity in these environments, since it is affected by the ambient surrounding such as obstacles and landscape. In such environments, due to the limited communication range of each sensor node, it becomes difficult to transfer data to the sink node. Moreover, since the sensing range of each sensor node is limited, it becomes difficult to monitor the whole region.

To solve this problem, there have been many studies on sensor nodes with a moving facility (*mobile sensors*) [4,17,18,21,23,36,38,64,87]. By introducing mobile sensors to a wireless sensor network, it is expected that a large region can be monitored with a small number of sensor nodes. For example in Figure 1.1, each mobile sensor can monitor an arbitrary area by moving to the area of interest. In addition, data transfer to the sink node can be achieved by the movement of mobile sensors. When multiple mobile sensors cooperate with each other, a multi-hop communication route to the sink node can be constructed. Due to these characteristics, mobile sensors are well suited for a 'sparse' environment, where the number of mobile sensors is small compared with the size of the region. In this thesis, we call sensor networks which fully or partially include mobile

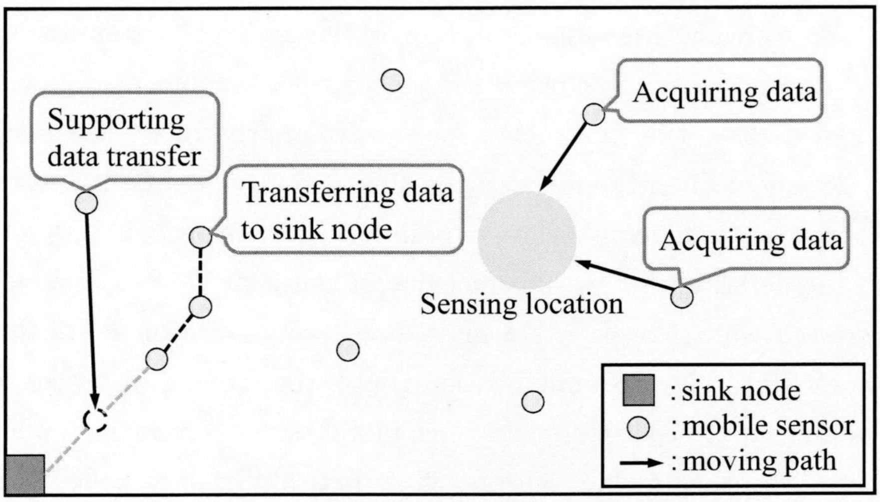


Figure 1.1: Mobile sensor networks.

sensors as *mobile sensor networks*.

1.2 Research Issues on Mobile Sensor Networks

As described in Section 1.1, mobile sensor networks are expected to be applied to a wide variety of applications. Among them, one of the typical applications is monitoring a target region. In this kind of application, the system has to acquire data from the whole region. For example, in an investigation of ocean floor, in order to obtain the map of landscape or temperature, or to detect a sunken ship or mineral vein, the whole region has to be monitored.

In a sparse environment, the number of nodes becomes too small to acquire data in the whole region at once. In order to acquire a sufficient amount of data from the whole region, it is important to design the cooperative behavior (i.e., to control the movement) of multiple mobile sensors. For example, it is redundant that multiple mobile sensors simultaneously acquire data at the same location (sensing point). To avoid such a redundant monitoring, it is necessary to share information of sensing points among mobile sensors. In addition, in order for a mobile sensor to acquire more data in a short time, reducing the moving distance between sensing points is expected to be effective.

Moreover, in a sparse environment, the connectivity between mobile sensors becomes extremely low due to the limitation of the communication range of nodes. Thus, it is also necessary to control the movement of mobile sensors to transfer the acquired data to the sink node. As a naive approach, we can assume that each mobile sensor directly moves to the sink node every time it acquires data. In this approach, the moving distances of mobile sensors become very large. This results in the deterioration in the efficiency of sensing. We can assume another naive approach which constructs a multi-hop communication route to the sink node. By constructing a communication route, the moving distances of mobile sensors can be suppressed. However, since mobile sensors have less opportunities to connect with other mobile sensors, it is quite difficult to collect an enough number of multiple mobile sensors for constructing a communication route.

Furthermore, in the case where mobile sensors monitor in an unexplored region, it is impossible to determine the movement of mobile sensors in advance. Thus, the movement of each mobile sensor has to be determined by its autonomous behaviors.

In this thesis, we focus on achieving efficient sensing and data transfer in sparse sensor networks by control the mobility of mobile sensors.

1.3 Assumptions

Before presenting the details of our study, we make clear the assumed environment in this thesis.

First, as described earlier, we assume an application which monitors a vast and hazardous square region with a small number of nodes. To acquire data from the whole region with a small number of nodes, mobile sensors are introduced into the network. Following the conventional works [42,54,56,88], we assume that each mobile sensor can move freely and autonomously. Each mobile sensor acquires data whose sizes are relatively large such as pictures or movies. Moreover, we assume that the cost of preparing storage media is very large. This is because, it is well known that high-density storage media tend to be easily broken due to several effects such as radiation in hazardous environments [7]. Thus, it is difficult to prepare large storage media for all mobile sensors.

Each mobile sensor knows its present location by using GPS or other location detection methods [6, 31]. The sensor data acquired by mobile sensors are transfered to the sink node located at a corner of the area similar to the assumption in many conventional works such as target detecting, tracking and environment monitoring [40, 57, 90]. Each mobile sensor has a unique identifier in the network. In addition, we assume that all mobile sensors have the same sensor and radio devices. Thus, the sensing and wireless communication ranges are same among all mobile sensors. Following the conventional works [12, 35, 49, 67, 70, 71, 76, 90, 91], for simplicity, we assume that there are no obstacles in the region.

1.4 Related Work

Until now, many studies on mobile sensor networks have been conducted.

In [30, 56, 65, 89], assuming relatively dense environments, the authors proposed algorithms to control mobile sensors in order to acquire data in the whole target region. These studies adjust the locations of mobile sensors in order to improve the coverage of the entire region while keeping the connectivity of the network. These studies cannot be applied to sparse environments.

Some studies [58, 69] introduce mobile sensors to help operations of tiny sensor nodes without mobility. In these studies, mobile sensors transport energy between sensor nodes. Specifically, each mobile sensor receives (collects) energy from sensor nodes with high residual energies. Then, a mobile sensor moves to an area where there are many sensor nodes with low residual energies, and sends (gives) energy to these nodes. In there studies, the mobility of mobile sensors is used for transporting energy, not for sensing and data transfer.

There also have been many studies on controlling the mobility of mobile sensors for sensing and data transfer. These studies can be categorized into the following three types.

1.4.1 Data Transfer Using Nearby Nodes

In [41, 92], Wang et al. proposed a simple and efficient data transfer method named DFT-MSN(Delay/Fault Tolerant Mobile Sensor Network). In DFT-MSN, mobile sensors are classified into two types of nodes, *mobile sensor nodes* and *high-end sink nodes*. The high-end sink nodes are deployed at strategic locations where mobile sensor nodes visit with high probability. In addition, they can directly connect to the sink node all the time by changing their transmission power if necessary. Each mobile sensor node acquires data and sends it to a nearby high-end sink node by flooding with a probability in order to prevent duplication of transmitted data. After that, the high-end sink node transmits the received data to the sink node. This method assumes that each high-end sink node has to connect to the sink node all the time. As described in Section 1.1, this assumption cannot be applied to the environment assumed in this thesis.

In [71], Son et al. proposed a data transfer method assuming random and uncontrolled mobility of mobile sensors. In this method, each mobile sensor periodically sends information on its location to nearby sensors. Using this information, each mobile sensor predicts the locations of other sensors in the future and determines a mobile sensor to send the data. In [43], Liu and Wu proposed a data transfer method assuming predetermined mobility of mobile sensors. In this method, each mobile sensor determines a mobile sensor to which it forwards its holding data based on the moving paths of nearby mobile sensors. In [95], Yuan et al. proposed an area-based data transfer method. In this method, mobile sensors exchange their past moving paths with each other. Using this information, each mobile sensor predicts the area to which each nearby sensor will move in the future. When a mobile sensor wants to send data to an area, it forwards the data to a nearby sensor that is expected to move to the area. However, these methods do not work in a sparse network due to the low connectivity between mobile sensors.

1.4.2 Data Transfer Using Communication Routes

In [66, 67], Shinjo et al. proposed a data gathering method considering a sparse network. This method utilizes a broadcasting system to control the movement of mobile sensors. Specifically, the information on the locations of mobile sensors connected with the sink node by single or multi-hop communication routes is broadcasted to all mobile sensors. By using this broadcasted information, each mobile sensor can adjust the moving destination in order to decrease the moving distance to connect with the sink node or a mobile sensor which has already connected with the sink node. However, since this method needs a broadcasting system, it is difficult to be applied to the environments assumed in this thesis. For example, in a building or underwater, it is generally difficult for mobile sensors to surely receive broadcasted information.

In [59], Rao and Kesidis proposed an algorithm which governs the behaviors of mobile sensors in order to minimize the total energy consumption for moving and communication. In this algorithm, all mobile sensors have an associated clock cycling and each mobile sensor determines with a probability whether it acquires data or transmits data acquired by another mobile sensor in every cycle. In the latter case, the mobile sensor moves to construct a communication route between the sink node and a mobile sensor that determines to transmit data. By doing so, this algorithm can construct a communication route to the sink node and transfer the acquired data by autonomous behaviors of mobile sensors. However, this algorithm does not work well in a sparse network because it becomes difficult to construct a communication route due to the low connectivity between mobile sensors.

In [5], Arboleda and Nasser proposed a clustering algorithm which divides the region into several zones and clusters mobile sensors based on the divided zones. In this method, each mobile sensor periodically sends information on the frequency that it traverses the borders between zones. The mobile sensor with the smallest frequency among all sensors in a zone is elected as the cluster head. The cluster head in each cluster acts as a router in the network and transmits the data acquired in the corresponding cluster to the cluster head in an adjacent zone to which the multi-hop communication route is most stable. This algorithm has to periodically exchange information in order to elect cluster heads. In addition, data are transferred to the sink node via multi-hop communication routes between cluster heads. Thus, this algorithm does not work well in a sparse environment.

1.4.3 Data Transfer by Moving to the Sink Node

In [12,26,74,75,76,77,87,90], other data transfer methods in mobile sensor networks were proposed. In these methods, data transfer is conducted basically by the movement of mobile sensors to the sink node. Unlike the above two types of data transfer methods, these methods can be applied to the environment assumed in this thesis.

RAMOS

In [74,77], Suzuki et al. proposed RAMOS (Routing Assisted by Moving Objects). In RAMOS, data are categorized into two levels according to the emergency: high emergency level and low emergency level. Moreover, RAMOS defines the following six modes for each mobile sensor. Each mobile sensor autonomously controls its behavior by changing its mode according to the existence of data and the emergency level of the data.

- Absolutely Static (*AS*) mode

  A mobile sensor in *AS* mode performs a sensing operation without moving. After that, it transmits the data to the connected mobile sensor which is the nearest to the sink node. Otherwise, it waits for another mobile sensor to connect.

- Semi Static (*SS*) mode

  A mobile sensor in *SS* mode performs a sensing operation without moving. When a mobile sensor holds data and connects to mobile sensors except that in *AS*, it transmits the data to one of them. Otherwise, it moves directly to the sink node when the emergency level of its acquired data is high. Moreover, it transfers data to the connected mobile sensor while moving to the sink node. After transfered the data, a mobile sensor returns to its original position. On the other hand, if the emergency of its acquired data is low, the mobile sensor waits for another mobile sensor to connect. Then if no mobile sensor connects for a certain period, the mobile sensor changes its mode into *LS* mode.

Table 1.1: Conditions for changing mode in RAMOS

Changing of mode	Condition
AS	(no change)
SS→*LS*	No neighboring mobile sensor exists and emergency level of the data is low.
LS→*SS*	Cannot connect to any mobile sensor for a certain period.
DN	(no change)
SM→*RP*	Receives data from a node in *AS* or *SS* mode.
RP	(no change)

- Limited Search (*LS*) mode

  A mobile sensor in *LS* mode moves in a limited area in order to find another mobile sensor to which it can transfer its holding data. If it cannot connect to any other mobile sensor for a certain period, it returns to its original position and changes its mode into *SS* mode.

- Dynamic Node (*DN*) mode

  A mobile sensor in *DN* mode moves around the area and transfers the data. When a sensor holds data and it connects to mobile sensors in *SM* or *RP* mode, it transmits the data to one of them. Otherwise, it directly moves to the sink node to transfer data when the emergency level of the data is high. On the other hand, if the emergency level of the data is low, it moves to the sink node after waiting for other mobile sensors in *SM* or *RP* mode to connect. When the mobile sensor connects to a mobile sensor in *SM* or *RP* mode while moving to the sink node, it transmits the data to the connected mobile sensor and returns to its original position.

- Spontaneously Moving (*SM*) mode

  A mobile sensor in *SM* mode moves around the area to find other mobile sensors that hold data. When a mobile sensor holds data and connects to mobile sensors in *SM* mode, it transmits the data to one of them. When it connects to a mobile sensor in *RP* mode, it transmits only data with the high emergency level. On the other hand, when it connects to mobile sensors in *AS* or *SS* mode, it receives data from them and changes its mode into *RP* mode.

- Round Patrol (*RP*) mode

  A mobile sensor in RP mode moves around mobile sensors in *AS* and *SS* modes. When it receives data from them, it moves to the sink node. When the mobile sensor connects to a mobile sensor in *SM* mode while moving to the sink node, it transmits the data to the connected mobile sensor and returns to its original route.

Here, mobile sensors in *SS*, *SM* and *RP* modes move directly to the sink node until they connect to another mobile sensor. Table 1.1 shows the conditions for changing mode in RAMOS.

In RAMOS, each mobile sensor transfers data to the sink node by changing its mode autonomously. However, since each mobile sensor has to move to the sink node to transfer data until it connects to another mobile sensor, the moving cost much increases. Moreover, in a sparse environment, since each mobile sensor has few opportunities to connect to other sensors, the performances of sensing and data transfer become low.

UM

In [90], Wang and Ramanathan proposed a sensing method using uncoordinated mobile sensors (*UM nodes*). In this method, each UM node acquires data until the amount of the acquired data reaches the memory capacity. Moreover, each UM node exchanges information on the acquired data with a connected UM node and deletes the data which were acquired by the connected node at the same location and time. By doing so, duplicated sensing (sensing a location by multiple mobile sensors) can be suppressed and more data can be accumulated in the memory space of a UM node.

In addition, this method proposes two mobility models of UM nodes, the multi-homed random way point model and the controlled mobile nodes model.

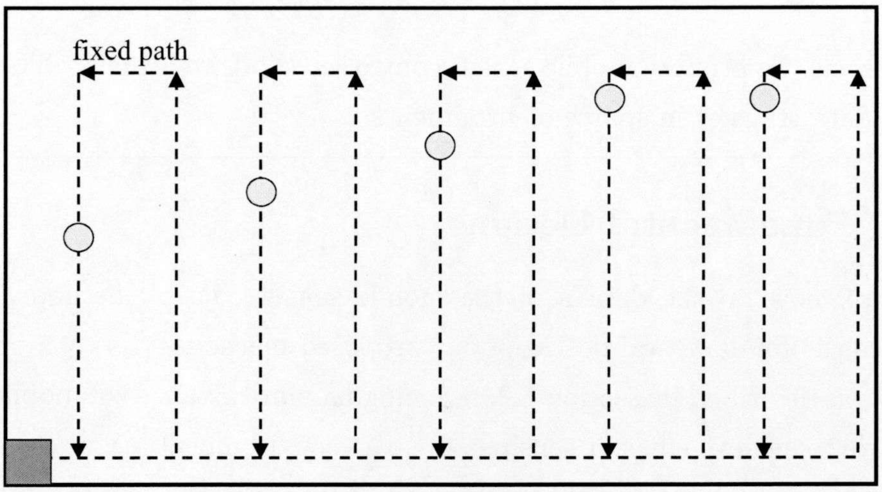


Figure 1.2: Controlled mobile nodes model.

In the multi-homed random way point model, each UM node randomly chooses the destination and moves there. After reached the destination, it stochastically determines whether it returns to the sink node or it moves to a new destination. In the latter case, since each UM node selects the destination randomly from the whole region, the moving distance to the destination tends to increase. Thus, the efficiency of sensing decreases especially in a wide region.

On the other hand, in the controlled mobile nodes model, the moving path of each UM node is determined in advance and each node does the sensing operation while moving its path as shown in Figure 1.2. Furthermore, by setting moving paths of UM nodes to reduce unnecessary movement, the efficiency of sensing can be further improved. However, it is necessary to calculate the moving paths of all UM nodes in advance. Thus, the computational cost becomes very large. Moreover, this model cannot handle dynamic changes of conditions such as the existence of obstacles. Thus, moving paths cannot be determined in an unknown or highly dynamic region (e.g. planetary or underwater).

1.5 Organization of Thesis

In this thesis, we propose mobile sensor control methods to achieve efficient sensing and data transfer in sparse environments.

1.5.1 Fundamental Design

Before we move to the details of the mobile sensor control methods, here we introduce the fundamental design of our proposed methods.

First, as described in Section 1.2, reducing moving distances of mobile sensors is effective to achieve efficient sensing. In sparse environment, each mobile sensor does not know the locations of other sensor nodes, since the connectivity of each mobile nodes is very low. Therefore, each mobile node has to move to the sink nodes to transfer data. To reduce the moving distance, we focus on the approach in [41,92]. As described in Section 1.4.1, mobile sensors in this approach transfer their acquired data to a nearby high-end sink node that does not move. By doing so, mobile sensors can finish the data transfer faster than the approaches in [74,77,90] where mobile sensors directly move to the sink node, and can quickly return to the sensing operations. However, the approach in [41,92] cannot be applied to the environment assumed in this thesis because the direct connections between high-end sink nodes and the sink node are needed.

To solve this problem, we focus on the approach in [59]. As described in Section 1.4.2, this approach constructs a multi-hop communication route to transfer data acquired by a mobile sensor far from the sink node. Thus, when a (multi-hop) communication route from a high-end sink node to the sink node can be constructed, it is expected that efficient data transfer can be achieved.

Based on this idea, our proposed methods introduce *fixed node*s which do not move like high-end sink nodes in [41,92]. As shown in Figure 1.3, a fixed node has two roles in addition to sensing, 1) temporarily accumulating data acquired by nodes, and 2) constructing a communication route between fixed nodes for transferring the accumulated data toward the sink node. In order to achieve the above roles, a fixed node has to be equipped with high-performing resources (e.g. larger memory capacity). This causes the increase of the cost of preparing fixed nodes. Thus, we assume that a small number of fixed nodes are introduced. Note

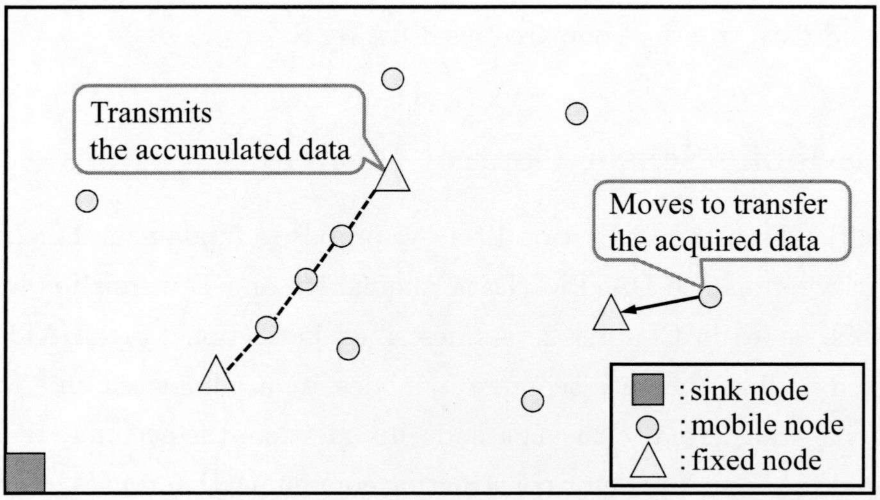


Figure 1.3: System model of our proposed methods.

that the cost of preparing and deploying fixed nodes is discussed in Chapter 4. Here, to distinguish from fixed nodes, we call other mobile sensors *mobile nodes*. As in the methods proposed in [41, 92], locations of all fixed nodes are known by all nodes. By using fixed nodes, each mobile node can transfer its acquired data to a nearby fixed node, and thus, can quickly return to the next sensing operation. This is expected to result in efficient sensing.

In addition, to achieve efficient data transfer to the sink node even in a sparse environment, our proposed methods construct communication routes between fixed nodes using mobile nodes. To reduce the difficulty to collect mobile nodes for transferring data, our proposed methods allow fixed nodes to construct a communication route not only to the sink node but also to another nearby fixed node. By doing so, the required number of mobile nodes for constructing a communication route becomes smaller.

Here, nodes in *AS* mode in RAMOS (described in Section 1.4.3) are similar to fixed nodes in our proposed methods. These do not move, and control other mobile sensors to transfer data toward the sink node. Unlike these nodes, however, fixed nodes in our methods have additional roles described above, that is, 1) temporarily accumulating data acquired by nodes, and 2) constructing a communication route between fixed nodes for transferring the accumulated data toward the sink node. By these characteristics, our methods achieve more efficient sensing and data transfer compared with RAMOS.

1.5.2 Mobile Sensor Control Method

Based on the discussion in Section 1.5.1, we propose a fundamental mobile sensor control method, named DATFM (Data Acquisition and Transmission with Fixed and Mobile node) in Chapter 2. As described in Section 1.5.1, DATFM introduces fixed nodes. The data acquired by nodes are accumulated on a fixed node before being transferred to the sink node. In addition, the accumulated data are transferred to the sink node by constructing communication routes between fixed nodes.

Moreover, in order to make it easier for fixed nodes to construct communication routes, DATFM defines the moving strategy of mobile nodes. This strategy enables fixed nodes to connect with mobile nodes frequently, and to collect mobile nodes that are required to construct communication routes.

Furthermore, DATFM defines another data transfer strategy using the cooperative movement and communication of mobile nodes in order to transfer data even when a communication route cannot be constructed.

We also conduct simulation experiments to verify that our method can achieve efficient sensing and data transfer compared with the conventional methods described in Section 1.4.3.

Chapter 2 is based on our works published in [78, 80, 82].

1.5.3 Extended Methods of DATFM

In DATFM, fixed nodes can be deployed to arbitrary locations. Here, when there are some points which should be carefully monitored (e.g. coral reefs in underwater explorations or sources of toxic gas in the observation of a disaster site), it is effective to deploy fixed nodes to such important points. By doing so, these important points can be continuously monitored by the fixed nodes. On the other hand, when there are no important points (e.g. monitoring a dessert or detection of a mineral vein), fixed nodes can be freely deployed in the region.

In such environments, further improvement of efficiencies of sensing and data transfer is expected by strategically determining locations of fixed nodes.

Therefore, in Chapter 3, which is based on our works published in [81,83,85], we propose an extended method, named DATFM/DF (DATFM with deliberate Deployment of Fixed nodes), to further improve the performance of DATFM focusing on the locations of fixed nodes. First, we analyze the effects of locations of fixed nodes on the performance of DATFM. Based on the result of the analysis, DATFM/DF strategically determines the location of each fixed node. We also conduct simulation experiments to verify that DATFM/DF further improves the performances of sensing and data transfer.

On the other hand, in DATFM, mobile nodes move around the whole region in order to acquire data in the whole region. By doing so, the whole region can be monitored even when some mobile nodes become unavailable due to the battery exhaustion or physical damages. However, in some environments where mobile nodes with much higher durability are deployed, mobile nodes do not need to move around the whole region, i.e., the movement of mobile nodes in DATFM becomes redundant. Therefore, it is expected that we can further improve efficiency of sensing by decreasing moving distances of mobile nodes.

Therefore, in Chapter 4, which is based on our works published in [79,84], we propose another extended method, named DATFM/DA (DATFM with Deployment Adjusting), focusing on sensing operations of mobile nodes. First, we divide the target region into multiple areas and statically deploys mobile nodes to each divided area. Each mobile node basically moves within the divided area where it is deployed. Next, we analyze the effects of number of mobile nodes in each divided area on the performance. Based on the analysis result, DATFM/DA adjusts the number of mobile nodes deployed in each divided area. We also conduct simulation experiments to verify that this method further improves the performances of sensing and data transfer. Moreover, we also discuss the integration of DATFM/DF and DATFM/DA.

Chapter 2

Mobile Sensor Control Method for Sparse Sensor Networks

2.1 Introduction

As described in Chapter 1, conventional data transfer methods in mobile sensor networks [12,74,75,76,77,87,90] cannot achieve efficient sensing and data transfer in a sparse environment because data transfer in these methods are performed mainly by the movement of mobile sensors.

In this chapter, in order to solve this problem, we propose DATFM (Data Acquisition and Transmission with Fixed and Mobile node), which is the fundamental method proposed in this thesis. DATFM uses two types of sensor nodes, fixed node and mobile node. The data acquired by nodes are once accumulated on a fixed node, and then, transferred to the sink node by using a communication route constructed by cooperative behaviors of multiple mobile nodes. Moreover, in order to make it easier for fixed nodes to construct communication routes, DATFM defines the moving strategy of mobile nodes. This strategy enables fixed nodes to connect with mobile nodes frequently, and to collect an enough number of mobile nodes to construct a communication route.

The remainder of this chapter is organized as follows. In Section 2.2, we present the definitions of nodes, fixed nodes and mobile nodes, introduced in our proposed method. In Section 2.3, we explain the details of DATFM. The results of simulation experiments are presented in Section 2.4. Finally, we conclude this chapter in Section 2.5.

2.2 Definitions of Nodes

Before we move to the details of DATFM, here we present the detailed functions of fixed and mobile nodes.

2.2.1 Fixed node

In order to reduce the moving distance of mobile sensors, we introduce fixed nodes which do not move in DATFM. a fixed has larger memory space compared with a mobile node and accumulates data acquired by itself and other nodes. In addition, it controls nearby mobile nodes to construct a communication route when transmitting the accumulated data toward the sink node. Here, the sink node is classified as a fixed node in DATFM. The locations of all fixed nodes are known by all nodes.

DATFM divides the region into several areas based on a Voronoi diagram in which fixed nodes are the site points. Here, the Voronoi diagram of a set of points partitions the region into convex polygons that consist of the vertical bisectors of the points. Every point in a polygon is closer to the site point in the corresponding polygon than to any other site points. In DATFM, each fixed node has charge of the corresponding area. In other words, each fixed node has a role for collecting data acquired in the area it exists. We call the area for each fixed node as its *territory*. Figure 2.1 shows an example of Voronoi diagram and divided territories. In each territory, mobile nodes connect to the corresponding fixed node before performing sensing operations (The details are described in Section 2.3.1). Every time a fixed node connects with a mobile node, it determines the sensing point of the mobile node, and sends the information of the sensing point. In this thesis, we assume that a fixed node controls the sensing point of each mobile node in order for the entire of its territory to be monitored uniformly.

Each fixed node holds information on nearby mobile nodes that directly connected to it (entered wireless communication range) for a certain period; the

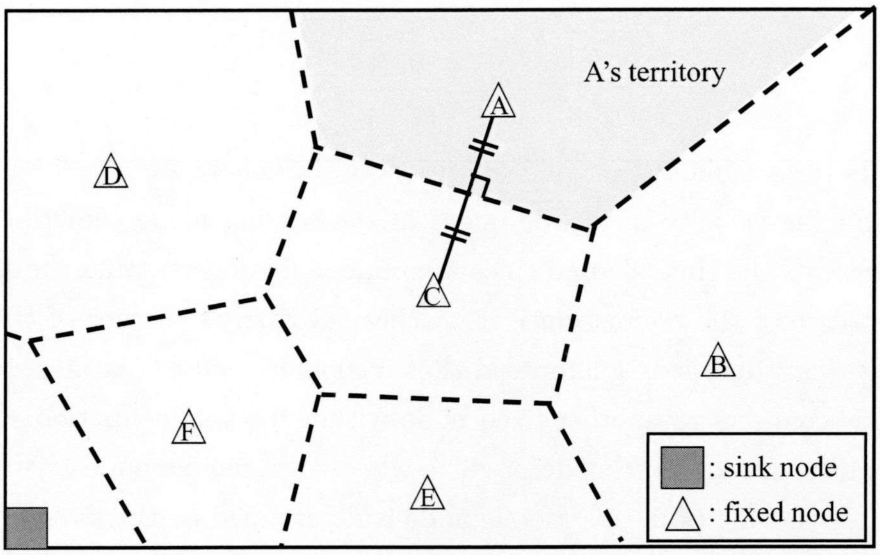


Figure 2.1: Dividing the region.

Table 2.1: Information of mobile nodes held by a fixed node.

Node ID	Connected time	Next fixed node	Destination
a	1,232	*E*	(832,324)
c	1,266	*F*	(542,255)
f	1,335	*E*	(832,324)
i	1,552	*C*	(754,743)
o	1,632	*C*	(754,743)

information for each mobile node includes the identifier of the node, the time when the node connected to it, the fixed node that the node goes to next, and the next destination (sensing point or location of the next fixed node to go) of the node. Table 2.1 shows an example of the information held by a fixed node. Each fixed node sets the validity period for each record in the information (each row in Table 2.1), and removes it when the validity period expires. Here, the validity period for mobile node i is calculated by the following formula:

$$VP_i = \frac{\sqrt{2 \cdot S_{\text{region}}}}{\nu_m}. \tag{2.1}$$

In the above formula, S_{region} and ν_m respectively denote the size of the whole region and the velocity of mobile nodes in the sensing mode (described later). VP_i represents the time elapsed since a mobile node departs from the sink node (i.e. a corner of the region) until it reaches the farthest corner of the region. By using this value, it is guaranteed that a mobile node which departs from a fixed node connects to another fixed node within the validity period. By using the information, each fixed node predicts where each mobile node exists. This is because the moving path of a mobile node is determined by the moving strategy in DATFM (The detail will be described in Section 2.3.1). Thus, each fixed node can calculate the current location of a mobile node using its holding information on the corresponding mobile node.

2.2.2 Mobile node

A mobile node moves around the region. In addition, it has the following three modes:

Sensing mode (*SM*): A node selects a territory to perform sensing and moves there. After performing the sensing operation in the territory, it determines new territory to move.

Collecting mode (*CM*): When a node in *SM* receives a *route request packet (RReq)* from a fixed node, it changes its mode into *collecting mode* (*CM*). In *CM*, a node moves faster than that in *SM* in order to collect other mobile nodes to construct a communication route.

Transmission mode (*TM*): When a node in *SM* receives a *route construction request packet (RCReq)* from a fixed node or a mobile node in *CM*, it changes its mode into *transmission mode* (*TM*). In *TM*, a node constructs a route and transfers the data.

Figure 2.2 shows the mode transition of a mobile node.

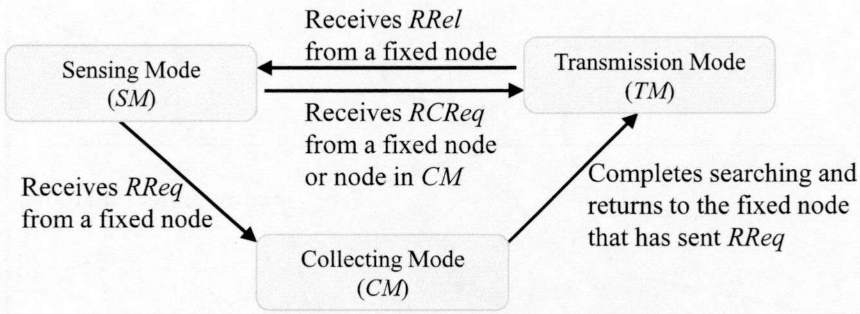


Figure 2.2: Mode transition of a mobile node.

2.3 DATFM

In this section, we explain the details of DATFM.

DATFM defines the moving strategy of mobile nodes in order to make it easier for fixed nodes to construct communication routes to transfer data. Furthermore, DATFM defines a data transfer strategy using the cooperative movement and communication of mobile nodes in order to transfer data even when a communication route cannot be constructed.

2.3.1 Moving Strategy of Mobile Nodes

A mobile node basically sets its mode as *SM*. It selects a territory to perform sensing according to the probability which is proportional to the size of each territory, which can be derived using the locations of fixed nodes. By doing so, DATFM aims to monitor the whole region. After that, it moves to the territory by the following steps:

1. The mobile node moves to connect to the nearest fixed node (i.e. the fixed node in the current territory) and transmits its acquired data.

2. It calculates the distances between the fixed node in the selected territory and all those in the territories adjacent to the territory that the mobile nodes currently exists, and moves to the fixed node that is the nearest to the fixed node in the selected territory. Figure 2.3 shows an example in which

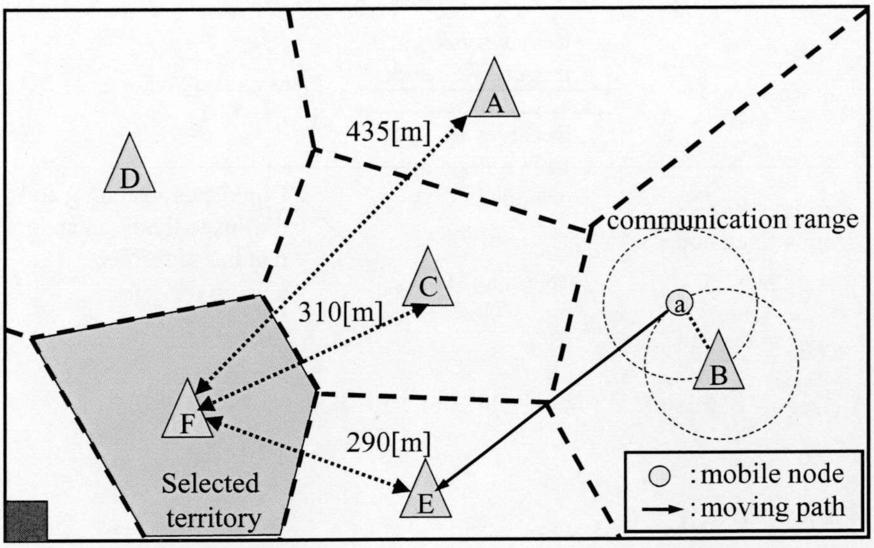


Figure 2.3: Moving path of a mobile node.

mobile node a chooses the next fixed node to move. After transmitting its acquired data to fixed node B, mobile node a calculates the distances between fixed node F that has charge of the selected territory and fixed nodes A, C, and D, which are adjacent to the current territory. After that, it chooses fixed node E that is the nearest to fixed node F and moves there. This procedures is repeated until the mobile node connects to the fixed node in the selected territory.

3. It receives the information of the sensing point from the connected fixed node. Then it moves there and performs the sensing operation.

This moving strategy enables each fixed node to have many opportunities to connect with mobile nodes. Moreover, each fixed node can predict the locations of the recently connected mobile nodes by using the information shown in Table 2.1. Therefore, this strategy makes it easy for fixed nodes to collect mobile nodes in the data transmission process.

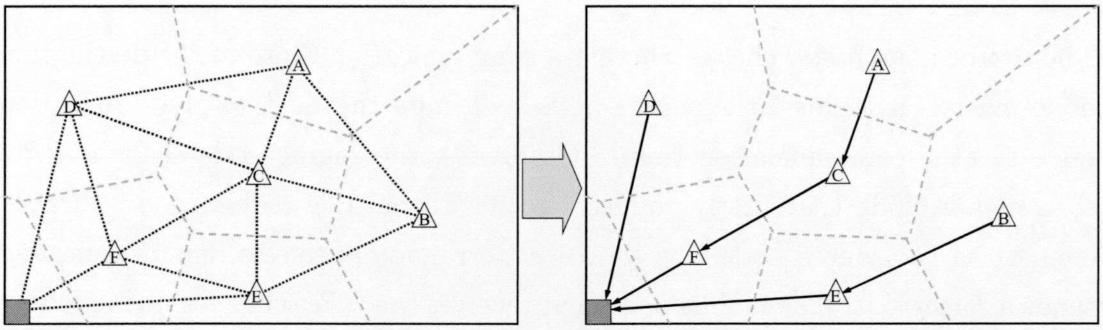


Figure 2.4: Selection of the next fixed node.

2.3.2 Data Transmission

A fixed node starts to transmit the accumulated data when the amount of the accumulated data in its memory exceeds the predetermined threshold, Th. In what follows, we explain the procedures for transferring the accumulated data to the sink node.

Selection of the next fixed node

The fixed node that starts data transmission process (the *source node*) selects a next fixed node to transmit the data (the *destination node*) by using the Delaunay triangulation. The Delaunay triangulation can be performed by connecting the site points in the Voronoi diagram whose polygons share a common edge.

The source node creates Delaunay triangles which include itself, and selects another fixed node that is a vertex of a Delaunay triangle and is the nearest to the sink node. This node is set as the destination node. When the sink node is a vertex of the created Delaunay triangles, the source node selects the sink node as the destination. Figure 2.4 shows an example of selecting the destination node. Fixed node A selects the destination node from fixed nodes B, C and D. In this figure, since fixed node C is the nearest to the sink node, it becomes the destination node. On the other hand, when fixed node D starts data transmission process, it selects the sink node as the destination node since the sink node is a vertex of the created Delaunay triangles.

Request for collecting mobile nodes

The source node firstly checks whether a communication route to the destination node exists. If it does, the source node transmits the data to the destination node via the communication route. Otherwise, the source node sends a *RReq* to a mobile node that firstly connects to it. If multiple mobile nodes already connect to the source node, the source node randomly selects one of them and sends a *RReq* to it. The mobile node that receives the *RReq* changes its mode into *CM*. Here, a *RReq* includes the identifiers of the source and destination nodes, the *required number of mobile nodes*, N_{req}, and the *time limit*, T_{lim}, for collecting mobile nodes. N_{req} is the number of mobile nodes which is required to construct the communication route, and is calculated by the following formula:

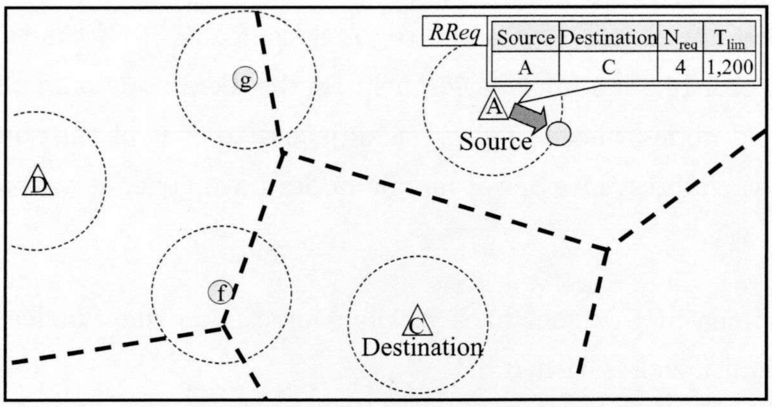
$$N_{\mathrm{req}} = \lfloor \frac{|\mathbf{L}_{\mathrm{src}} - \mathbf{L}_{\mathrm{dst}}|}{R_{com}} \rfloor. \tag{2.2}$$

$\mathbf{L}_{\mathrm{src}}$ and $\mathbf{L}_{\mathrm{dst}}$ are the locations of the source and destination nodes. R_{com} is the wireless communication range. T_{lim} is the time that terminates collecting mobile nodes. The detail of the time limit is described later in this section. Figure 2.5(a) shows an example of *RReq* sent from the source node A.

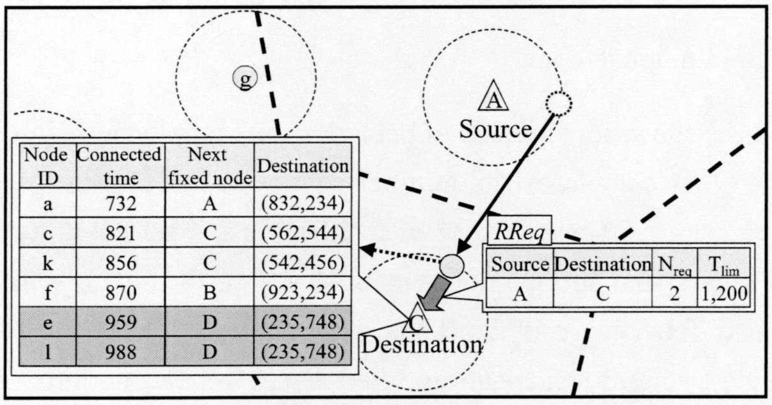
The mobile node in *CM* travels around the nearby fixed nodes to collect the other mobile nodes. First, the mobile node in *CM* moves to the destination node. When the mobile node connects with a fixed node, it performs the following procedures:

1. The mobile node calculates the number of mobile nodes which are performing a sensing operation in the territory of the connected fixed node. This can be done by checking the 'next destination' record in the information held by the fixed node. Specifically, when the 'next destination' is in the territory of the fixed node, the corresponding mobile node is performing the sensing operation in the territory.

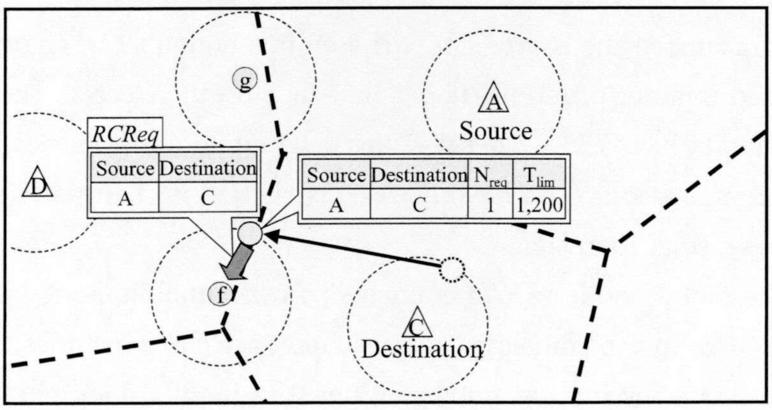
2. It decreases N_{req} in the *RReq* by that calculated number of nodes. This is because the mobile nodes performing a sensing operation in a territory will connect to the fixed node in the territory after the sensing operation. As described later, those mobile nodes will receive *RCReq* from the fixed node, and move to the source node to help the data transmission.



(a) Receiving *RReq* from the source node.



(b) Sending *RReq* to connected fixed node.



(c) Sending *RCReq* to connected mobile node.

Figure 2.5: The operations of mobile node in *CM*.

3. It sends the updated *RReq* to the fixed node.

4. It chooses the next fixed node to go. Specifically, it checks the 'next fixed node' record in the information held by the fixed node, and chooses one of the fixed nodes whose territory is adjacent to that of the source node, to which the largest number of mobile nodes have gone, and which it has not visited yet.

The mobile node in *CM* goes back to the source node when at least one of the following conditions is satisfied:

- The next fixed node to go cannot be found.
- N_{req} in the *RReq* becomes zero.
- The time limit has come.

Figure 2.5(b) shows an example of behaviors of a mobile node in *CM* connected to fixed node C. It calculates the number of mobile nodes which are performing a sensing operation in the territory of fixed node C. In this figure, the mobile node in *CM* knows that mobile nodes c and k are performing a sensing operation since their next fixed node is C. Thus, the mobile node in *CM* decreases N_{req} in the *RReq* by two and sends the updated *RReq* to fixed node C. After that, it checks the 'next fixed node' record in the information held by fixed node C, and chooses fixed node D as the next destination to go since the largest number of nearby mobile nodes (i.e., e and l) have gone.

After returning to the source node, the mobile node in *CM* changes its mode. If the required number of mobile nodes have not been collected, the node in *CM* changes its mode into *TM*. Otherwise, if N_{req} mobile nodes have been collected and the communication route have been constructed, it changes back to *SM* and returns to a sensing operation.

When the mobile node in *CM* connects to other mobile nodes while moving, it sends a *RCReq* to the connected nodes. Here, a *RCReq* includes the identifiers of the source and destination nodes. When the mobile node that receives the *RCReq* is in *SM*, it moves to the source node and changes its mode into *TM*. After that, the mobile node in *CM* decreases the N_{req} in the *RReq* by one. For example in Figure 2.5(c), when a mobile node in *CM* connects with mobile node f, it sends a *RCReq* to f and decreases N_{req}. Here, if the N_{req} in the mobile node in *CM* becomes zero, the mobile node in *CM* returns to the source node.

On the other hand, when a mobile node in *SM* connects to the source node or a fixed node which received the *RReq* from a mobile node in *CM*, the source node or the fixed node sends a *RCReq* to the connected mobile node. The mobile node that receives the *RCReq* moves to the source node and changes its mode into TM. Here, a fixed node which received the *RReq* from the mobile node in *CM* continues to send *RCReqs* until the time limit comes.

Data transmission using the collected node

The source node starts data transmission by using *train transmission*, when it firstly connects to a mobile node in *TM*. In train transmission, data are transfered by cooperative movement and communication among the collected mobile nodes. Specifically, the collected mobile nodes firstly form a line segment (train). To do so, the source node calculates the location, (x_k, y_k), of each collected mobile node m_k by the following formulae:

$$x_k = x_{\text{src}} + \frac{k}{n+1} \cdot (x_{\text{dst}} - x_{\text{src}}), \quad (2.3)$$

$$y_k = y_{\text{src}} + \frac{k}{n+1} \cdot (y_{\text{dst}} - y_{\text{src}}). \quad (2.4)$$

Here, n is the number of collected mobile nodes and each of the collected mobile nodes is assigned an identifier $(m_1, m_2, ..., m_n)$. $(x_{\text{src}}, y_{\text{src}})$ and $(x_{\text{dst}}, y_{\text{dst}})$ are the locations of the source and destination nodes. Then, the source node sends information on the calculated locations to the collected mobile nodes. The collected mobile nodes move to their locations. Figure 2.6 shows an example of calculating the locations of mobile nodes when two mobile nodes are collected.

After that, the collected nodes repeats the following procedures:

1. The source node transmits a part of the accumulated data so that the amount of the transmitted data equals to the sum of memory spaces of the collected nodes.

2. The collected nodes move toward the destination node while keeping same

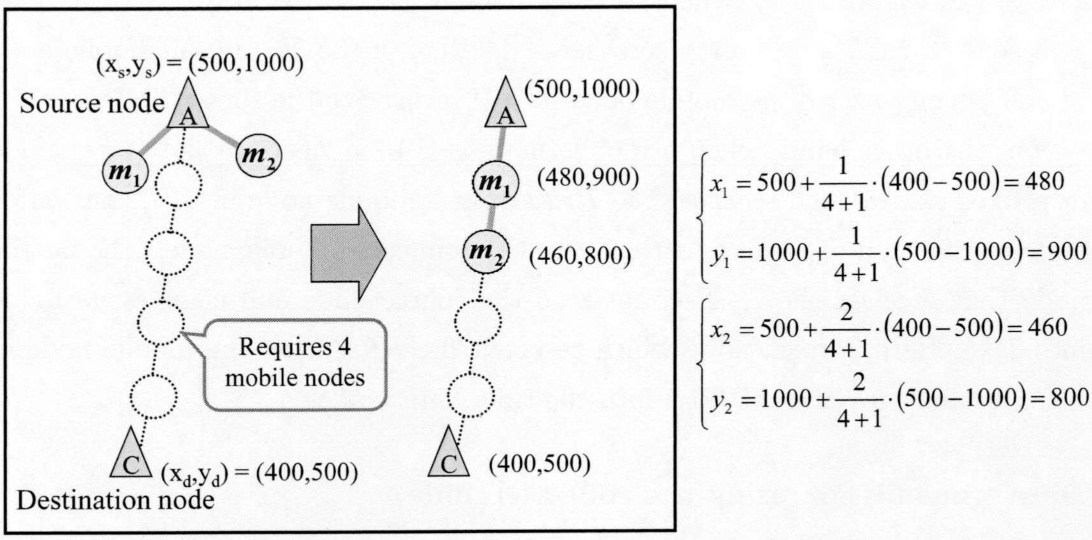


Figure 2.6: Construction of a train.

distances between adjacent nodes, and stop when the node at the other end of the line segment connects to the destination node.

3. The collected nodes transmit all the data to the destination node through the line segment (communication route), and then, move back toward the source node.

Figure 2.7 shows an example of train transmission with two mobile nodes.

Moreover, when another mobile node in *TM* connects to the source node after started the train transmission, the source node adds the connected mobile node to the train. Specifically, the source node increments the identifiers of mobile nodes (i.e., $m_k \rightarrow m_{k+1}$) in the current train and assigns identifier m_1 to the newly connected node. Then, the source node recalculates the location of each mobile node and sends information on the recalculated locations to all the collected mobile nodes. The mobile nodes which received the information move to their new locations and restart the train transmission. Figure 2.8 shows an example when another mobile node connects to the source node. In this figure, the newly connected node is assigned identifier m_1 and added to the train.

Here, when the number of the collected nodes reaches N_{req}, the complete

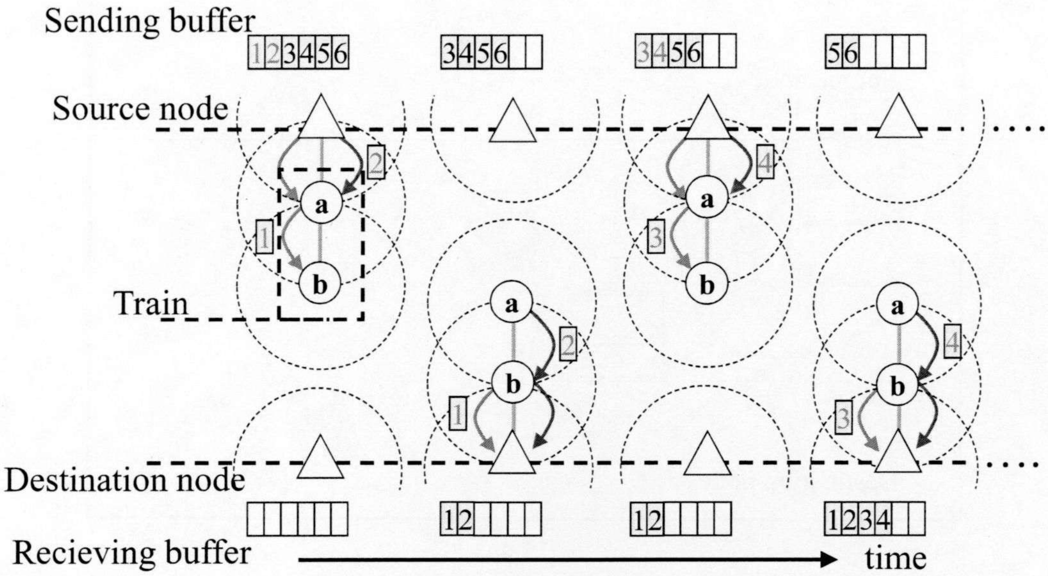
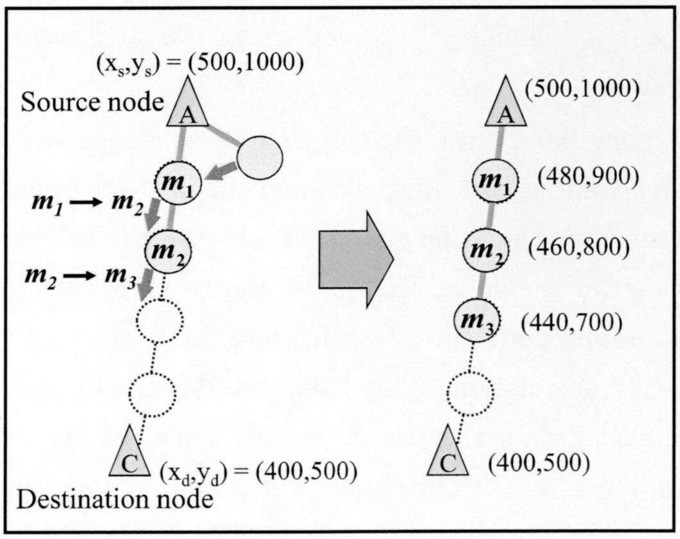


Figure 2.7: Train transmission.



$$\begin{cases} x_3 = 500 + \frac{3}{4+1} \cdot (400 - 500) = 440 \\ y_3 = 1000 + \frac{3}{4+1} \cdot (500 - 1000) = 700 \end{cases}$$

Figure 2.8: Reconstruction of a train.

communication route is constructed. In that case, the source node stops the train transmission and start transmitting the data via the constructed route.

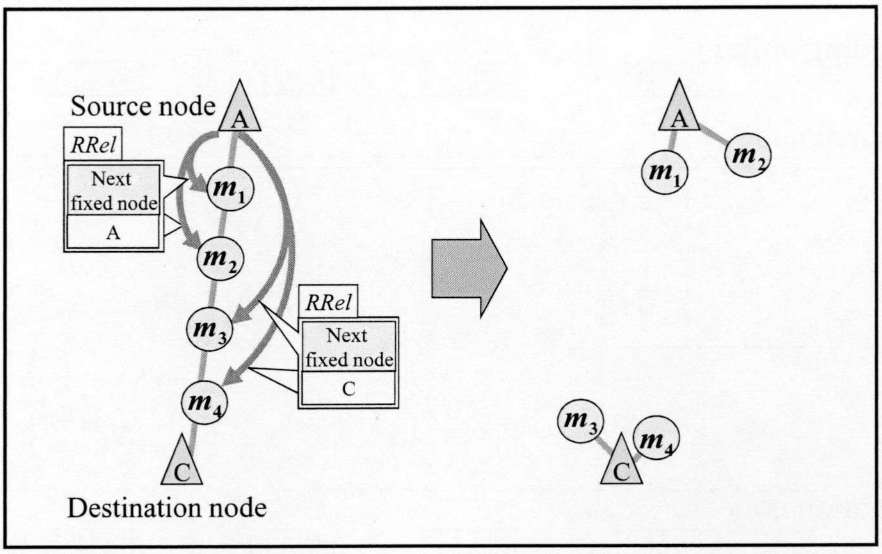


Figure 2.9: Release a communication route.

Moreover, the source node sends a *route release packet (RRel)* to nodes in *TM* which newly connects after constructed the complete communication route. The mobile nodes that received the *RRel* change their mode into *SM* and restart moving to their next sensing points.

After transmitting all the accumulated data, the source node sends a *RRel* to each mobile node in *TM*. A *RRel* includes the information on the next fixed node to go. Each mobile node that receives the *RRel* changes its mode into *SM* and moves to the destination specified in the *RRel*. Here, the source node sets the next fixed node for the half of mobile nodes in *TM* which are far from itself as the destination node of the data transmission. This helps the destination node to collect mobile nodes for the next data transmission process since the amount of data in its memory space may exceed the threshold. On the other hand, for each of the other mobile nodes, the source node sets the next fixed node as itself. Figure 2.9 shows an example when the source node A has transmitted all the accumulated data to the destination node C. The source node A sends *RRels* including C as the next fixed node to mobile nodes m_3 and m_4. It also sends *RRels* including itself as the next fixed node to mobile nodes m_1 and m_2. After that, mobile nodes m_1 and m_2 restart to move to the destination after connected

Table 2.2: Parameters in calculating the time limit.

Parameter	Description
i	IDs of fixed nodes.
$\mathbf{L}_i$	location of fixed node i.
$\mathbf{T}_i$	territory of fixed node i.
$\mathbf{d}_i$	location of the sensing point in $\mathbf{T}_i$ (i.e. $\mathbf{d}_i \in \mathbf{T}_i$).
$\mathbf{S}_F$	set of fixed nodes in the whole region.
S_{region}	size of the whole region (i.e. $S_{\text{region}} = \sum_i \lvert\mathbf{T}_i\rvert$).
ν_m	velocity of mobile nodes in *SM*.
T_{acq}	time for a sensing operation.
N_{mov}	total number of mobile nodes in the whole region.
Th	threshold for starting data transmission.

to fixed node A.

Decision of the time limit

In DATFM, the source node sets the time limit and notifies it to the mobile node in *CM* and other nearby fixed nodes. This value is used for terminating collecting mobile nodes for the corresponding data transmission. Here, when the time limit is set small, it becomes difficult for the source node to collect the sufficient number of mobile nodes. In contrast, when the time limit is set large, the mobile node in *CM* and fixed nodes that received *RReq* continue to collect mobile nodes for the data transmission for a long time. In this case, excessive numbers of mobile nodes are collected even after the complete communication route is constructed. Therefore, it is necessary to set the appropriate time limit.

In order to set the appropriate time limit, we use the estimated time elapsed to collect the required number of mobile nodes. We assume that the parameters in Table 2.2 are given.

Let us denote T_{est_i} as the estimated time in which the required number of mobile nodes connect to fixed node i. Then, the time limit of fixed node i, T_{lim_i}, is calculated by the following formula:

$$T_{\text{lim}_i} = T_{\text{current}} + T_{\text{est}_i}. \tag{2.5}$$

Here, T_{current} is the current time. In what follows, we describe how to calculate T_{est_i}.

First, let us denote T_{ave_i} as the average interval for fixed node i that mobile nodes connect to i. By using this value, we can derive the following formula:

$$T_{\text{est}_i} = N_{\text{req}_i} \cdot T_{ave_i}. \tag{2.6}$$

In order to calculate T_{ave_i}, we define *sensing cycle* of a mobile node engaged in a sensing operation in $\mathbf{T}_i$. The *sensing cycle* is defined as the sequence of a sensing operation in which the mobile node departs from the fixed node in the region including the previous sensing point, moves to the sensing point, performs the sensing operation, and moves to fixed node i. We also define the *average sensing cycle time*, T_{sense}, as the average time elapsed for a sensing cycle of a mobile node that performs a sensing operation. Here, since the moving path of a mobile node to the selected territory can be roughly approximated as the straight line, we assume that mobile nodes go straight to the fixed node of the selected territory (do not go through fixed nodes in the neighboring territories) in order to simplify the calculation. Actually, the effect of the difference of moving to the analysis is small.

As an example, we assume a mobile node in Figure 2.10 which departs form fixed node F and performs a sensing operation in $\mathbf{T}_B$. First, since the distance between fixed nodes F and B is $|\mathbf{L}_F - \mathbf{L}_B|$, the time elapsed for moving to fixed node B, T_{mov_B}, becomes $|\mathbf{L}_F - \mathbf{L}_B|/\nu_m$. Next, the time elapsed for moving from fixed node B to the sensing point is $|\mathbf{L}_B - \mathbf{d}_B|/\nu_m$. Finally, after the sensing operation at a destination (elapsed time is T_{acq}), the time elapsed for moving back to fixed node B becomes $|\mathbf{d}_B - \mathbf{L}_B|/\nu_m$. Thus, the total time elapsed of the sensing cycle becomes:

$$\frac{|\mathbf{L}_F - \mathbf{L}_B|}{\nu_m} + \frac{|\mathbf{L}_B - \mathbf{d}_B|}{\nu_m} + T_{\text{acq}} + \frac{|\mathbf{d}_B - \mathbf{L}_B|}{\nu_m} = \frac{|\mathbf{L}_F - \mathbf{L}_B|}{\nu_m} + T_{\text{acq}} + 2 \cdot \frac{|\mathbf{d}_B - \mathbf{L}_B|}{\nu_m}. \tag{2.7}$$

Here, when the fixed node controls the sensing point in order for the entire of its territory to be monitored uniformly, the average elapsed time for moving

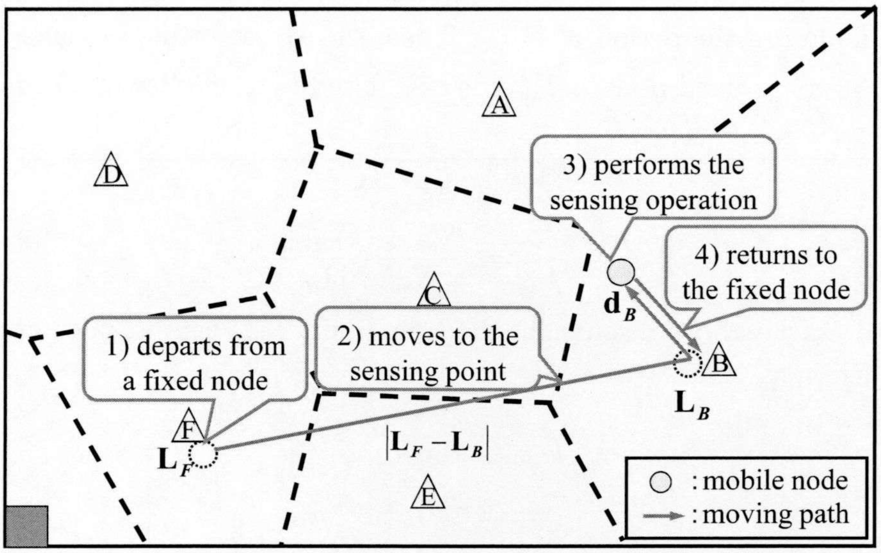


Figure 2.10: The operations in a sensing cycle.

between $\mathbf{L}_B$ and $\mathbf{d}_B$ becomes the average of the moving time form $\mathbf{L}_B$ to any location in $\mathbf{T}_B$. Moreover, since the probability that a mobile node selects a territory to perform sensing depends on the size of the territory, the average time which a mobile node departs from any fixed node (other than B) and moves to fixed node B becomes $\sum_{j\in\mathbf{S}_F, j\neq B}(|\mathbf{T}_j|/S_{\text{region}})\cdot(|\mathbf{L}_j-\mathbf{L}_B|/\nu_m)$.

Therefore, the *average sensing cycle time* in $\mathbf{T}_i$, T_{sense_i}, is derived by the following formula:

$$T_{\text{sense}_i} = \sum_{j\in\mathbf{S}_F, j\neq i}\frac{|\mathbf{T}_j|}{S_{\text{region}}}\cdot(\frac{|\mathbf{L}_i-\mathbf{L}_j|}{\nu_m}) + T_{\text{acq}} + 2\cdot\frac{ave(|\mathbf{d}_i-\mathbf{L}_i|)}{\nu_m}. \quad (2.8)$$

Here, the average sensing cycle time in the whole region, T_{sense}, is derived by the average of the T_{sense_i} for all i-s. Thus,

$$T_{\text{sense}} = \sum_{i\in\mathbf{S_F}}\frac{|\mathbf{T_i}|}{S_{\text{region}}}\cdot T_{\text{sense}_i}. \quad (2.9)$$

Since the mobile node connects to the fixed node after the sensing operation, all mobile nodes connect to a fixed node in the period of T_{sense}. Furthermore, since the number of mobile nodes which choose the destination in $\mathbf{T}_i$ is

$N_{\text{mov}} \cdot |\mathbf{T}_i|/S_{\text{region}}$, we can estimate that $N_{\text{mov}} \cdot |\mathbf{T}_i|/S_{\text{region}}$ mobile nodes connect to fixed node i in the period of T_{sense}. Thus, the average time in which a mobile node connects to fixed node i, T_{ave_i}, is calculated by the following formula:

$$\begin{aligned} T_{ave_i} &= \frac{T_{\text{sense}}}{\frac{N_{\text{mov}} \cdot |\mathbf{T}_i|}{S_{\text{region}}}} \\ &= \frac{T_{\text{sense}} \cdot S_{\text{region}}}{N_{\text{mov}} \cdot |\mathbf{T}_i|}. \end{aligned} \tag{2.10}$$

Based on the above discussion, the time limit, T_{lim_i} can be calculated by the following formula:

$$\begin{aligned} T_{\text{lim}_i} &= T_{\text{current}} + T_{\text{est}_i} \\ &= T_{\text{current}} + N_{\text{req}_i} \cdot T_{ave_i} \\ &= T_{\text{current}} + N_{\text{req}_i} \cdot \frac{T_{\text{sense}} \cdot S_{\text{region}}}{N_{\text{mov}} \cdot |\mathbf{T}_i|}. \end{aligned} \tag{2.11}$$

2.4 Performance Evaluation

In this section, we show results of simulation experiments regarding performance evaluation of DATFM. In the simulation experiments, we compare the performances of DATFM, UM with random way point model (UM-random), UM with controlled mobile node model (UM-controlled) [90], and RAMOS [74, 77].

2.4.1 Simulation Environment

We assume an application of planetary exploration in which each sensor acquires the picture, information of minerals or the temperature in the region. Sensor nodes are deployed in a 2,100[m]×2,100[m] flatland. Each sensor node performs a sensing operation with the rate of 100[bit/sec·m^2] and T_{acq} is 30[sec]. The wireless communication range of all nodes and the channel bandwidth are 100[m] and 11[Mbps], respectively.

In DATFM, there are N_{fix} fixed nodes and N_{mov} mobile nodes randomly deployed in the region. Each mobile node moves with velocity of ν_m[m/s] in *SM* and $2\nu_m$[m/s] in *TM* and *CM*. Each fixed and mobile node has a memory space whose size is respectively 2,000[Mbit] and 10[Mbit]. Each fixed node starts data transmission process when the amount of the accumulated data exceeds Th[Mbit].

Each mobile node performs a sensing operation every time it arrives the sensing point. Each fixed node performs a sensing operation every 1,500[sec]. The rate and the duration of a sensing operation by a fixed node are also 100[bit/sec$\cdot m^2$] and 30[sec].

In RAMOS, there are N_{fix} nodes in *AS* mode, $(N_{\text{mov}} - N_{\text{fix}})$ nodes in *RP* mode, and N_{fix} nodes in *DN* mode. This parameter setting is to make the total number of nodes in *RP* mode and *DN* mode in RAMOS equal to N_{mov} and to guarantee all nodes in *AS* mode can transfer data to the sink node by using nodes in *DN* mode. Nodes in *AS* mode are deployed to the same locations of fixed nodes in DATFM, and have a memory space of 1,000[Mbit]. Nodes in *RP* mode and *DN* mode have a memory space of 10[Mbit]. Nodes in *RP* mode and *DN* mode move with the velocity of ν_m[m/s] when sensing and gathering data, and $2\nu_m$[m/s] when transferring data to the sink node. Nodes in *AS* mode perform a sensing operation every 1,500[sec] in the same way as fixed nodes in DATFM.

In UM-random and UM-controlled, there are $(N_{\text{mov}} + N_{\text{fix}})$ UM nodes. Each UM node has a memory space of 10[Mbit]. Each node moves with velocity of ν_m[m/s] when sensing, and $2\nu_m$[m/s] when transferring data. Each UM node starts transferring data to the sink node when the amount of the accumulated data exceeds 10[Mbit] (i.e. each node transfers data after a sensing operation). In UM-controlled, moving paths of UM nodes are predetermined as shown in Figure 2.11. In this figure, all UM nodes form a line segment (like a train in DATFM) and move on a fixed path. More specifically, after performing a sensing operation, UM nodes move so that the node at the end of the line segment which is closer to the sink node, while keeping the shape of the segment. Then, the accumulated data are transferred to the sink node via the multi-hop communication route. Finally, UM nodes move to the next sensing point.

We set parameters N_{fix}, N_{mov}, ν_m, and Th as shown in Table 2.3. In the experiments, we change one of the parameters and set others as default values in order to verify the effect of each parameter. In this environment, we run 100 simulations each of which consists of 1[week] changing the locations of fixed nodes, and evaluate the averages of the following four criteria:

- Throughput

  The average amount of data that arrive at the sink node per 1[sec].

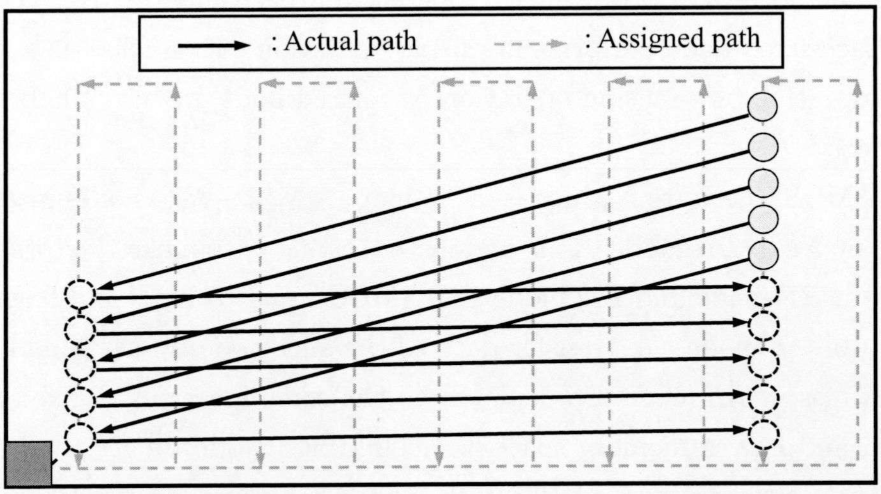


Figure 2.11: Moving path in UM-controlled ($N_{\text{mov}} + N_{\text{fix}} = 5$).

Table 2.3: Parameters in the experiments.

Parameter	Value (Default)
N_{fix}	5∼10 (10)
N_{mov}	15∼50 (40)
ν_m	1.0∼5.0 (5.0)
Th	0∼2,000 (1,000)

- Average moving distance
  The average of moving distances of all nodes during the simulation period.

- Average delay
  The average of the elapsed time after data are acquired (sensed) until the data arrive at the sink node.

- Average control traffic
  The average number of packets exchanged for controlling mobile sensors during the simulation period. Specifically, the control traffic in DATFM is the total number of *RReqs*, *RCReqs*, and *RRels* exchanged during the

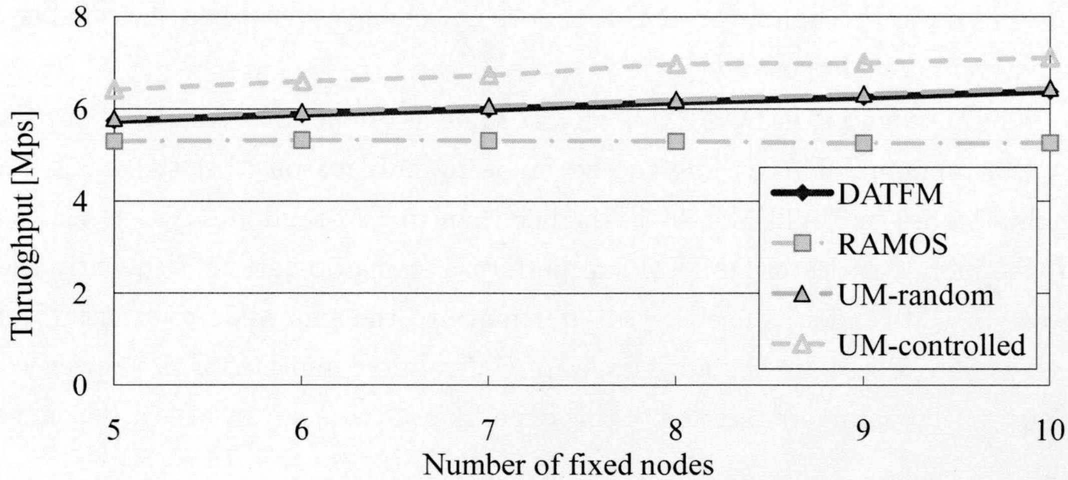


Figure 2.12: N_{fix} and throughput.

simulation period. On the other hand, the control traffic in RAMOS is the total number of packets sent from nodes in *RP* mode to those in *DN* mode. Note that there are no packets for controlling mobile sensors in UM. Thus, we only evaluate control traffics in DATFM and RAMOS.

2.4.2 Effects of the Number of Fixed Nodes

Figures 2.12, 2.13, 2.14, and 2.16 show the simulation results when changing parameter N_{fix}. The horizontal axis on all graphs indicates the number of fixed nodes N_{fix}. The vertical axis indicates throughput in Figure 2.12, average moving distance in Figure 2.13, average delay in Figure 2.14, and average control traffic in Figure 2.16.

Throughput

Figure 2.12 shows that the throughput in DATFM is always larger than that in RAMOS. This shows that DATFM can acquire and transfer data more efficiently than RAMOS. As the number of fixed nodes gets larger, the difference in throughput between DATFM and RAMOS slightly increases because it becomes easier for mobile nodes to transfer data to the nearby fixed nodes in DATFM. On the other hand, in RAMOS, since the mobile node in *AS* mode does not move to transfer data, throughput in RAMOS does not change even when the number of fixed nodes gets larger.

The throughput in DATFM is almost same as (sometimes smaller than) that in UM-random. This is due to the following two reasons: a) the number of mobile nodes in DATFM (40) is smaller than in UM-random (N_{fix} + 40), and b) the mobile nodes in DATFM can perform sensing operations frequently than those in UM-random since they do not move to the sink node to transfer data every time. These are the advantages of UM-random and DATFM, respectively. Thus, in the parameter setting in this experiment, these advantages are balanced out.

The throughput in UM-controlled is always larger than those in other methods. This is because the number of UM nodes (N_{fix} + 40) is larger than that of mobile nodes in DATFM. Moreover, compared with UM-random, mobile sensors in UM-controlled can perform sensing operations more frequently since the moving distance for transferring data can be reduced by using a multi-hop communication route.

Average moving distance

Figure 2.13 shows that the moving distances in RAMOS, UM-random, and UM-controlled are larger than that in DATFM. This is because sensor nodes in those methods move to the sink node every time they acquire data. On the other hand, since DATFM accumulates the acquired data on a fixed node before transmitting them to the sink node, each mobile node that acquires the data does not need to move to the sink node.

The moving distance in RAMOS increases as the number of fixed nodes gets larger. This is because, the nodes in *DN* mode have to transfer data received from nodes in *AS* mode more frequently.

Average delay

Figure 2.14 shows that the delay in DATFM is always larger than those in RAMOS, UM-random, and UM-controlled. This is because DATFM needs much

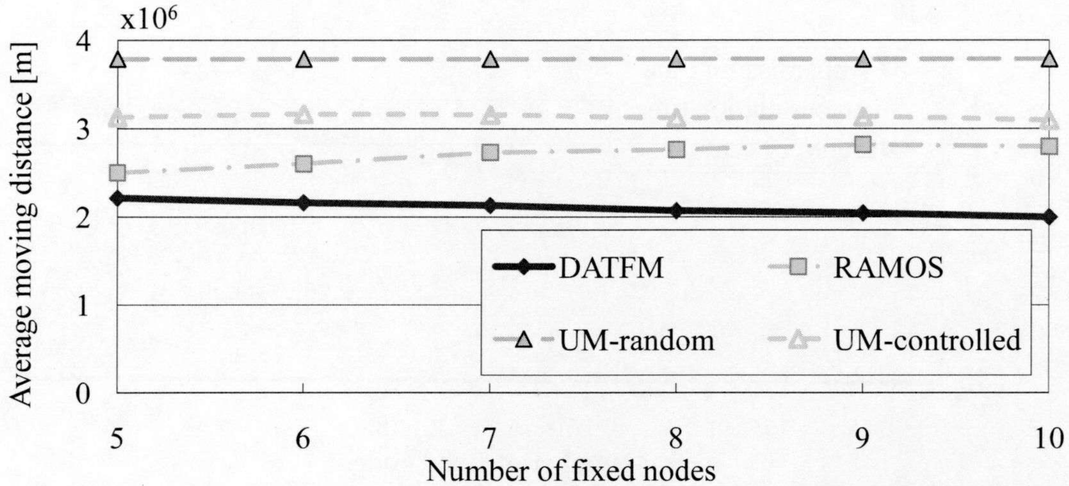


Figure 2.13: N_{fix} and average moving distance.

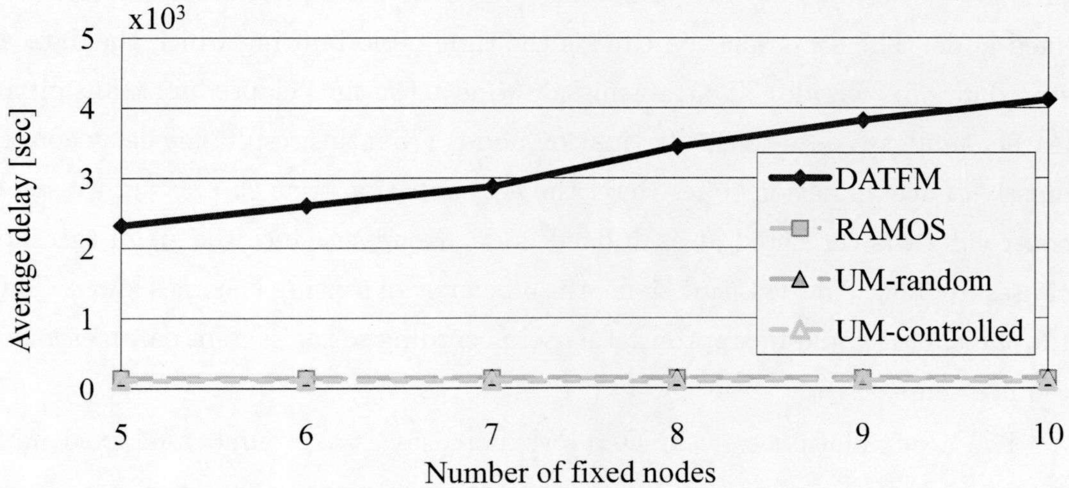


Figure 2.14: N_{fix} and average delay.

time for accumulating data on fixed nodes.

On the other hand, the average delays in other methods do not change even when the number of mobile nodes increases. This is because the mobile nodes in these methods transfer data without accumulation.

Figure 2.15 shows the detail of the delay in DATFM. In this figure, the delivery

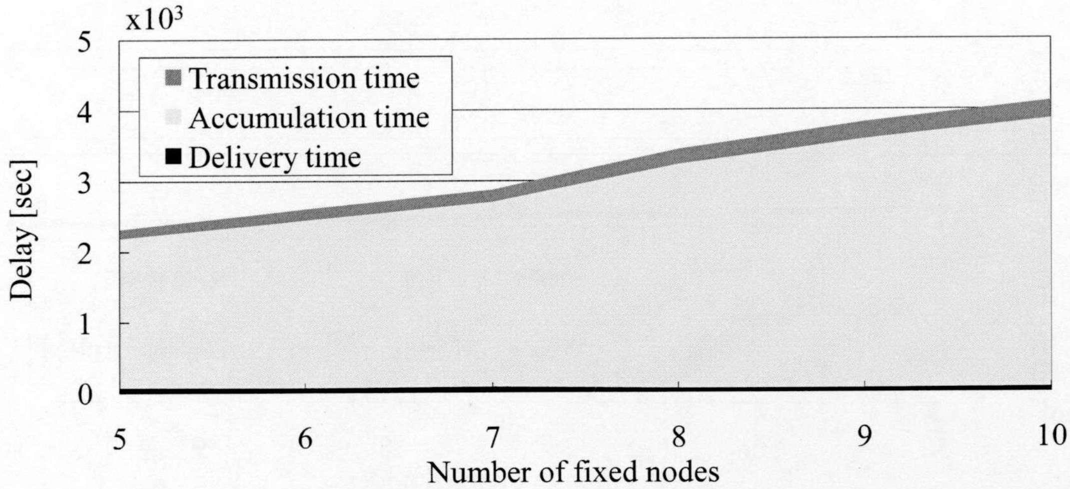


Figure 2.15: N_{fix} and details of delay in DATFM.

time is the time period after a mobile node acquires data until it connects to a fixed node. The accumulation time is the time period during which the data are stored at a fixed node. The transmission time is the time elapsed for transmitting the accumulated data to the destination node. From this result, the delay consists mostly of accumulation time. Here, the accumulation time can be suppressed by controlling the threshold at each fixed node. However, since the small threshold causes frequent data transmissions, the efficiency of sensing becomes lower. Thus, the threshold should be appropriately set according to the system parameters and requirements.

The accumulation time in DATFM increases as the number of fixed nodes gets larger. This is because the number of mobile nodes that perform sensing operations in its territory decreases due to the decrease of the territory size. In this case, it takes much time until the amount of accumulated data reaches the threshold.

Average control traffic

Figure 2.16 shows that the average control traffic in DATFM is smaller than that in RAMOS. In RAMOS, each node in *RP* mode sends control packets every time

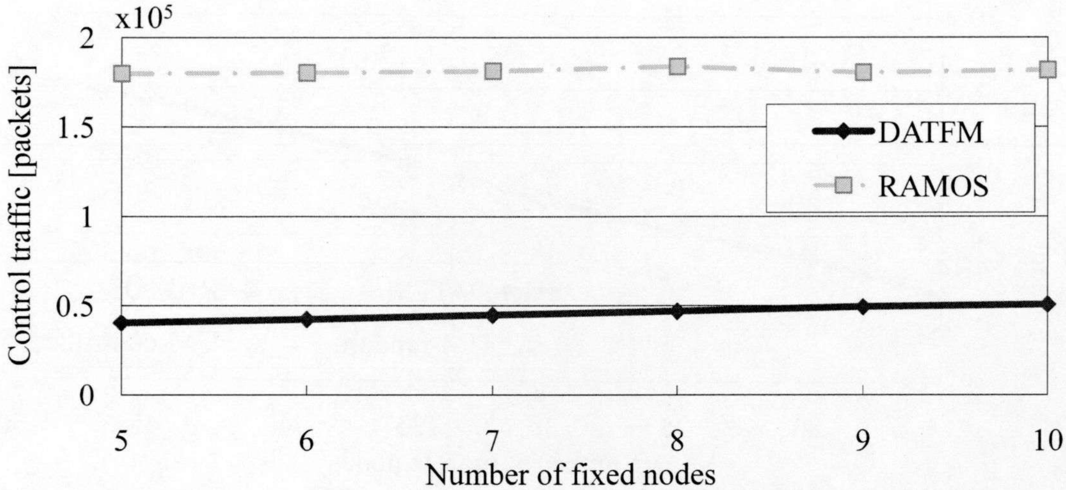


Figure 2.16: N_{fix} and average control traffic.

it connects to nodes in *DN* mode which hold data. On the other hand, in DATFM, the frequency of data transmission processes becomes small by accumulating data at a fixed node. Thus, the number of *RReqs*, *RCReqs*, and *RRels*, that are sent only in data transmission processes, becomes small.

2.4.3 Effects of the Number of Mobile Nodes

Figures 2.17, 2.18, 2.19, and 2.21 show the simulation results when changing parameter N_{mov}. The horizontal axis on all graphs indicates the number of mobile nodes N_{mov}. The vertical axis indicates throughput in Figure 2.17, average moving distance in Figure 2.18, average delay in Figure 2.19, and average control traffic in Figure 2.21.

Throughput

Figure 2.17 shows that the throughput in DATFM is always larger than that in RAMOS. Moreover, the throughput in DATFM is always smaller than that in UM-controlled. These are due to the same reason as that in Figure 2.12.

Compared with UM-random, the throughput in DATFM is smaller when the number of mobile nodes is small. This is because it becomes impossibly difficult

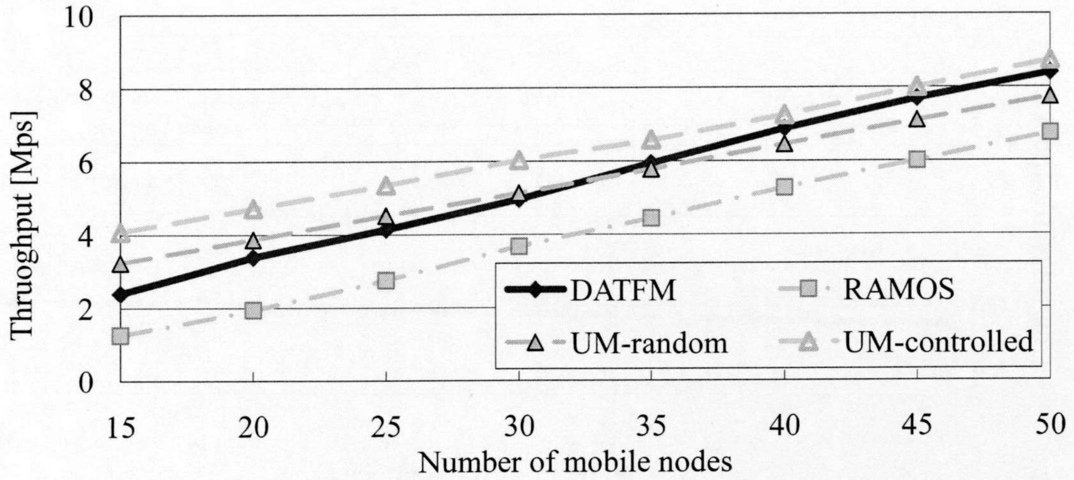


Figure 2.17: N_{mov} and throughput.

for some fixed nodes to construct a communication route. When a fixed node cannot collect the sufficient number of mobile nodes, it transmits the data by the train transmission with a small number of mobile nodes. Thus, these mobile nodes are kept for transferring data and cannot return to perform sensing operations for a long time. On the other hand, mobile sensors in UM-random can perform sensing operations more frequently than DATFM since such an inefficient situation does not occur.

On the other hand, when the number of mobile node is large, the throughput in DATFM is larger than that in UM-random. This is because the number of train transmissions with a small number of nodes decreases since it becomes easier for each fixed node to collect the sufficient number of mobile nodes to construct a communication route.

Note that, even when the number of mobile nodes equals to 50, the network is still sparse in which the connectivity between nodes is very low.

Average moving distance

Figure 2.18 shows that the moving distances in RAMOS, UM-random, and UM-controlled are larger than that in DATFM. This is due to the same reason as that

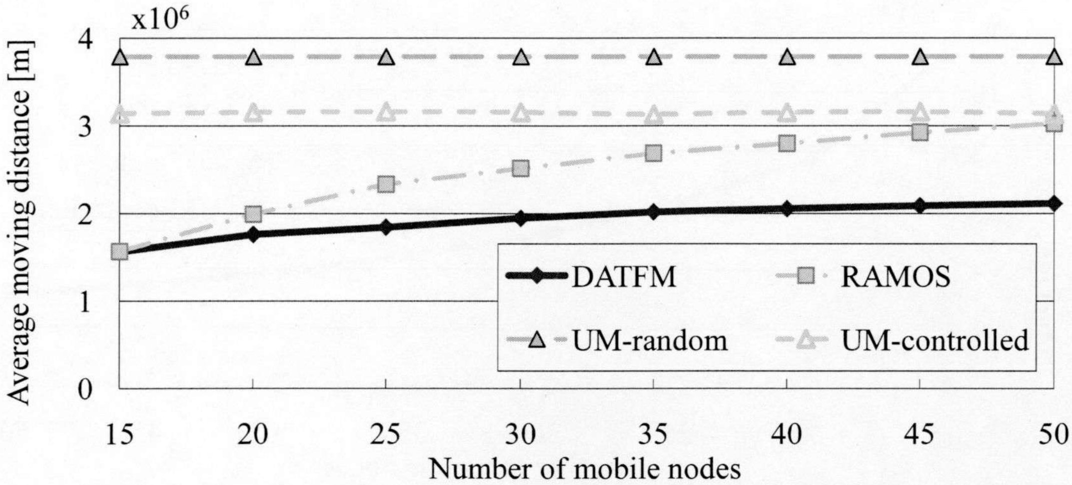


Figure 2.18: N_{mov} and average moving distance.

in Figure 2.13.

The moving distance in DATFM increases as the number of mobile nodes gets larger. This is because it becomes easier for each fixed node to construct a communication route according to the increase of mobile nodes. In such an environment, the opportunities to transfer data increase.

On the other hand, the moving distance in RAMOS increases as the number of sensor nodes gets larger. This is because the frequency that nodes in *DN* mode connect to nodes in *RP* mode increases. This makes moving distances of nodes in *DN* mode large because they have to transfer the data received from nodes in *RP* mode.

Average delay

Figure 2.19 shows that the delay in DATFM is always larger than those in RAMOS, UM-random, and UM-controlled. This is due to the same reason as that in Figure 2.14.

Figure 2.20 shows the detail of the delay in DATFM. From this result, the accumulation time decreases as the number of mobile nodes gets larger. This is because the fixed nodes have many opportunities to receive data from mo-

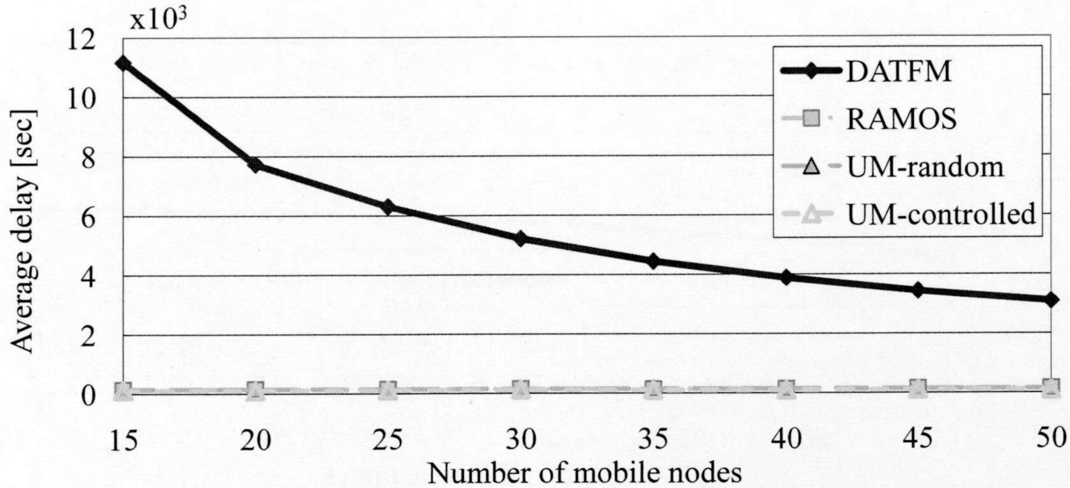


Figure 2.19: N_{mov} and average delay.

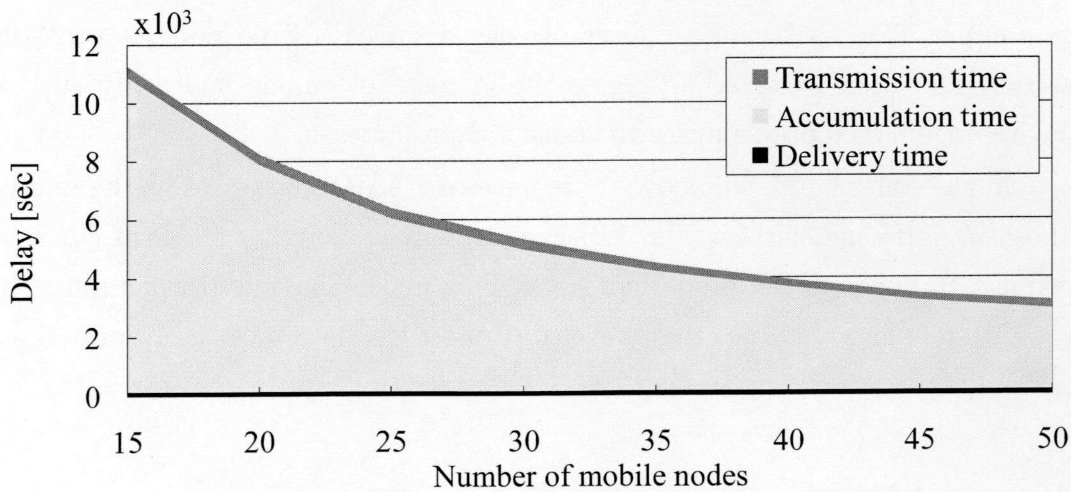


Figure 2.20: N_{mov} and details of delay in DATFM.

bile nodes. In this case, the amount of accumulated data quickly reaches the threshold.

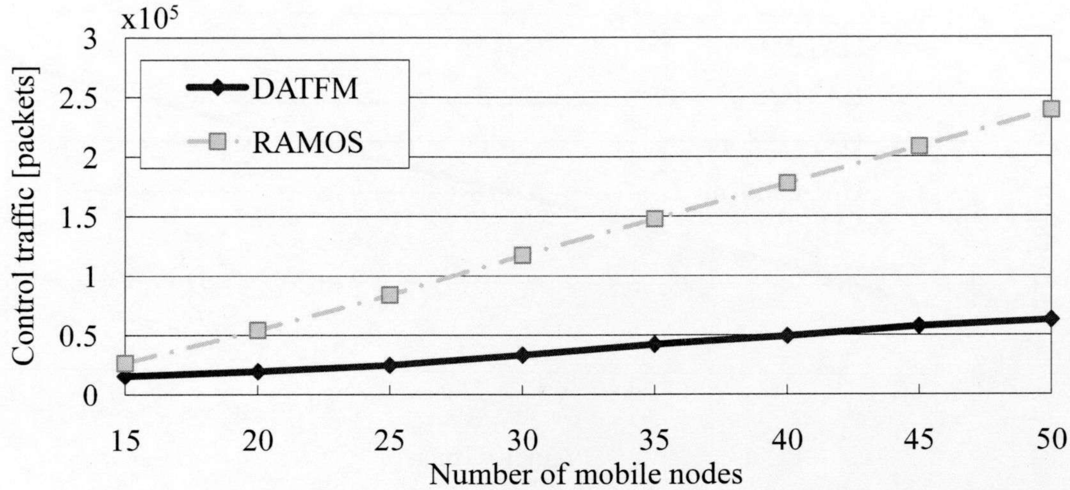


Figure 2.21: N_{mov} and average control traffic.

Average control traffic

Figure 2.21 shows that the average control traffic in DATFM is smaller than that in RAMOS. This is due to the same reason as that in Figure 2.16.

The average control traffic in DATFM increases as the number of mobile nodes increases. When the number of mobile nodes is large, each fixed node has many opportunities to receive data from mobile nodes. This increases the frequency of data transmission processes, and thus, many *RReqs*, *RCReqs*, and *RRels* are sent.

2.4.4 Effects of the Velocity of Node

Figures 2.22, 2.23, 2.24, and 2.26. show the simulation results when changing parameter ν_m. The horizontal axis on all graphs indicates the velocity (movement speed) of sensor nodes ν_m. The vertical axis indicates throughput in Figure 2.22, average moving distance in Figure 2.23, average delay in Figure 2.24, and average control traffic in Figure 2.26.

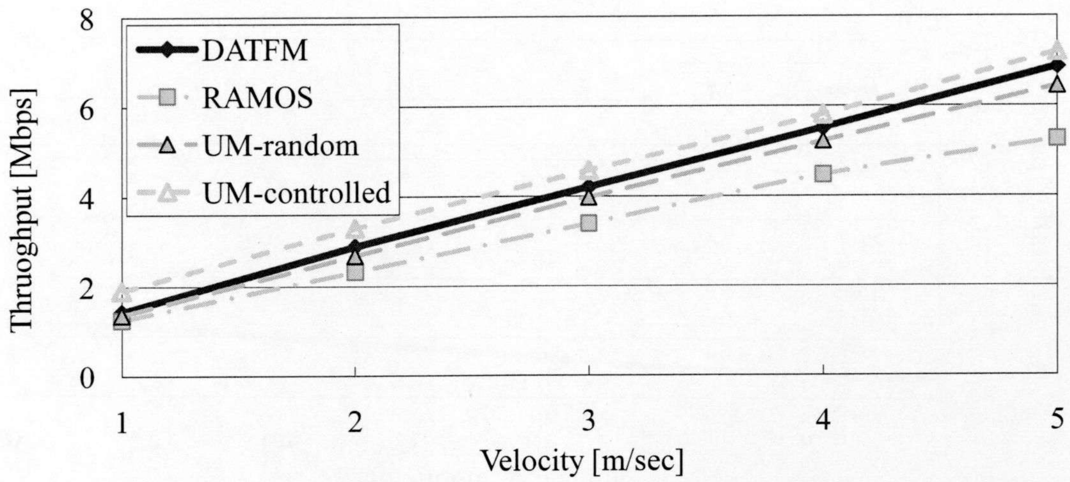


Figure 2.22: Velocity and throughput.

Throughput

Figure 2.22 shows that the throughput in DATFM is always larger than those in RAMOS and UM-random. This shows that DATFM can acquire and transfer data more efficiently than RAMOS and UM-random. On the other hand, the throughput in DATFM is always smaller than that in UM-controlled. However, as ν_m increases, the difference of throughput in DATFM and UM-controlled gets smaller. Mobile nodes in DATFM transfer the acquired data to a nearby fixed node while UM nodes in UM-controlled transfer data to the sink node every time they acquired data. Therefore, the frequency that mobile nodes in DATFM perform sensing operations gets larger when the mobile nodes move with higher speed.

Average moving distance

Figure 2.23 shows that the moving distances in all methods increase as ν_m gets larger, which is an obvious result.

In addition, the moving distance in DATFM is smaller than those in RAMOS, UM-random, and UM-controlled. This is due to the same reasons as those in Figures 2.13 and 2.18.

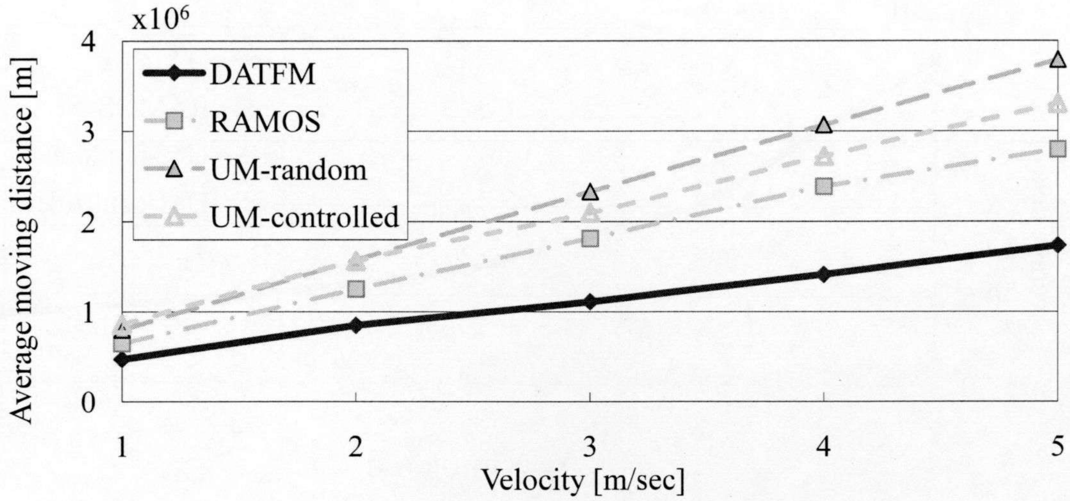


Figure 2.23: Velocity and average moving distance.

Furthermore, the differences in moving distance between four methods increase as ν_m gets larger. This is because mobile nodes in all methods can perform more sensing operations as ν_m gets larger, and thus, the numbers of movements between the sensing point of each sensor node and the sink node in RAMOS, UM-random, and UM-controlled become larger.

Average delay

Figure 2.24 shows that the delay for transmitting data in DATFM is always larger than those in RAMOS, UM-random, and UM-controlled. This is due to the same reasons as those in Figures 2.14 and 2.19.

As ν_m increases, average delays of all methods decrease. This is obvious since each sensor node can transfer data faster.

Figure 2.25 shows the detail of delay in DATFM. When ν_m is very small, the accumulation time is very large. This is because mobile nodes spend long time to transfer the acquired data to the fixed node. This makes the rate of data accumulation low. Moreover, the delivery time decreases as ν_m increases, which is an obvious result.

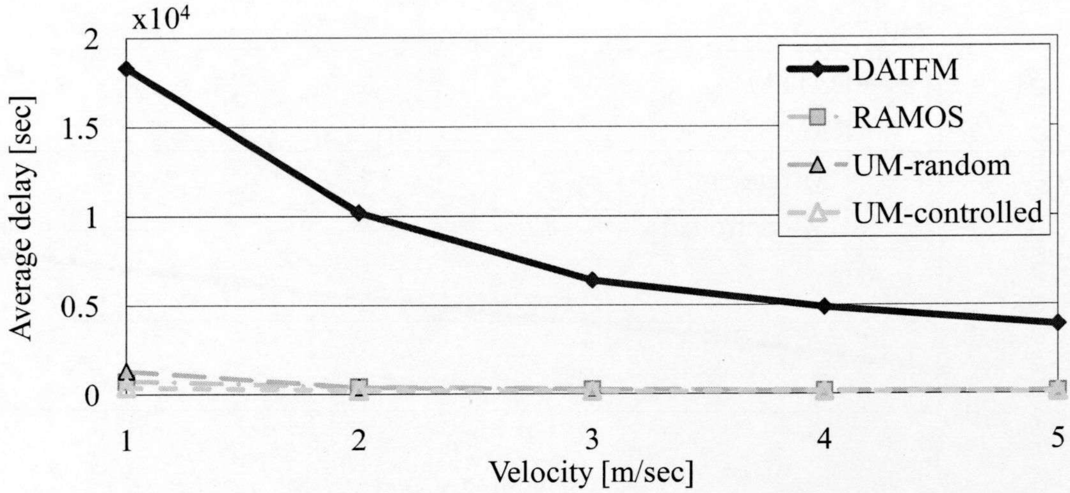


Figure 2.24: Velocity and average delay.

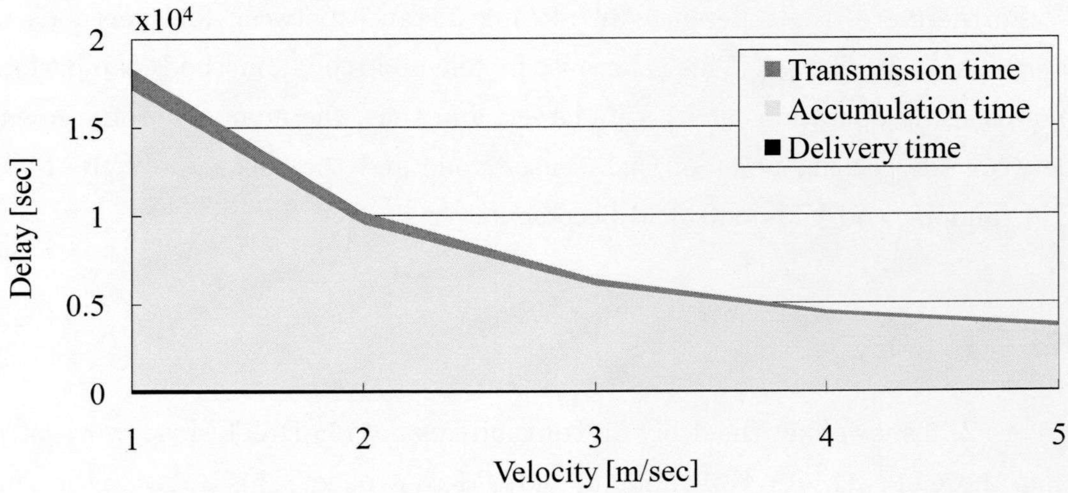


Figure 2.25: Velocity and details of delay in DATFM.

Average control traffic

Figure 2.26 shows that the average control traffic in DATFM is smaller than that in RAMOS. This is due to the same reason as those in Figures 2.16 and 2.21.

The average control traffic in DATFM increases as ν_m increases. This is

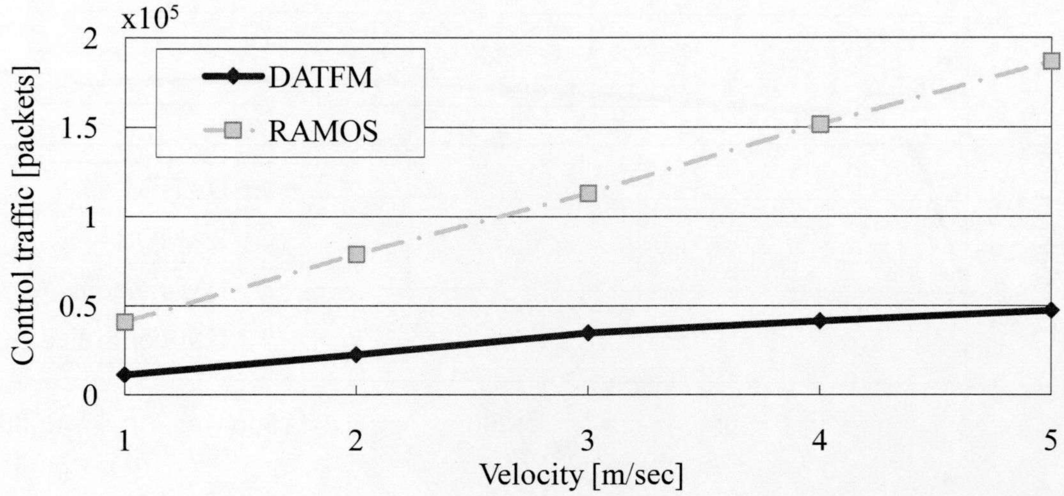


Figure 2.26: Velocity and average control traffic.

because, when a mobile node can move fast, it can perform sensing operation and transfer data to a nearby fixed node more frequently. This increases the frequency of data transmission processes.

2.4.5 Effects of the Threshold for Starting Data Transmission

Figures 2.27, 2.28, 2.29, and 2.31 show the simulation results when changing parameter Th. The horizontal axis on all graphs indicates the threshold for starting data transmission in DATFM, Th. The vertical axis indicates throughput in Figure 2.27, average moving distance in Figure 2.28, average delay in Figure 2.29, and average control traffic in Figure 2.31.

Throughput

Figure 2.27 shows that the throughput in DATFM is smaller than those in RAMOS and UM-random when Th is very small. This is because fixed nodes frequently start data transmission processes. In this case, many mobile nodes are used for transmitting the accumulated data. This decreases the frequency of sensing operations performed by mobile nodes.

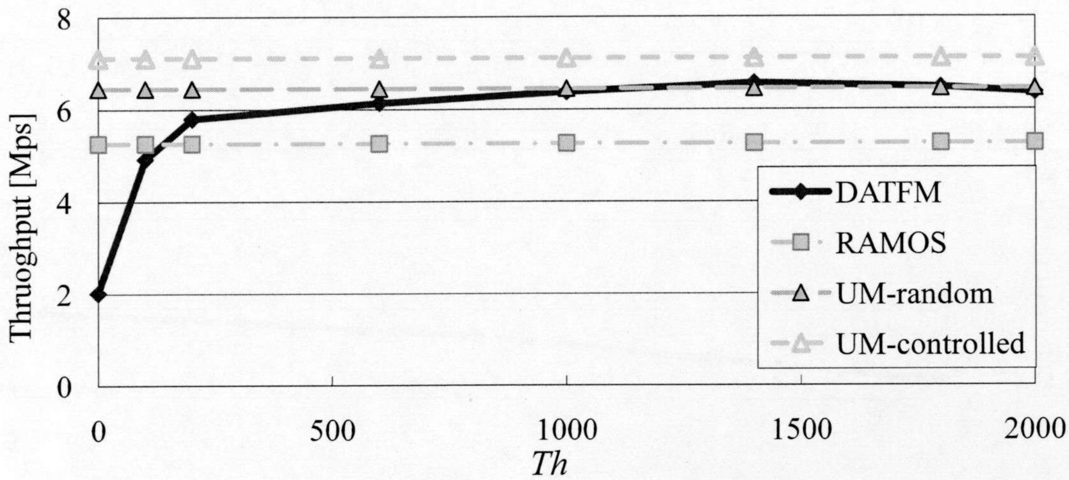


Figure 2.27: *Th* and throughput.

As *Th* increases, the throughput in DATFM increases and becomes larger than that in RAMOS. This is because the frequency of data transmission processes decreases. However, the increase in the throughput in DATFM is saturated when *Th* is larger than about 200[Mbit]. This is because the accumulated data are discarded before being transferred to the sink node due to the limited amount of memory space in each fixed node.

Average moving distance

Figure 2.28 shows that the moving distance in DATFM is smaller than those in RAMOS, UM-random, and UM-controlled. This is due to the same reasons as those in Figures 2.13, 2.18, and 2.23.

When *Th* is very small, the moving distance in DATFM decreases. This is due to the decrease in the frequency of sensing operations performed by mobile nodes. Here, when fixed nodes frequently start data transmission processes, it is expected that the moving distance of mobile nodes for transferring accumulated data increases. However, since *Th* (i.e. the accumulated data in each fixed node) is very small, each data transmission process is finished only with a small moving distance. For example, when *Th* equals to the memory space of a mobile node,

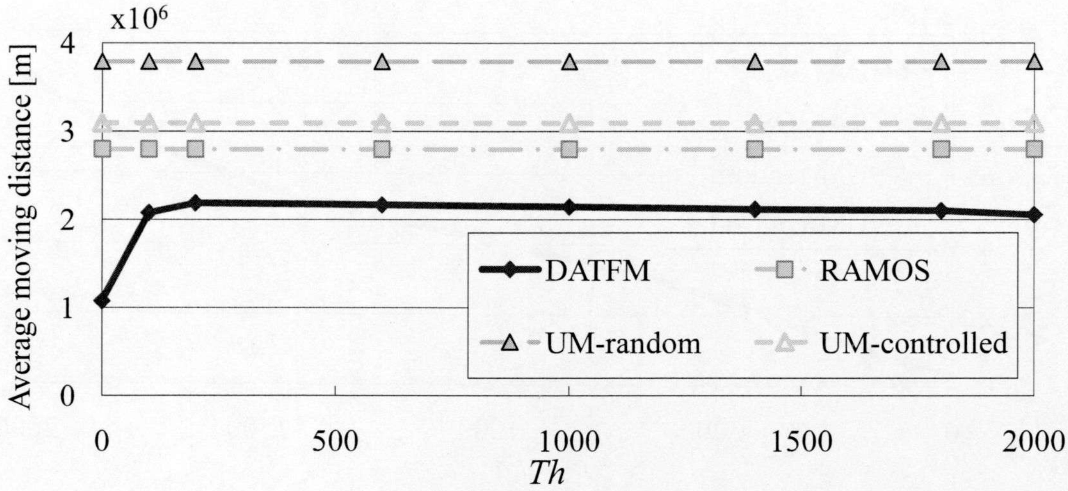


Figure 2.28: Th and average moving distance.

a mobile node needs to move from the source node to the destination node only once. These cause the decrease of the total moving distance.

Average delay

Figure 2.29 shows that the delay for transmitting data in DATFM is always larger than those in RAMOS, UM-random, and UM-controlled. This is due to the same reasons as those in Figures 2.14, 2.19, and 2.24.

As Th increases, the average delay in DATFM increases. As shown in Figure 2.30, this is due to the increase in the accumulation time, which is an obvious result.

Average control traffic

Figure 2.31 shows that the average control traffic in DATFM is basically smaller than that in RAMOS. This is due to the same reasons as those in Figures 2.16, 2.21, and 2.26. However, when Th is very small, the control traffic in DATFM becomes larger than that in RAMOS. This is due to the increase in the frequency of data transmission processes.

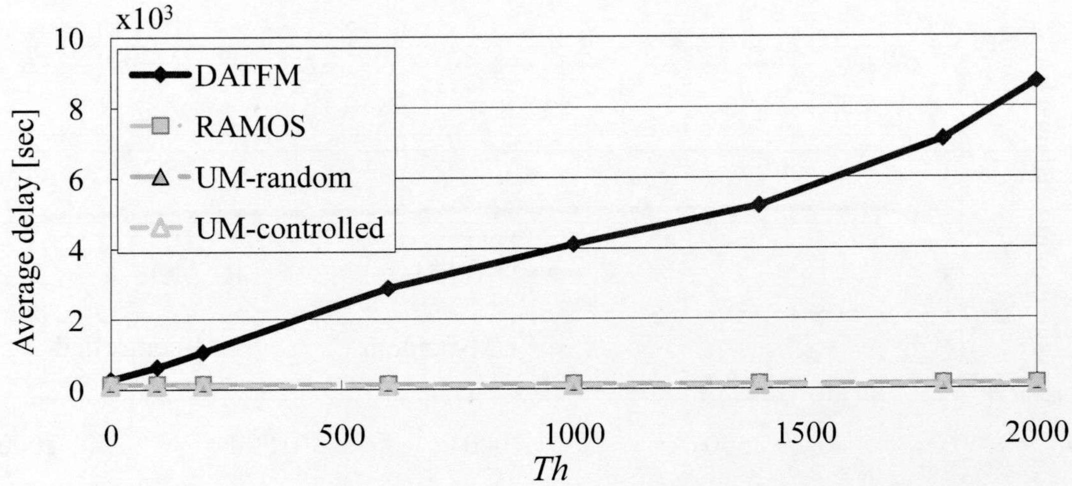


Figure 2.29: Th and average delay.

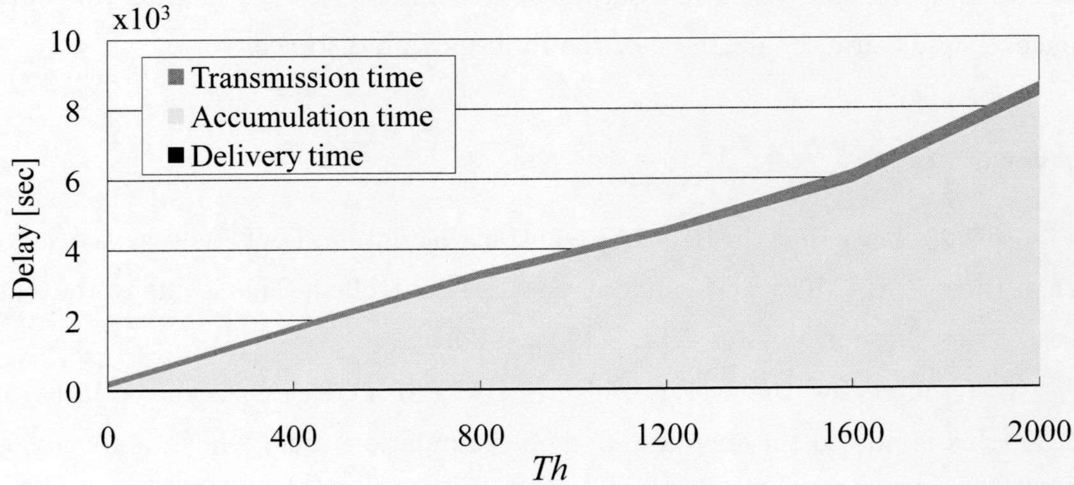


Figure 2.30: Th and details of delay in DATFM.

2.4.6 Summary

From the above results, we can see that DATFM basically achieves high throughput in sparse environments. However, throughput in DATFM is always smaller than that in UM-controlled where the movement of mobile sensors is fully-controlled.

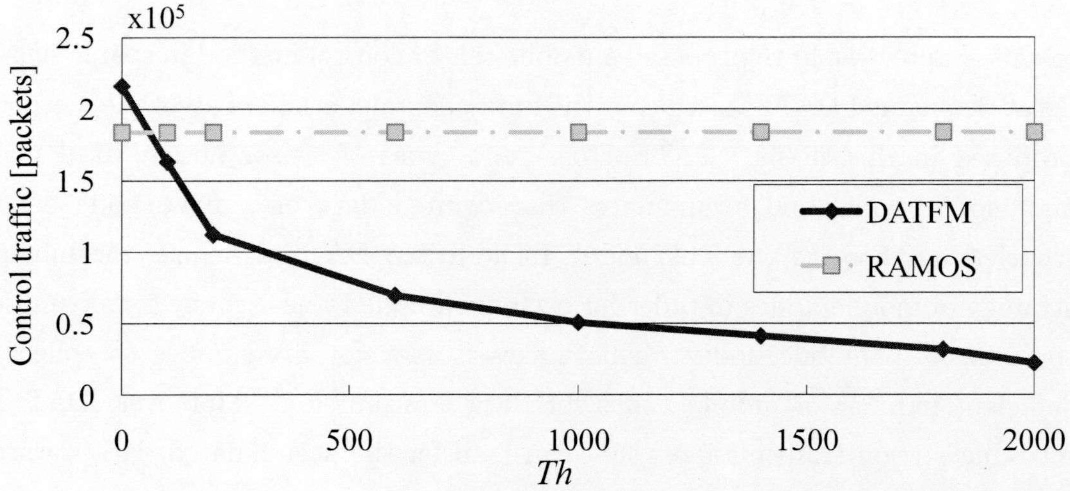


Figure 2.31: Th and average control traffic.

Moreover, as shown in Figure 2.17, throughput in DATFM becomes smaller than that in UM-random when the number of mobile nodes is very small. As described in Section 2.4.2, this is due to the difference of the number of mobile nodes (mobile nodes in DATFM and UM nodes in UM-controlled and UM-random). In addition, DATFM is just the fundamental method that introduces the ideas described in Section 1.5.1. Note that throughputs in our extended methods described in Chapters 3 and 4 are drastically improved and become larger than those in UM-controlled and UM-random.

However, the moving distance in DATFM is much smaller than those in RAMOS, UM-random, and UM-controlled, since mobile nodes in RAMOS, UM-random, and UM-controlled have to move for longer distance in order to transfer the acquired data to the sink node. Here, in mobile sensor networks, the energy consumed by movement is much larger than those by communication and computation [58]. Therefore, it is meaningful that DATFM achieves higher energy efficiency than those in RAMOS, UM-random, and UM-controlled.

2.5 Conclusion

In this chapter, we have proposed a mobile sensor control method in sparse sensor networks, named DATFM, which is the fundamental method of all of the methods proposed in this thesis. DATFM uses two types of sensor nodes, fixed node and mobile node, and accumulates the acquired data on a fixed node before transferring them to the sink node. In addition, DATFM defines the moving strategy of mobile nodes in order for each fixed node to have many opportunities to connect to mobile nodes. This makes it easy for fixed nodes to collect a sufficient number of mobile nodes for data transmission. Moreover, DATFM introduces train transmissions, that can transfer the accumulated data towards the sink node with a small number of mobile nodes.

We have also conducted simulation experiments to evaluate the performance of DATFM. The results showed that DATFM can reduce the moving distance compared with conventional methods while keeping high throughput.

Chapter 3

A Node Deployment Strategy for Efficient Data Collection in Sparse Sensor Networks

3.1 Introduction

In Chapter 2, we proposed DATFM which is an effective mobile sensor control method for sparse sensor networks. DATFM achieves efficient sensing and data transfer by introducing fixed nodes which accumulate data acquired by nodes and manage mobile nodes for transferring data to the sink node.

In DATFM, fixed nodes can be deployed to arbitrary locations. Here, when there are some points which should be carefully monitored, it is effective to deploy fixed nodes to such important points. By doing so, these important points can be continuously monitored by the fixed nodes. On the other hand, in an application where fixed nodes can be freely deployed in the region, further improvement of efficiencies of sensing and data transfer is expected by strategically determining the locations of fixed nodes. This idea comes from the following two features of DATFM:

- Since each fixed node accumulates data generated in its territory, the sizes of territories affect the performance of sensing.
- Since each fixed node transfers the accumulated data by using multiple

mobile nodes, the distance of the route (between source and destination nodes) affects the performance of data transmission.

Based on the above features, in this chapter, we propose an extended method of DATFM, named DATFM/DF (DATFM with deliberate Deployment of Fixed nodes), in order to further improve the efficiencies of sensing and data transfer. This method focuses on the deployment locations of fixed nodes in the region. First, we analyze the effects of the deployment of fixed nodes on the performance. Based on the result of the analysis, we propose a deliberate deployment strategy to improve the performance.

The remainder of this chapter is organized as follows. In Section 3.2, we present DATFM/DF, which is proposed in this chapter. The results of simulation experiments are presented in Section 3.3. The discussions of proposed method DATFM/DF are described in Section 3.4. Finally, we conclude this chapter in Section 3.5.

3.2 DATFM/DF

In this section, we analyze the effects of the locations of fixed nodes, and propose a deliberate deployment strategy to further improve the performance of DATFM.

3.2.1 Analysis of Effects of the Deployment of Fixed Nodes

In this analysis, we discuss the *sensing rate*, R_{sense}, which is defined as the number of sensing operations per unit time in the whole region. The sensing rate depends on the time for each mobile node to move to its sensing point.

Assumptions

Before starting the analysis, we show the assumptions in this chapter. We do not consider the notions of collection of mobile nodes by using mobile nodes in *CM* described in Section 2.3.2 for simplicity. In addition, as in the analysis in Section 2.3.2, we assume that mobile nodes do not go through fixed nodes in the adjacent territories.

Table 3.1: Given parameters in the analysis.

Parameter	Description
ν_r	velocity of mobile nodes in *TM*.
N_{fix}	number of fixed nodes in the whole region.
D	data size acquired in a sensing operation.
r_{com}	communication range of all nodes.

We assume that the parameters in Table 3.1 are given in addition to those in Table 2.2. Moreover, we define the *transfer cycle*, which is the sequence of a data transmission. In a transfer cycle, a fixed node accumulates data and transmits the accumulated data to the destination node. We also define the *average transfer cycle time*, T_{transfer}, as the average time elapsed for a transfer cycle.

Analysis of R_{sense}

From the definition of the sensing cycle, R_{sense} is calculated as the inverse of T_{sense}. As described in Section 2.3.2, the average sensing cycle time of a mobile node that performs sensing operation in $\mathbf{T}_i$, T_{sense_i}, is derived by the following formula:

$$\begin{aligned} T_{\text{sense}_i} &= \sum_{j \in \mathbf{S}_F, j \neq i} \frac{|\mathbf{T}_j|}{S_{\text{region}}} \cdot \left(\frac{|\mathbf{L}_i - \mathbf{L}_j|}{\nu_m}\right) \\ &\quad + T_{\text{acq}} + 2 \cdot \frac{ave(|\mathbf{d}_i - \mathbf{L}_i|)}{\nu_m}. \end{aligned} \tag{3.1}$$

Moreover, the average sensing cycle time in the whole region, T_{sense}, is derived by the following formula:

$$T_{\text{sense}} = \sum_{i \in \mathbf{S}_F} \frac{|\mathbf{T}_i|}{S_{\text{region}}} \cdot T_{\text{sense}_i}. \tag{3.2}$$

After fixed node i accumulates data, it starts a data transmission. During the data transmission, the mobile nodes which construct the communication route cannot perform the sensing operation. Thus, we should consider the frequency of data transmissions, R_{route}, and the number of mobile nodes used for data transmission.

Let us define the average number of free nodes, N_{free}, that are not used for data transmission in a unit time. Then, R_{sense} is represented by using the ratio of free nodes to all mobile nodes per one average sensing cycle time, that is,

$$R_{\text{sense}} = \frac{N_{\text{free}}}{N_{\text{mov}}} \cdot \frac{1}{T_{\text{sense}}}. \tag{3.3}$$

N_{free} can be represented by using the required number of mobile nodes to construct a communication route, N_{req}, and the frequency of data transmission, R_{route}, that is,

$$N_{\text{free}} = N_{\text{mov}} - \sum_{i \in \mathbf{S}_F} (R_{\text{route}_i} \cdot N_{\text{req}_i}). \tag{3.4}$$

Therefore, R_{sense} is represented by the following formula:

$$R_{\text{sense}} = \frac{1}{T_{\text{sense}}} \cdot (1 - \frac{\sum_{i \in \mathbf{S}_F} (R_{\text{route}_i} \cdot N_{\text{req}_i})}{N_{\text{mov}}}). \tag{3.5}$$

In the above formulae, N_{req_i} is represented by $|\mathbf{L}_i - \mathbf{L}_d|/r_{\text{com}}$ where $\mathbf{L}_d$ is the location of the destination node of the data transmission.

The frequency of route constructions in fixed node i, R_{route_i}, is the ratio of times elapsed for data transmission, T_{transmit_i}, to the average transfer cycle time, T_{transfer_i}, that is, $T_{\text{transmit}_i}/T_{\text{transfer}_i}$. T_{transfer_i} is the sum of times elapsed for accumulating data, T_{acc_i}, and data transmission. T_{acc_i} is derived by using the required number of times that fixed node i connects to mobile nodes which holds data, and the average time for each mobile node to connect to fixed node i, T_{ave_i}. The former is represented as Th/D. The latter is represented as the product of average sensing cycle time and the inverse of the number of mobile nodes that exist in this territory. Here, mobile nodes in DATFM selects a territory to perform sensing according to the probability which is proportional to the size of each territory. Thus, it is expected that there are $N_{\text{mov}} \cdot |\mathbf{T}_i|/S_{\text{region}}$ mobile nodes in $\mathbf{T}_i$. Therefore, T_{acc_i} is represented by the following formula:

$$T_{\text{acc}_i} = \frac{Th}{D} \cdot T_{\text{ave}_i} = \frac{Th}{D} \cdot \frac{S_{\text{region}} \cdot T_{\text{sense}_i}}{N_{\text{mov}} \cdot |\mathbf{T}_i|}. \tag{3.6}$$

T_{transmit_i} is the total time from when fixed node i starts data transmission until the accumulated data are transferred to the destination. Here, we define the *round* as a sequence of operations in a data transmission, that is, the train departs from the source node, moves to the destination, transmits data, and returns to the source node. We assume that the first round is conducted by the train that contains one mobile node. The elapsed time for this round, $T_{\mathrm{train}_{i_1}}$, becomes $2 \cdot (|\mathbf{L}_i - \mathbf{L}_j| - 2 \cdot r_{\mathrm{com}})/\nu_r$. The number of mobile nodes which newly connect to fixed node i during this round, $N_{\mathrm{train}_{i_1}}$, is $T_{\mathrm{train}_{i_1}}/T_{\mathrm{ave}_i}$. Thus, in the second round, the train transmission is conducted by $N_{\mathrm{train}_{i_1}} + 1$ mobile nodes. The elapsed time for this round is represented by the following formula:

$$\begin{aligned} T_{\mathrm{train}_{i_2}} &= \frac{2 \cdot (|\mathbf{L}_i - \mathbf{L}_j| - r_{\mathrm{com}} \cdot (2 + \frac{T_{\mathrm{train}_{i_1}}}{T_{\mathrm{ave}_i}})}{\nu_r} \\ &= A \cdot T_{\mathrm{train}_{i_1}}. \qquad (A = 1 - \frac{2 \cdot N_{\mathrm{mov}} \cdot |\mathbf{T}_i| \cdot r_{\mathrm{com}}}{\nu_r \cdot S_{\mathrm{region}} \cdot T_{\mathrm{sense}_i}}) \end{aligned} \tag{3.7}$$

In the same way, the elapsed time for N-th round, $T_{\mathrm{train}_{i_N}}$, is represented by the following formula:

$$\begin{aligned} T_{\mathrm{train}_{i_N}} &= \frac{2 \cdot (|\mathbf{L}_i - \mathbf{L}_j| - r_{\mathrm{com}} \cdot (2 + N_{\mathrm{train}_{N-1}})}{\nu_r} \\ &= A^{N-1} \cdot T_{\mathrm{train}_{i_1}}. \end{aligned} \tag{3.8}$$

Therefore, when a data transmission is conducted by N rounds, T_{transmit_i} is derived by the following formula:

$$\begin{aligned} T_{\mathrm{transmit}_i} &= T_{\mathrm{train}_{i_1}} + T_{\mathrm{train}_{i_2}} + \cdots + T_{\mathrm{train}_{i_N}} \\ &= T_{\mathrm{train}_{i_1}} + A \cdot T_{\mathrm{train}_{i_1}} + \cdots + A^{N-1} \cdot T_{\mathrm{train}_{i_1}} \\ &= \frac{(1 - A^{N-1}) \cdot T_{\mathrm{train}_{i_1}}}{(1 - A)}. \end{aligned} \tag{3.9}$$

Here, since the required time for transferring data between connected nodes is much smaller than that for moving between fixed nodes, we neglect the time elapsed for transferring data after constructed the complete communication route. Thus, we suppose T_{transmit_i} is equal to the time after starting the first round until the communication route is constructed. Here, as shown in Formula (3.7), A is smaller than 1. In addition, since we assume a sparse environment, N tends to be large. Thus, we assume that $A^{N-1} \approx 0$ and T_{transmit_i} can be represented by the following formula:

$$T_{\mathrm{transmit}_i} \approx \frac{T_{\mathrm{train}_{i_1}}}{1 - A} = \frac{T_{\mathrm{train}_{i_1}} \cdot \nu_r \cdot T_{\mathrm{sense}_i} \cdot S_{\mathrm{region}}}{2 \cdot r_{\mathrm{com}} \cdot N_{\mathrm{mov}} \cdot |\mathbf{T}_i|}. \tag{3.10}$$

From the above discussions, R_{route_i} is represented by the following formula:

$$R_{\text{route}_i} = \frac{\frac{T_{\text{train}_{i_1}} \cdot \nu_r \cdot T_{\text{sense}_i} \cdot S_{\text{region}}}{2 \cdot r_{\text{com}} \cdot N_{\text{mov}} \cdot |\mathbf{T}_i|}}{\frac{Th}{D} \cdot \frac{S_{\text{region}} \cdot T_{\text{sense}_i}}{N_{\text{mov}} \cdot |\mathbf{T}_i|} + \frac{T_{\text{train}_{i_1}} \cdot \nu_r \cdot T_{\text{sense}_i} \cdot S_{\text{region}}}{2 \cdot r_{\text{com}} \cdot N_{\text{mov}} \cdot |\mathbf{T}_i|}} = \frac{T_{\text{train}_{i_1}} \cdot \nu_r}{(T_{\text{train}_{i_1}} \cdot \nu_r) + (2 \cdot r_{\text{com}} \cdot \frac{Th}{D})}. \quad (3.11)$$

3.2.2 Guideline for Deploying Fixed Node

It is impossible to derive the optimal locations of fixed nodes directly from the result of the analysis, i.e. Formula (3.5). This is because the aim of the analysis of the sensing rate, R_{sense} itself is not to derive the optimal locations of fixed nodes but to derive the effects of the locations of fixed nodes. Thus, the formulae in the analysis are based not only on the locations of fixed nodes but on other parameters that indirectly depend on the locations of fixed nodes. Specifically, some given parameters for the analysis such as $\mathbf{T}_i$ and $\mathbf{d}_i$ depend on the locations of fixed nodes. In addition, the distance between fixed nodes, $|\mathbf{L}_i - \mathbf{L}_j|$, appears in the result of the analysis. This value also indirectly depends on the locations of fixed nodes.

Thus, we firstly extracted the abstract guidelines from the result of the analysis in order to make it easier to determine the locations of fixed nodes.

From Formula (3.5), we can see that R_{sense} depends on the total of required numbers of mobile nodes to construct a communication route, $\sum_{i=0}^{N_{\text{fix}}} N_{\text{req}_i}$, and the average sensing cycle time, T_{sense}.

First, from Formulae (3.1) and (3.2), T_{sense} is represented by the following formula:

$$T_{\text{sense}} = \sum_{i=0}^{N_{\text{fix}}} \frac{|\mathbf{T}_i|}{S_{\text{region}}} \Big(\frac{2 \cdot ave(|\mathbf{L}_i - \mathbf{d}_i|)}{\nu_m} + \sum_{j=0, j \neq i}^{N_{\text{fix}}} \frac{|\mathbf{T}_j|}{S_{\text{region}}} \cdot \frac{|\mathbf{L}_i - \mathbf{L}_j|}{\nu_m} \Big) + T_{\text{acq}}. \quad (3.12)$$

This formula indicates that, T_{sense} depends on the average distance between $\mathbf{L}_i$ and $\mathbf{d}_i$, $ave(|\mathbf{L}_i - \mathbf{d}_i|)$, the size of each territory, $|\mathbf{T}_i|$, and the distance between fixed nodes, $|\mathbf{L}_i - \mathbf{L}_j|$. The smaller these parameters are, the smaller T_{sense} becomes.

First, in order to decrease $ave(|\mathbf{L}_i - \mathbf{d}_i|)$, we suppose that, the location of a fixed node should be the center of the corresponding territory .

Next, we discuss the way to minimize the value of $\sum_{i=0}^{N_{\text{fix}}}(|\mathbf{T}_i| \cdot ave(|\mathbf{L}_i - \mathbf{d}_i|))$. Since $ave(|\mathbf{L}_i - \mathbf{d}_i|)$ depends on $|\mathbf{T}_i|$, we can regard $ave(|\mathbf{L}_i - \mathbf{d}_i|)$ as a function of $|\mathbf{T}_i|$. Here, we regard $ave(|\mathbf{L}_i - \mathbf{d}_i|)$ as a proportional to $|\mathbf{T}_i|$ for simplicity. Thus, we can express the value as $\sum_{i=0}^{N_{\text{fix}}}(\beta|\mathbf{T}_i|^2)$. In order to minimize this value, we discuss the partial differentiation of $\sum_{i=0}^{N_{\text{fix}}}|\mathbf{T}_i|^2$ with respect to $|\mathbf{T}_i|$. First, we differentiate partially with respect to $|\mathbf{T}_0|$.

$$\frac{\partial}{\partial|\mathbf{T}_0|}\sum_{i=0}^{N_{\text{fix}}}(|\mathbf{T}_i|^2) = \frac{\partial}{\partial|\mathbf{T}_0|}(|\mathbf{T}_0|^2 + |\mathbf{T}_1|^2 + \cdots + |\mathbf{T}_{N_{\text{fix}}}|^2). \tag{3.13}$$

Here, since $|\mathbf{T}_i|$ can be expressed as $S_{\text{region}} - \sum_{j=0,j\neq i}^{N_{\text{fix}}}|\mathbf{T}_j|$, the partial differentiation of $|\mathbf{T}_i|^2$ can be expressed as $2 \cdot (S_{\text{region}} - \sum_{j=0,j\neq i}^{N_{\text{fix}}}|\mathbf{T}_j|)$. Therefore,

$$\begin{aligned}\frac{\partial}{\partial|\mathbf{T}_0|}\sum_{i=0}^{N_{\text{fix}}}(|\mathbf{T}_i|^2) &= |\mathbf{T}_0| - (N_{\text{fix}} - 1) \cdot S_{\text{region}} + (N_{\text{fix}} - 1) \cdot |\mathbf{T}_0| \\ &\quad + (N_{\text{fix}} - 2) \cdot S_{\text{region}} \\ &= N_{\text{fix}} \cdot |\mathbf{T}_0| - S_{\text{region}}. \end{aligned} \tag{3.14}$$

In order to minimize this value, $|\mathbf{T}_0|$ should be $S_{\text{region}}/N_{\text{fix}}$. In the same way, we can derive $|\mathbf{T}_i|$ for all $i = \{0, 1, \cdots, N_{\text{fix}}\}$ as the same value $S_{\text{region}}/N_{\text{fix}}$. Therefore, considering when the location of a fixed node is the center of the corresponding territory, we can derive the guideline:

Guideline 1: *The distance between fixed nodes on each communication route and the territory size of each fixed node should be uniform.*

Next, in order to decrease $\sum_{i=0}^{N_{\text{fix}}}\sum_{k=0,k\neq i}^{N_{\text{fix}}}|\mathbf{L}_i - \mathbf{L}_k|$, the distance between fixed nodes should be small. This term represents the moving distance of a mobile node from the connected fixed node to the fixed node that has charge of the next sensing point. Since each mobile node in DATFM goes through the fixed nodes of adjacent territories until arriving the territory which contains the next sensing point, we can derive the following guideline:

Guideline 2: *The total length of moving paths of mobile nodes between all pairs of fixed nodes should be small.*

Next, we discuss the relation between N_{mov} and $\sum_{i=0}^{N_{\text{fix}}} R_{\text{route}_i} \cdot N_{\text{req}_i}$. From Formula (3.4), the ratio of free nodes to all mobile nodes is represented by the following formula:

$$\frac{N_{\text{free}}}{N_{\text{mov}}} = 1 - \frac{\sum_{i=0}^{N_{\text{fix}}} R_{\text{route}_i} \cdot N_{\text{req}_i}}{N_{\text{mov}}}. \tag{3.15}$$

When this value becomes lower than zero, all mobile nodes cannot perform sensing operations, and thus, the network does not work. Therefore, the value must be larger than zero. Here, in the data transmission from fixed node i to j, $|\mathbf{L}_i - \mathbf{L}_j|$, can be expressed as $(N_{\text{req}_i} + 1) \cdot r_{\text{com}}$. Therefore, from Formula (3.12) we can derive the following formula:

$$R_{\text{route}_i} = \frac{2 \cdot (N_{\text{req}_i} - 1) \cdot r_{\text{com}}}{(2 \cdot (N_{\text{req}_i} - 1) \cdot r_{\text{com}}) + (2 \cdot r_{\text{com}} \cdot \frac{Th}{D})} \approx \frac{N_{\text{req}_i}}{N_{\text{req}_i} + \frac{Th}{D}}. \tag{3.16}$$

Thus,

$$\sum_{i=0}^{N_{\text{fix}}} R_{\text{route}_i} \cdot N_{\text{req}_i} = \sum_{i=0}^{N_{\text{fix}}} \frac{N_{\text{req}_i}^2}{N_{\text{req}_i} + \frac{Th}{D}}. \tag{3.17}$$

Here, if Guideline 1 is satisfied, N_{req_i} becomes uniform among all i-s. Thus,

$$\sum_{i=0}^{N_{\text{fix}}} R_{\text{route}_i} \cdot N_{\text{req}_i} = \frac{N_{\text{fix}} \cdot N_{\text{req}_i}^2}{N_{\text{req}_i} + \frac{Th}{D}}. \tag{3.18}$$

In order for $N_{\text{free}}/N_{\text{mov}}$ to be larger than zero,

$$\begin{aligned} N_{move} &> \frac{N_{\text{fix}} \cdot N_{\text{req}_i}^2}{N_{\text{req}_i} + \frac{Th}{D}}. \\ \frac{N_{move} - P}{2 \cdot N_{\text{fix}}} &< N_{\text{req}} < \frac{N_{move} + P}{2 \cdot N_{\text{fix}}}. \\ (P &= \sqrt{N_{move}^2 + 4 \cdot N_{\text{fix}} \cdot N_{move} \cdot \frac{Th}{D}}) \end{aligned} \tag{3.19}$$

Let us define the right-hand-side term in Formula (3.19) as N_{max}. From the above formula, we can derive the guideline:

Guideline 3: *The required number of mobile nodes to construct the communication route from each fixed node should be smaller than* N_{max}.

However, if we follow Guidelines 2 and 3, most fixed nodes are deployed near the sink node. This may violate Guideline 1. Therefore, we should set the following guideline in order to suppress such an undesirable increase of the difference of territory sizes:

Guideline 4: *Some fixed nodes should be deployed at locations which are far from the sink node with high priority.*

From the analysis, we can see that some guidelines can be derived intuitively from the results of the analysis. For example, Formula (3.12), which derives the average sensing cycle time, T_{sense}, includes the distance between fixed nodes, i.e. $|\mathbf{L}_i - \mathbf{L}_j|$. Thus, we can see that T_{sense} becomes smaller by reducing the value $|\mathbf{L}_i - \mathbf{L}_j|$. This idea corresponds to Guideline 2. In addition, from Formula (3.15), we can see that the distance of the route from each fixed node should be small in order to improve the efficiency of data transmission processes. This idea corresponds to Guideline 3. Although this guideline is intuitive, it derives the specific value $N_{\max}$ that indicates the upper limit of the distance between fixed nodes in order for the network to work. Thus, the strategy adopts this value for determining the locations of fixed nodes. This value is an important parameter that can be derived only from the analysis.

On the other hand, other guidelines, i.e. Guidelines 1 and 4, cannot be intuitively derived because they do not follow the above discussion. For example, we derive Guideline 1 (*The distance between fixed nodes on each communication route and the territory size of each fixed node should be uniform*) by the partial differentiation shown in Formula (3.13). This guideline indicates that the performance is improved not only by reducing the distances between fixed nodes, but also by uniforming the territory size and the moving distance in each territory. In addition, Guideline 4 also cannot be intuitively derived because this is derived to avoid the violation of Guideline 1 due to (intuitive) Guidelines 2 and 3.

3.2.3 Strategy for Deploying Fixed Nodes

Based on the above guidelines, we devise the following strategy for deploying fixed nodes. Note that we design the strategy assuming the rectangular region with the sink node deployed at a corner of the region. Let us use an example where 6 fixed nodes (including the sink node) and 40 mobile nodes are deployed in 2,100[m]×2,100[m] flatland, and the communication range is 100 [m] for explaining the strategy.

The strategy for deploying fixed nodes in DATFM/DF is divided into two steps, determining the pattern of deployment, and adjusting the distance between all pairs of fixed nodes in the determined pattern.

Determining pattern of deployment

First, DATFM/DF determines the pattern of deployment based on Guidelines 1 and 2. Here, Guideline 2 only considers the moving distance between neighboring territories in a sensing cycle. On the other hand, Guideline 1 considers not only the moving distance between territories but also that in each territory and the efficiency of data transmission processes. Therefore, if we apply Guideline 2 prior to Guideline 1, the performances of operations that Guideline 1 considers may deteriorate. Thus, DATFM/DF first derives multiple candidates of deployment patterns considering Guideline 1 and selects the one considering Guideline 2.

1. Draw the line from the sink node to the farthest point. Let us define the length of the line as L. Next, calculate the ideal length of the communication route l by the following formulae:

$$\frac{S_{\text{region}}}{N_{\text{fix}}} = \pi \cdot (\frac{l}{2})^2$$
$$l = 2 \cdot \sqrt{\frac{S_{\text{region}}}{\pi \cdot N_{\text{fix}}}}. \tag{3.20}$$

   In the above formulae, we assume the shape of each territory as a circle for simplicity. This formula indicates that all regions have the same size ($S_{\text{region}}/N_{\text{fix}}$) (from Guideline 1). Next, calculate the required number of fixed nodes in order to deploy a fixed node at the farthest point from the sink node when the distance between fixed nodes is set as l (as shown in Figure 3.1). We define this value as N_{hop}. In the example, L becomes 2,970[m] ($= 2100 \cdot \sqrt{2}$) and l becomes 970[m] ($= 2 \cdot \sqrt{(2100^2/\pi \cdot 6)}$). Thus, $N_{\text{hop}} \approx 3$ (2970/970).

2. Divide the region into $N_{\text{hop}} + 1$ zones as shown in Figure 3.2. Then, calculate the number of fixed node deployed in zone i, N_{zone_i}, according to the following formula:

$$N_{\text{zone}_i} = \lceil N_{\text{fix}} \cdot \frac{|\mathbf{Z}_i|}{S_{\text{region}}} \rceil. \tag{3.21}$$

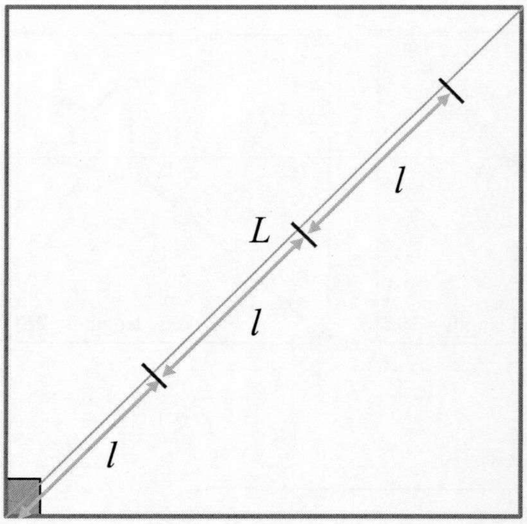


Figure 3.1: Calculate N_{hop}.

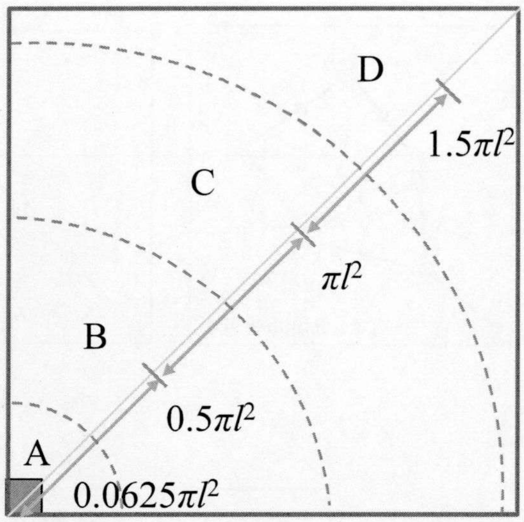


Figure 3.2: Divide the region.

$|\mathbf{Z}_i|$ is the size of zone i. In the example, the region is divided into four zones, Z_A, Z_B, Z_C and Z_D whose sizes are respectively $0.0625\pi l^2$, $0.5\pi l^2$, πl^2, and $1.5\pi l^2$. Since the number of fixed nodes is 6, the number of fixed nodes deployed in each zone respectively becomes 1 ($= \lceil 6 \cdot 0.0625/3.0625 \rceil$), 1 ($= \lceil 6 \cdot 0.5/3.0625 \rceil$), 2 ($= \lceil 6 \cdot 1/3.0625 \rceil$) and 3 ($= \lceil 6 \cdot 1.5/3.0625 \rceil$).

3. Make all patterns of deployment that satisfy the following conditions:

   - The distance between adjacent fixed nodes is uniform.
   - The required number of the routes for each fixed node to transfer its holding data to the sink node (we define this value as the *hop-count*) is equal to or more than N_{hop}.
   - The number of fixed nodes in each zone i is equal to or less than N_{zone_i} calculated in the previous step.

   Figure 3.3 shows the patterns of deployment that satisfy the above conditions in the example. In this figure, the distance between adjacent fixed nodes is expressed as K.

4. Calculate the total length of moving paths of mobile nodes between all pairs

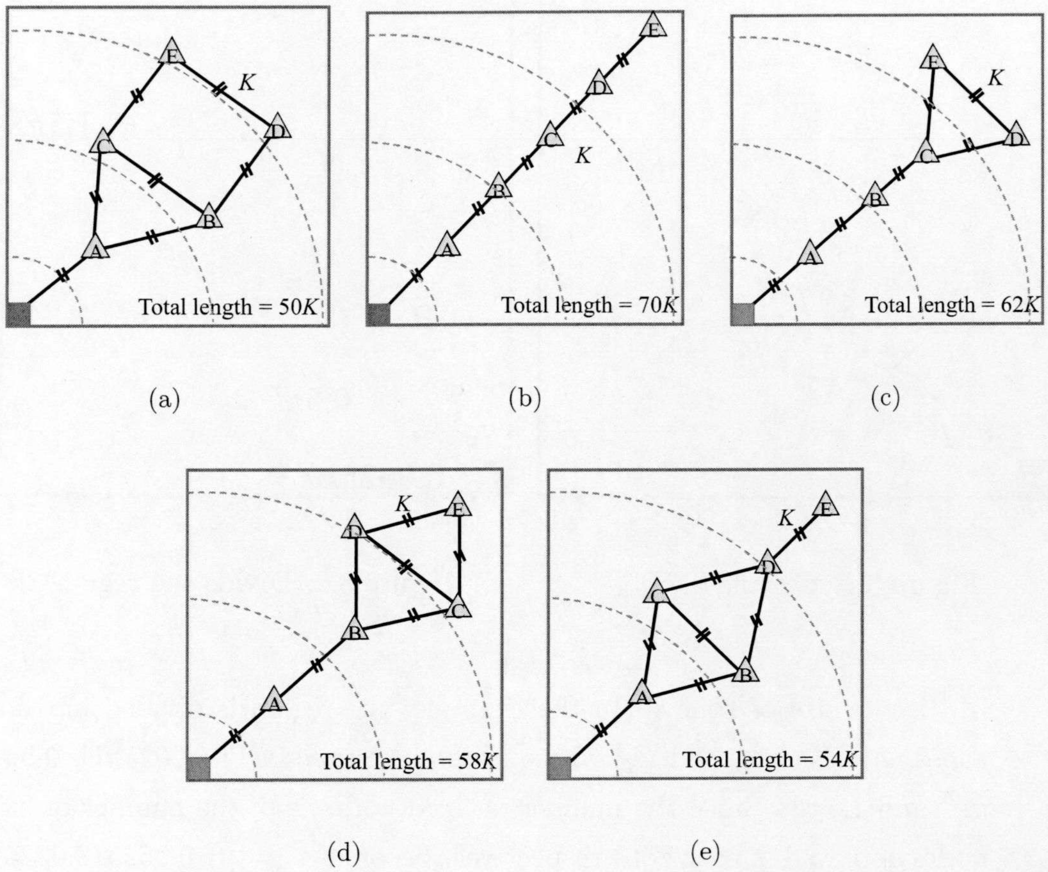


Figure 3.3: Deployment patterns.

of fixed nodes. For example in Figure 3.3(a), the lengths of moving paths between all pairs of fixed nodes become as shown in Table 3.2. Thus, the total length in this pattern becomes $50K$. After that, select the pattern with the minimum length as the initial deployment pattern (from Guideline 2). In the example, Figure 3.3(a) is adopted as the initial pattern.

Adjusting distance between fixed nodes

Next, DATFM/DF adjusts the distance between all pairs of fixed nodes in the determined pattern.

5. Derive the optimal distance between adjacent fixed nodes in order to deploy

Table 3.2: Lengths of moving paths in Figure 3.3(a).

from\to	Sink	A	B	C	D	E
Sink	-	$K(\text{Sink}\rightarrow A)$	$2K(\text{Sink}\rightarrow A \rightarrow B)$	$2K(\text{Sink}\rightarrow A \rightarrow C)$	$3K(\text{Sink}\rightarrow A \rightarrow B \rightarrow D)$	$3K(\text{Sink}\rightarrow A \rightarrow C \rightarrow E)$
A	$K(A \rightarrow \text{Sink})$	-	$K(A \rightarrow B)$	$K(A \rightarrow C)$	$2K(A \rightarrow B \rightarrow D)$	$2K(A \rightarrow C \rightarrow E)$
B	$2K(B \rightarrow A \rightarrow \text{Sink})$	$K(B \rightarrow A)$	-	$K(B \rightarrow C)$	$K(B \rightarrow D)$	$2K(B \rightarrow D \rightarrow E)$
C	$2K(C \rightarrow A \rightarrow \text{Sink})$	$K(C \rightarrow A)$	$K(C \rightarrow B)$	-	$2K(C \rightarrow B \rightarrow D)$	$K(C \rightarrow E)$
D	$3K(D \rightarrow B \rightarrow A \rightarrow \text{sink})$	$2K(D \rightarrow B \rightarrow A)$	$K(D \rightarrow B)$	$2K(D \rightarrow B \rightarrow C)$	-	$K(D \rightarrow E)$
E	$3K(F \rightarrow C \rightarrow A \rightarrow \text{sink})$	$2K(E \rightarrow C \rightarrow A)$	$2K(E \rightarrow C \rightarrow B)$	$K(E \rightarrow C)$	$K(E \rightarrow D)$	-

some fixed nodes far from the sink node (from Guideline 4). First, fixed nodes with the maximum hop-count are deployed at the farthest points from the sink node. Thus, we adjust the distance between adjacent fixed nodes according to the maximum hop-count. Specifically, when the distance between adjacent fixed nodes and the maximum hop-count are respectively defined as l_{opt} and $H_{\max}$, the distance between the sink node and fixed nodes with the maximum hop-count becomes $H_{\max} \cdot l_{\text{opt}}$. Here, when this value is too large, some fixed nodes may located outside of the target region. To avoid such situation, we introduce the distance between the sink node and the nearest corner of the region, $L_{\min}$. Specifically, we set l_{opt} in order for the territory of a fixed node with the maximum hop-count to be located within the range of $L_{\min}$ from the sink node (see Figure 3.4). When we assume that the location of a fixed node is the center of the corresponding territory and that the shape of each territory becomes a circle for simplicity, the distance between a fixed node and the boundary of its territory becomes $2/l_{\text{opt}}$. Based on this discussion, l_{opt} is calculated by the following formula:

$$l_{\text{opt}} = 2 \cdot \frac{L_{\min}}{(2 \cdot H_{\max} + 1)}. \tag{3.22}$$

In the example, since $H_{\max}$ is 3 and $L_{\min}$ is 2,100, l_{opt} becomes 600[m] ($= 2 \cdot 2100/(2 \cdot 3 + 1)$).

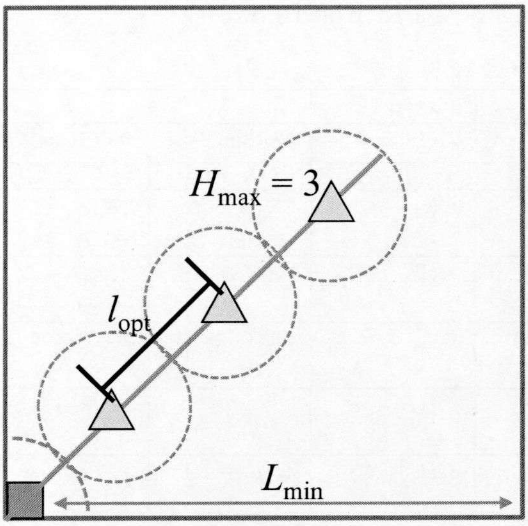


Figure 3.4: Adjust the distance.

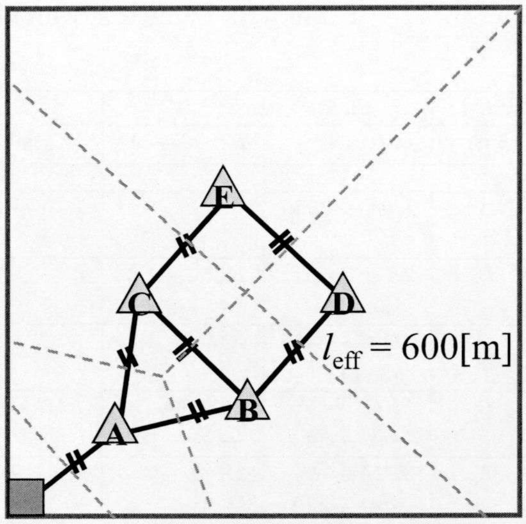


Figure 3.5: Deployment of fixed node.

6. Derive the maximum distance between adjacent fixed nodes as l_{max} according to the following formula (from Guideline 3):

$$l_{max} = N_{max} \cdot r_{com}. \tag{3.23}$$

Next, compare l_{max} with l_{opt} and adopt the smaller one as distance between adjacent fixed nodes l_{eff}. In the example, when Th/D is 30, l_{max} becomes 2000[m] (= $((40+\sqrt{1600+24000})/10)\cdot$ 100: see Formula (3.19)). Thus, l_{eff} becomes l_{opt}, which is smaller than l_{max}.

7. Finally, adjust the locations of fixed nodes. Specifically, adjust the distance between adjacent fixed nodes to l_{eff} while keeping the shape of deployment pattern selected in step (4). In the example, since l_{eff} is 600[m] (calculated in step (5)), the deployment becomes as shown in Figure 3.5.

As described in Section 3.2.2, when the condition of Guideline 3 is not satisfied, the network does not work. On the other hand, Guideline 4 is derived for preventing the violation of Guideline 1 due to Guidelines 2 and 3. Thus, the importance of Guideline 4 becomes lower compared with Guideline 3. In the above strategy, l_{max} and l_{opt} are calculated respectively based on Guidelines 3 and 4.

According to Guideline 3, the distance between fixed nodes should be smaller than $l_{\max}$ in order for the network to work. On the other hand, according to Guideline 4, the distance should be at least l_{opt} in order to avoid the violation of Guideline 1. Thus, the strategy adopts the smaller one among $l_{\max}$ and l_{opt} as the distance between adjacent fixed nodes l_{eff} in order for l_{eff} not to be larger than $l_{\max}$.

3.2.4 Verification of the Guidelines

In order to verify the validity of the guidelines, we conduct the simulation experiments. In the experiments, we verify the validities of $l_{\max}$ and l_{opt}, which are respectively derived from Guidelines 3 and 4. The simulation environment is basically same as that in Section 2.4.1. There are 6 fixed nodes (including the sink node) which are deployed according to the deployment pattern determined in the former part of the strategy in Section 3.2.3. On the other hand, 40 mobile nodes are deployed in the region. Each mobile node moves with velocity of 5[m/s] in *SM* and 10[m/s] in *TM* and *CM*. In this environment, we change the distance between adjacent fixed nodes (K in Figure 3.3) and evaluate the throughput, which is defined as the amount of data that arrive at the sink node in 1[sec]. Figure 3.6 shows the simulation result. From this result, the throughput becomes highest when the distance between fixed nodes equals to l_{opt} (= l_{eff} in the strategy). Moreover, when the distance is larger than $l_{\max}$, the throughput begins to decrease drastically. This result indicates that the guidelines derived from the result of the analysis appropriately work in determining the deployment of fixed nodes.

3.3 Performance Evaluation

In this section, we show the results of simulation experiments regarding performance evaluation of DATFM/DF. In the simulation experiments, we compare the performances of DATFM/DF with the following two methods:

- **AverageDATFM**: The average performances of DATFM in all the patterns of deployment of fixed nodes on the grid whose interval is 350[m].

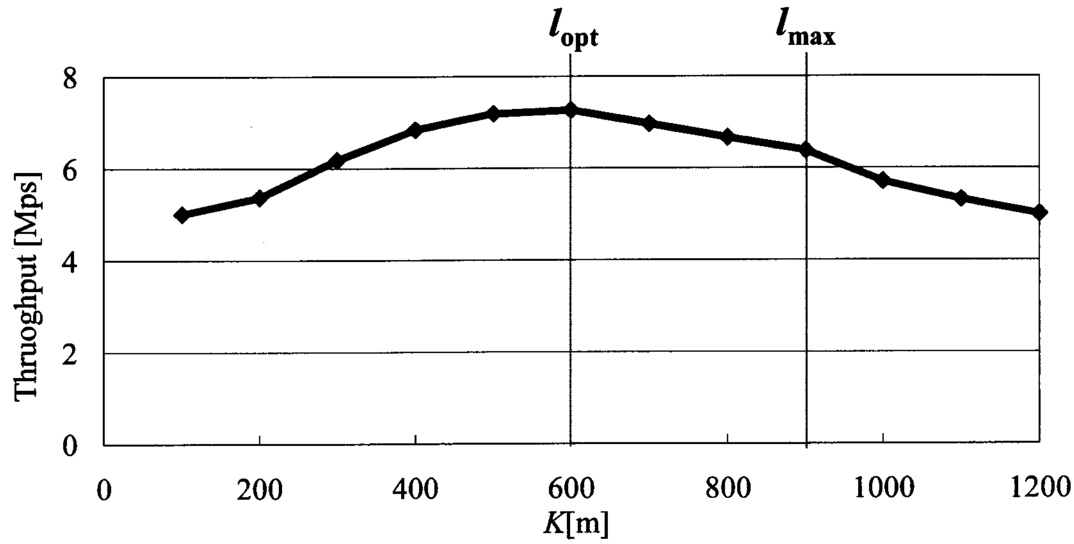


Figure 3.6: The effects of the distance between fixed nodes.

- **Semi-optimal**: The performances of DATFM when the throughput is the largest in all the patterns of deployment of fixed nodes on the grid whose interval is 350[m].

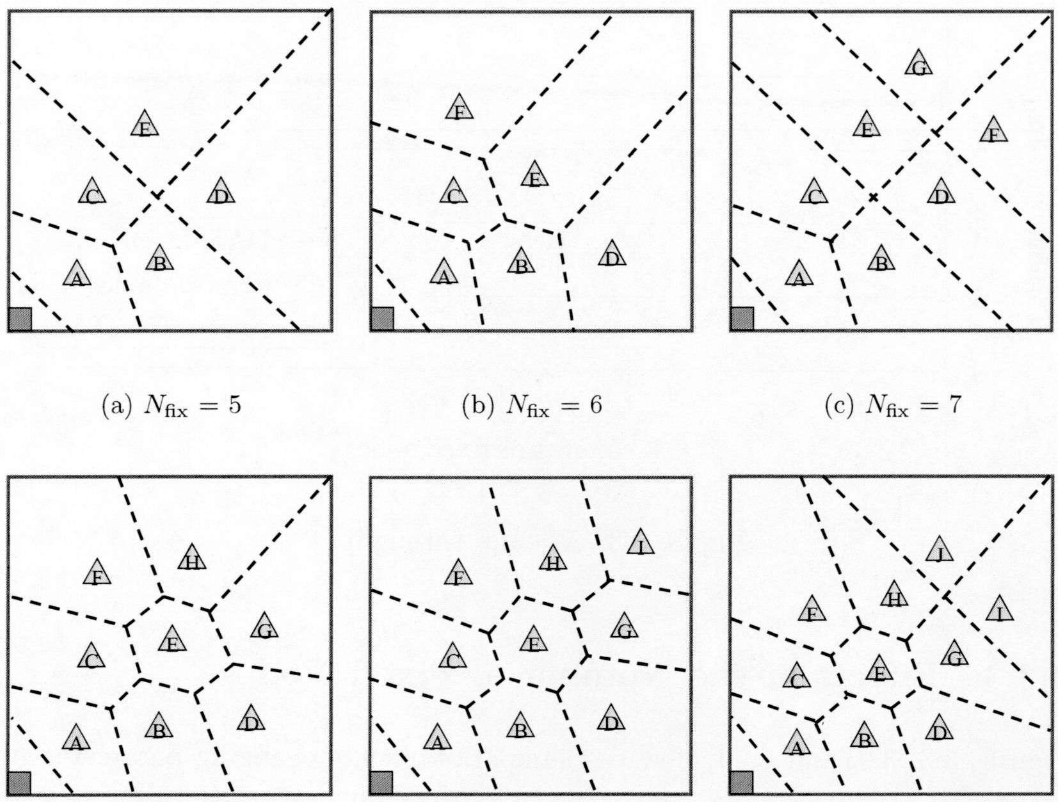
3.3.1 Simulation Environment

The simulation environment is basically same as that in Section 2.4.1. Each fixed node starts data transmission process when the amount of the accumulated data exceeds 1,000[Mbit]. Each mobile node moves with velocity of 5[m/s] in *SM* and 10[m/s] in *TM* and *CM*.

In this environment, we evaluate the averages of the following three criteria during 1[week]:

- Throughput
  The average amount of data that arrive at the sink node per 1[sec].

- Average moving distance
  The average of moving distances of all nodes during the simulation period.



(a) $N_{\text{fix}} = 5$ (b) $N_{\text{fix}} = 6$ (c) $N_{\text{fix}} = 7$

(d) $N_{\text{fix}} = 8$ (e) $N_{\text{fix}} = 9$ (f) $N_{\text{fix}} = 10$

Figure 3.7: The locations of fixed nodes in DATFM/DF.

- Average delay
  The average of the elapsed time after data are acquired (sensed) until the data arrive at the sink node.

3.3.2 Locations of Fixed Nodes

Figure 3.7 shows the locations of fixed nodes in DATFM/DF changing parameter N_{fix}. From this figure, we can see that the size of each territory becomes uniform while keeping the distances between adjacent fixed nodes uniform, especially when the number of fixed nodes is large. This indicates that DATFM/DF appropriately deploys fixed nodes based on the guidelines described in Section 3.2.2.

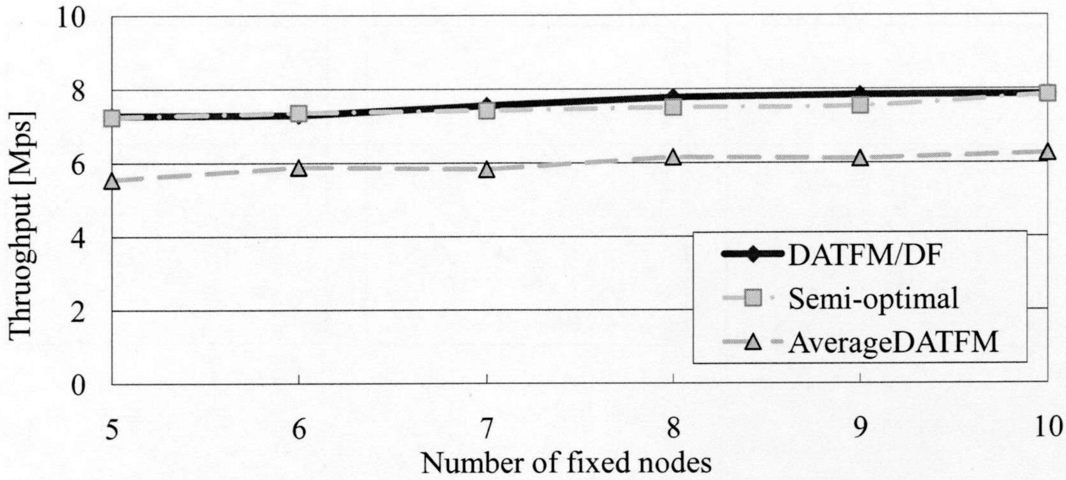


Figure 3.8: N_{fix} and throughput.

3.3.3 Effects of the Number of Fixed Nodes

Figures 3.8, 3.9, and 3.10 show the simulation results changing parameter N_{fix}. The horizontal axis on all graphs indicates the number of fixed nodes N_{fix}. The vertical axis indicates throughput in Figure 3.8, average moving distance in Figure 3.9, and average delay in Figure 3.10.

Figure 3.8 shows that the throughput in DATFM/DF is always larger than that in AverageDATFM, and almost same as (sometimes larger than) that in semi-optimal deployment. In addition, the throughput in DATFM/DF becomes larger than that in UM-controlled (see Figure 2.12) although the total number of mobile sensors in DATFM/DF is smaller than that in UM-controlled. This indicates that the strategy in DATFM/DF is effective to improve the efficiencies of sensing and data transfer. Throughputs in all methods slightly increase as the number of fixed nodes increases. This is because the increase of fixed nodes makes the average distance between fixed nodes smaller, and thus, it becomes easier for each mobile nodes to connect to fixed nodes.

Figure 3.9 shows that the moving distance in DATFM/DF is slightly smaller than those in semi-optimal deployment and AverageDATFM. This is because the strategy in DATFM/DF aims to reduce the average distance between fixed nodes.

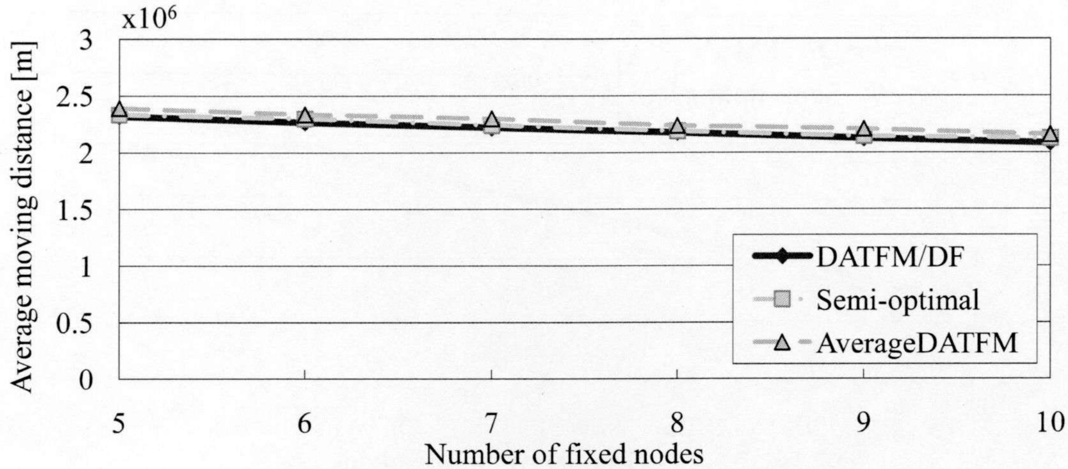


Figure 3.9: N_{fix} and average moving distance.

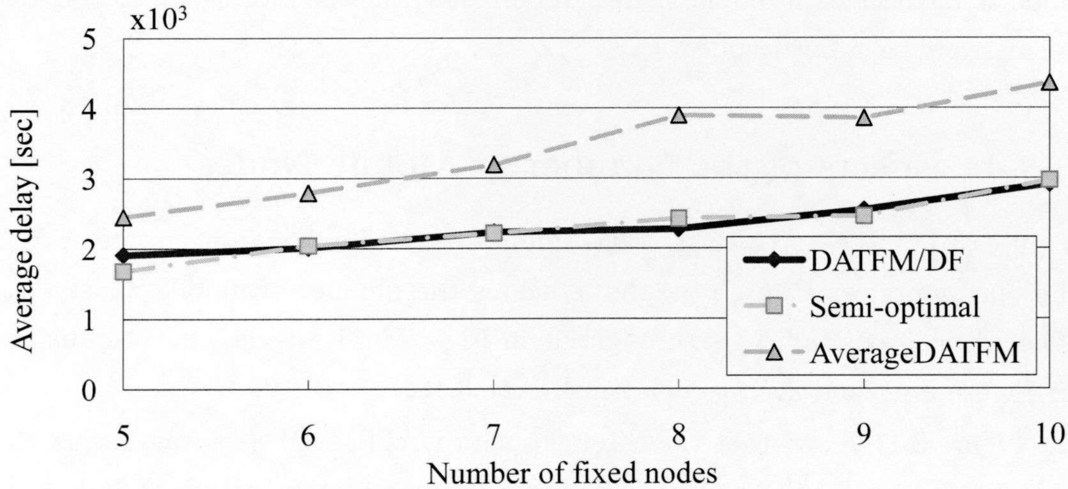


Figure 3.10: N_{fix} and average delay.

Figure 3.10 shows that the delays in DATFM/DF is always smaller than that in AverageDATFM. This result shows that the strategy in DATFM/DF improves the efficiencies of data accumulation and transmission. Moreover, delays in all methods become larger as the number of fixed nodes increases. This is because the larger the number of fixed nodes is, the lower the frequency for each fixed

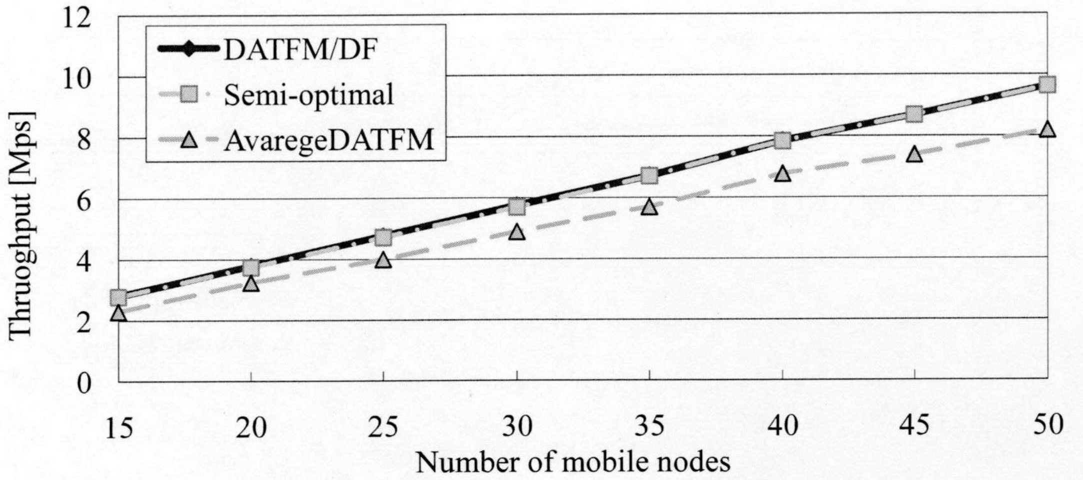


Figure 3.11: N_{mov} and throughput.

node to connect with mobile nodes. As a result, the time for accumulating data increases in each fixed node.

3.3.4 Effects of the Number of Mobile Nodes

Figures 3.11, 3.12, and 3.13 show the simulation results changing parameter N_{mov}. The horizontal axis on all graphs indicates the number of mobile nodes N_{mov}. The vertical axis indicates throughput in Figure 3.11, average moving distance in Figure 3.12, and average delay in Figure 3.13.

Figure 3.11 shows that the throughput in DATFM/DF is always larger than that in AverageDATFM. This is due to the same reason as that in Figure 3.8. Moreover, throughputs in all methods increase as the number of mobile nodes increases. This is because it becomes easier for each fixed node to collect the sufficient number of mobile nodes to construct a communication route.

Figure 3.12 shows that the moving distance in DATFM/DF is always smaller than that in AverageDATFM. This is due to the same reason as that in Figure 3.9. Also, the moving distance in DATFM/DF is smaller than that in semi-optimal deployment when the number of mobile nodes is small. This is because DATFM/DF tries to deploy fixed nodes at the center of territories. This results in

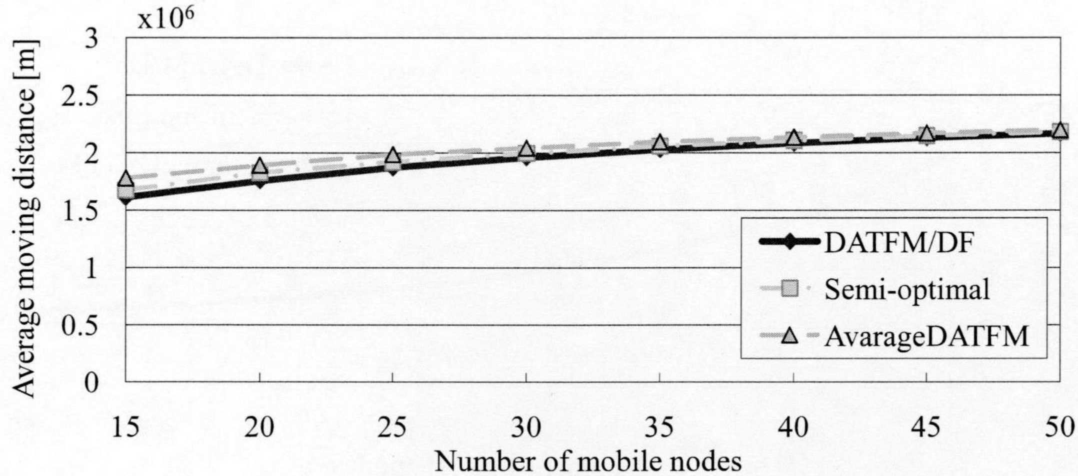


Figure 3.12: N_{mov} and average moving distance.

the reduction of the moving distance from fixed node to sensing point. Moreover, the moving distances in all methods increase as the number of sensor nodes gets larger. This is because the frequency of data transmissions in each fixed node increases due to the increase of the frequency that a fixed node receives data from mobile nodes. In such an environment, the movement of mobile nodes for data transmissions increases.

Figure 3.13 shows that the delay in DATFM/DF is always smaller than that in AverageDATFM. This is due to the same reason as that in Figure 3.10. Moreover, delays in all methods decrease as the number of sensor nodes gets larger. This is because the times elapsed for accumulating data in each fixed node (T_{acc_i}) decreases.

3.4 Discussion

From the results of the simulation experiments, we can see that DATFM/DF deploys fixed nodes to appropriate locations to achieve high throughput. However, the deployment in DATFM/DF is not the optimal since the strategy is designed by simple guidelines considering the sensing rate. This is because it is very difficult to find the optimal deployment to maximize the sensing rate. For example,

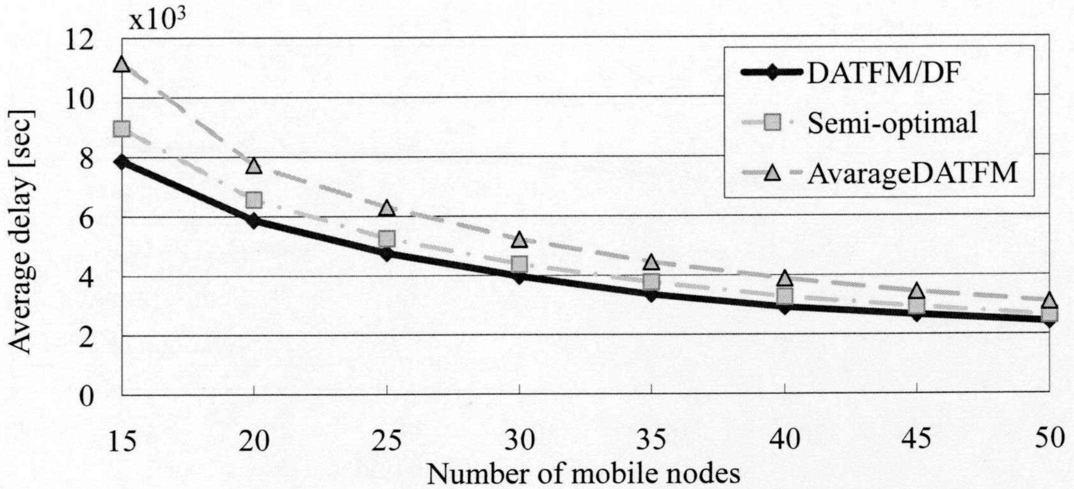


Figure 3.13: N_{mov} and average delay.

to derive the semi-optimal deployment with 7 fixed nodes, we have evaluated throughputs in more than 3,000,000 patterns. When the numbers of fixed nodes and mobile nodes become larger, the number of trials becomes much larger. On the other hand, DATFM/DF can derive the deployment which achieves good performance without such trials. In addition, DATFM/DF can work in an environment where the number of nodes dynamically changes.

3.5 Conclusion

In this chapter, we have proposed an effective deployment strategy for fixed nodes, named DATFM/DF, to further improve the efficiencies of sensing and data transfer. DATFM/DF focuses on the locations of fixed nodes in DATFM which was proposed in Chapter 2. First, we analyzed the effects of the locations of fixed nodes on the performance. Based on the result of the analysis, we derived the guidelines for deploying fixed nodes in order to achieve high sensing rate, R_{sense}. After that, we proposed a deliberate deployment strategy based on the derived guidelines.

We have also conducted the simulation experiments to evaluate the performance of DATFM/DF. The results show that DATFM/DF improves the performance of DATFM by introducing the deployment strategy described in Section 3.2.3.

Chapter 4

Adjustment the Number of Nodes for Efficient Data Collection in Sparse Sensor Networks

4.1 Introduction

In DATFM, mobile nodes move around the whole region in order to acquire data in the whole region. By doing so, the whole region can be monitored even when some mobile nodes become unavailable due to the battery exhaustion or physical damages. However, in some environments where mobile nodes with much higher durability are deployed, the moving distances of mobile nodes can be suppressed, which can further improve the efficiency of sensing in DATFM.

In this chapter, we propose another extended method of DATFM, named DATFM/DA (DATFM with Deployment Adjusting) in order to alleviate the increase of moving distance of mobile nodes. Unlike DATFM/DF proposed in Chapter 3, This method focuses on the sensing operations of mobile nodes.

As described in Section 2.3.1, each mobile node in DATFM selects a territory to perform sensing according to the probability which is proportional to the size of each territory. However, the moving distance for performing a sensing operation increases when a mobile node selects a territory which is far from its current location. This causes the decrease in the frequency of sensing operations, and thus, the efficiencies of sensing and data transfer become low. To alleviate the increase of moving distance, DATFM/DA statically assigns a territory for each mobile node. Specifically, each mobile node performs sensing operations only in its assigned territory. Moreover, when a fixed node starts data transmission, it constructs the communication route only using mobile nodes deployed in its territory. Thus, DATFM/DA does not use the collection of mobile nodes by a mobile node in *CM*, described in Section 2.3.2. By doing so, it is expected that the delay for a data transmission decreases.

For this aim, it is important to determine appropriate number (ratio) of mobile nodes deployed to each territory. Intuitively, the larger the size of a territory is, the more mobile nodes should be deployed to the territory. However, since some mobile nodes have to stop their sensing operations for transferring data in a data transmission process, the efficiency of sensing deteriorates in a particular territory where data transmissions frequently occur. For example, fixed nodes near the sink node receive data accumulated by other fixed nodes far from the sink node, and thus, frequently perform data transmissions. Therefore, mobile nodes in such 'busy' territories have lower chance to perform sensing operations. Such a skew of the amount of acquired data cannot be allowed in some applications which need to acquire data in the whole region uniformly, or those which specify the required (minimum) amount of acquired data in the entire region. In order to avoid such a skew, DATFM/DA determines the number of mobile nodes deployed to each territory considering not only the size of territory but also the frequency of data transmissions.

Moreover, we discuss the integration of DATFM/DF and DATFM/DA in this chapter. As described above, each of DATFM/DF and DATFM/DA focuses on different aspects of DATFM, that is, locations of fixed nodes and the number of mobile nodes in each territory, respectively. Thus, further improvement of efficiencies of sensing and data transfer is expected by considering both of these aspects together. Furthermore, we discuss the cost for deploying fixed and mobile nodes through the performance evaluation of the integrated method.

The remainder of this chapter is organized as follows. In Section 4.2, we present DATFM/DA, which is proposed in this chapter. The results of simulation experiments are presented in Section 4.3. Next, we propose an integrated method of DATFM/DF and DATFM/DA, and evaluate the performances in Section 4.4. Finally, we conclude this chapter in Section 4.5.

4.2 DATFM/DA

In this section, we first present the details of moving strategy of mobile nodes and data transmission in DATFM/DA. After that, we explain the strategy to determine the number of mobile nodes deployed to each territory.

4.2.1 Sensing Operations of Mobile Nodes

As described above, DATFM/DA statically assigns a territory for each mobile node. Thus, each mobile node performs sensing operations only in its assigned territory. After performing a sensing operation, a mobile node moves to the fixed node in its assigned territory, receives the information on the next sensing point (in the same territory) from the fixed node, and moves to the point. By doing so, DATFM/DA reduces the moving distances of mobile nodes and improves the efficiency of sensing.

4.2.2 Data Transmission

Similar to DATFM, a fixed node starts data transmission to its destination node when the amount of the accumulated data in its memory exceeds the threshold. First, the fixed (source) node sends a *RCReq* to a mobile node that firstly connects to itself. The mobile node which receives the *RCReq* immediately changes its mode into *TM*. Then the source node starts train transmission using the connected mobile node. When another mobile node connects to the source node, the source node adds the connected node to the train until the complete communication route is constructed.

Unlike DATFM, DATFM/DA does not collect mobile nodes using a mobile node in *CM*. By doing so, the moving distance of the mobile node which firstly connects to the source node can be reduced. In addition, this may reduce the delay for data transmission.

4.2.3 Adjusting the Number of Mobile Nodes

As described in Section 4.1, it is desirable to deploy more mobile nodes in a territory where data transmissions frequently occur. By doing so, even when some mobile nodes stop sensing for transferring data in a territory, efficient sensing can be performed by using the remaining mobile nodes.

In order to determine the number of mobile nodes, we analyze the *sensing rate*, R_{sense}, and *sensing amount*, A_{sense}, in each territory. The sensing rate in a territory is defined as the number of sensing operations performed in a unit time. The sensing amount is defined as the total amount of data acquired in a unit area in a unit time. After analyzing these two metrics, we propose a strategy to deploy mobile nodes to each territory.

Assumptions

Before starting the analysis, we show the assumed environment in this section. As described in Section 4.2.1, in DATFM/DA, each mobile node performs sensing operations only in its assigned territory. Similar to DATFM/DF described in Chapter 3, we assume that the parameters in Table 3.1 are given.

Unlike the *sensing cycle* in Sections 2.3.2 and 3.2.1. The sensing cycle in DATFM/DA includes the operations shown in Figure 4.1, in which a mobile node departs from the fixed node, moves to the next sensing point, performs a sensing operation, and returns to the fixed node.

Analysis of R_{sense} and A_{sense}

Unlike DATFM and DATFM/DF, each mobile node in DATFM/DA performs sensing operation only in its assigned territory. Therefore, the average sensing cycle time of a mobile node that performs a sensing operation in $\mathbf{T}_i$, T_{sense_i}, becomes:

$$T_{\text{sense}_i} = T_{\text{acq}} + 2 \cdot \frac{ave(|\mathbf{d}_i - \mathbf{L}_i|)}{\nu_m}. \tag{4.1}$$

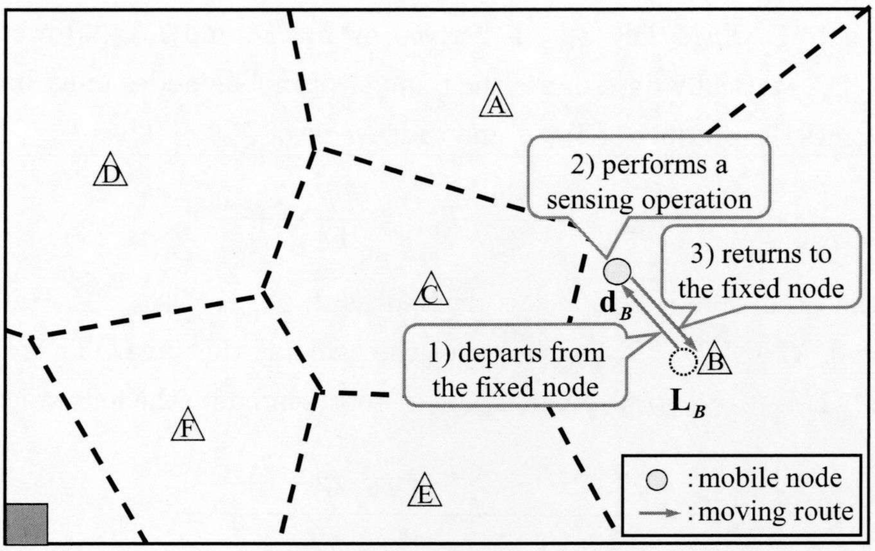


Figure 4.1: Operations in a sensing cycle.

After fixed node i accumulates data, it starts a data transmission. During the data transmission, mobile nodes which construct the communication route cannot perform sensing operations. Similar to the analysis in Section 3.2.1, we consider the frequency of data transmissions, R_{route_i}, and the number of mobile nodes used for a data transmission.

Let us define the average number of free nodes, N_{free_i}, that are not used for data transmission from fixed node i in a unit time. Then, R_{sense_i} is represented by using the ratio of free nodes to all mobile nodes in one average sensing cycle time, that is, $(N_{\text{free}_i}/N_{\text{mov}_i}) \cdot (1/T_{\text{sense}_i})$.

N_{free_i} can be represented by using the required number of mobile nodes to construct a communication route, N_{req_i}, and the frequency of data transmission from fixed node i, R_{route_i}, that is,

$$N_{\text{free}_i} = N_{\text{mov}_i} - (R_{\text{route}_i} \cdot N_{\text{req}_i}). \tag{4.2}$$

Therefore, R_{sense_i} is represented by the following formula:

$$R_{\text{sense}_i} = \frac{1}{T_{\text{sense}_i}} \cdot (1 - \frac{R_{\text{route}_i} \cdot N_{\text{req}_i}}{N_{\text{mov}_i}}). \tag{4.3}$$

Similar to DATFM/DF, R_{route_i} is derived by $T_{\text{transmit}_i}/T_{\text{transfer}_i}$ and T_{transfer_i} is the sum of T_{acc_i} and data transmission.

Similar to DATFM/DF, T_{acc_i} is derived by Th/D, and T_{ave_i}. However, since DATFM/DA statically determines the number of mobile nodes in each territory, T_{ave_i} becomes the product of T_{sense_i} and the inverse of N_{mov_i}. Thus, T_{acc_i} becomes:

$$T_{\text{acc}_i} = \frac{Th}{D} \cdot T_{\text{ave}_i} = \frac{Th}{D} \cdot \frac{T_{\text{sense}_i}}{N_{\text{mov}_i}}. \tag{4.4}$$

On the other hand, T_{transmit_i} is derived as $T_{\text{train}_{i_1}} + T_{\text{train}_{i_2}} + \cdots + T_{\text{train}_{i_N}}$, similar to DATFM/DF. $T_{\text{train}_{i_1}}$ becomes the same as that in DATFM/DF, that is, $2 \cdot (|\mathbf{L}_i - \mathbf{L}_j| - 2 \cdot r_{\text{com}})/\nu_r$. Next, $T_{\text{train}_{i_2}}$ is represented by the following formula:

$$\begin{aligned} T_{\text{train}_{i_2}} &= \frac{2 \cdot (|\mathbf{L}_i - \mathbf{L}_j| - r_{\text{com}} \cdot (2 + \frac{T_{\text{train}_{i_1}}}{T_{\text{ave}_i}})}{\nu_r} \\ &= A' \cdot T_{\text{train}_{i_1}} \qquad (A' = 1 - \frac{2 \cdot N_{\text{mov}} \cdot r_{\text{com}}}{\nu_r \cdot T_{\text{sense}_i}}). \end{aligned} \tag{4.5}$$

In the same way, the elapsed time for N-th round, $T_{\text{train}_{i_N}}$, is represented by the following formula:

$$\begin{aligned} T_{\text{train}_{i_N}} &= \frac{2 \cdot (|\mathbf{L}_i - \mathbf{L}_j| - r_{\text{com}} \cdot (2 + N_{\text{train}_{N-1}})}{\nu_r} \\ &= A'^{N-1} \cdot T_{\text{train}_{i_1}}. \end{aligned} \tag{4.6}$$

Therefore, T_{transmit_i} in DATFM/DA can be represented by the following formula:

$$T_{\text{transmit}_i} = \frac{(1 - A'^{N-1}) \cdot T_{\text{train}_{i_1}}}{1 - A'}. \tag{4.7}$$

Similar to A in Formula 3.9, we assume that $A'^{N-1} \approx 0$. Thus, T_{transmit_i} can be represented by the following formula:

$$T_{\text{transmit}_i} = \frac{T_{\text{train}_{i_1}} \cdot \nu_r \cdot T_{\text{sense}_i}}{2 \cdot r_{\text{com}} \cdot N_{\text{mov}}}. \tag{4.8}$$

Here, since the accumulated data are transmitted toward the sink node via nearby fixed nodes, some fixed nodes receive data accumulated by other fixed nodes. Such fixed nodes accumulate data faster than others. Thus, data transmissions frequently occur in the territories of these fixed nodes, and thus the sensing data in such territory cannot acquired for a long time. Compared with DATFM/DF, which the mobile nodes travel around the region and acquires data

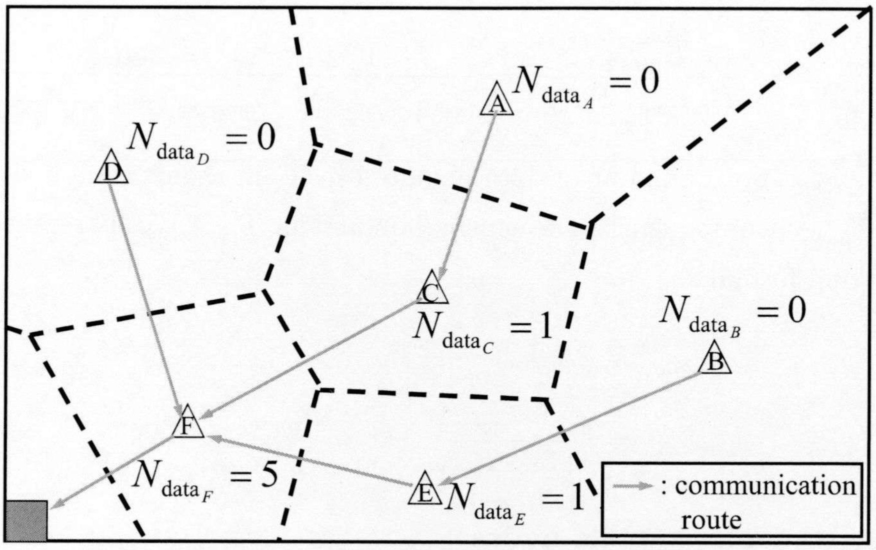


Figure 4.2: N_{data} of fixed nodes.

throughout the region, this problems becomes seriously. This is because, the mobile nodes moves only in its assigned territory, and the fixed nodes uses only mobile nodes in its territory to transfers data. Base on this discussion, DATFM/DA consider the effect of the frequency of data transmission when receiving data by other fixed nodes.

To consider such a skew of frequency of data transmissions, let us define the number of fixed nodes which transmit data via fixed node i as N_{data_i}. For example in Figure 4.2, N_{data_F} becomes 5 because fixed node F receives data accumulated by five fixed nodes $\{A, B, C, D, E\}$. Using N_{data_i}, we adjust the time elapsed for accumulating data, T_{acc_i}, and that elapsed for data transmission, T_{transmit_i}. Fixed node i receives data accumulated by N_{data_i} fixed nodes in addition to that acquired by mobile nodes in its territory. Thus, we assume that fixed node i accumulates data $(N_{\text{data}_i} + 1)$ times faster than other fixed nodes whose N_{data} equals to 0. Thus, we express the time elapsed for accumulating data, T'_{acc_i} as $T_{\text{acc}_i}/(N_{\text{data}_i} + 1)$. On the other hand, fixed node i transmits data $(N_{\text{data}_i} + 1)$ times larger than other fixed nodes whose N_{data} equals to 0. Thus, we express the time elapsed for data transmission (T'_{transmit_i}) as $(N_{\text{data}_i} + 1) \cdot T_{\text{transmit}_i}$. From the above discussion, R_{route_i} is represented by the following formula:

$$R_{\text{route}_i} = \frac{T'_{\text{transmit}_i}}{T'_{\text{transmit}_i} + T'_{\text{acc}_i}} = \frac{(N_{\text{data}_i}+1)\cdot T_{\text{transmit}_i}}{((N_{\text{data}_i}+1)\cdot T_{\text{transmit}_i}) + (\frac{T_{\text{acc}_i}}{(N_{\text{data}_i}+1)})}. \quad (4.9)$$

Finally, the total amount of data acquired in $\mathbf{T}_i$ in a unit time is derived by $R_{\text{sense}_i} \cdot N_{\text{mov}_i}$. Consequently, the sensing amount in $\mathbf{T}_i$, A_{sense_i} is represented by the following formula:

$$\begin{aligned} A_{\text{sense}_i} &= R_{\text{sense}_i} \cdot \frac{N_{\text{mov}_i}}{|\mathbf{T}_i|}. \\ &= \frac{N_{\text{mov}_i}}{|\mathbf{T}_i| \cdot T_{\text{sense}_i}} \cdot \left(1 - \frac{R_{\text{route}_i} \cdot N_{\text{req}_i}}{N_{\text{mov}_i}}\right). \end{aligned} \quad (4.10)$$

Strategy to deploy mobile nodes

Using the result of the analysis, DATFM/DA deploys mobile nodes to each territory. First, DATFM/DA deploys one mobile node to each territory. This is because, when no mobile nodes are deployed in a territory, the territory cannot be monitored. After that, DATFM/DA deploys remaining mobile nodes one by one according to the requirement specified by the application. In this section, we deploy mobile nodes by the following steps assuming an application that monitors the whole region uniformly:

1. Calculate A_{sense} of each territory by using formula (4.10).
2. Add one mobile node to the territory with the lowest A_{sense} among all territories.
3. Repeat step 2 until all mobile nodes are deployed.

By doing so, DATFM/DA can achieve higher sensing amount even in a territory where data transmissions frequently occur.

4.2.4 Validation of the Analysis

In order to verify the validity of the analysis, we conduct the simulation experiments. In the experiments, we compare A_{sense} in each territory with that calculated in the analysis. The simulation environment is basically same as that

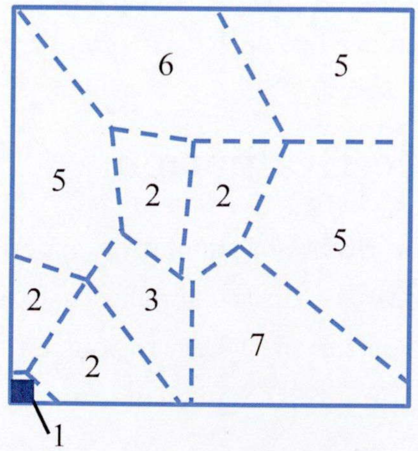
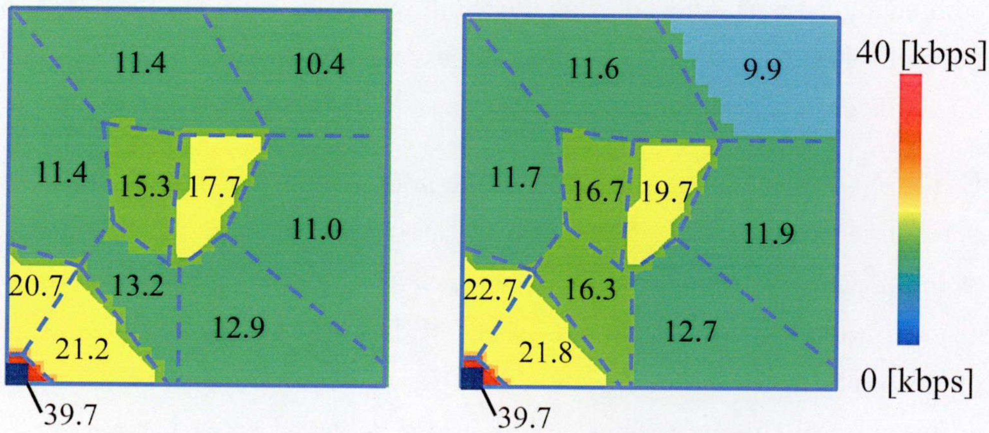


Figure 4.3: Number of mobile nodes.



(a) Analysis.

(b) Simulation.

Figure 4.4: A_{sense} in each territory.

in Section 3.3.1. There are 11 fixed nodes (including the sink node) which are deployed randomly in the region. On the other hand, 40 mobile nodes are deployed in each territory according to the strategy in Section 4.2.3. Figure 4.3 shows the divided territories and the number of mobile nodes deployed to each territory.

Figure 4.4 shows the comparison of A_{sense}. From this result, we can see that the simulation result shows the same tendency as the result of the analysis. This indicates that the result of the analysis appropriately works for determining the number of mobile nodes deployed to each territory.

4.3 Performance Evaluation

In this section, we show the results of simulation experiments regarding performance evaluation of DATFM/DA. In the simulation experiments, we assume an application that aims to monitor the whole region uniformly. We compare the performances of DATFM/DA with the following two methods:

- **DATFM-w/oDA**: A mobile sensor control method we have proposed in Chapter 2. In this method, each mobile node randomly selects the next sensing point (territory to perform sensing) from the whole region. By comparing performance of this method with those in DATFM/DA and DATFM-area, we verify the effectiveness of the statical territory assignment for each mobile node.

- **DATFM-area**: The number of mobile nodes deployed to each territory is determined only based on the size of territories. Specifically, the number of mobile nodes assigned to $\mathbf{T}_i$ is set as $\lfloor N_{\mathrm{mov}} \cdot \mathbf{T}_i / S_{\mathrm{region}} \rfloor$. The behavior of each node is same as that in DATFM/DA. By comparing performance of this method with that in DATFM/DA, we verify the effectiveness of the adjustment of the number of mobile nodes deployed to each territory.

4.3.1 Simulation Environment

The simulation environment is the same as that in Section 3.3.1. The fixed nodes in DATFM/DA and DATFM-area are deployed to the same locations of fixed nodes in DATFM-w/oDA.

In this environment, we run 100 simulations each of which consists of 1[week] changing the locations of fixed nodes, and evaluate the averages of the following three criteria:

- Throughput

  The average amount of data that arrive at the sink node per 1[sec].

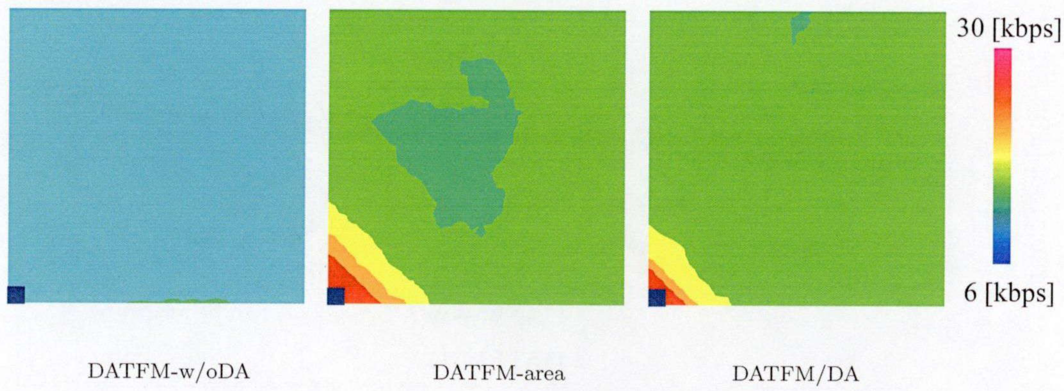


Figure 4.5: Distribution of the average throughput ($N_{\rm fix} = 10, N_{\rm mov} = 40$).

- Average moving distance

  The average of moving distances of all nodes during the simulation period.

- Average delay

  The average of the elapsed time after data are acquired (sensed) until the data arrive at the sink node.

4.3.2 Comparison of Throughput

Since we assume an application that aims to monitor the whole region uniformly in this experiment, we first compared the amount of acquired data in the region among DATFM/DA, DATFM-w/oDA, and DATFM-area, in order to verify the effectiveness of the strategy to deploy mobile nodes in DATFM/DA. In the experiment, we set $N_{\rm fix}$ and $N_{\rm mov}$ as 10 and 40, respectively.

Figure 4.5 shows the distribution of the average throughput in the whole region (The dark blue square shows the sink node). From the results, throughputs in DATFM-area and DATFM/DA become much larger than that in DATFM-w/oDA. This is due to the reduction of moving distance of each mobile node for a sensing operation. On the other hand, we can see the skew of throughput in DATFM/DA and DATFM-area. Especially, the skew of throughput in DATFM-area is larger than that in DATFM/DA. This is because DATFM-area deploys mobile nodes considering only the size of each territory. Thus, the throughput in

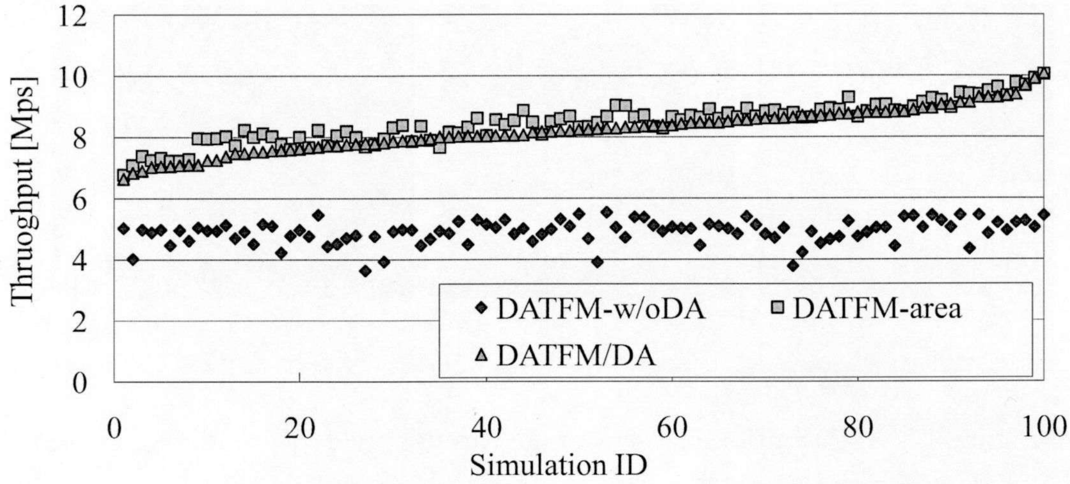


Figure 4.6: The average throughput in each simulation.

the territory where fixed nodes frequently transfer data (e.g. the central part of the region) becomes small.

Figures 4.6, 4.7, and 4.8 show the average, maximum, and minimum throughput among the whole region in each simulation. In these figures, the horizontal axis on all graphs indicates the simulation ID which is sorted by the average throughput in DATFM/DA. The vertical axis indicates the average throughput in Figure 4.6, the maximum throughput in Figure 4.7, and the minimum throughput in Figure 4.8.

From these results, we can see that the throughput in DATFM/DA is always larger than that in DATFM-w/oDA. The maximum throughput in DATFM-area is larger than (sometimes same as) that in DATFM/DA. However, the minimum throughput in DATFM-area is smaller than that in DATFM/DA, and is sometimes smaller than that in DATFM-w/oDA. On the other hand, DATFM/DA achieves the largest minimum throughput while achieving almost the same average throughput as that in DATFM-area.

This result shows that DATFM/DA can achieve effective sensing and data transfer in the whole region.

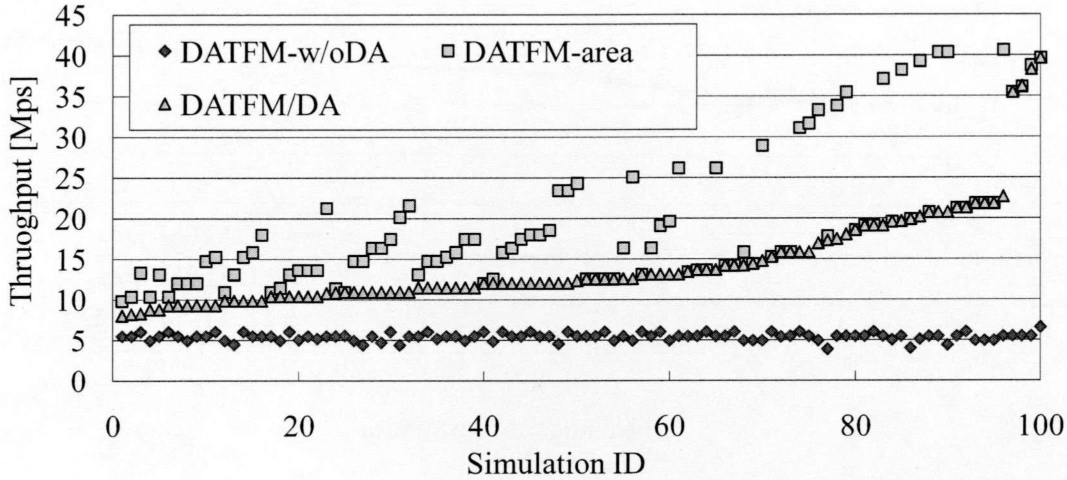


Figure 4.7: The maximum throughput in each simulation.

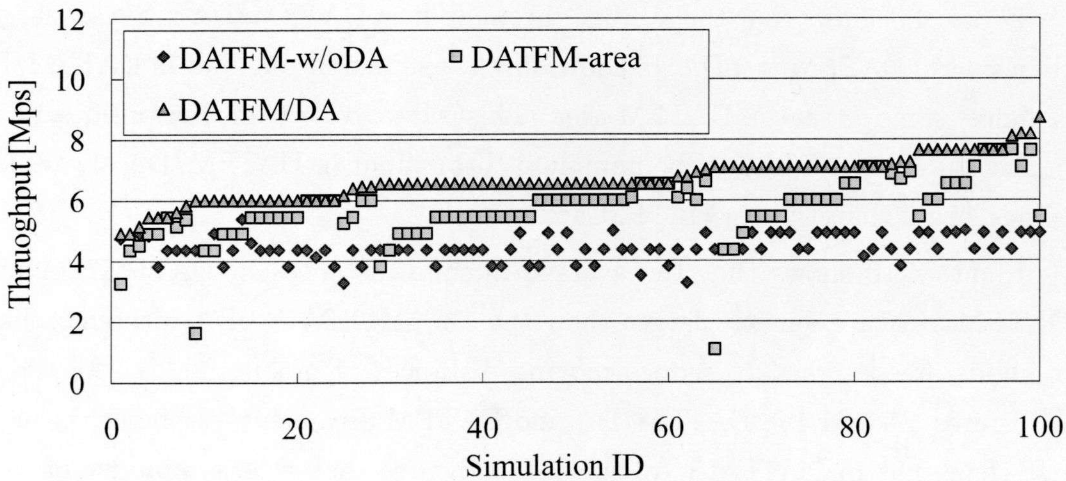


Figure 4.8: The minimum throughput in each simulation.

4.3.3 Effects of the Number of Fixed Nodes

Figures 4.9, 4.10, and 4.12 show the simulation results when changing parameter N_{fix}. The horizontal axis on all graphs indicates the number of fixed nodes N_{fix}. The vertical axis indicates throughput in Figure 4.9, average moving distance in

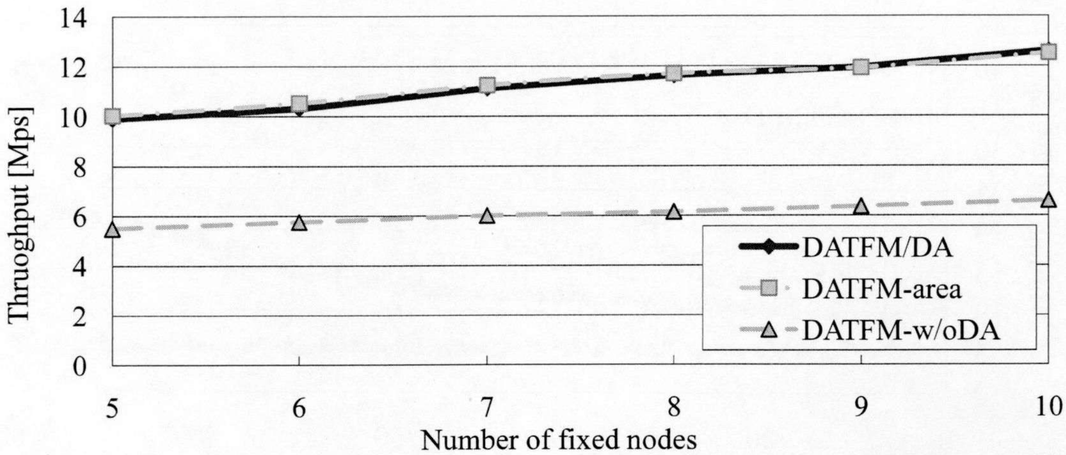


Figure 4.9: N_{fix} and throughput.

Figure 4.10, and average delay in Figure 4.12.

Figure 4.9 shows that the average throughput in DATFM/DA is always larger than that in DATFM-w/oDA. In addition, the average throughput in DATFM/DA is almost same as that in DATFM-area. This is due to the same reasons described in Section 4.3.2. Note that the minimum throughput in DATFM/DA always becomes larger than that in DATFM-area.

Figure 4.10 shows that the average moving distances in DATFM/DA and DATFM-area are slightly larger than that in DATFM-w/oDA although these methods are designed to reduce moving distances of mobile nodes. As shown in Figures 4.5 and 4.9, DATFM/DA and DATFM-area achieves higher throughput than that in DATFM-w/oDA. This indicates that a large amount of data are quickly accumulated in each fixed node, and thus, data transmissions frequently occur. Thus, the average moving distances in these methods become much larger than that in DATFM-w/oDA due to the frequent data transmissions. In order to validate this discussion, we illustrate the detail of the moving distance in Figure 4.11. In this figure, the horizontal axis on all graphs indicates the number of fixed nodes N_{fix}. The vertical axis indicates the average moving distance for sensing operations in Figure 4.11(a), and that for data transmissions in Figure 4.11(b). Figure 4.11(a) shows that the moving distances for sens-

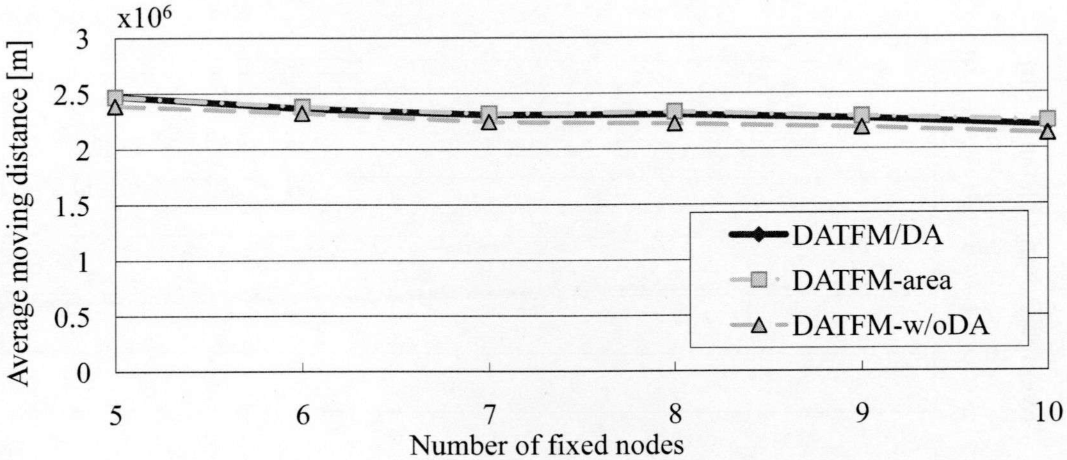


Figure 4.10: N_{fix} and average moving distance.

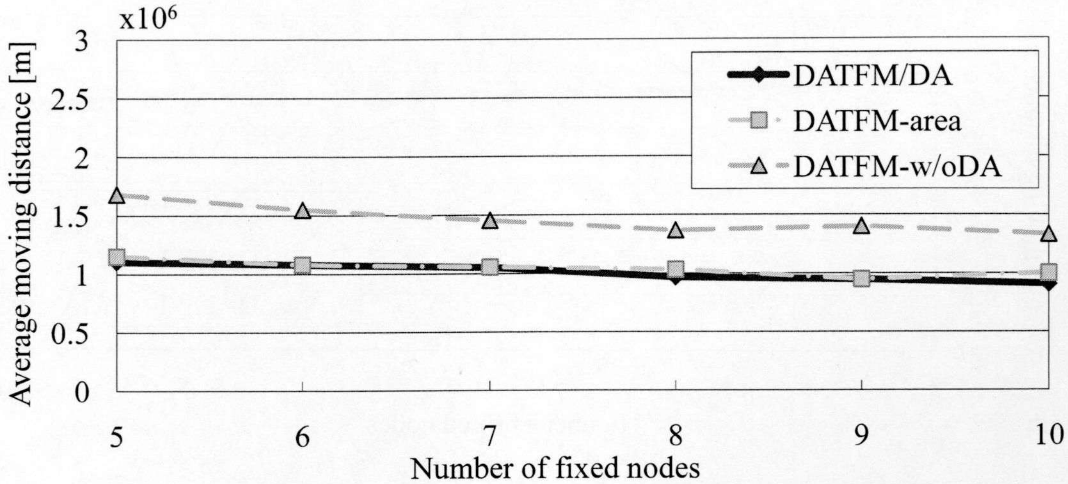
ing operations in DATFM/DA and DATFM-area are always smaller than that in DATFM-w/oDA. This results in the increase of throughput. On the other hand, Figure 4.11(b) shows that the moving distances for data transmissions in DATFM/DA and DATFM-area are always larger than that in DATFM-w/oDA. This results in the increase of moving distance.

Figure 4.12 shows that the delays in DATFM/DA and DATFM-area are always smaller than that in DATFM-w/oDA. This is due to the decrease in time elapsed for accumulating data at each fixed node. As the number of fixed nodes increases, delays in all methods get larger. This is because the larger the number of fixed nodes is, the lower the frequency for each fixed node to connect with mobile nodes. As a result, time for accumulating data increases at each fixed node.

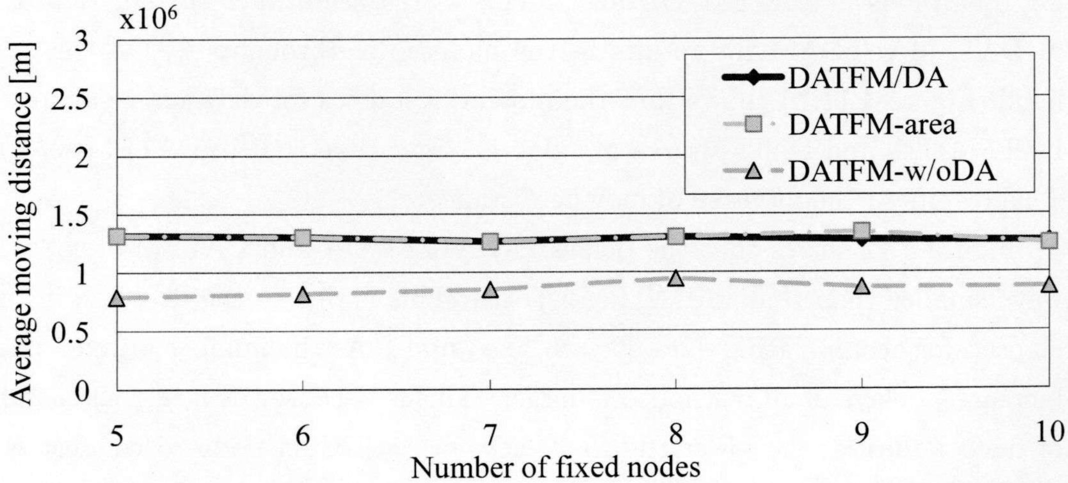
4.3.4 Effects of the Number of Mobile Nodes

Figures 4.13, 4.14, and 4.16 show the simulation result when changing parameter N_{mov}. The horizontal axis on all graphs indicates the number of fixed nodes N_{mov}. The vertical axis indicates throughput in Figure 4.13, average moving distance in Figure 4.14, and average delay in Figure 4.16.

Figure 4.13 shows that the average throughputs in DATFM/DA and DATFM-



(a) Moving distance for sensing operations.



(b) Moving distance for data transmissions.

Figure 4.11: N_{fix} and details of average moving distance.

area are always larger than that in DATFM-w/oDA. This is due to the same reason as that in Figure 4.9. Moreover, the average throughput in all methods increases as the number of mobile nodes increases. This is obvious because more

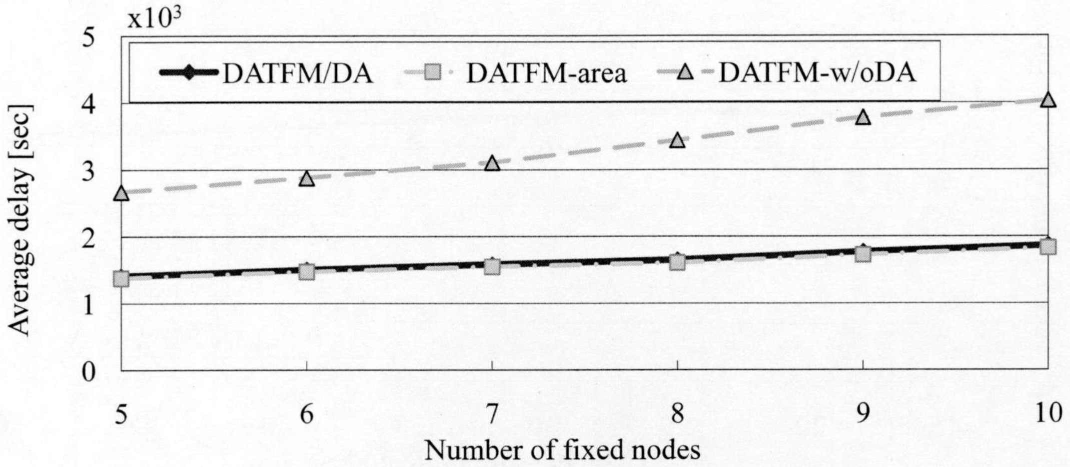


Figure 4.12: N_{fix} and average delay.

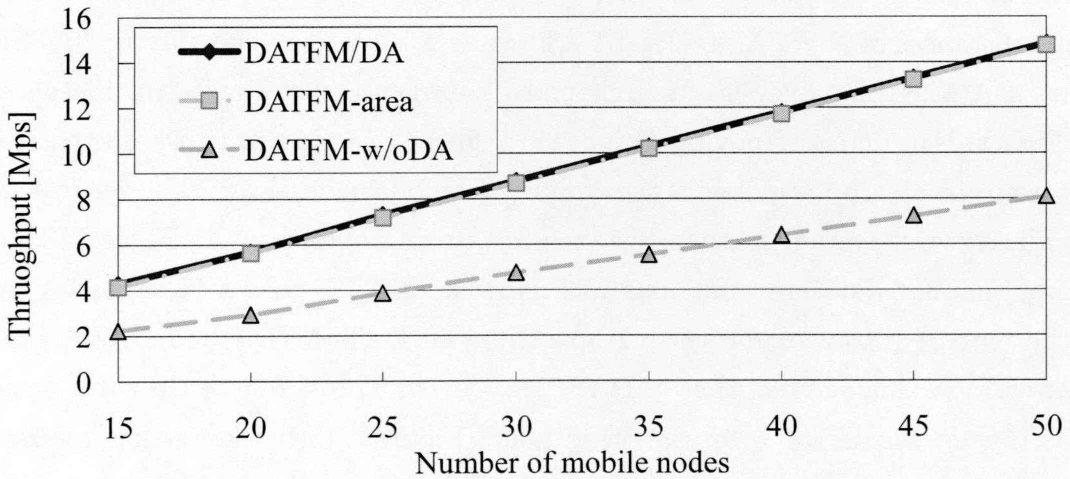


Figure 4.13: N_{mov} and throughput.

sensing operations are performed with a larger number of mobile nodes. In addition, it becomes easier for each fixed node to collect mobile nodes for transferring the accumulated data.

Figure 4.14 shows that the moving distances in DATFM/DA and DATFM-area are larger than that in DATFM-w/oDA when the number of mobile nodes is

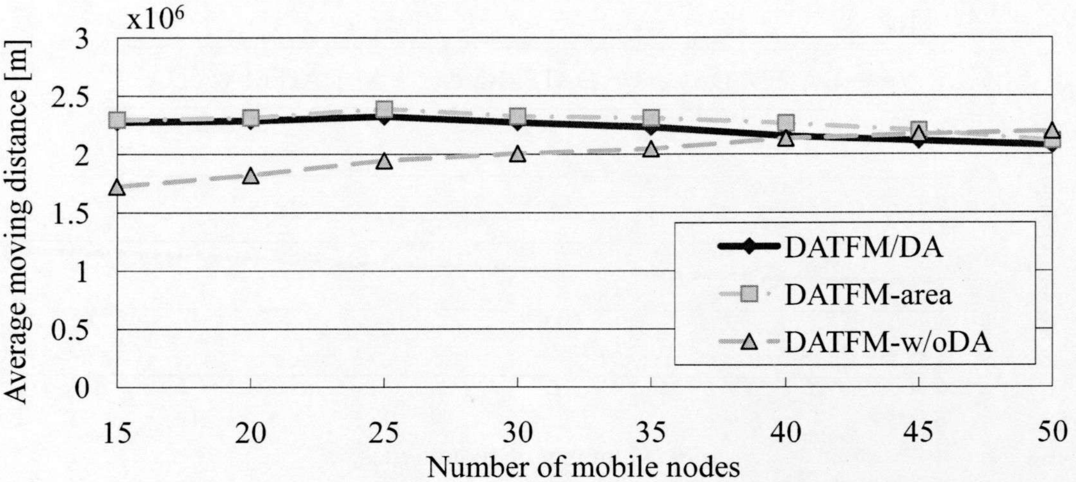
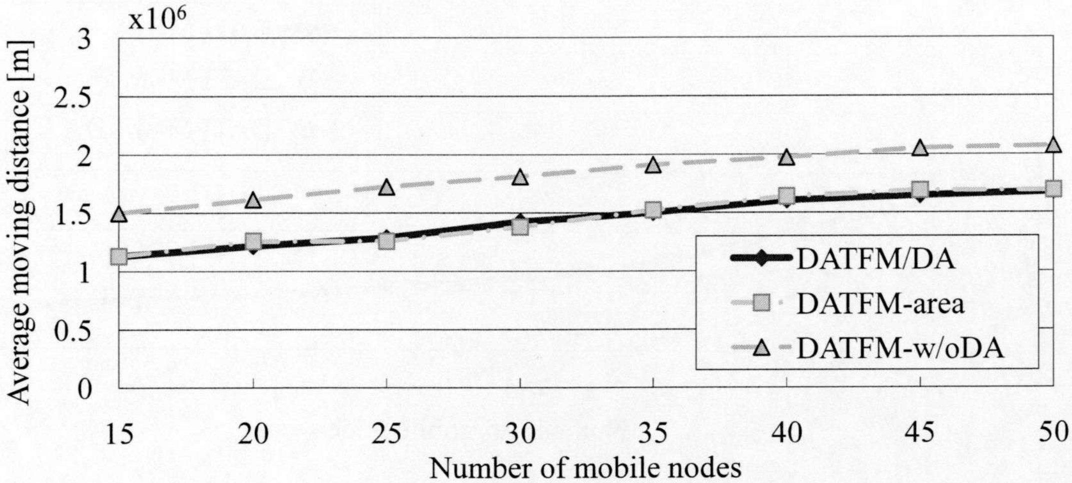


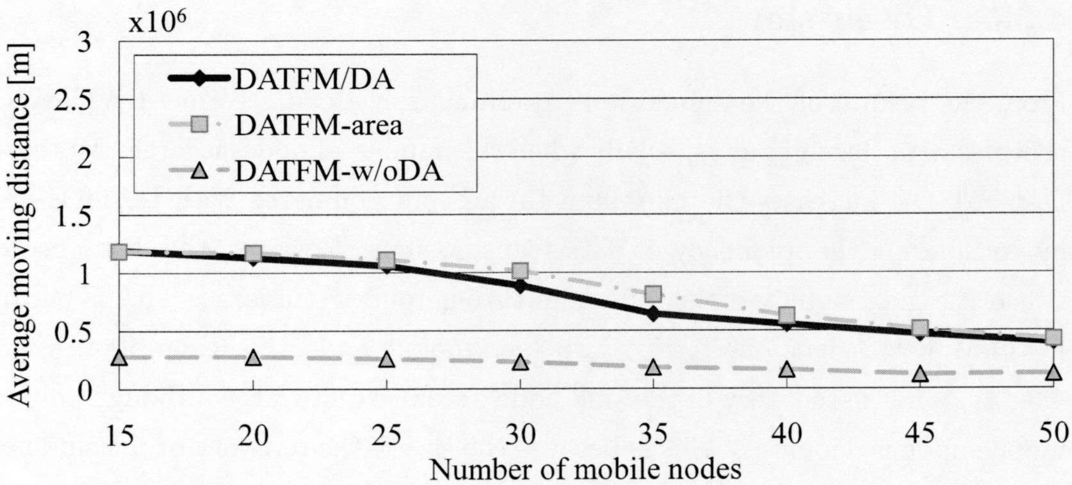
Figure 4.14: N_{mov} and average moving distance.

small. This is due to the same reason as that in Figure 4.10. However, the moving distances in DATFM/DA and DATFM-area is smaller than that in DATFM-w/oDA when the number of mobile nodes is large. To investigate the reason, we illustrate the detail of moving distance in Figures 4.15. In this figure, the horizontal axis on all graphs indicates the number of mobile nodes N_{mov}. The vertical axis indicates the average moving distance for sensing operations in Figure 4.15(a), and that for data transmissions in Figure 4.15(b). Figure 4.15(a) shows that the moving distances for sensing operations in DATFM/DA and DATFM-area are always smaller than that in DATFM-w/oDA. This is due to the same reason as that in Fig.4.11(a). On the other hand, Figure 4.15(b) shows that the moving distances for data transmissions in DATFM/DA and DATFM-area decrease more drastically than that in DATFM-w/oDA when the number of mobile nodes increases. This is because it becomes easier for each fixed node to construct a complete communication route to its destination when there are many mobile nodes in its territory. In this case, the moving distance for train transmissions becomes smaller.

Figure 4.16 shows that the delays in DATFM/DA and DATFM-area are always smaller than that in DATFM-w/oDA. This is due to the same reason as that in Figure 4.12. Moreover, delays in all methods decrease as the number of



(a) Moving distance for sensing operations.



(b) Moving distance for data transmissions.

Figure 4.15: N_{mov} and details of average moving distance.

mobile nodes increases. This is because time elapsed for accumulating data at each fixed node decreases.

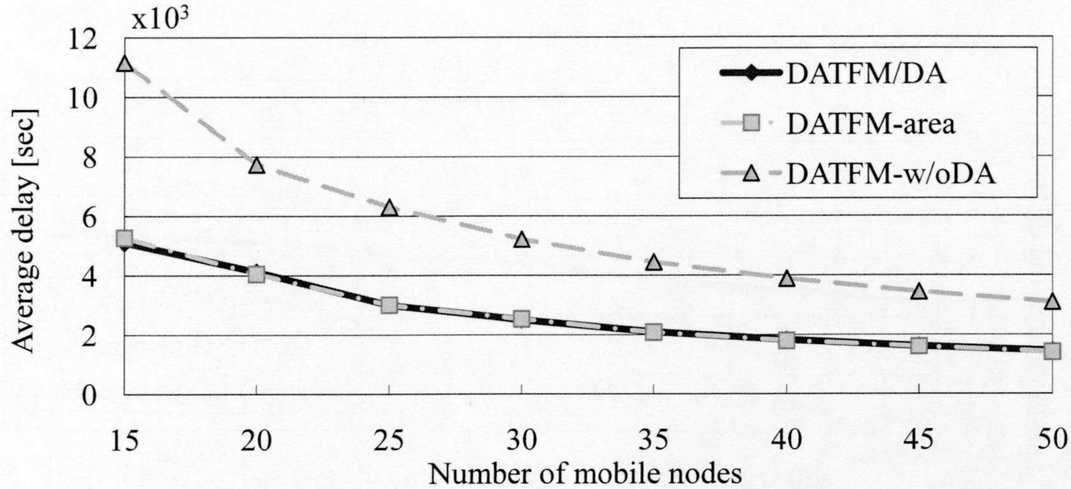


Figure 4.16: N_{mov} and average delay.

4.3.5 Discussion

From the results of the simulation experiments, we can see that DATFM/DA achieves high throughput especially when the number of nodes is large. Moreover, DATFM/DA increases the minimum throughput compared with DATFM-area, by considering the frequency of data transmissions. However, the difference between the maximum and the minimum throughput is still large. This is because DATFM/DA assigns one territory to each mobile node. For example in Figure 4.3, A_{sense} in the territory of the sink node becomes quite large although only one mobile node is deployed. This is because the size of the territory of the sink node is quite smaller than those of other territories. Such a skew of A_{sense} cannot be avoided especially when the number of mobile nodes is small. Thus, in order to apply our method to some applications which need to acquire data in the whole region uniformly, it is necessary to further extend our method. For example, it is effective to assign multiple adjacent territories to a mobile node when the difference of A_{sense} between adjacent territories is large.

In addition, some applications may specify the required (minimum) throughput in the entire region. In such applications, it is not necessary for all the mobile nodes to perform sensing operations. For these applications, we can extend our strategy to control each mobile node. For example, it might be good to deploy some mobile nodes only for moving for data transmission.

4.4 Integration of DATFM/DF and DATFM/DA

In this section, we present the integration of DATFM/DF and DATFM/DA. We call the integrated method DATFM/DF-DA (DATFM with deliberate Deployment of Fixed nodes and Deployment Adjusting).

4.4.1 Design

DATFM/DF-DA first deploys fixed nodes based on the strategy in DATFM/DF. After that, DATFM/DF-DA deploys mobile nodes to each territory based on DATFM/DA. Each mobile node in DATFM/DF-DA behaves as in DATFM/DA. Specifically, each mobile node performs sensing operations only in its assigned territory.

Similar to DATFM/DF, we analyze the sensing rate, R_{sense} of DATFM/DF-DA, in order to define the strategy to determine the locations of fixed nodes. However, unlike DATFM/DF, since each mobile node in DATFM/DF-DA moves only in its assigned territory, it becomes unnecessary to consider the effects of the distance of moving paths for sensing operations. Thus, Guideline 2 (*The total length of moving paths of mobile nodes between all pairs of fixed nodes should be small*) becomes useless in DATFM/DF-DA.

In addition, we consider the effects of data that a fixed node receives from other fixed nodes, as discussed in Section 4.2.3, and derive another guideline. From Formula (4.3), we can see that R_{sense_i} depends on the frequency of data transmission in territory i. Here, in the data transmission from fixed node i to j, the distance between fixed nodes i and j, $|\mathbf{L}_i - \mathbf{L}_j|$, is expressed as $(N_{\text{req}_i}+1)\cdot r_{\text{com}}$. Therefore, from Formula (4.9), we can derive the following formula:

$$\begin{aligned} R_{\text{route}_i} &= \frac{(N_{\text{data}_i}+1)\cdot 2\cdot (N_{\text{req}_i}-1)\cdot r_{\text{com}}}{((N_{\text{data}_i}+1)\cdot 2\cdot (N_{\text{req}_i}-1)\cdot r_{\text{com}}) + (\frac{2\cdot r_{\text{com}}\cdot \frac{Th}{D}}{(N_{\text{data}_i}+1)})} \\ &\approx \frac{(N_{\text{data}_i}+1)^2\cdot N_{\text{req}_i}}{(N_{\text{data}_i}+1)^2\cdot N_{\text{req}_i} + \frac{Th}{D}}. \end{aligned} \tag{4.11}$$

Thus,

$$R_{\text{route}_i} = \frac{(N_{\text{data}_i}+1)^2 \cdot N_{\text{req}_i}}{(N_{\text{data}_i}+1)^2 \cdot N_{\text{req}_i} + \frac{Th}{D}} = 1 - \frac{\frac{Th}{D}}{(N_{\text{data}_i}+1)^2 \cdot N_{\text{req}_i} + \frac{Th}{D}}. \quad (4.12)$$

From this formula, we can see that the frequency of data transmissions in $\mathbf{T}_i$, R_{route_i}, decreases when N_{data_i} becomes smaller. In other words, mobile nodes in $\mathbf{T}_i$ can perform more sensing operations when N_{data_i} is small. As the result, we can derive the following guideline:

Guideline 5: N_{data_i} *should be small for all fixed nodes.*

4.4.2 Strategy for Deploying Fixed Nodes

Based on the above discussion, we devise the following strategy for deploying fixed nodes in DATFM/DF-DA.

Similar to DATFM/DF, the strategy for deploying fixed nodes in DATFM/DF-DA is divided into two steps, determining the pattern of deployment, and adjusting the distance between all pairs of fixed nodes in the determined pattern.

Determining pattern of deployment

First, DATFM/DF-DA determines the pattern of deployment based on Guidelines 1 and 5. Here, Guideline 5 only considers to improve the efficiency of sensing. On the other hand, Guideline 1 considers not only to improve the efficiency of data transfer but also to reduce the moving distance for sensing operations. Therefore, DATFM/DF-DA first derives multiple candidates of deployment patterns considering Guideline 1 and selects the one considering Guideline 5.

Here, the procedures 1 to 3 in this step are the same as those in DATFM/DF since they derived by considering Guideline 1.

1. Draw the line L from the sink node to the farthest point, and calculate the ideal length of the communication route l by the following formulae:

$$\frac{S_{\text{region}}}{N_{\text{fix}}} = \pi \cdot \left(\frac{l}{2}\right)^2$$

$$l = 2 \cdot \sqrt{\frac{S_{\text{region}}}{\pi \cdot N_{\text{fix}}}}. \quad (4.13)$$

Next, calculate N_{hop} in order to deploy a fixed node at the farthest point from the sink node when the distance between fixed nodes is set as l.

2. Divide the region into $N_{\text{hop}} + 1$ zones. Then, calculate the number of fixed node deployed in zone i, N_{zone_i}, according to the following formula:

$$N_{\text{zone}_i} = \lceil N_{\text{fix}} \cdot \frac{|\mathbf{Z}_i|}{S_{\text{region}}} \rceil. \tag{4.14}$$

3. Make all patterns of deployment that satisfy the following conditions:

   - The distance between adjacent fixed nodes is uniform.
   - The required number of intermediate fixed nodes for each fixed node to transfer its holding data to the sink node (we define this value as the *hop-count*) is equal to or more than N_{hop}.
   - The number of fixed nodes in each zone i is equal to or less than N_{zone_i} calculated in the previous step.

   For example, when the number of fixed nodes (except for the sink node) is five, five candidates are made as shown in Figure 4.17.

4′. Calculate the total of N_{data_i} for all fixed nodes. Then, select the pattern in which the total of N_{data_i} is minimum among all patterns as the initial deployment pattern (from Guideline 5). For example in Figure 4.17, Figure 4.17(a) is adopted as the initial pattern.

Adjusting distances between fixed nodes

Next, DATFM/DF-DA adjusts the distances between all pairs of fixed nodes in the determined pattern. Since this step is based on Guidelines 3 and 4, the procedures in this step are the same as those in DATFM/DF.

4.4.3 Performance Evaluation of DATFM/DF-DA

In this section, we show the results of simulation experiments regarding performance evaluation of DATFM/DF-DA. In the simulation experiments, we compare the performances of DATFM/DF-DA with DATFM/DA, DATFM/DF and DATFM. The simulation environment is the same as that in Section 4.3.1.

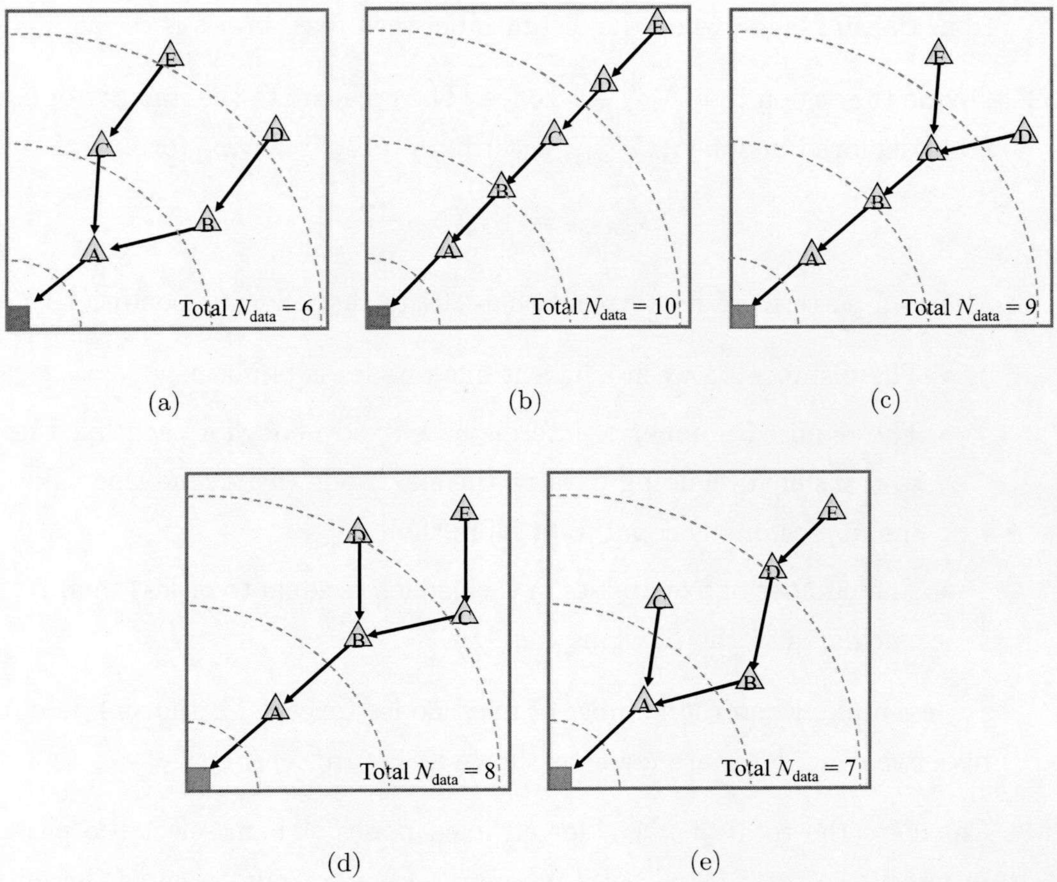


Figure 4.17: Deployment patterns.

In this environment, we evaluate the averages of the following three criteria:

- Throughput
  The average amount of data that arrive at the sink node per 1[sec].
- Average moving distance
  The average of moving distances of all nodes during the simulation period.
- Average delay
  The average of the elapsed time after data are acquired (sensed) until the data arrive at the sink node.

Here, in DATFM and DATFM/DA, we ran 100 simulations changing the lo-

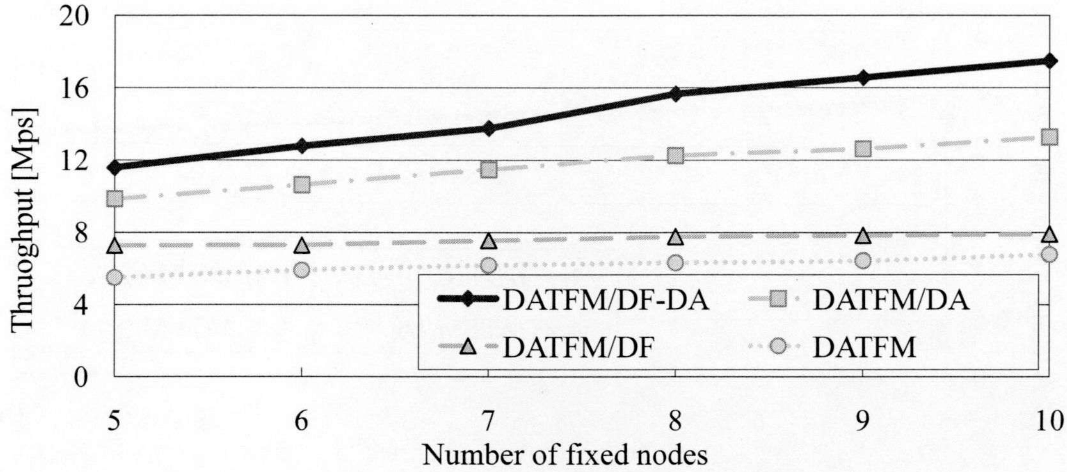


Figure 4.18: N_{fix} and throughput.

cations of fixed nodes which are randomly determined, and evaluated the average of the above criteria. Moreover, the locations of fixed nodes in DATFM are the same as those in DATFM/DA.

Effects of the number of fixed nodes

Figures 4.18, 4.19, and 4.20 show the simulation results when changing parameter N_{fix}. The horizontal axis on all graphs indicates the number of fixed nodes N_{fix}. The vertical axis indicates throughput in Figure 4.18, average moving distance in Figure 4.19, and average delay in Figure 4.20.

Figure 4.18 shows that the throughput in DATFM/DF-DA is always higher than those in other methods. This result shows that the combination of DATFM/DF and DATFM/DA is effective to further improve the efficiencies of sensing and data transfer.

Figure 4.19 shows that the moving distance in DATFM/DF-DA is slightly larger than that in DATFM. This is because mobile nodes in DATFM/DF-DA behave as in DATFM/DA. However, the increase in the moving distance in DATFM/DF-DA always smaller than that in DATFM/DA. This is because the strategy in DATFM/DF-DA aims to reduce the average distance between fixed node and the sensing points.

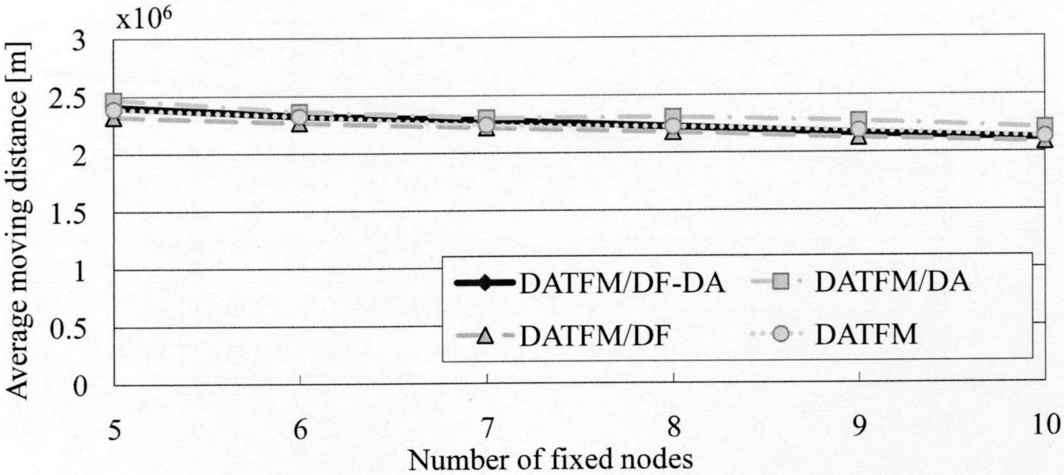


Figure 4.19: N_{fix} and average moving distance.

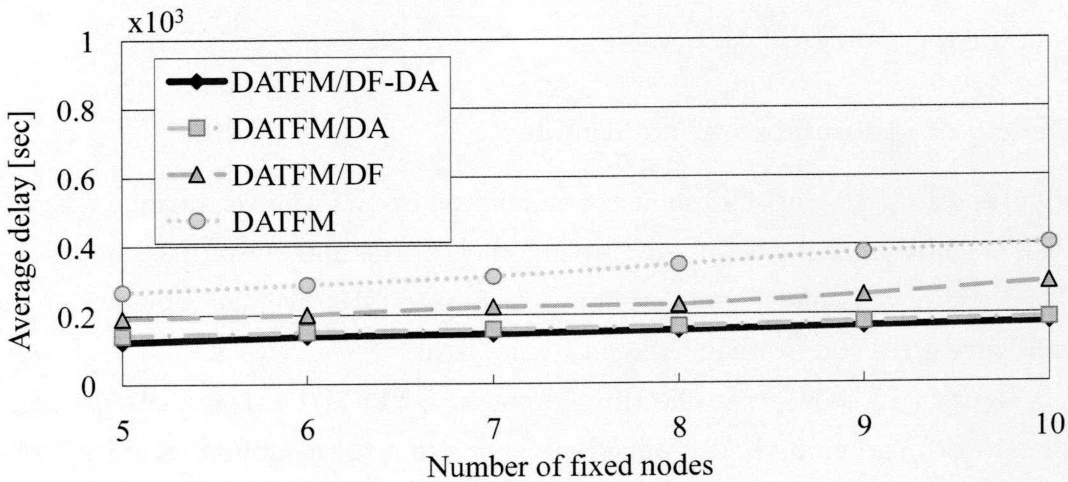


Figure 4.20: N_{fix} and average delay.

Figure 4.20 shows that the delay in DATFM/DF-DA is always smaller than those in other methods. This result indicates that the combination of DATFM/DF and DATFM/DA is also effective to reduce the moving distance of mobile nodes.

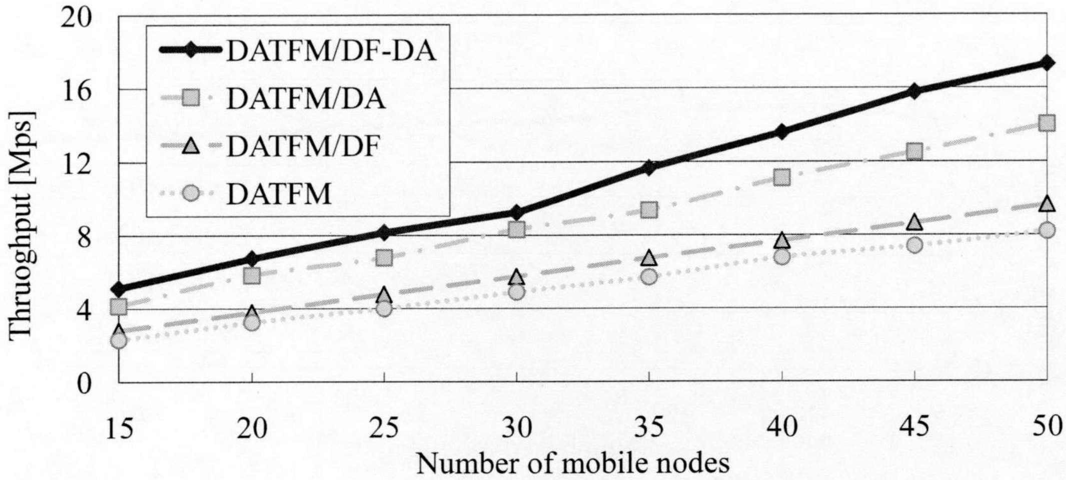


Figure 4.21: N_{mov} and throughput.

Effects of the Number of Mobile Nodes

Figures 4.21, 4.22, and 4.23 show the simulation results when changing parameter N_{mov}. The horizontal axis on all graphs indicates the number of fixed nodes N_{mov}. The vertical axis indicates throughput in Figure 4.21, average moving distance in Figure 4.22, and average delay in Figure 4.23.

Figure 4.21 shows that the throughput in DATFM/DF-DA is always higher than those in other methods. This is due to the same reason as that in Figure 4.18. Moreover, the differences of throughput between DATFM/DF-DA and other methods increase as the number of mobile nodes gets larger. This is because in DATFM/DF-DA, each mobile sensor can acquire data more frequently than other methods, due to the combination of DATFM/DF and DATFM/DA. Specifically, when the number of mobile nodes gets larger, the fixed nodes can finish data transmission earlier, and thus, mobile nodes can concentrate more on sensing operations. This makes the advantage of acquiring data in DATFM/DF-DA larger.

Figure 4.22 shows that the moving distance in DATFM/DF-DA shows similar tendency to that in DATFM/DA. In addition, the moving distance in DATFM/DF-DA is smaller than that in DATFM/DA. This is due to the same reason as that

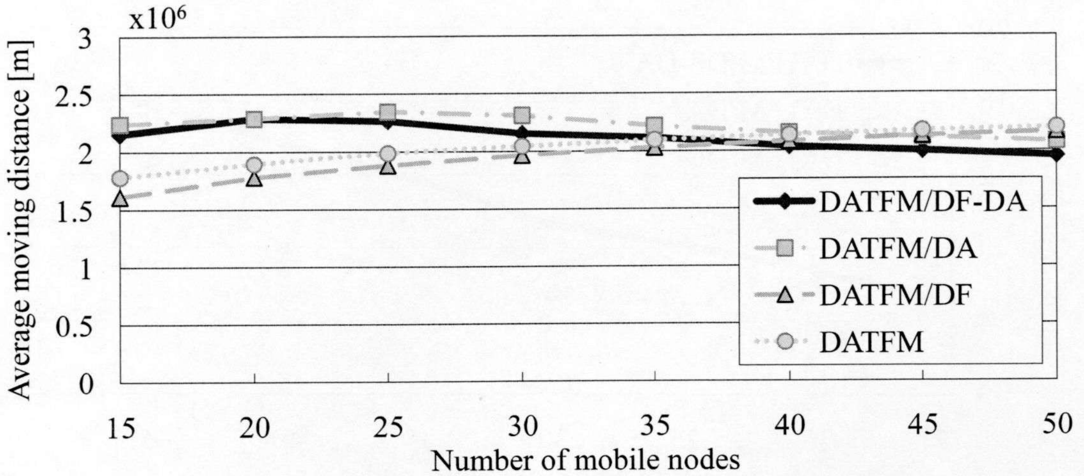


Figure 4.22: N_{mov} and average moving distance.

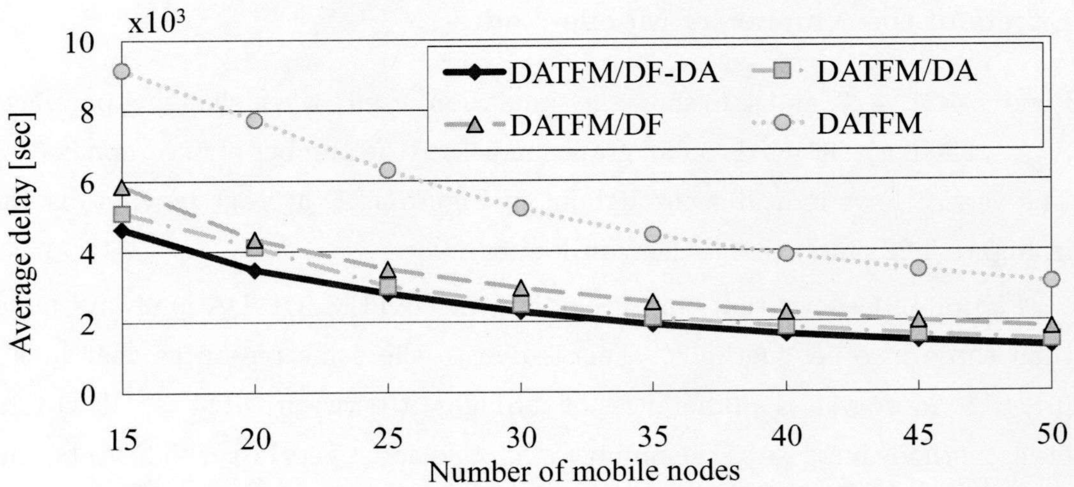


Figure 4.23: N_{mov} and average delay.

in Figure 4.19.

Figure 4.23 shows that the delay in DATFM/DF-DA is always smaller than those in other methods. This is due to the decrease in time elapsed for accumulating data at each fixed node by strategically deploying fixed nodes and assigning a territory to each mobile node.

4.4.4 Discussion of Cost Constrains

Our proposed methods uses two types of mobile sensors. In a real environment, the cost of preparing each of fixed and mobile nodes should be different due to the costs of modules with which each node is equipped. For example, in some environments where the cost of high-density storage becomes large [7], the cost of a fixed node becomes higher than that of a mobile node. In addition, in order to provide reliable communication with multiple mobile nodes, fixed nodes should be equipped with high-performing communication device. This also increases the cost of a fixed node. On the other hand, in some other environments where high-performing actuator for moving is required (e.g. underwater [87]), the cost of a mobile node becomes higher than that of a fixed node. Moreover, the cost of deploying fixed nodes also depends on the environment. For example, in planetary exploration, mobile sensors can be deployed from the air (e.g. spacecraft drops mobile sensors before landing). In such an environment, the cost of deploying a fixed node becomes relatively small. On the other hand, in a polluted plant or building, fixed nodes cannot be deployed from the air. In this case, fixed nodes have to be equipped with a moving facility. Thus, the cost of deploying a fixed node becomes very large. Therefore, when the total cost for the entire system is limited, we should optimize the deployment of fixed and mobile nodes, i.e., the ratio of the numbers of fixed and mobile nodes. Here, we define the optimal ratio as the ratio of the number of mobile nodes to that of fixed nodes when the throughput becomes the highest under the given cost constraint.

In this subsection, assuming DATFM/DF-DA, we investigate the optimal ratio through simulation experiments. In the experiments, we prepared four scenarios in each of which the cost for deploying a fixed node is a) half of, b) same as, c) two times of, and d) five times of, that for deploying a mobile node. In addition, we assume several cost constraints for each scenario. Under the given cost constraint, we derived all patterns of the numbers of fixed and mobile nodes, and evaluated the throughput for each pattern in the environment in the same way as that in Section 4.3. Among them, we plotted the pattern of the ratio that achieved the highest throughput.

Figure 4.24 shows the result. The horizontal axis indicates the number of

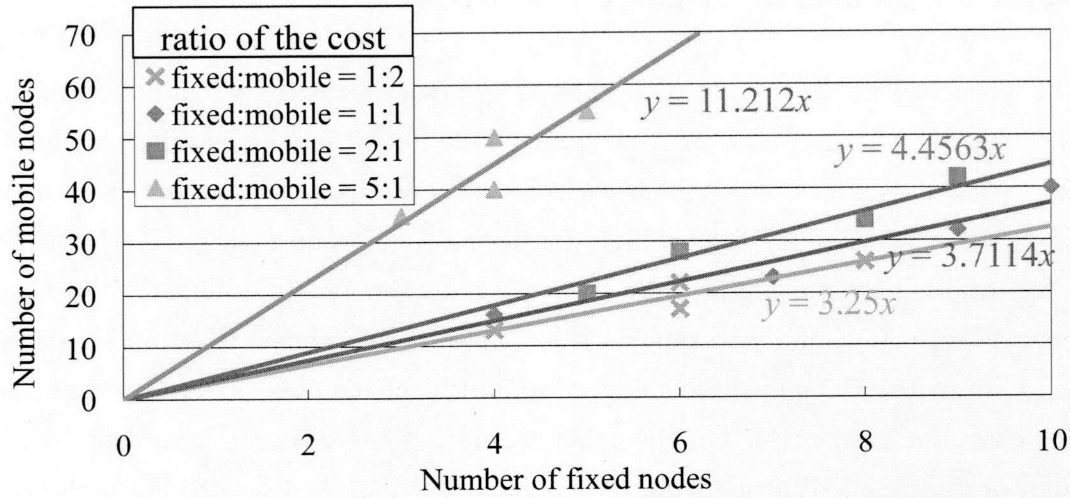


Figure 4.24: The optimal ratio.

fixed nodes. The vertical axis indicates the number of mobile nodes. From this result, we can derive the optimal ratio in each scenario. For example, when the cost for deploying a fixed node is same as that for deploying a mobile node, the optimal ratio becomes 3.71, i.e. N_{mov} should be $3.71 \times N_{\mathrm{fix}}$. In addition, we can see that the optimal ratio increases when the cost for deploying fixed node gets larger. This result indicates that we can decide the numbers of fixed and mobile nodes when the total cost constraint and the ratio of cost for deploying each node are given.

4.5 Conclusion

In this chapter, we proposed DATFM/DA, which is an extension of DATFM. In order to improve the efficiency of sensing, DATFM/DA statically assigns a territory to each mobile node, i.e., each mobile node performs sensing operations only in its assigned territory. By doing so, the moving distance of each mobile node can be suppressed. In addition, DATFM/DA determines the number of mobile nodes deployed to each territory in order to suppress the skew of sensing amount in the whole region. We also conducted the simulation experiments to evaluate the performance of DATFM/DA. The results show that DATFM/DA improves the performance by statically assigning a territory to each mobile node, and adjusting the number of mobile nodes deployed to each territory.

Moreover, we proposed DATFM/DF-DA as an integration of DATFM/DF and DATFM/DA. We also conducted the simulation experiments to verify that the integrated method further improves the efficiencies of sensing and data transfer. Furthermore, we discussed the cost for deploying nodes and derived that the optimal ratio can be determined when the total cost constraint and the ratio of cost for deploying each of fixed and mobile nodes are given.

Chapter 5

Conclusion

5.1 Summary of Thesis

In this thesis, we have discussed about mobile sensor control methods in sparse sensor networks.

In Chapter 1, we presented the importance of mobile sensor networks and made clear research issues for achieving efficient sensing and data transfer. In addition, we introduced the related work and discussed the problems of the conventional studies. We also presented the fundamental ideas of our proposed methods.

In Chapter 2, we proposed a mobile sensor control method DATFM for sparse sensor networks, which is the fundamental method proposed in this thesis. DATFM uses two types of sensor nodes, fixed node and mobile node, and accumulates the acquired data on a fixed node before transferring them to the sink node. In addition, DATFM defines the moving strategy of mobile nodes, that enables fixed nodes to connect with mobile nodes frequently, and to collect mobile nodes for transferring the accumulated data toward the sink node. Moreover, DATFM defines another data transfer strategy using the cooperative movement and communication of mobile nodes in order to transfer data even when a communication route cannot be constructed. We also conducted the simulation experiments to evaluate the performance of DATFM. The results showed that DATFM improves the efficiencies of sensing and data transfer compared with conventional methods.

In Chapter 3, assuming applications where fixed nodes can be deployed any-where in the region, we proposed an extended method of DATFM, named DATFM/DF. In this method, we first analyzed the effects of the locations of fixed nodes on the performance. Based on the result of the analysis, DATFM/DF strategically deploys fixed nodes to further improve the performance. We also conducted the simulation experiments to evaluate the performance of DATFM/DF. The results showed that DATFM/DF improves the performance of DATFM by introducing the deployment strategy.

In Chapter 4, assuming applications where mobile nodes with higher durability can be deployed, we proposed another extended method of DATFM, named DATFM/DA. DATFM/DA statically assigns a territory for each node in order to decrease the moving distance of mobile nodes for sensing operations. In addition, we analyzed the effects of the number of mobile nodes deployed in each territory on the performance. Based on the result of the analysis, DATFM/DA adjusts the number of mobile nodes deployed in each territory. We also conducted the simulation experiments to evaluate the performance of DATFM/DA. The results showed that DATFM/DA improves the performance compared with DATFM. In this chapter, we also discussed the integration of DATFM/DF and DATFM/DA. On integrating these two methods, we modified the deployment strategy defined in DATFM/DF considering the moving strategy of mobile nodes in DATFM/DA. We also conducted the simulation experiments to evaluate the performance of the integrated method, named DATFM/DF-DA. The results showed that DATFM/DF-DA further improves the performances of sensing and data transfer. Furthermore, we discussed the cost for deploying fixed and mobile nodes.

In summary, our proposed mobile sensor control methods successfully achieve efficient sensing and data transfer in sparse sensor networks. This achievement is expected to enlarge the availability of wireless sensor network systems. Therefore, we believe our work makes a contribution to the development of new services of wireless sensor networks in the future.

5.2 Future Work

Through this thesis work, we found the following issues open to our future work.

5.2.1 Handling Node Failures

In this thesis, we do not consider the failure of mobile sensors. However, in hazardous environments, failures of mobile sensors do occur due to the battery exhaustion or physical damages. Thus, we plan to extend our methods to handle failures of nodes.

In our methods, the failure of a fixed node has a great impact on the system performance. This is because data acquired mobile nodes are accumulated and transferred by fixed nodes. If a fixed node disappears, neither data accumulation nor data transfer cannot be performed. To handle a disappearance of a fixed node, some mechanisms are needed to detect a disappearance of a node and share the information on the disappearance among nodes.

On the other hand, DATFM/DA deploys mobile nodes to each territory in order to satisfy the required amount of acquired data (i.e. A_{sense_i}). Thus, if failures of mobile nodes occur, the system fails to satisfy the requirement. To handle a failure of a mobile node, some mechanisms are needed to detect a failure and autonomously readjust the number of mobile nodes deployed to each territory.

5.2.2 Applying to Real Environments

As described in Section 1.3, we assumed that there are no obstacles in the region and the communication ranges of all nodes are the same. Since those assumptions are basically not true in real environments, the performance of our proposed methods may deteriorate. For example, when there are obstacles in an environment, the moving path of a mobile node between fixed nodes cannot be set as the straight line. In addition, it becomes difficult to conduct a train transmission due to the existence of obstacles between the source and destination nodes. On the other hand, it is well known that radio waves show complicated characteristics according to several effects such as fading and shadowing. Thus, the communication range of each node may change according to these effects.

Therefore, we plan to extend our methods to handle these factors in order for our methods to be applied to real environments. For example, in order to handle the effects of the obstacles, it is necessary to change moving paths of mobile nodes for sensing operations and data transmissions. In addition, in order to handle the complicated characteristics of radio propagation, it is necessary to construct a communication route (or train) considering the quality of radio communication in the environment.

Acknowledgment

Upon the completion of this thesis, I am very thrilled and would like to take this opportunity to express my sincerest gratitude to those who have done their best to offer me assistances to the thesis.

First and foremost, I am deeply indebted to my honors supervisor, Prof. Shojiro Nishio, who not only accepted me as his student but also gave me insightful guidance to my research work and great encouragement to overcome all the obstacles I encountered. This thesis would never have been completed without his unfailingly wise advice and support.

Meanwhile, it is my pleasure and fortunate that Associate Prof. Takahiro Hara becomes my direct advisor, assisting me to overcome the limitation of my knowledge and converting me into a researcher in this novel research field, as well as providing me valuable advice and guidance throughout my undergraduate, master, and doctoral research work.

I would also like to acknowledge the committee members of my thesis, Prof. Toru Fujiwara, Prof. Koh Hosoda, and Prof. Norihisa Komoda at the Department of Multimedia Engineering of the Graduate School of Information Science and Technology of Osaka University. Their insightful and constructive comments considerably improved the quality of the thesis.

I would like to express my appreciation to Prof. Fumio Kishino at the Department of Human System Interaction of the School of Science and Technology of Kwansei Gakuin University, Prof. Shinji Shimojo at Osaka University (Cybermedia Center) / Executive Researcher at National Institute for Information and Communications Technology, Dr. Yutaka Kidawara at National Institute of Information and Communications Technology, and Principal Researcher Mr. Toshiyasu Nakao at NEC Corporation for their valuable advices and discussions as doctoral advisories.

I would also like to express my gratitude to Assistant Prof. Akimitsu Kanzaki for his continuous support and guidance throughout my undergraduate, master, and doctoral research work.

I am very grateful for the help and support from Associate Prof. Yuichi Teranishi, Associate Prof. Tomoki Yoshihisa, and Associate Prof. Kaname Harumoto at Osaka University, Associate Prof. Tsutomu Terada at Kobe University, Associate Prof. Takefumi Ogawa and Assistant Prof. Kotaro Nakayama at the University of Tokyo, and Mr. Yoshimasa Ishi at Osaka University.

I sincerely appreciate Dr. Hideki Hayashi at Central Research Laboratory, Hitachi, Ltd., Dr. Masako Shinohara and Dr. Shinya Kitajima at Fujitsu Laboratories Limited, Tetsuya Taniguchi at Fujitsu Limited, Yohei Sawai and Yuuki Iima at Nomura Research Institute Limited, Hironori Nagaishi and Ryo Hagihara at IBM Japan, Ltd., Hiroyuki Sago, Masaki Nakamiya, and Yasuhiro Nose at NTT DATA Corporation, and Tatsuya Shinjo at Hitachi Management Partner Corporation for discussion of my thesis.

I would also like to thank my team members, Minoru Kaji, Yoshihiro Kosaka, Yuya Sasaki, Shinya Kondo, and Ayana Yamamoto for discussion of my thesis.

With pleasure of studying and working with a number of very talented and warm hearted people at Nishio laboratory, I would like to express thanks to them.

At this moment, I have to mention Thailand Government. Just for their supports and encouragements, I could have this good opportunity to pursue my study in Japan. I would hereby give my special thanks to them.

Last but by no means least it gives me immense pleasure to offer my hearty thanks to my family. No words can express my appreciation to their support and love in my life. My parents have provided me with the greatest concerns and cares, for which I am eternally grateful.

Finally, with all my best wishes to those wonderful people who ever gave me assistances, supports and encouragements throughout my research experience.

Reference

[1] Akyildiz, I.F., Pompili, D., and Melodia, T.: Underwater Acoustic Sensor Networks: Research Challenges, *Ad Hoc Networks*, Vol. 3, No. 3, pp. 257–279 (May 2005).

[2] Alena, R., Gilbaugh, B., Glass, B., and Braham, S.: Communication System Architecture for Planetary Exploration, in *Proceedings of International Conference on Aerospace (Aerospace Conference 2001)*, Vol. 3, pp. 1075–1084 (Mar. 2001).

[3] Alena, R., Ossenfort, J., Lee, C., Walker, E., and Stone, T.: Design of Hybrid Mobile Communication Networks for Planetary Exploration, in *Proceedings of International Conference on Aerospace (Aerospace Conference 2004)*, Vol. 2, pp. 1161–1179 (Mar. 2004).

[4] Allred, J., Hasan, A., Panichsakul, S., Pisano, W., Gray, P., Huang, J., Han, R., Lawrence, D., and Mohseni, K.: SensorFlock: an Airborne Wireless Sensor Network of Micro-air Vehicles, in *Proceedings of International Conference on Embedded Networked Sensor Systems (SenSys 2007)*, pp. 117–129 (Nov. 2007).

[5] Arboleda, C.L.M. and Nasser, N.: Cluster-based Routing Protocol for Mobile Sensor Networks. in *Proceedings of International Conference on Quality of Service in Heterogeneous Wired/Wireless Networks (QShine 2006)* (Aug. 2006).

[6] Bahl, P. and Padmanabhan, V.: RADAR: an In-building RF-based User Location and Tracking System, in *Proceedings of IEEE Infocom 2000*, Vol. 2, pp. 775–784 (Mar. 2000).

[7] Baker, D., Braby, L., Curtis, S., Jokipii, J., Lewis, W., Miller, J., Schimmerling, W., Singer, H., Strachan, L., Townsend, L., Turner, R., and Zurbuchen, T.: Space Radiation Hazards and The Vision for Space Exploration: A Report on The October 2005 Wintergreen Conference, *Space Weather*, Vol. 5, No. 2 (Feb. 2007).

[8] Barnes, D., Pugh, S., and Tyler, L.: Autonomous Science Target Identification and Acquisition (ASTIA) for Planetary Exploration, in *Proceedings of International Conference on Intelligent Robots and Systems (IROS 2009)*, pp. 3329–3335 (Oct. 2009).

[9] Bin Tariq, M., Ammar, M., and Zegura, E.: Message Ferry Route Design for Sparse Ad hoc Networks with Mobile Nodes, in *Proceedings of International Symposium on Mobile Ad hoc Networking and Computing (MobiHoc 2006)*, pp. 37–48 (May 2006).

[10] Bjorklund, P. and Varbrand, P.: Resource Optimization of Spatial TDMA in Ad hoc Radio Networks: A Column Generation Approach, in *Proceedings of IEEE Infocom 2006*, Vol. 2, pp. 818–824 (Mar. 2003).

[11] Bruhn, F., Kratz, H., Warell, J., Lagerkvist, C., Kaznov, V., Jones, J., and Stenmark, L.: A Preliminary Design for a Spherical Inflatable Microrover for Planetary Exploration, *Acta Astronautica*, Vol. 63, No. 5-6, pp. 618–631 (Sept. 2008).

[12] Chin, T., Ramanathan, P., and Saluja, K.: Optimal Sensor Distribution for Maximum Exposure in a Region with Obstacles, in *Proceedings of The Global Telecommunications Conference (GLOBECOM 2006)*, pp. 1–5 (Nov. 2006).

[13] Chong, C. and Kumar, S.: Sensor Networks: Evolution, Opportunities, and Challenges, *Sensor Network Applications*, Vol. 91, No. 8, pp. 1247–1256 (Aug. 2003).

[14] Clouqueur, T., Phipatanasuphorn, V., Ramanathan, P., and Saluja, K.K.: Sensor Deployment Strategy for Target Detection, in *Proceedings of ACM*

International Workshop on Wireless Sensor Networks and Applications (WSNA 2002), pp. 42–48 (Sept. 2002).

[15] Corke, P., Hrabar, S., Peterson, R., Rus, D., Saripalli, S., and Sukhatme, G.: Autonomous Deployment and Repair of a Sensor Network Using an Unmanned Aerial Vehicle, in *Proceedings of IEEE International Conference on Robotics and Automation (ICRA 2004)*, Vol. 4, pp. 3602–3608 (Apr. 2004).

[16] Cui, J., Kong, J., Gerla, M., and Zhou, S.: The Challenges of Building Mobile Underwater Wireless Networks for Aquatic Applications, *IEEE Network*, Vol. 20, No. 3, pp. 12–18 (May 2006).

[17] Dahlberg, T., Nasipuri, A., and Taylor, C.: Explorebots: A Mobile Network Experimentation Testbed, in *Proceedings of ACM SIGCOMM 2005 Workshop on Experimental Approaches to Wireless Network Design and Analysis (E-WIND 2005)*, pp. 76–81 (Aug. 2005).

[18] Dantu, K., Rahimi, M., Shah, H., Babel, S., Dhariwal, A., and Sukhatme, G.: Robomote: Enabling Mobility in Sensor Networks, in *Proceedings of International Symposium on Information Processing in Sensor Networks (IPSN 2005)*, pp. 404–409 (Apr. 2005).

[19] Das, S., Pucha, H., and Hu, Y.: Microrouting: A Scalable and Robust Communication Paradigm for Sparse Ad hoc Networks, *International Journal of Distributed Sensor Networks*, Vol. 2, No. 1, pp. 79–100 (Jan. 2006).

[20] Dubois, P., Botteron, C., Mitev, V., Menon, C., Farine, P., Dainesi, P., and Ionescu, A.: Ad Hoc Wireless Sensor Networks for Exploration of Solar-system Bodies, *Acta Astronautica*, Vol. 64, No. 5-6, pp. 626–643 (Mar. 2009).

[21] Eriksson, J., Girod, L., Hull, B., Newton, R., Madden, S., and Balakrishnan, H.: The Pothole Patrol: Using A Mobile Sensor Network for Road Surface Monitoring, in *Proceeding of International Conference on Mobile Systems, Applications, and Services (MobiSys 2006)*, pp. 29–39 (June 2006).

[22] Ghosh, A.: Estimating Coverage Holes and Enhancing Coverage in Mixed Sensor Networks, in *Proceeding of International Conference on Local Computer Networks (LCN 2004)*, pp. 68–76 (Nov. 2004).

[23] Gu, Y., Bozdag, D., Brewer, R., and Ekici, E.: Data Harvesting with Mobile Elements in Wireless Sensor Networks, *Computer Networks*, Vol. 50, No. 17, pp. 3449–3465 (Dec. 2006).

[24] Guinard, A., McGibney, A., and Pesch, D.: A Wireless Sensor Network Design Tool to Support Building Energy Management, in *Proceedings of ACM Workshop on Embedded Sensing Systems for Energy-efficiency in Buildings (BuildSys 2009)*, pp. 25–30 (Nov. 2009).

[25] Hamilton, M., Graham, E., Rundel, P., Allen, M., Kaiser, W., Hansen, M., and Estrin, D.: New Approaches in Embedded Networked Sensing for Terrestrial Ecological Observatories, *Environmental Engineering Science*, Vol. 24, No. 2, pp. 192–204 (Feb. 2007).

[26] Harras, K. and Almeroth, K.: Inter-regional Messenger Scheduling in Delay Tolerant Mobile Networks, in *Proceedings of International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM 2006)*, pp. 93–102 (June 2006).

[27] Hart, J. and Martinez, K.: Environmental Sensor Networks: A Revolution in the Earth System Science, *Earth-Science Reviews*, Vol. 78, No. 3-4, pp. 177–191 (Oct. 2006).

[28] Heidemann, J., Silva, F., Intanagonwiwat, C., Govindan, R., Estrin, D., and Ganesan, D.: Building Efficient Wireless Sensor Networks with Low-level Naming, in *Proceedings of ACM Symposium on Operating Systems Principles (SOSP 2001)*, Vol. 35, No. 5, pp. 146–159 (Oct. 2001).

[29] Holman, R., Stanley, J., and Ozkan-Haller, T.: Applying Video Sensor Networks to Nearshore Environment Monitoring, *IEEE Pervasive Computing*, Vol. 2, No. 4, pp. 14–21 (Oct. 2003).

[30] Howard, A., Mataric, M., and Sukhatme, G.: An Incremental Self-deployment Algorithm for Mobile Sensor Networks, *Autonomous Robots*, Vol. 13, No. 2, pp. 113–126 (Sept. 2002).

[31] Hu, L. and Evans, D.: Localization for Mobile Sensor Networks, in *Proceedings of International Conference on Mobile Computing and Networking (MobiCom 2004)*, pp.45–57 (Sept. 2004).

[32] Hull, B., Bychkovsky, B., Zhang, Y., Chen, K., Goraczko, M., Miu, A., Shih, E., Balakrishnan, H., and Madden, S.: CarTel: a Distributed Mobile Sensor Computing System, in *Proceedings of International Conference on Embedded Networked Sensor Systems (SenSys 2006)*, pp.125–138 (Oct. 2006).

[33] Iima, Y., Kanzaki, A., Hara, T., and Nishio, S.: Overhearing-based Data Transmission Reduction for Periodical Data Gathering in Wireless Sensor Networks, in *Proceedings of International Workshop on Data Management for Information Explosion in Wireless Networks (DMIEW 2009)*, pp. 1048–10553 (Mar. 2009).

[34] Iima, Y., Kanzaki, A., Hara, T., and Nishio, S.: An Evaluation of Overhearing-based Data Transmission Reduction in Wireless Sensor Networks, in *Proceedings of International Workshop on Sensor Network Technologies for Information Explotion Era (SeNTIE 2009)*, pp. 519–524 (Apr. 2009).

[35] Jea, D., Somasundara, A., and Srivastava, M.: Multiple Controlled Mobile Elements (Data Mules) for Data Collection in Sensor Networks, in *Proceedings of International Conference on Distributed Computing in Sensor Systems (DCOSS 2005)*, pp. 244–257 (July 2005).

[36] Kansal, A., Yuen, E., Kaiser, W., Pottie, G., and Srivastava, M.: Sensing Uncertainty Reduction Using Low Complexity Actuation, in *Proceedings of International Symposium on Information Processing in Sensor Networks (IPSN 2004)*, pp. 388–395 (Apr. 2004).

[37] Kanzaki, A., Hara, T., and Nishio, S.: An Efficient TDMA Slot Assignment Protocol in Mobile Ad hoc Networks, in *Proceedings of the 2007 ACM Symposium on Applied Computing (SAC 2007)*, pp. 891–895 (Mar. 2007).

[38] Kataoka, S. and Honiden, S.: Multi-robot Positioning Model: Multi-agent Approach, in *Proceedings of International Conference on Computational Intelligence for Modelling, Control and Automation (CIMCA 2006)*, pp. 112–119 (Nov. 2006).

[39] Lee, C., Alena, R., Stone, T., Ossenfort, J., Walker, E., and Notario, H.: Software Architecture of Sensor Data Distribution in Planetary Exploration, in *Proceedings of International Conference on Aerospace (Aerospace Conference 2006)*, pp. 1–9 (Mar. 2006).

[40] Lin, C., Peng, W., and Tseng, Y.: Efficient In-Network Moving Object Tracking in Wireless Sensor Networks, *IEEE Transactions on Mobile Computing*, Vol. 5, No. 8, pp. 1044–1056 (Aug. 2006).

[41] Lin, F., Wang, Y., and Wu, H.: Testbed Implementation of Delay/fault Tolerant Mobile Sensor Network (DFT-MSN), in *Proceedings of IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops 2006)*, pp. 321–327 (Mar. 2006).

[42] Liu, B., Brass, P., Dousse, O., Nain, P., and Towsley, D.: Mobility Improves Coverage of Sensor Networks, in *Proceedings of International Symposium on Mobile Ad hoc Networking and Computing (MobiHoc 2005)*, pp. 300–308 (May 2005).

[43] Liu, C. and Wu, J.: Scalable Routing in Delay Tolerant Networks, in *Proceedings of International Symposium on Mobile Ad hoc Networking and Computing (MobiHoc 2007)*, pp. 51–60 (Sept. 2007).

[44] Liu, J., Zhao, F., and Petrovic, D.: Information-directed Routing in Ad Hoc Sensor Networks, in *Proceedings of International Conference on Wireless Sensor Networks and Applications (WSNA 2003)*, pp. 88–97 (Sept. 2003).

[45] Lorincz, K., Malan, D., Fulford-Jones, T., Nawoj, A., Clavel, A., Shnayder, V., Mainland, G., Welsh, M., and Moulton, S.: Sensor Networks for Emergency Response: Challenges and Opportunities, *IEEE Pervasive Computing*, Vol. 3, No. 4, pp. 16–23 (Oct. 2005).

[46] Mainwaring, A., Culler, D., Polastre, J., Szewczyk, R., and Anderson, J.: Wireless Sensor Networks for Habitat Monitoring, in *Proceedings of International Workshop on Wireless Sensor Networks and Applications (WSNA 2002)*, pp. 88–97 (Sept. 2002).

[47] Marasigan, D. and Rommel, P.: MV Routing and Capacity Building in Disruption Tolerant Networks, in *Proceedings of IEEE Infocom 2005*, Vol. 1, pp. 398–408 (Mar. 2005).

[48] Marcelloni, F. and Vecchio, M.: Enabling Energy-efficient and Lossy-aware Data Compression in Wireless Sensor Networks by Multi-objective Evolutionary Optimization, *Journal Information Sciences*, Vol. 180, No. 10, pp. 1924–1941 (May 2010).

[49] Nickerson, J.: Flying Sinks: Heuristics for Movement in Sensor Networks, in *Proceedings of Hawaii International Conference on System Sciences (HICSS 2006)*, Vol. 9, pp. 237–246 (Jan. 2006).

[50] Nittel, S.: A Survey of Geosensor Networks: Advances in Dynamic Environmental Monitoring, *Sensors*, Vol. 9, pp. 5664–5678 (July 2009).

[51] Nittel, S., Trigoni, N., Ferentinos, K., Neville, F., Nural, A. and Pettigrew, N.: A Drift-tolerant Model for Data Management in Ocean Sensor Networks, in *Proceedings of ACM International Workshop on Data Engineering for Wireless and Mobile Access (MobiDE 2007)*, pp. 49–58 (June 2007).

[52] Nose, Y., Kanzaki, A., Hara, T., and Nishio, S.: Route Construction Based on Measured Characteristics of Radio Propagation in Wireless Sensor Networks, in *Proceedings of International Conference on Advanced Information Networking and Applications (AINA 2009)*, pp. 560–567 (May 2009).

[53] Nose, Y., Kanzaki, A., Hara, T., and Nishio, S.: A Route Construction Method Based on Received Signal Strength in Wireless Sensor Networks, in *Proceedings of International Conference on Computers And Their Applications (CATA 2010)*, pp. 130–137 (Mar. 2010).

[54] Ogren, P., Fiorelli, E., and Leonard, N.: Cooperative Control of Mobile Sensor Networks: Adaptive Gradient Climbing in a Distributed Environment, *IEEE Transactions on Automatic Control*, Vol. 49, No. 8, pp. 1292–1302 (Aug. 2004).

[55] Partan, J., Kurose, J., and Levine, B.: A Survey of Practical Issues in Underwater Networks, *ACM SIGMOBILE Mobile Computing and Communications Review*, Vol. 11, No. 4, pp. 23–33 (Oct. 2007).

[56] Poduri, S. and Sukhatme, G.: Constrained Coverage for Mobile Sensor Networks, in *Proceedings of International Conference on Robotics and Automation (ICRA 2004)*, Vol. 1, pp. 165–171 (Apr. 2004).

[57] Pompili, D., Melodia, T., and Akyildiz, I.: Routing Algorithms for Delay-insensitive and Delay-sensitive Applications in Underwater Sensor Networks, in *Proceedings of International Conference on Mobile Computing and Networking (MobiCom 2006)*, pp. 298–309 (Sept. 2006).

[58] Rahimi, M., Shah, H., Sukhatme, G., Heideman, J., and Estrin, D.: Studying the Feasibility of Energy Harvesting in a Mobile Sensor Network, in *Proceedings of IEEE International Conference on Robotics and Automation (ICRA 2003)*, Vol. 1, pp. 19–24 (Sept. 2003).

[59] Rao, R. and Kesidis, G.: Purposeful Mobility for Relaying and Surveillance in Mobile Ad hoc Sensor Networks, *IEEE Transactions on Mobile Computing*, Vol. 3, No. 3, pp. 225–232 (July 2004).

[60] Reinhardt, A., Hollick, M., and Steinmetz, R.: Stream-oriented Lossless Packet Compression in Wireless Sensor Networks, in *Proceedings of IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON 2009)*, pp. 99–107 (June 2009).

[61] Roy, S., Arabshahi, P., Rouseff, D., and Fox, W.: Wide Area Ocean Networks: Architecture and System Design Considerations, in *Proceedings of ACM International Workshop on Underwater Networks (WUWNet 2006)*, pp. 25–32 (Sept. 2006).

[62] Sago, H., Shinohara, M., Hara, T., and Nishio, S.: A Data Dissemination Method for Information Sharing Based on Inter-vehicle Communication, *Journal of Interconnection Networks (JOIN)*, Vol. 8, No. 4, pp. 337–354 (Dec. 2007).

[63] Savvides, A., Han, C., and Strivastava, M.: Dynamic Fine-grained Localization in Ad-hoc Networks of Sensors, in *Proceedings of International Conference on Mobile Computing and Networking (MobiCom 2001)*, pp. 166–179 (July 2001).

[64] Schenker, P., Huntsberger, T., Pirjanian, P., Baumgartner, E., and Tunstel, E.: Planetary Rover Developments Supporting Mars Exploration, Sample Return and Future Human-robotic Colonization, *Autonomous Robots*, Vol. 14, No. 2, pp. 103–126 (Mar. 2003).

[65] Sekhar, A., Manoj, B., and Murthy, C.: Dynamic Coverage Maintenance Algorithms for Sensor Networks with Limited Mobility, in *Proceedings of IEEE International Conference on Pervasive Computing and Communications (PerCom 2005)*, pp. 51–60 (Mar. 2005).

[66] Shinjo, T., Kitajima, S., Ogawa, T., Hara, T., and Nishio, S.: Mobile Sensor Control Methods for Reducing Power Consumption in Sparse Sensor Network, in *Proceedings of International Workshop on Sensor Network Technologies for Information Explotion Era (SeNTIE 2008)*, pp.133–140 (May 2008).

[67] Shinjo, T., Kitajima, S., Ogawa, T., Hara, T., and Nishio, S.: A Mobile Sensor Control Method Considering Node Failures in Sparse Sensor Network, in *Proceedings of International Workshop on Data Management for Information Explosion in Wireless Networks (DMIEW 2009)*, pp. 1054-1059 (Mar. 2009).

[68] Shinohara, M., Hara, T., and Nishio, S.: Data Replication Considering Power Consumption in Ad hoc Networks, in *Proceedings of International Conference on Mobile Data Management (MDM 2007)*, pp. 118-125 (May. 2009).

[69] Somasundara, A., Kansal, A., Jea, D., Estrin, D., and Srivastava, M.: Controllably Mobile Infrastructure for Low Energy Embedded Networks, *IEEE Transactions on Mobile Computing*, Vol. 5, No. 8, pp. 958–973 (Aug. 2006).

[70] Somasundara, A., Ramamoorthy, A., and Srivastava, M.: Mobile Element Scheduling for Efficient Data Collection in Wireless Sensor Networks with Dynamic Deadlines, in *Proceedings of IEEE International Real-Time Systems Symposium (RTSS 2004)*, pp. 296–305 (Dec. 2004).

[71] Son, D., Helmy, A., and Krishnamachari, B.: The Effect of Mobility-induced Location Errors on Geographic Routing in Mobile Ad hoc Sensor Networks: Analysis and Improvement Using Mobility Prediction, *IEEE Transactions on Mobile Computing*, Vol. 3, No. 3, pp. 233–245 (July 2004).

[72] Spyropoulos, T., Psounis, K., and Raghavendra, C.: Spray and Wait: an Efficient Routing Scheme for Intermittently Connected Mobile Networks, in *Proceedings of ACM SIGCOMM Workshop on Delay-tolerant Networking (WDTN 2005)*, pp. 252–259 (Aug. 2005).

[73] Spyropoulos, T., Psounis, K., and Raghavendra, C.: Spray and Focus: Efficient Mobility-assisted Routing for Heterogeneous and Correlated Mobility, in *Proceedings of IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops 2007)*, pp. 79–85 (Mar. 2007).

[74] Suzuki, R., Makimura, K., Saito, H., and Tobe, Y.: Prototype of a Sensor Network with Moving Nodes, in *Proceedings of International Workshop on Networked Sensing Systems (INSS 2004)* (June 2004).

[75] Suzuki, R., Suzuki, T., Sezaki, K., and Tobe, Y.: Billiards: Policy-based Session Control Protocol in Disruption Tolerant Sensor Networks, in *Pro-*

ceedings of International Workshop on Networked Sensing Systems (INSS 2007), pp. 275–282 (June 2007).

[76] Suzuki, R., Tobe, Y., and Sezaki, K.: Session Control Protocol Exploiting Controlled Mobility in Multi-robot Sensor Networks, in *Proceedings of IEEE Conference on Local Computer Networks (LCN 2007)*, pp. 395–402 (Oct. 2007).

[77] Tobe, Y. and Suzuki, T.: WISER: Cooperative Sensing Using Mobile Robots, in *Proceedings of International Conference on Parallel and Distributed Systems (ICPADS 2005)*, pp.388–392 (July 2005).

[78] Treeprapin, K., Kanzaki, A., Hara, T., and Nishio, S.: A Mobile Sensor Control Method for Sparse Sensor Networks, in *Proceedings of DICOMO*, Vol. 2006, No. 6, pp. 173–176 (June 2006) (in Japanese).

[79] Treeprapin, K., Kanzaki, A., Hara, T., and Nishio, S.: A Mobile Sensor Control Method in Wide-area Sensor Networks, in *Proceedings of IEICE SIG Notes SN2006-47*, pp. 59–66 (Dec. 2006) (in Japanese).

[80] Treeprapin, K., Kanzaki, A., Hara, T., and Nishio, S.: A Mobile Sensor Control Method for Sparse Sensor Networks, in *Proceedings of the 2007 ACM Symposium on Applied Computing (SAC 2007)*, pp. 886–890 (Mar. 2007).

[81] Treeprapin, K., Kanzaki, A., Hara, T., and Nishio, S.: A Node Deployment Method for Efficient Sensing with Mobile Sensors in Sparse Sensor Networks, in *Proceedings of DPSWS*, Vol. 2008, No. 14, pp. 291–296 (Dec. 2008) (in Japanese).

[82] Treeprapin, K., Kanzaki, A., Hara, T., and Nishio, S.: An Effective Mobile Sensor Control Method for Sparse Sensor Networks, *Sensors*, Vol.9, No.1, pp.327–354 (Jan. 2009).

[83] Treeprapin, K., Kanzaki, A., Hara, T., and Nishio, S.: An Efficient Node Deployment Strategy in Sparse Sensor Networks, in *Proceedings of Interna-*

tional Workshop on Data Management for Information Explosion in Wireless Networks (DMIEW 2009), pp. 1060–1065 (Mar. 2009).

[84] Treeprapin, K., Kanzaki, A., Hara, T., and Nishio, S.: On a Mobile Sensor Control Method for Uniform Sensing in Sparse Sensor Networks, in *Proceedings of International Workshop on Sensor Network Technologies for Information Explotion Era (SeNTIE 2009)*, pp. 125–132 (Apr. 2009).

[85] Treeprapin, K., Kanzaki, A., Hara, T., and Nishio, S.: A Node Deployment Strategy for Efficient Sensing with Mobile Sensors in Sparse Sensor Networks, *IPSJ Journal*, Vol. 19, No. 3, (Mar. 2011, to appear).

[86] Tsujita, W., Yoshino, A., Ishida, H., and Moriizumi, T.: Gas Sensor Network for Air-pollution Monitoring, *Sensors and Actuators B: Chemical*, Vol. 110, No. 2, pp. 304–311 (Oct. 2005).

[87] Vasilescu, I., Kotay, K., Rus, D., Dunbabin, M., and Corke, P.I.: Data Collection, Storage, and Retrieval with an Underwater Sensor Network. in *Proceedings of International Conference on Embedded Networked Sensor Systems (SenSys 2005)*, pp.154–165 (Nov. 2005).

[88] Wang, G., Cao, G., La Porta, T., and Zhang, W.: Sensor Relocation in Mobile Sensor Networks, in *Proceedings of IEEE Infocom 2005*, Vol. 4, pp. 2302–2312 (Mar. 2005).

[89] Wang, G., Cao, G., and LaPorta, T.: Movement-assisted Sensor Deployment, *IEEE Transactions on Mobile Computing*, Vol. 6 No. 5, pp. 640–652 (June 2006).

[90] Wang, K. and Ramanathan, P.: Collaborative Sensing Using Sensors of Uncoordinated Mobility, *Proceedings of International Conference on Distributed Computing in Sensor Systems (DCOSS 2005)*, pp.293–306 (June 2005).

[91] Wang, Y. and Wu, H.: DFT-MSN: The Delay Fault Tolerant Mobile Sensor Network for Pervasive Information Gathering, in *Proceedings of IEEE Infocom 2006*, pp. 1–12 (Apr. 2006).

[92] Wang, Y. and Wu, H.: Delay/fault-tolerant Mobile Sensor Network (DFT-MSN): a New Paradigm for Pervasive Information Gathering, *IEEE Transactions on Mobile Computing*, Vol. 6, No. 9, pp. 1021–1034 (Sept. 2007).

[93] Xu, Y. and Lee, W.: On Localized Prediction for Power Efficient Object Tracking in Sensor Networks, in *Proceedings of International Conference on Distributed Computing Systems Workshops (ICDCS 2003)*, pp. 434–439 (May 2003).

[94] Xu, Y., Winter, J., and Lee, W.: Dual Prediction-based Reporting for Object Tracking Sensor Networks, in *Proceedings of International Conference on Mobile and Ubiquitous Systems (Mobiquitous 2004)*, pp. 154–163 (Aug. 2004).

[95] Yuan, Q., Cardei, I., and Wu, J.: Predict and Relay: an Efficient Routing in Disruption-tolerant Networks, in *Proceedings of ACM International Symposium on Mobile Ad hoc Networking and Computing (MobiHoc 2009)*, pp.95–104 (May 2009).

[96] Zhao, W., Ammar, M., and Zegura, E.: A Message Ferrying Approach for Data Delivery in Sparse Mobile Ad hoc Networks, in *Proceedings of ACM International Symposium on Mobile Ad hoc Networking and Computing (MobiHoc 2004)*, pp. 187–198 (May 2004).