

Copyright Undertaking

This thesis is protected by copyright, with all rights reserved.

By reading and using the thesis, the reader understands and agrees to the following terms:

- 1. The reader will abide by the rules and legal ordinances governing copyright regarding the use of the thesis.
- 2. The reader will use the thesis for the purpose of research or private study only and not for distribution or further reproduction or any other purpose.
- 3. The reader agrees to indemnify and hold the University harmless from and against any loss, damage, cost, liability or expenses arising from copyright infringement or unauthorized usage.

If you have reasons to believe that any materials in this thesis are deemed not suitable to be distributed in this form, or a copyright owner having difficulty with the material being included in our database, please contact lbsys@polyu.edu.hk providing details. The Library will look into your claim and consider taking remedial action upon receipt of the written requests.

Pao Yue-kong Library, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong

http://www.lib.polyu.edu.hk

The Hong Kong Polytechnic University Department of Electronic and Information Engineering

System Design for Wireless Sensor Networks

by

Chi Tsun CHENG

A thesis submitted in partial fulfilment

of the requirements for the degree of

Doctor of Philosophy

Initial Submission: 17th July, 2009

CERTIFICATION OF ORIGINALITY

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degree or diploma, except where due acknowledgment has been made in the text.

_____(Signed)

_____(Name of student)

"Go to the ant, O sluggard, observe her ways and be wise,

which, having no chief, officer or ruler, ..."

-Proverb 6:6-7, Holy Bible.

Abstract

The advent of wireless electronics and sensing technologies has made the production of versatile low-cost wireless sensor nodes possible. A wireless sensor network typically consists of a large number of wireless sensor nodes. The main use of wireless sensor networks is to collect data from sensing areas where human beings are not able to access. In contrast to conventional sensing systems, sensor networks utilize a huge volume of low-cost wireless sensor nodes to perform close-range sensing. The data collected will undergo in-network processes and then return to the user who is located in a remote site. This high redundancy of sensing power can greatly enhance the sensing resolution and make sensor networks robust to any adverse environmental conditions. However, these large number of wireless sensor nodes have also introduced a lot of challenging problems in system design. Some major problems are network lifetime, data collection efficiency, interference among wireless sensor nodes, and retransmissions due to noise and interference. Among all the problems mentioned above, the network lifetime problem and the data collection efficiency problem have been selected as the focus of this thesis.

Since wireless sensor nodes are power-constrained devices, the number of distant transmissions should be minimized in order to reduce energy consumption in wireless sensor nodes and prolong the network lifetime. An effective approach to improve efficiency is to divide the network into several clusters. By raising the sensing power of a sensor network in excess of the necessary level, the overall target tracking capability can be increased. However, a high sensing power also implies a greater interference to the network in addition to higher energy consumption. Such disadvantages can be relieved by adopting appropriate scheduling schemes and putting unnecessary sensor nodes into a sleep state. As mentioned earlier, clustering can provide a significant improvement in energy saving. In practice, most nodes in a sensor network are only capable of handling a single connection at any one time. With such configurations, cluster heads may become the bottlenecks in the data collection process. This bottleneck problem can be alleviated by modifying the network structure.

The contributions of this thesis are threefold. First of all, an energy-efficient clustering algorithm is proposed to tackle the network lifetime problem. Second, an energyaware scheduling scheme is proposed to alleviate the surplus sensing power problem. Finally, a delay-aware network structure and its formation algorithms are proposed to tackle the data collection efficiency problem. Compared with other existing algorithms in wireless sensor networks, the proposed algorithms are shown to be more efficient in extending network lifetime and improving data collection efficiency.

Publications Arising From the Thesis

Journal Papers

[1] C. T. Cheng, C. K. Tse and F. C. M. Lau, "A clustering algorithm for wireless sensor networks based on social insect colonies," (Submitted)

[2] C. T. Cheng, C. K. Tse and F. C. M. Lau, "A delay-aware data collecting network structure for wireless sensor networks," (Submitted)

[3] C. T. Cheng, C. K. Tse and F. C. M. Lau, "An energy efficient soft deployment scheme for wireless sensor networks," (Submitted)

[4] C. T. Cheng, C. K. Tse and F. C. M. Lau, "A scheduling scheme for wireless sensor networks based on social insect colonies," *IET Communications*, vol. 3, no. 5, pp. 714–722, May 2009.

[5] C. T. Cheng, C. K. Tse and F. C. M. Lau, "A tree-based data collecting network structure for wireless sensor networks," *Journal of Electronic Science and Technology of China*, (Invited), vol. 6, no. 3, pp. 274–278, September 2008.

Conference Papers

[6] C. T. Cheng, C. K. Tse and F. C. M. Lau, "An efficient data collecting network structure in wireless sensor networks," *The Sixth International Conference on Communications, Circuits and Systems 2008*, (ICCCAS 2008), Xiamen, China, vol. 1, May 2008, pp. 497–500.

[7] C. T. Cheng, C. K. Tse and F. C. M. Lau, "A bio-inspired scheduling scheme for wireless sensor networks," 2008 IEEE 67th Vehicular Technology Conference, (VTC 2008-Spring), Singapore, May 2008, pp. 223–226.

[8] C. T. Cheng, C. K. Tse and F. C. M. Lau, "A bio-inspired algorithm for performance optimization in wireless sensor networks," *International Symposium on Nonlinear Theory and Its Applications*, (NOLTA '07), Vancouver, Canada, September 2007, pp. 309–312.

[9] C. T. Cheng, C. K. Tse and F. C. M. Lau, "A clustering algorithm for sensor networks based on bee colony structure," *The Third Shanghai International Symposium on Nonlinear Science and Applications*, (NSA '07), Shanghai, China, June 2007, pp. 47–49.

[10] C. T. Cheng, C. K. Tse and F. C. M. Lau, "A bio-inspired clustering algorithm for wireless sensor networks," 20th China Symposium on Circuits and Systems, (CSCAS 2007), Guangzhou, China, June 2007.

Acknowledgements

I would like to express my profound gratitude to my chief supervisor, Prof. Michael Tse, and my co-supervisor, Dr. Francis Lau, for their continual support, encouragement, supervision and valuable suggestions throughout this research work. Their moral support and continuous guidance enabled me to complete my work successfully. I am also very thankful to Prof. Henry Leung, my co-supervisor when I was attached to the University of Calgary, Canada, for his valuable suggestions throughout this study.

This work would not have been possible without the support and assistance of many individuals. I would like to take this opportunity to thank all members of the Applied Nonlinear Circuits and Systems Research Group. My interactions and discussions with the group members over the years have helped to formulate and develop this research work.

I am, as ever, indebted to my parents for their love and support throughout my life. I also wish to thank Ms. Glory Lee for her support and understanding during my study.

Contents

Al	ostrac	et		iv
Acknowledgements vii				
Li	st of [Fables		xiv
Li	st of l	Figures		XV
1	Intr	oductio	n	1
	1.1	Backg	round	1
	1.2	Wirele	ess Sensor Nodes	7
		1.2.1	Sensing Module	8
		1.2.2	Computational Module	9
		1.2.3	Memory Module	9
		1.2.4	Communication Module	9
		1.2.5	Battery and Power Management Module	10
	1.3	Applic	cations of Wireless Sensor Networks	11
		1.3.1	Surveillance	11
		1.3.2	Monitoring	11

		1.3.3	Health Care	12
	1.4	Proble	ms in Wireless Sensor Networks	13
		1.4.1	Clustering	13
		1.4.2	Surplus Sensing Power	19
		1.4.3	Delay in Data Collecting Process	22
	1.5	Chapte	er Outline	27
	1.6	Chapte	er Summary	27
2	Clus	stering		29
	2.1	Introdu	uction	29
	2.2	Review	v of Existing Clustering Algorithms	31
	2.3	Biolog	cical Decentralized Systems	33
	2.4	Cluster	ring Algorithm Based on Social Insect Colonies	35
		2.4.1	Phase I – Recruiting Cluster Members	36
		2.4.2	Phase II – Neighbors Exploration	37
		2.4.3	Phase III – Local Scheduling	39
		2.4.4	Phase IV – Data Aggregation	39
		2.4.5	Phase V – Cluster Head Reelection	40
	2.5	Optim	ization	41
	2.6	Simula	ation Study	48
	2.7	Result	s and Evaluations	51
		2.7.1	Network Lifetime	51
		2.7.2	Network Coverage	54
		2.7.3	Network Robustness	57

x

		2.7.4	Averaged	Data Collecting Time	59
	2.8	Chapte	r Summar	y	61
3	Sche	eduling			62
	3.1	Introdu	iction		62
	3.2	Proble	m Formula	tion	64
			3.2.0.1	Target 3-Coverage Hit-Rate (THR)	65
			3.2.0.2	Detection Delay (DD)	65
			3.2.0.3	Energy Consumption Per Successful Detection (ECSD)	65
	3.3	Task S	witching in	n Ant Colonies	66
	3.4	Operat	ion of the	Proposed Soft Deployment Scheme	67
	3.5	Probab	ilistic Fori	nulation	71
		3.5.1	Probabili	stic Model of the ROF Soft Deployment Scheme	71
		3.5.2	Probabili	stic Model of the SOF Soft Deployment Scheme	72
		3.5.3	Probabili	stic Model of the Proposed Soft Deployment Scheme .	76
	3.6	Simula	tions and l	Evaluations	81
	3.7	Discus	sions of R	esults	88
		3.7.1	Expected	Target 3-Coverage Hit-Rate	89
			3.7.1.1	Effect of n	89
			3.7.1.2	Effect of <i>a</i>	90
			3.7.1.3	Effect of <i>b</i>	90
			3.7.1.4	Effect of κ	91
		3.7.2	Expected	Detection Delay	92
			3.7.2.1	Effect of <i>n</i>	92

			3.7.2.2	Effect of a	92
			3.7.2.3	Effect of <i>b</i>	93
			3.7.2.4	Effect of κ	93
		3.7.3	Expected	Energy Consumption Per Successful Target Detection	93
			3.7.3.1	Effect of n	93
			3.7.3.2	Effect of <i>a</i>	93
			3.7.3.3	Effect of <i>b</i>	94
			3.7.3.4	Effect of κ	94
	3.8	Chapte	er Summar	у	94
4	Netv	vork Sti	ructure		96
	4.1	Introdu	uction		96
	4.2	Related	d Work .		99
	4.3	The Pr	oposed Ne	etwork Structure	101
	4.4	Netwo	rk Formati	on Algorithm	104
		4.4.1	Top-Dow	n Approach	104
		4.4.2	Bottom-	Up Approach	110
	4.5	Numer	ical Analy	zsis	114
	4.6	Simula	tions		119
	4.7	Analys	sis		123
	4.8	Chapte	er Summar	y	125
5	Con	clusions	5		127
	5.1	Bio-ins	spired Solu	utions	128
	5.2	Non B	io-inspired	l Solutions	130

5.3	Future	Work	
	5.3.1	Short Te	rm Goals
		5.3.1.1	Mobile Wireless Sensor Networks
		5.3.1.2	Heterogeneous Wireless Sensor Networks 132
		5.3.1.3	Localization in Wireless Sensor Networks 133
		5.3.1.4	System Integration
	5.3.2	Long Ter	rm Goals
		5.3.2.1	Monitoring Clinical Signs of Avian Influenza in Poultry 135
		5.3.2.2	A Flooding Monitoring System
		5.3.2.3	A Man-Made Slope Monitoring System 137

Bibliography

139

List of Tables

1.1	Categorizing sensor networks	3
3.1	Value of constants c_1 and c_2 used in (3.7)	73
3.2	Value of constants c_3 , c_4 , c_5 , and c_6 used in (3.8)	73
4.1	Cluster members degree distribution of the proposed network structure	
	with network size $N = 2^q$, where $q = 1, 2, \cdots$.	101
4.2	The edge weight matrix of graph G in Example 1 \ldots \ldots \ldots \ldots	107

List of Figures

1.1	Top: Remote sensing in conventional sensing systems. Bottom: Close		
	sensing in wireless sensor networks.	5	
1.2	The use of heterogeneous thresholds in wireless sensor nodes to enhance		
	data resolution.	6	
1.3	The use of homogeneous thresholds in wireless sensor nodes to dilute		
	spatial interference	7	
1.4	The five basic building modules of a modern wireless sensor node	8	
1.5	Wireless sensor networks used in health care applications	13	
1.6	Wireless sensor nodes communicate with the base station directly	14	
1.7	Wireless sensor nodes communicate with the base station in a multi-hop		
	manner	16	
1.8	Wireless sensor nodes communicate with the base station through cluster		
	heads	18	
1.9	Wireless sensor networks with poor on-off arrangements	20	
1.10	An improved wireless sensor nodes on-off arrangement	21	

- 1.11 Data collecting processes in wireless sensor networks with different network structures. Circles represent the wireless sensor nodes. Circles in gray represent the cluster heads. Dashed arrows illustrate the flow of data packets. Number associated with each arrow is the time slot number. 23
- 1.12 Data collecting processes in wireless sensor networks with different network structures. Circles represent the wireless sensor nodes. Circles in gray represent the cluster heads. Dashed arrows illustrate the flow of data packets. Number associated with each arrow is the time slot number. 24
- 1.13 Data collecting processes in wireless sensor networks with different network structures. Circles represent the wireless sensor nodes. Circles in gray represent the cluster heads. Dashed arrows illustrate the flow of data packets. Number associated with each arrow is the time slot number. 26

2.1	A cluster with 2 layers	42
2.2	Averaged separation of nodes across adjacent layers	44
2.3	Orientation of the base station (BS) to a rectangular sensing terrain	46
2.4	The optimized values of $r_{\rm com}$ (solid line) and l (dash line) for different	
	values of <i>n</i>	48
2.5	Network lifetime of networks with (a) $\rho = 0.12 \text{ nodes/m}^2$; (b) $\rho = 0.2$	
	nodes/m ² . Results shown are averaged values taken over 100 indepen-	
	dent simulations.	52
2.6	Sensing coverage of networks with (a) $\rho = 0.12 \text{ nodes/m}^2$; (b) $\rho = 0.2$	
	nodes/m ² . Results shown are averaged values taken over 100 indepen-	
	dent simulations.	55

2.7	Lifetime standard deviations across 100 simulations of networks with	
	(a) $\rho = 0.12 \text{ nodes/m}^2$; (b) $\rho = 0.2 \text{ nodes/m}^2$	58
2.8	Averaged data collecting time of networks with (a) $\rho = 0.12$ nodes/m ² ;	
	(b) $\rho = 0.2 \text{ nodes/m}^2$. Results shown are averaged values taken over 100	
	independent simulations.	60
3.1	A generalized state diagram of sensor nodes in the proposed soft deploy-	
	ment scheme.	68
3.2	Translating a coverage problem into a clean tile problem	74
3.3	Target hit-rate of the random on-off soft deployment scheme	82
3.4	Detection delays of the random on-off soft deployment scheme	83
3.5	Energy consumption per successful detection of the random on-off soft	
	deployment scheme.	84
3.6	Target hit-rate of the selective on-off soft deployment scheme	85
3.7	Detection delays of the selective on-off soft deployment scheme	86
3.8	Energy consumption per successful detection of the selective on-off soft	
	deployment scheme. Note that in the simulations, ECSD becomes in-	
	finite when r_{bcast} is beyond 7.5 m (where $n=300,500$) and 8 m (where	
	<i>n</i> =700)	87
3.9	Target hit-rate of the proposed soft deployment scheme	88
3.10	Detection delays of the proposed soft deployment scheme	89
3.11	Energy consumption per successful detection of the proposed soft de-	
	ployment scheme.	90

3.12	The minimum ECSD of the proposed, the generic SOF, and the generic	
	ROF soft deployment schemes with THR \geq 98% and DD \leq 1	91
3.13	The minimum ECSD of the proposed, the generic SOF, and the generic	
	ROF soft deployment schemes with THR \geq 96% and DD \leq 2	92
4.1	(a) Data collection in a single cluster 2-hop network and (b) data collec-	
	tion in an improved single cluster multi-hop network. Circles with CM	
	represent the cluster members. Circles with CH represent the cluster	
	heads. Filled circles with BS represent the base stations. Dashed arrows	
	show the direction of data flow	98
4.2	Proposed network structure with network size $N = 16$. Circles with	
	CM represent the cluster members. Circle with CH represents the cluster	
	head. Filled circle with BS represents the base station. Degree of each	
	node is represented by the variable k . Dashed arrows show the direction	
	of data flow	102
4.3	Network formation of the proposed network structure using centralized	
	top-down approach ($N \ge 4$)	106
4.4	Using dynamic programming to solve the heaviest k-subgraph problem	
	in Example 1	108
4.5	Averaged data collecting time of different single tree structures. Note	
	that results obtained from the proposed algorithm using the top-down	
	approach agree perfectly with those obtained from the bottom-up approach	.119
4.6	Averaged ψ of different single tree structures	120
4.7	Averaged data collecting time of different multiple-cluster structures	122

4.8	Averaged ψ of different multiple-cluster structures	123
5.1	Wireless sensor networks with the proposed scheduling scheme and the	
	proposed clustering algorithm. Filled circle represents the cluster head	129
5.2	Localization processes in wireless sensor networks with different dis-	
	tribution of wireless sensor nodes with GPS modules. Empty circles	
	represent normal wireless sensor nodes. Circles with crosses represent	
	wireless sensor nodes with GPS modules. Dashed lines indicate com-	
	munications among wireless sensor nodes	134
5.3	Using wireless sensor networks to monitor clinical signs of avian in-	
	fluenza in poultry	136
5.4	Using wireless sensor networks to monitor man-made slopes	138

Chapter 1

Introduction

1.1 Background

In the previous decades, sensor networks and their applications have greatly improved the quality of our life. Sensor networks, or distributed sensing systems, are employed by power companies to monitor their power transmission and distribution networks [1]. Thanks to these sensing systems, faults in power lines can nowadays be detected and isolated within minutes or even seconds. Apart from the industries on ground, the aviation industries have also benefited from the use of wireless sensor networks. The radar networks used in air traffic control (ATC) have been keeping our air journeys safe for more than 70 years [2].

War has been a great driving force for engineering breakthroughs. Like many advanced technologies, sensor networks were first developed for military applications. An acoustic wired sensor network, namely the Sound Surveillance System (SOSUS), was developed by the US army to detect USSR submarines during Cold War [3, 4]. In the same period of time, the Semi-Automatic Ground Environment (SAGE) system was developed [5]. The SAGE is an air defense system used to protect the US from long-range attacks. It consists of a large number of radars which are connected using telephone lines. In the SAGE system, data are collected from the radars and transmitted to the centralized control center for further analysis. The SOSUS and the SAGE systems have built frameworks for the development of modern sensor networks.

Even though computers have been invented for more than 60 years, computing systems can never be mobilized until the early 1980s. In 1986, a distributed acoustic sensor network was set up by the MIT Lincoln Laboratory to trace low-flying aircrafts [6]. In the 1980s, each "mobile" node was as big as a cargo container which was carried by trucks and powered by generators. Regardless of its portability, the system had demonstrated the possibility of setting up a wireless sensor network.

The discovery of gallium arsenide (GaAs), the invention of the very-large-scale integration (VLSI) technology, the breakthroughs in the micro-electro-mechanical system (MEMS) and the remarkable strides in wireless technology have catalyzed the growth of wireless sensor networks in the early 1990s. The Tactical Remote Sensor System (TRSS) [7] of the US Navy has battery-powered mobile nodes in shoe box size. The mobile nodes are light enough for airborne missions.

Thanks to advanced electronic technologies, wireless sensor nodes can nowadays be produced in compact size and at low cost, bringing the development of sensor networks from a theoretical consideration to practical implementation. Off-the-shelf sensor nodes nowadays can be as small as coins which can operate for weeks to months [8]. Notice that technology has not only changed the appearance of the wireless sensor nodes, but also changed their connection style. In wired sensor networks, sensor nodes are usually

Features	Options
Connection	Wired/Wireless
Network architectures	Flat/Clustered/Hierarchical
Sensor nodes' mobility	Stationary/Mobile
Data collection mechanism	Event-driven/Query-driven/Source-driven

Table 1.1: Categorizing sensor networks

connected in a pre-defined hierarchical structure to ease maintenance. In early wireless sensor networks, due to the limited number of mobile nodes, star topologies are usually adopted for better centralized control. Modern sensor networks usually contain massive numbers of wireless sensor nodes. As the number of nodes increases, scalability becomes a severe problem in the system design of wireless sensor networks. Decentralized control methods are highly welcomed and ad-hoc connections are usually employed.

General speaking, all networks with networked sensors can be regarded as sensor networks. Sensor networks can be mainly classified as wired or wireless. A typical wireless sensor network comprises a large number of wireless sensor nodes. These wireless sensor nodes are deployed into a sensing terrain intentionally or randomly. Wireless sensor nodes will make use of their embedded sensors to collect information from the terrain. Captured data are then reported to a base station through wireless communications. Network architectures commonly employed in wireless sensor networks are flat, clustered, and hierarchical. Wireless sensor networks can be further categorized into stationary or mobile as shown in Table 1.1. Here, the terms stationary and mobile are used to describe the physical movements of the wireless sensor nodes. In stationary wireless sensor networks, once deployed, wireless sensor nodes will not change their geographical locations. In mobile wireless sensor networks, wireless sensor nodes are usually mounted on vehicles, such as unmanned aerial vehicles (UAVs), which will move around in the sensing area [9]. In this thesis, the focus of investigation will be on stationary wireless sensor networks.

Wireless sensor networks can be further categorized by their data collecting mechanisms. Basically, event-driven, query-driven, and source-driven are the main data collecting mechanisms commonly used in wireless sensor networks [10]. In an event-driven data collecting mechanism, wireless sensor nodes will report data only if some predefined events have been triggered. The predefined events can be the invasion of an intruder or other abnormal activities inside the sensing terrain. The query-driven data collecting mechanism is totally different to the event-driven mechanism. In the query-driven mechanism, a wireless sensor node will report its captured data on an on-demand basis. Users will enter their request at the base station or other access points of the system. A request will become a query which is then broadcast to all wireless sensor nodes within the network. Only wireless sensor nodes with related data will reply. Source-driven data collecting mechanism is commonly used to collect global data such as temperature, UV level, and humidity. Wireless sensor nodes with source-driven data collecting mechanism will collect data on a regular basis and report data to the base station from time to time. This thesis will be mainly focused on wireless sensor networks with event-driven and source-driven data collecting mechanisms.

In conventional sensing systems (Figure 1.1), sophisticated sensors are installed far away from the phenomenon to perform remote sensing. The data obtained from targets are highly vulnerable to noise contamination. Large amount of computational efforts are



Figure 1.1: Top: Remote sensing in conventional sensing systems. Bottom: Close sensing in wireless sensor networks.

therefore needed in signal processing. In some extreme scenarios, signals can be blocked completely due to obstacles, landscapes of the environment or even poor weather conditions. All these have limited the practical value of remote sensing systems. From the viewpoint of network security, the networks of conventional sensing systems are highly centralized. Sensors are closely packed together, making them highly fragile to intentional attacks.

On the other hand, wireless sensor networks deploy wireless sensor nodes into the interested phenomenon to perform close sensing (Figure 1.1). By reducing the distance among sensors and targets, a high signal-to-noise ratio (SNR) can be obtained [11]. The distributed nature of wireless sensor networks has also made them highly robust to failures and attacks. The large number of sensor nodes has provided redundant sensing



Sensor nodes with heterogeneous thresholds

Figure 1.2: The use of heterogeneous thresholds in wireless sensor nodes to enhance data resolution.

power and gives wireless sensor networks a certain level of fault tolerance. Nevertheless, wireless sensor nodes nowadays are advanced enough to handle simple data processing tasks. By exchanging data among sensor nodes, sensor networks are capable to perform in-network data processing and provide feedbacks or decisions in real time.

Due to the massive number of wireless sensor nodes in a network, inexpensive sensors with low resolution are usually used in building wireless sensor nodes. These lowcost sensors are usually of low precision. Thanks to the redundant sensing power in wireless sensor networks, the problem mentioned above can be alleviated by employing heterogeneous thresholds in wireless sensor nodes as shown in Figure 1.2. The idea is like concatenating several quantizers in series. The signal amplitude is divided into several segments and each segment is sampled by a particular quantizer. The outputs of the quantizers are then merged to reconstruct the sampled data.

However, if sensing reliability is the major concern, homogeneous thresholds can be



Sensor nodes with homogeneous thresholds

Figure 1.3: The use of homogeneous thresholds in wireless sensor nodes to dilute spatial interference.

used in wireless sensor nodes as shown in Figure 1.3. Each sensor is receiving the same signal with different attenuations and noise levels. Data obtained from various locations are used to dilute the error in sampling and provide data with high reliability.

1.2 Wireless Sensor Nodes

Typically, five basic building modules can be identified in a modern wireless sensor node. They are the sensing module, the computational module, the memory module, the communication module, and most importantly the battery and power management module (Figure 1.4).



Figure 1.4: The five basic building modules of a modern wireless sensor node.

1.2.1 Sensing Module

The sensing module acts as the gateway connecting our physical world to the information highway. It consists of sensors and analog to digital converters (ADC) which capture physical phenomena and convert them into digital data. Sensors of different attributes, such as acoustic, optical, pressure and temperature, can be embedded in the wireless sensor nodes as long as they are light and compact. If the characteristics of the monitoring targets are unknown, the sensing module on each sensor node should be turned on all the time so as to avoid missing the targets. However, depending on the natures of the sensors, a sensing module can be power consuming. If energy conservation is given the highest priority in the optimization process, periodical or event trigger sensing schemes can be used.

1.2.2 Computational Module

The computational module is like the central processing unit in a modern personal computer. The main functions of the computational module are to process data obtained from the sensing module, control all the on-board modules, and coordinate with other wireless sensor nodes. To reduce energy consumption and building cost of a wireless sensor node, the computational module usually employs a low-cost and low-power micro-controller with limited computational power. These physical limitations have imposed extra challenges to the design of sensor networks algorithms [12].

1.2.3 Memory Module

The memory module is mainly used for storing captured data which are scheduled for transmission. The memory module can also store historical captured data which can be used for simple signal processing functions, such as averaging. The stored data can also be used for forecasting, such as estimating the motion of a target. One-time programmable (OTP) memories, flash memories and other solid state storages are usually used in typical wireless sensor nodes [13, 14].

1.2.4 Communication Module

The communication module consists of a transceiver, a phase locked loop (PLL) and amplifier circuitries. In a free space scenario, the energy consumed by a communication module is proportional to the communication distance with an exponent equaling 2. In an urban area, the exponent may even rise to 4. These have made the communication module the most power hungry module in a wireless sensor node. With the enormous volume of wireless sensor nodes, a medium access control (MAC) protocol plays a very important role in the data broadcasting and collecting process. Obviously, frequency division multiple access (FDMA) scheme will never be a good choice in such a bandwidth constrained system. Code division multiple access (CDMA) scheme is being implemented in some middle to high-end wireless sensor node models [15], while time division multiple access (TDMA) scheme is widely used in low-end wireless sensor node models [16].

1.2.5 Battery and Power Management Module

Wireless sensor nodes are usually deployed in areas which are inaccessible for battery replacement. This means that the node lifetime is highly correlated with the battery capacity. One trivial way to extend network lifetime is to use batteries with higher capacities. However, high capacity batteries are usually too bulky to be embedded in wireless sensor nodes. Wireless sensor nodes are usually limited to coin size. Nevertheless, the extreme temperature in the operating environment will also affect the chemical reaction inside the batteries and shorten their lives. It is possible to employ energy harvesting techniques such as installing solar panels on wireless sensor nodes [17, 18]. However, most energy harvesting techniques available today are unreliable and with low efficiencies. This explains why energy saving has become one of the hottest research area in the studies of wireless sensor networks recently.

1.3 Applications of Wireless Sensor Networks

In the previous decades, due to some practical limitations, sensor networks were mainly applied in military applications. Thanks to the advances in technologies, the production cost of wireless sensor nodes nowadays has been greatly reduced. The diversity of sensors' sensing attributes has also broadened the functions of sensor networks. Wireless sensor networks can now be found in our everyday living.

1.3.1 Surveillance

Today, security systems are not simply in the form of closed circuit television (CCTV) cameras. Wireless sensor nodes are installed on window frames to detect vibrations. Any abnormal behaviors will be reported to the security control room without delay. Acoustic sensors are planted around buildings to discover suspicious visitors. Wall-mounted temperature and smoke sensors are widely used to detect fire incidents. Chemical detectors are attached to selected areas to spot gas leakages [19].

1.3.2 Monitoring

In the old days, wildlife monitoring surveys were mainly conducted by human observers and most wildlife behaviors were studied through observations. Sometimes, it was unavoidable for the observers to be subjective on certain behaviors' descriptions, such as the emotion of the subject. The presence of the observers around the subject might also introduce artificial disturbances and affect observation outcomes. The emergence of wireless sensing technologies has eliminated all these problems. Wireless sensor networks are now deployed near the organisms under study to perform close sensing [20]. Collected data are then reported to the user at the remote base station. Interactions among humans and wildlife are therefore minimized. For large animals, sensor nodes are attached to their bodies to perform continuous monitoring. The monitoring radius can be as long as a few kilometers which can never be achieved before.

Besides being used in wildlife monitoring, wireless sensor networks are applied in buildings for structure monitoring. Pressure and tension sensor are installed in buildings and bridges for diagnosis purposes [14]. Preventive measures can be applied before any disaster takes place. Wireless sensor networks are also setup by manufacturers to monitor their production lines. In manufacturing machines, wireless sensors are installed in moving parts to ensure their reliability. Components can always be replaced before they are worn.

1.3.3 Health Care

Biomedical sensors are widely used by medical personnel to perform diagnoses on patients [21]. The clinical data obtained have greatly facilitated the diagnosis processes. However, most sensors in early medical diagnosis systems were connected with cables. The cables have bounded the movement of patients and made the diagnosis an unpleasant experience. It is also impractical to have continuous monitoring. The situation has totally changed with the development of wireless sensing technologies. Medical sensor nodes are now compact enough which can be attached to clothing seamlessly. The sensors can monitor patients' different kinds of clinical signs for weeks without interruption [22]. Collected data are then transmitted to the hospital mainframe as shown in Figure 1.5, such that medical personnel can perform any further analysis when necessary [23, 24]. In the future, wireless medical sensor nodes are expected to get smaller and



Figure 1.5: Wireless sensor networks used in health care applications.

smaller, which can finally be implanted on the human body.

1.4 Problems in Wireless Sensor Networks

1.4.1 Clustering

A sensor network usually contains a large number of wireless sensor nodes. These massive numbers of wireless sensor nodes will collect information from the environment. The collected data will then be reported to the remote base station. Wireless sensor nodes can connect with the base station using different topologies. The simplest way is to connect each wireless sensor node directly with the base station (Figure 1.6). If the



A wireless sensor network with single-hop topology

Figure 1.6: Wireless sensor nodes communicate with the base station directly.

number of wireless sensor nodes is relatively small, some basic media access control (MAC) protocols, such as Aloha or slotted Aloha, can be employed. The advantage of this strategy is that the implementation can be simple and straight forward. The draw-back of such configuration is that it lacks scalability. When the number of wireless sensor nodes increases, those basic MAC protocols become inefficient and affect the system throughput. Connecting wireless sensor nodes with the base station directly will also affect the sensing ability of a sensor network. The reasons will be explained shortly.

Before explaining the relation between network topologies and the sensing ability of a sensor network, the concept of sensing coverage should be elaborated first. Consider

an ordinary wireless sensor network. Each wireless sensor node is equipped with some sensors. The on-board sensors usually have a limited sensing range. Assume that the on-board sensors are omni-directional sensors with a sensing range of $r_{\rm sen}$ m. The dish of radius $r_{\rm sen}$ m, with its center located at the position of a wireless sensor node j, is defined as the *detectable area* of node j. Any target within the detectable area can be detected by node j. A terrain is said to be *covered* if its area is fully covered by the detectable area of the wireless sensor nodes deployed in the area.

To communicate with the base station, a wireless sensor node will have to make use of its communication module. To ensure an adequate signal-to-noise ratio (SNR) at the receiver side, the transmission power has to be proportional to the communication distance. Wireless sensor nodes are battery-powered devices. Wireless sensor nodes may be deployed in hostile environments which are inaccessible for charging or replacement of batteries. The nodes will stop operating once their batteries are fully drained. The areas which are used to be covered by these "dead" nodes are then uncovered and become some blind-spots of the network. The sensing ability of the network is therefore affected. If the ratio between the uncovered area and the covered area keeps on rising and reaches a certain threshold, the remaining sensor nodes will no longer be able to provide valid information, and the network is considered as out of service or "dead".

As mentioned earlier, the transmission power of a wireless sensor node is a function proportional to the communication distance. Considering the configuration in Figure 1.6, wireless sensor nodes located far away from the base station will have to consume more energy in communications than those nodes located closer to the base station. Therefore, sensor nodes far from the base station will use up their energy quickly. More and more regions are not being covered and thus the sensing ability of the network is deteriorated.



A wireless sensor network with multi-hop topology

Figure 1.7: Wireless sensor nodes communicate with the base station in a multi-hop manner.

To solve this problem, communication distances for nodes far away from the base station should be reduced. This can be done by applying a multi-hop network topology.

Consider the sensor network as shown in Figure 1.7. Instead of communicating directly with the base station, each sensor node will try to forward its data to one of its neighbors which is closer to the base station. Upon receiving the data from a neighboring node, a wireless sensor node will make use of its computational module to combine the incoming data with its own data by means of data or decision fusion. The fused data will then again be forwarded to a neighbor which is closer to the base station. The process goes on until the fused data has reached the base station. The main advantage
of this network topology over the direct connection method is that the communication distances of the sensor nodes can be greatly reduced. Energy consumed in wireless communications can now be distributed among the sensor nodes more evenly. Unfortunately, this multi-hop flat network structure also has its own imperfection.

In a multi-hop flat network structure, besides transmitting the data to the base station, wireless sensor nodes are also required to receive data from their neighbors. The energy consumed in receiving data can be considerable when the number of neighboring nodes increases. Nevertheless, wireless sensor nodes are also required to fuse the incoming data with their own data. The energy consumed by the computational units in wireless sensor nodes is not negligible. Such extra energy consumption will become more and more significant when the node density of the network is getting higher and higher. Another problem is the uneven distribution of loading among wireless sensor nodes. Although most wireless sensor nodes are only required to communicate with their nearest neighbors and have very short communication distances, wireless sensor nodes close to the base station are required to communicate with the base station directly. Since the base station is usually located at a remote site, the averaged transmission distance for nodes close to the base station are much longer than any other nodes in the network. Also, wireless sensor nodes close to the base station will consumed a lot of energy in fusing data from far away sensor nodes. As a result, wireless sensor nodes close to the base station will run out of energy sooner than the others and lead to the similar coverage problem that exists in the direct connection method.

The problems mentioned above can be alleviated by applying clustering techniques. The basic idea of clustering in wireless sensor networks can be explained with the help of an example. Consider the network in Figure 1.8. The network is divided into three



A wireless sensor network with clustering

Figure 1.8: Wireless sensor nodes communicate with the base station through cluster heads.

clusters. Within each cluster, a wireless sensor node is elected as the cluster head. The remaining nodes in the cluster are regarded as the cluster members. Data collected by the cluster members will be forwarded to the cluster head directly or in a multi-hop manner. The cluster head will perform data or decision fusion on all incoming data. When all data within the cluster have reached the cluster head, the cluster head will forward the fused data to the base station. With such arrangements, the cluster heads will be the only group of nodes involved in long distance transmission. In order to distribute the load evenly among the wireless sensor nodes, each cluster will re-elect its cluster head from time to time.

Clustering techniques can distribute the loading evenly among wireless sensor nodes which can greatly extend the lifetimes of wireless sensor nodes. To maximize the lifetime of the network, however, we will have to solve the following problems:

- 1. How many clusters should be formed?
- 2. How many nodes should there be in each cluster?
- 3. What kinds of cluster head election mechanisms should be employed?
- 4. What kinds of cluster formation algorithms should be used?
- 5. How frequently should the cluster heads be re-elected?

All these problems are non-trivial and most of them still remain unsolved. Therefore, further study on the clustering problems will be given in Chapter 2. In Chapter 2, a decentralized clustering algorithm based on social insect colonies is proposed. Procedures for optimizing the proposed algorithm will be given in the chapter. The proposed algorithm will be compared against other existing clustering algorithms using computer simulations.

1.4.2 Surplus Sensing Power

As mentioned in the previous section, each wireless sensor node is equipped with sensors of limited sensing range. To minimize the setup cost of the network, wireless sensor nodes are required to be planted strategically to ensure that the sensing terrain can be fully covered with a minimum amount of nodes. Unfortunately, in most sensor networks applications, wireless sensor nodes are distributed over the sensing field randomly. Extra wireless sensor nodes are required to guarantee a 100% coverage of the sensing terrain.



(b) Sensing power in deficit

Figure 1.9: Wireless sensor networks with poor on-off arrangements.

These extra wireless sensor nodes will also provide the redundant sensing power and give the network extra robustness. Therefore, it is a common practice to deploy more than enough wireless sensor nodes when setting up a wireless sensor network.

However, under normal circumstances, these extra sensing power will have very limited contribution to the sensing quality of the system. Furthermore, communications from these extra wireless sensor nodes may cause severe interference to other nodes. It is desirable to have these extra wireless sensor nodes being switched off, reserving their energy for later use. Still, the wireless sensor nodes to be switched off cannot be chosen arbitrarily.

According to the example in Figure 1.9(a), a number of wireless sensor nodes (solid circles) are placed close to a phenomenon (cloud shape). Operating nodes are shaded in gray while non-operating nodes are not shaded. The detectable area of an operating node is represented by a dotted circle. In this example, all wireless sensor nodes are



Figure 1.10: An improved wireless sensor nodes on-off arrangement.

operating. The target detection ability and the network robustness of the network are both high. However, the total energy consumption of the network is unnecessarily high. By turning off more than 70% of the wireless sensor nodes (Figure 1.9(b)), the total energy consumption of the network can be greatly reduced. Communication interference among wireless sensor nodes is also alleviated. However, by trimming down the total energy consumption, the network is sacrificing its sensing coverage. A better solution is shown in Figure 1.10.

To maintain a balance between the sensing capability and the energy consumption of the network, we will have to answer the following questions:

- 1. How many wireless sensor nodes should be switched off?
- 2. Which wireless sensor nodes should be switched off?
- 3. For how long should a wireless sensor node be switched off?

4. What is the relation between the on-off arrangement of wireless sensor nodes and the characteristics of the targets?

To answer these questions, a comprehensive study on the on-off arrangement of wireless sensor nodes is given in Chapter 3. In this chapter, a bio-inspired scheduling scheme is proposed. The proposed scheme is a kind of decentralized scheduling scheme. Wireless sensor nodes with the proposed scheme will use only local information to arrange their on-off durations. The proposed scheme will be analyzed against other existing schemes using computer simulations.

1.4.3 Delay in Data Collecting Process

The clustering techniques described previously can greatly extend the lifetime of wireless sensor nodes. However, clustering techniques can also create bottlenecks in network structures and impose extra delays in data collection processes.

Wireless sensor nodes are simple communication devices. To reduce energy consumption and production cost, the communication module on a wireless sensor node is an un-sophisticated circuit which has only one transceiver. Even for advanced wireless sensor nodes running on dual transceivers, one transceiver is usually dedicated for the control channel. As a result, typical wireless sensor nodes are capable of handling one channel at a time.

Consider the situation in Figure 1.11(a) where a clustering algorithm is employed in a wireless sensor network. The clustering algorithm forms one single cluster with a 2-hop topology (cluster member \rightarrow cluster head \rightarrow base station). Assume that each data transmission costs a transmission delay of 1 time slot. It is also assumed that the



Figure 1.11: Data collecting processes in wireless sensor networks with different network structures. Circles represent the wireless sensor nodes. Circles in gray represent the cluster heads. Dashed arrows illustrate the flow of data packets. Number associated with each arrow is the time slot number.

data inside the data packets are highly correlated. By means of data or decision fusion techniques, a node is always capable of fusing all incoming packets into a single packet which is in the same size of an incoming packet. It will take 8 time slots for the cluster head to collect all data from its cluster members and 9 time slots for the base station to collect all data from the network. With such network topology, it can be easily observed that the time delay in the data collecting process will go linearly with the number of wireless sensor nodes in the network.

In the previous example, the bottleneck of the network is located at the cluster head. The problem can be alleviated if the network can be transformed carefully into a multilayer cluster as shown in Figure 1.11(b). Since the network has been partitioned into multiple layers, sensor nodes belonging to different layers can transmit data simultaneously as long as they are not communicating with the same parent node. In this example,



Figure 1.12: Data collecting processes in wireless sensor networks with different network structures. Circles represent the wireless sensor nodes. Circles in gray represent the cluster heads. Dashed arrows illustrate the flow of data packets. Number associated with each arrow is the time slot number.

the number of incoming connections at the cluster head has been greatly reduced from 8 to 4 and now it takes 5 time slots for the base station to collect all data from the network. This network structure is better than the 2-hop single cluster network mentioned earlier, however, it is still far from being optimized. In a multi-layer network structure, wireless sensor nodes, especially those located at the middle layers, may have to wait for the availability of their parent node. These extra delays may cancel out the time slots gained by employing simultaneous transmissions.

Another way to shorten the delays in the data collecting processes is to employ multiple clusters. Consider the configuration in Figure 1.12(c), where the network is partitioned into 3 clusters with equal number of wireless sensor nodes in each of them. Each cluster head will perform time division multiple access (TDMA) within its cluster. In this example, a cluster head will take 2 time slots to collect data from all its belonging sensor nodes. The cluster heads will then report the collected data to the base station. The multiple access scheme between cluster heads and base station is also in a time division manner. With the multiple clusters network structure, the total delay in the data collecting process of the base station is 5 time slots. The bottleneck of the network is now located at the base station. A straight forward solution to such a problem is to install multiple transceivers at the base station. However, such assignment will impose a limitation on the scalability of the system and greatly increase the complexity of the base station. Another problem of the multiple-cluster network is that more sensor nodes will be involved in long distance transmission, which implies that the wireless sensor nodes will have shorter lifetime.

Both problems can be tackled by choosing a suitable number of clusters. By reducing the number of clusters from 3 to 2 as shown in Figure 1.12(d), the time slots required by the base station to collect a full set of data from the network is further reduced to 4. The extra time slot is gained by arranging a different number of wireless sensor nodes to each cluster, such that cluster heads can finish their data collecting processes at different times and avoid accessing the base station simultaneously. The remaining problem is that it is difficult to determine the suitable number of clusters and the suitable number of wireless sensor nodes in each cluster. The problem becomes even more complicated when the number of wireless sensor nodes gets larger.

From the previous examples, it is shown that both 1) the number of layers in each cluster and 2) the number of clusters in a network play important roles in reducing delays in the data collecting process of wireless sensor networks. However, in order to achieve a short data collecting delay, both parameters cannot be chosen arbitrarily. Using excess number of layers in clusters (as shown in Figure 1.13(e)) or using excess number of clusters (as shown in Figure 1.13(e)).



Figure 1.13: Data collecting processes in wireless sensor networks with different network structures. Circles represent the wireless sensor nodes. Circles in gray represent the cluster heads. Dashed arrows illustrate the flow of data packets. Number associated with each arrow is the time slot number.

ters in a network (as shown in Figure 1.13(f)) will impose extra delays on data collecting processes. Therefore, the questions of this section will be:

- 1. What are the characteristics of a delay-aware network structure?
- 2. Can the delay-aware network structure be used in both single cluster and multiplecluster networks?
- 3. How the delay-aware network can be formed?
- 4. Can the delay-aware network be formed in both centralized and decentralized manners?
- 5. While delays in data collecting process are being minimized, will the total communication distance be largely extended? How to avoid that?

All these questions will be further studied in Chapter 4. In Chapter 4, a delay-aware data collecting network structure for wireless sensor networks is proposed. The proposed network structure can obtain very low delays in data collecting processes. Two different network formation algorithms for the proposed network structure are also introduced. The proposed network structure will be analyzed against other existing network structures using computer simulations.

1.5 Chapter Outline

This thesis is organized as follows. In Chapter 1 (the current chapter), an introduction to wireless sensor networks is given and the three selected problems in wireless sensor networks are briefly addressed. Details of the three selected problems, namely, clustering, surplus sensing power, and delay in data collection process, are elaborated with their corresponding solutions in Chapters 2, 3, and 4, respectively. Finally, conclusions and future work are given in Chapter 5.

1.6 Chapter Summary

As discussed in this chapter, sensor networks can bring a lot of benefits to human society. With the advances in technologies, inexpensive and compact wireless sensor nodes can be produced in the foreseeable future. Applications of sensor networks are expected to increase exponentially. In the near future, wireless sensor networks will become part of our everyday living. However, such massive numbers of wireless sensor nodes in sensor networks have also introduced a number of engineering problems. Some typical problems in sensor networks are synchronization, localization, routing, and medium access control problems. These problems are commonly found in ad-hoc networks which have been well defined and tackled. However, some unique problems in wireless sensor networks have never been studied until recently. Among these problems, three problems in wireless sensor networks, namely, the clustering problem, the surplus sensing power problem, and the problem of delays in data collecting process have been selected as the focuses of this thesis. The nature and characteristics of each problem have been briefly elaborated in this chapter.

Chapter 2

Clustering

2.1 Introduction

The advent of wireless electronics and sensing technologies has made the production of versatile low-cost wireless sensor nodes possible. A wireless sensor network typically consists of a large number of wireless sensor nodes. These wireless sensor nodes collect data from a sensing area which is possibly inaccessible for human beings. Data collected from the sensing field are usually reported to a remote base station. This high redundancy of sensing power can greatly enhance the sensing resolution and make sensor networks robust to the rapidly changing environment. Some applications of wireless sensor networks are wildlife habit study, environment observation and health care monitoring.

Since wireless sensor nodes are power-constrained devices, long-distance transmissions should be kept to minimum in order to prolong the network lifetime [25, 26]. Thus, direct communications between nodes and the base station are not encouraged. As discussed in Chapter 1, an effective approach to improve efficiency is to arrange the network into several clusters, with each cluster electing one node as its cluster head [27]. A cluster head collects data from other wireless sensor nodes in its cluster, directly or through a multi-hop manner. Typically, data collected from nodes of the same cluster are highly correlated. Data can be fused during the data aggregation process. The fused data will then be transmitted to the base station directly. In such an arrangement, only cluster heads are required to transmit data over a long distance. The rest of the nodes will need to do only short-distance transmission. To distribute the workload of the cluster heads among the wireless sensor nodes, cluster heads will be re-elected from time to time. Energy consumption of wireless sensor nodes is greatly reduced and the overall network lifetime can thus be prolonged. In this chapter, a decentralized clustering algorithm based on social insect colonies is proposed. Sensor networks adopting the proposed algorithm will form clusters with multiple layers. The primary objective of the proposed algorithm is to extend network lifetime. The proposed algorithm will be evaluated against three existing algorithms, namely, LEACH (Low-Energy Adaptive Clustering Hierarchy) [28], PEGASIS (Power-Efficient Gathering in Sensor Information Scheme) [29], and PEDAP (Power Efficient Data Gathering and Aggregation Protocol) [30]. The rest of the chapter is organized as follows. A review of existing clustering algorithms is given in Section 2.2. The biological phenomenon giving inspiration to this work is described in Section 2.3. Section 2.4 elaborates the details of the proposed algorithm. Methods for optimizing the proposed algorithm are given in Section 2.5. Simulation results and their evaluations are shown in Section 2.6 and Section 2.7 respectively. Finally, Section 2.8 gives the chapter summary.

2.2 Review of Existing Clustering Algorithms

Unlike in ordinary wireless networks, routing in sensor networks will either be from sensor nodes to the base station (many to one) or from the base station to sensor nodes (one to many). Routing in a sensor network can therefore be divided into three main categories, namely, direct transmission, planar routing and cluster based routing. It has been shown by Jamal and Ahmed [25] that within the three categories, cluster based routing can effectively prolong network lifetimes. The main problem in deriving a cluster based routing algorithm is the formation of clusters. Cluster formation techniques were first studied by researchers in data mining and data clustering, and the main process is to divide a set of data into partitions so that data having similar features are grouped in the same partition [31]. Cluster formation techniques such as fuzzy C-mean algorithm, K-mean algorithm, subtractive clustering algorithm, mountain method, genetic algorithm and particle swarm optimization have been considered by Malka, Siddeswara and Andrew [32] for applications in sensor networks for prolonging network lifetimes. However, all these methods require knowledge of sensor nodes' geographical locations which may not be suitable in many practical situations.

In recent years, a number of algorithms [28, 29, 30, 33, 34, 35] have been proposed specifically for forming clusters in sensor networks. Basically any clustering algorithm in sensor networks is concerned with the management of clusters, which includes forming a suitable number of clusters, selecting a cluster head for each cluster, and controlling the data transmission within clusters and from cluster heads to the base station [25]. The scope of this chapter is further limited to situations where data from every wireless sensor nodes is equally important and is required to be reported to the remote base station. The

algorithms considered in this chapter are required to maintain path(s) for each wireless sensor node to the base station from time to time. Therefore, event triggering algorithms such as TEEN [33], APTEEN [34] and the cluster head selection algorithm proposed in [35] will not be considered. The clustering algorithms which have fallen into the scope of this chapter are LEACH [28], PEGASIS [29] and PEDAP [30]. These algorithms organize networks with different network topologies. LEACH forms multiple clusters with 2-hop topology. PEGASIS forms a single multi-hop chain network while PEDAP forms a single minimum spanning tree network. The operations of LEACH, PEGASIS and PEDAP are briefly described as follows.

In LEACH, one single node will be randomly elected as the cluster head in each cluster. As the root of the cluster, the cluster head collects data from its cluster members and may combine several related data into one single unit. With fusion of data, fewer transmissions are required. In [28], it has been shown that networks with LEACH can have longer network lifetimes than those with minimum-transmission-energy (MTE) or direct transmission. LEACH is easy to implement and its decentralized properties make it robust to intentional attacks. However, LEACH has no control over the number of cluster heads. At some particular instances, systems with LEACH may have no cluster head.

PEGASIS is a widely studied centralized clustering scheme. In this scheme, sensor nodes are sorted and connected to form a chain. Only one node on the chain will be elected as the cluster head in a particular operation cycle. Each node on the chain receives data from its neighbors, fuses the data with its own and transmits it to a neighbor closer to the cluster head. Afterward, the cluster head forwards the data to the base station. PEGASIS can control the number of cluster heads effectively. However, being a centralized algorithm, PEGASIS is more vulnerable to attacks due to the presence of a centralized processing point. It also comes with other disadvantages such as having large communication overheads and a non-scalable nature.

PEDAP is another widely studied centralized clustering scheme. PEDAP organizes sensor nodes into a single minimum spanning tree. The minimum spanning tree is formed by Prim's algorithm [30]. The original cost function used in Prim's algorithm is replaced by a function in terms of distance among sensor nodes and sensor nodes' residual energy. The root of the tree structure will become the cluster head. Each node receives data from their child nodes, fuses the data with its own and transmits it to its parent node. Finally, all data will reach the cluster head and being forwarded to the base station. PEDAP can effectively reduce the total path length and reduce the number of connections to low energy nodes. However, as a centralized scheme, it shares the similar disadvantages as PEGASIS.

2.3 **Biological Decentralized Systems**

By evolution and natural selection, living organisms have become the most optimized or nearly optimized systems on earth. In constructing practical engineering systems, biological phenomena represent good sources of inspiration for achieving high efficiency and performance. Among many organizational structures in living organisms, the social structures of social insects are chosen as the sources of inspiration in this study because of their massive number of simple individuals and decentralized control mechanisms which show the greatest similarity to wireless sensor networks [36, 37].

Social insects make use of pheromones, a mixture of chemical, to communicate and

exchange information. Generally, pheromones used by social insects can be classified into two basic types, namely, *releaser* or *primer* [38]. A releaser pheromone will stimulate a response, such as the defensive behavior and the orientation behavior, instantaneously. A primer pheromone can change the behavioral repertoire of an individual, such as altering reproductive and endocrine systems.

In a typical social insect colony, for instance, we can identify *queens* and *workers* [39]. A queen has a similar structure as a worker. The main difference between queens and workers is that queens have well developed ovaries while workers do not. A queen releases to its surrounding a unique pheromone called *queen retinue pheromone*. The queen retinue pheromone is a mixture of the queen mandibular pheromone (QMP) and some retinue compounds. The mixture acts as both a releaser and a primer. The retinue compounds attract workers to the queen, induce the retinue behavior in workers and cause the workers to remove the QMP from the queen's body [40]. The QMP is then circulated throughout the colony via worker to worker transmissions [41]. High concentration of QMP indicates the presence of the queen and inhibits the queen rearing behaviors in workers. A drop in concentration of the QMP will trigger the re-development of ovaries in a small number of workers, transforming them into queens [42, 43]. A constant ratio among workers and queens is therefore maintained. In the following, we will adopt this decentralized organizational approach to derive an algorithm for forming and managing clusters for sensor networks.

In the proposed algorithm, each sensor node in a wireless sensor network is analogous to an individual in a social insect colony. The cluster heads are the queens and the reminding nodes are the workers. The retinue compounds are represented by the global control packets (GCP) while the QMP is represented by the local control packets (LCP). The cluster heads will make use of the GCP and LCP to form clusters, control the number of clusters, and re-elect cluster heads in a decentralized manner. A detailed explanation of the proposed algorithm will be given in the next section.

Note that the proposed clustering algorithm is different from Ant Colony Optimization (ACO) Algorithms [44]. ACO algorithms are general purpose optimization methods which are used to locate optimum points in search spaces. An ACO algorithm will release agents into a search space. The agents will perform random search and leave trails of artificial pheromones into the search space, which are leading to some optimum points. In contrast, the proposed algorithm is designed specifically for the clustering operations in wireless sensor networks. Instead of using the search space as a medium of communication, artificial pheromones are exchanged among individuals in a multi-hop manner.

2.4 Clustering Algorithm Based on Social Insect Colonies

In a network of N nodes, each node is assigned with a unique identity (UID) n, where $n = 1, 2, 3, \dots, N$. The UID will only serve as an identification which has no relation with sensor nodes' locations and connections. A cluster is organized in concentric layers with the cluster head located at the center. A node in a cluster will be assigned with a layer number (LN). The LN is an integer number starting from zero. Nodes in an inner layer (closer to the center) will have a lower LN while nodes at an outer layer will have a higher LN. Nodes belonging to different clusters may have the same LN. The cluster head will be the only node in a cluster with LN equal to 0.

Right after the deployment, wireless sensor nodes will wait for random durations.

For a wireless sensor node of UID *i*, its waiting duration τ_i is expressed as

$$\tau_i = \tau_{\max} \left(1 - t_r \frac{\mathbf{E}_{\text{res}_i}}{\mathbf{E}_{\max}} \right) \tag{2.1}$$

where t_r is a random number between 0 and 1, τ_{max} is the maximum waiting duration, E_{res_i} is the residual energy of the node *i*, and E_{max} is the maximum energy stored in a node's battery. This random duration is added to avoid having too many cluster heads at the initial phase. During the waiting period, a wireless sensor node will listen to the communication channel and look for GCP.

2.4.1 Phase I – Recruiting Cluster Members

If no GCP is received within the waiting period, a wireless sensor node will enter phase I. A wireless sensor node entering phase I is a cluster head. With (2.1), nodes with higher residual energy will have shorter waiting periods, and vice versa. Therefore, nodes with higher residual energy will have higher probabilities of being cluster heads. This is to go along with the observations in social insect colonies that a queen has more energy than a worker. The cluster head will broadcast a message to its neighbors within R m using the GCP, where R is equal to the cluster radius. The message includes the UID of the cluster head, the number of layers l in the cluster and the local communication radius $r_{\rm com}$. Procedures for obtaining R, l, and $r_{\rm com}$ will be given in Section 2.5. The objective of this message is to suppress other nearby nodes from turning into cluster heads. With the fact that a control packet is much smaller than a data packet, broadcasting the GCP will not burden the cluster head. According to (2.1), waiting periods of the wireless sensor nodes are highly randomized. Together with the small packet size of the GCP, the broadcasting can be carried out using CSMA/CA based methods.

Wireless sensor nodes receiving the GCP will stop their timers and become the cluster members of the cluster head. As mentioned earlier, clusters are arranged in concentric layers. Sensor nodes will make use of the messages in the packets and the following equation to calculate the bound b_z of the z^{th} layer, where z = 1, 2, ..., l.

$$b_{z} = \begin{cases} 0, & z \leq 0\\ \bar{b}_{z-1} + r_{\rm com}, & \text{otherwise} \end{cases}$$
(2.2)

where parameter \bar{b}_{z-1} is the averaged distance between the cluster head and the cluster members in the z - 1th layer. Here, \bar{b}_{z-1} is expressed as

$$\bar{b}_{z-1} = \sqrt{\frac{b_{z-1}^2 + b_{z-2}^2}{2}} \tag{2.3}$$

Detailed explanations of the equations will be given in Section 2.5. After the calculation, the cluster members will make use of the received signal strength (RSS) of the global control packets to estimate their distances to the cluster head and thus calculate their belonging layers.

2.4.2 Phase II – Neighbors Exploration

Once the belonging layer is known, a node (either a cluster member or a cluster head) will enter phase II. A node in phase II will adjust its transmitting power for local communications. The node will broadcast a message to its neighbors within $r_{\rm com}$ m, where $r_{\rm com} \ll R$, using the LCP. The message includes the UID, the approximated distance to the cluster head, and the belonging layer of the broadcasting cluster member. Since both the broadcasting area and the packet size of the LCP are small, CSMA/CA based methods can be used in coordinating the nodes.

Upon receiving the packets, a node can do a rough estimation on the node density ρ node/m² by simply counting the number of neighbors. The information will be saved for later use. A node can also estimate its distances to its neighbors using the RSS. Together with encapsulated messages, a node will be able to construct a database of its neighbors. According to the database, a node will then use the LCP to send a message to its nearest neighbor which is one layer closer to the cluster head. The message is an attachment request. If the request is accepted, the neighboring node will grant the request by again using the LCP. After the node has acknowledged the grant using the LCP, a connection is formed between the pair.

In case there is no neighbor which is one layer closer to an arbitrary node X, node X will increase its local communication radius to r'_{com} and transmit an attachment request to its neighbors using the LCP. The attachment request includes the UID, the belonging layer and the required communication radius r'_{com} of node X. When such request is received, nodes located at the inner layers will adjust their communication radius to r'_{com} , do a random back-off and then grant the request using the LCP. Node X will acknowledge the first grant received. The acknowledge message will also be used to suppress other nodes located at the inner layers from granting the previous attachment request. If no grant is received, node X will increase its local communication radius further and repeat the whole process again. The mechanism continues until a connection can finally be formed. The above process can be both time and energy consuming. Fortunately, such a scenario will rarely happen with a careful selection of the local communication radius r_{com} will be elaborated in the Section 2.5.

2.4.3 Phase III – Local Scheduling

For each connection pair, the node in the inner layer (or closer to the cluster head) is regarded as the parent node, while the node in the outer layer (or farther from the cluster head) is regarded as the child node. A parent node will wait for a period of time after a connection is formed. If no more request is received within the period, the parent node will enter phase III. The parent node will arrange its child nodes using TDMA-based methods and assign a time slot to each child node. Compared to the whole system, the number of child nodes to each parent node is relatively small. The parent node is always capable of scheduling its child nodes using TDMA-based methods.

2.4.4 Phase IV – Data Aggregation

Whenever the cluster head has finished scheduling all its direct connected child nodes, it will transmit a data request to all its child nodes using the LCP. Each cluster member receiving the data request will forward it to its child nodes once the local TDMA scheduling is done. When the data request reaches the outermost cluster member (i.e. a node with no child node), the outermost cluster member will return its parent node with data. Data are embraced in data packet which is much larger than a control packet. The parent node will acknowledge the receipt of data using the LCP. A parent will fuse all incoming data packets together with its own data and forward the fused data to its parent node. The data will finally reach the cluster head. The cluster head will transmit the fused data directly to the remote base station. The base station will broadcast a beacon message periodically. The cluster heads use RSS of the beacon message to estimate their distances to the base station and adjust their transmission power. In normal circumstances, a parent node is able to collect all data from its child nodes within a short period of time. However, if the channel condition changes rapidly or some of the nodes are out of service, the whole cluster may get into an infinite waiting state. To avoid such deadlock, if data packet is not received from a particular child node for a period of time, the parent node will poll the child node using the LCP. If the child node is reachable, it can either reply its parent node with a data packet or with a delay request using the LCP. If no reply can be heard from the child node, the node is regarded as unreachable and be skipped in the data aggregation process. On the other hand, if an acknowledgment is not received from the parent node will retransmit the acknowledgment immediately after the request is received. In case the parent node is unreachable, the child node will search its own database and try to attach to another favorable neighbor.

2.4.5 Phase V – Cluster Head Reelection

During the data aggregation process, all cluster members will estimate their lifetimes using the number of child nodes they have. The estimated lifetime is expressed in data aggregation cycles. The lifetime information together with the node density information obtained earlier are piggybacked by the data packet and forwarded to the cluster head. The cluster head can make use of the lifetime information to estimate the lifetime of the cluster. When the cluster reaches its estimated lifetime, the cluster head will broadcast a reset request to all neighbors within R m using the GCP. The reset request contains the node density information which is necessary for the upcoming cluster heads to compute the optimum cluster radius R, the number of layers l, and the local communication

radius $r_{\rm com}$. Methods for obtaining the parameters will be explained in Section 2.5. The reset request will put all belonging cluster members back to phase I. A wireless sensor node which has reached its own expected lifetime will not be involved in any upcoming operation, regardless of the amount of residual energy it has.

2.5 Optimization

The proposed algorithm organizes sensor nodes into multi-layer clusters. In [45], Bandyopadhyay and Coyle have provided a solution for finding the optimum number of layers l in a cluster, such that energy consumption of wireless sensor nodes can be minimized. In their work, $r_{\rm com}$ is assumed to be given and fixed. However, wireless sensor nodes in practice, as considered in this thesis, should be able to adjust their $r_{\rm com}$ as needed. This makes the solution suggested in [45] not applicable to the current scenario. Therefore, an optimization algorithm is designed for optimizing the proposed clustering algorithm.

In a social insect colony, workers are attracted by the retinue compounds released by the queen. The retinue compounds trigger the retinue behavior in workers and make them remove the QMP from the queen's body constantly. The QMP is then circulated throughout the colony via worker to worker transmissions. Since the social insect colony is crowded with individuals, the region of activities of each individual is limited. If the regions of activities are assumed to be in form of circles, a colony can be considered as layers of circles. This idea will be used in the design of the optimization algorithm. In the proposed clustering algorithm, a cluster head will receive LCP from its neighboring sensor nodes within $r_{\rm com}$ m. Sensor nodes within $r_{\rm com}$ m of the cluster head are defined as the first layer cluster members. In most practical situations, wireless sensor nodes



Figure 2.1: A cluster with 2 layers.

are distributed randomly and evenly across the sensing field as shown in Fig. 2.1. The first layer cluster members will have an averaged distance of $r_{\rm com}/\sqrt{2}$ m away from the cluster head. Cluster members in the first layer will receive LCP from their neighbors within $r_{\rm com}$ m. Wireless sensor nodes allocated with a distance of $r_{\rm com}$ m to $r_{\rm com} + r_{\rm com}/\sqrt{2}$ m away from the cluster head are defined as the second layer cluster members. In general, a sensor node allocated with a distance of b_{z-1} m to b_z m away from the cluster head is assigned as a $z^{\rm th}$ -layer CM, where the parameter b_z is described by (2.2) and (2.3).

If the sensor nodes are having a density of ρ nodes/m², the averaged number of

sensor nodes \bar{N}_z in the z^{th} -layer of a cluster can be written as

$$\bar{N}_{z} = \begin{cases} 1, & z = 0\\ \rho \pi b_{z}^{2} - \sum_{i=0}^{z-1} \bar{N}_{i}, & \text{otherwise} \end{cases}$$
(2.4)

In the data aggregation process, sensor nodes in the $z + 1^{\text{th}}$ -layer will forward their data to the sensor nodes in the z^{th} -layer. The node density ρ is normally very large and the sensor nodes are evenly distributed. Each sensor node in the z^{th} -layer will receive approximately $\bar{\gamma}_z$ data packets from sensor nodes in the $z + 1^{\text{th}}$ -layer, where $\bar{\gamma}_z$ is expressed as

$$\bar{\gamma}_z = \frac{\bar{N}_{z+1}}{\bar{N}_z} \tag{2.5}$$

A simple first-order radio model [28] is adequate for the purpose of calculating the energy dissipation of each sensor node in wireless communications. In the model, a sensor node will dissipate $E_{\text{TX/RX}} = 50 \times 10^{-9}$ J/bit to run its transceiver module and $E_{\text{AMP}} = 100 \times 10^{-12}$ J/bit/m² to run its amplifier module. In addition to the energy dissipated in wireless communications, a sensor node will dissipate $E_{\text{SEN}} = 50 \times 10^{-9}$ J/bit to run its sensor module [46]. Thus, a sensor node will dissipate 50×10^{-9} J of energy to capture every single bit of information from its surrounding. Furthermore, a sensor node will spend around $E_{\text{FUS}} = 0.3125 \times 10^{-9}$ J/bit to perform data/decision fusion on the incoming data and its own data [47]. Each data packet is ρ bits long. Since the size of a data packet is normally much larger than that of a control packet, energy consumption related to the control packets can be neglected in the optimization process. Therefore, in a cluster of *l* layers, the energy consumption $E_{\text{CM},z}$ of a *z*th-layer cluster member in



Figure 2.2: Averaged separation of nodes across adjacent layers.

each data aggregation process will be

$$E_{\text{CM}_z} = \begin{cases} \bar{\gamma}_z \varrho(E_{\text{TX/RX}} + E_{\text{FUS}}) + \\ \varrho(E_{\text{TX/RX}} + E_{\text{AMP}} \bar{\zeta}_z^2), & 1 \le z < l \\ \varrho(E_{\text{TX/RX}} + E_{\text{AMP}} \bar{\zeta}_z^2), & z = l \end{cases}$$
(2.6)

where $\bar{\zeta}_z$ is the averaged distance between a node in the z^{th} layer and its nearest neighbor in the $z - 1^{\text{th}}$ layer, which can be calculated as follows. Consider the situation in Figure 2.2 where a wireless sensor node (Point D) in the z^{th} layer of a cluster is going to transmit its data to one of the nodes in the $z - 1^{\text{th}}$ layer (Point B or C). Since there are \bar{N}_{z-1} nodes in the $z - 1^{\text{th}}$ layers and they are evenly distributed, $\angle \text{BAC}$ can be

expressed as

$$\angle BAC = \frac{2\pi}{\bar{N}_{z-1}} \tag{2.7}$$

With (2.7), $\bar{\zeta}_z$ is expressed as

$$\bar{\zeta}_{z} = \begin{cases} \bar{b}_{1}, & z = 1\\ \frac{1}{\angle BAC} \int_{-\frac{\angle BAC}{2}}^{\frac{\angle BAC}{2}} \left((\bar{b}_{z-1} \cos \theta - \bar{b}_{z})^{2} + (\bar{b}_{z-1} \sin \theta)^{2} \right)^{\frac{1}{2}} d\theta, & 2 \le z \le l \end{cases}$$
(2.8)

Similarly, the energy consumption $E_{\rm CH}$ of a cluster head in each data aggregation process will be

$$E_{\rm CH} = \bar{N}_1 \rho (E_{\rm TX/RX} + E_{\rm FUS}) + \rho (E_{\rm TX/RX} + E_{\rm AMP} \bar{\psi}^2)$$
(2.9)

where $\bar{\psi}$ is the averaged distance between the base station and a cluster head. If the base station is α m away from a rectangular sensing terrain of $w \times d$ m² as shown in Figure 2.3, $\bar{\psi}$ can be calculated as

$$\bar{\psi} = \frac{1}{wd} \int_{\alpha}^{\alpha+w} \int_{-\frac{d}{2}}^{\frac{d}{2}} \sqrt{x^2 + y^2} dy dx$$
(2.10)

In most wireless sensor networks, the base stations are usually located far away from the sensing terrain, making α very large. When α is much larger than the diagonal of the sensing terrain, the energy consumed by wireless sensor nodes in communicating with the base station will dominate the total energy consumption of the network. In these scenarios, in order to prolong network lifetime, the total number of clusters $n_{\rm C}$ should be set equal to 1. Therefore, to make sure a cluster can cover all the nodes in a sensing terrain of $w \times d$ m², the cluster radius *R* should be expressed as

$$R = \sqrt{w^2 + d^2} \tag{2.11}$$



Figure 2.3: Orientation of the base station (BS) to a rectangular sensing terrain.

Note that with s = 1, the cluster is always larger than the rectangular sensing terrain by a ratio of λ , where λ is given by

$$\lambda = \frac{\pi R^2}{wd} \tag{2.12}$$

Given *l*, and using (2.2) and (2.3), $r_{\rm com}$ can be determined by solving the following equation:

$$R = b_l$$

$$= \bar{b}_{l-1} + r_{\rm com}$$
(2.13)

Using the ratio λ , the total energy consumption E_{TOT} of a single cluster sensor network in a data aggregation process can be approximated as

$$E_{\rm TOT} = E_{\rm CH}' + \sum_{z=1}^{l} \frac{\bar{N}_z}{\lambda} E_{\rm CM_z}$$
(2.14)

where $E'_{\rm CH}$ is expressed as

$$E'_{\rm CH} = \frac{\bar{N}_1}{\lambda} \varrho(E_{\rm TX/RX} + E_{\rm FUS}) + \varrho(E_{\rm TX/RX} + E_{\rm AMP}\bar{\psi}^2)$$
(2.15)

The optimum value of l can be obtained by minimizing (2.14). Throughout the optimization, (2.8) and (2.10) involve integrations which may introduce extra computational burdens to a wireless sensor node. Fortunately, for stationary networks with fixed α , (2.10) is a constant. The constant can be defined on each node before a deployment or announced by the base station after a deployment. For (2.8), the computational effort can be reduced by employing numerical integration techniques with low resolution. In practical situations, transmitter power of a wireless sensor node is in finite scale. This can limit the choices on parameter l and further reduce the search space of the optimization. Since each node in the network may have a chance of being a cluster head, once the node density information ρ is obtained, the optimization will be carried out in a node regardless of it being idle or waiting. For a fresh deployment, since no ρ is yet obtained, ρ is predefined as N/wd initially. The optimized values of $r_{\rm com}$ and l for different values of n are shown in Figure 2.4. Note that the optimization procedures stated above are for networks with large α and relatively small $\sqrt{w^2 + d^2}$. For networks with comparable α and $\sqrt{w^2 + d^2}$, multiple clusters ($n_{\rm C} > 1$) can be more efficient. Each cluster can be arranged in the form of a circle. For a multi-clusters network, the first step in the optimization process is to find the optimum number of clusters $n_{\rm C}$ and the corresponding R to fully cover the sensing terrain. This is known as the *circle packing problem* [48]. The optimization will go through different values of $n_{\rm C}$. For each particular $n_{\rm C}$, the expected cluster radius R can be obtained. The cluster radius is then used to calculate the optimum values of l and $r_{\rm com}$ in each cluster. All these parameters will be used to calcu-



Figure 2.4: The optimized values of $r_{\rm com}$ (solid line) and l (dash line) for different values of n.

late the expected energy consumption of the whole network in a single data aggregation process. The set of parameters which gives the lowest expected energy consumption is regarded as the optimum set, which will be used to control the proposed algorithm. To reduce the computational effort in each wireless sensor node, different optimum sets can be calculated off-line and stored in a lookup table in each node before deployment.

2.6 Simulation Study

Simulations were carried out in MATLAB. In a simulation, the objectives of each clustering algorithm under test are 1) selection/election of cluster head(s), 2) collecting data from cluster members, 3) delivering the data to the remote base station, and 4) re-selection/election of cluster head(s). Each wireless sensor node is assumed to have data to report all the time. A data aggregation process is considered as completed only when all data from the operating nodes in the network has reached the remote base station. To keep the simulations fair, all algorithms under test will re-cluster after every data aggregation process. In each data aggregation process, energy consumption of each wireless sensor nodes, the number of operating nodes, the total coverage area, and the time used in the data aggregation process are recorded.

Sensor networks of node density ρ equal to 0.12 and 0.2 nodes/m² are used to show the effect of node density to different clustering algorithms. Note that the scope of this chapter is about energy efficient clustering algorithms. Energy efficient scheduling schemes for wireless sensor networks, such as random and selective on-off scheduling schemes [49], will not be employed in any algorithm under test. A node is always involved in data aggregation processes until all its energy has been used up. A network with a high node density will invoke more local communications and hence is expected to give poorer results than its counterpart with a low node density.

Sensor nodes are randomly and evenly distributed in a sensing field of $50 \times 50 \text{ m}^2$. The center of the sensing field is located at (x, y) = (25 m, 25 m). Once deployed, locations of the wireless sensor nodes are assumed to be fixed. In practical situations, the base station is usually located remotely from the sensing field. Therefore in the following simulations, the base station is assumed to be located at (x, y) = (25 m, -100 m) such that each sensor node will be at least 100 m away from the base station. Data packets and control packets (both GCP and LCP) used in the simulations are assumed to be 250 bytes and 8 bytes in size, respectively.

The base station is assumed to be a power unlimited device while a wireless sensor

node is powered by its embedded battery. Each battery is initially given 0.05 J of energy. When the energy drops to 0 J, a node is considered as non-operating. Discharge profiles and self-discharging characteristics of batteries are assumed to have no effect on extending the lifetime of batteries. A simulation is completed when all wireless sensor nodes in the network have stopped operating.

The radio model adopted in the simulations is the same as the one mentioned in Section 2.5. The radio channel is assumed to be symmetric. The channel condition is assumed to be perfect. Each clustering algorithm will adopt its own multiple access protocols. To make a fair comparison, probabilities of data collision and packet loss are assumed to be zero. Some clustering algorithms under test require precise synchronization among sensor nodes. All networks under test are assumed to be synchronized by some kinds of synchronization mechanisms such as those described in [50] and [51]. Since the focus of this chapter is mainly on the formation of clusters and their efficiencies, details of achieving synchronization is beyond the scope of this chapter and is omitted.

Results presented in this chapter are the averaged values obtained from 100 simulations. In each simulation, a new set of sensor nodes is distributed to a sensing field. As suggested in [28], the LEACH system is tuned with the desired percentage of each node to become a cluster head being set to 5%. Since PEGASIS and PEDAP are clustering algorithms which form only single cluster, PEGASIS and PEDAP inherently fix the number of clusters at 1 for all times. Moreover, the proposed algorithm is tuned according to the optimization procedures explained in Section 2.5 with $n_{\rm C} = 1$.

Routing varies with clustering algorithms. In LEACH, cluster members transmit their data packets toward their corresponding cluster head directly. Data fusions are carried out at the cluster heads only. In the PEGASIS system, cluster members forward packets to their adjacent neighbors on the chain which is closer to the cluster head. Data fusions are carried out along the chain. In PEDAP, data packets will propagate from the leaves to the root of the tree structure. Data fusion is carried out at the parent nodes. In the proposed algorithm, data packets will propagate from the outermost layer toward the inner layers of the cluster and finally reach the cluster head. Similar to PEDAP, data fusions are carried out at the parent nodes. The model for data aggregation used in the simulations assumes that, by means of data/decision fusion techniques, a node is always capable of fusing all incoming data packets into a single data packet which is of the same size of an incoming data packet.

2.7 **Results and Evaluations**

2.7.1 Network Lifetime

The lifetime of a wireless sensor network is highly correlated with the number of sensor nodes alive. Wireless sensor nodes are usually embedded with inexpensive sensors which return data with low accuracy and precision. These uncertainties can be diluted by using a large population of wireless sensor nodes. Therefore, in practice, the number of operating wireless sensor nodes is usually used to evaluate the lifetime of a network. Results in Figure 2.5 show that networks operated by the proposed algorithm can have longer lifetimes than those operated by LEACH, PEGASIS, and PEDAP. Nevertheless, performances of all algorithms under test are degraded when ρ is increased from 0.12 to 0.2. In LEACH, sensor nodes will only communicate with their nearest cluster heads. This renders the cluster heads with very high loading and drains their energy faster. There is no global control on the number of cluster heads. When the number of cluster



(b)

Figure 2.5: Network lifetime of networks with (a) $\rho = 0.12 \text{ nodes/m}^2$; (b) $\rho = 0.2 \text{ nodes/m}^2$. Results shown are averaged values taken over 100 independent simulations.
heads is small, distances between cluster members and cluster heads will increase by a lot. On the other hand, more energy is consumed in long distance communications when the number of cluster heads is too large. Note that during the formation of clusters, cluster heads are required to announce their existence to all the nodes in the network. A node has to receive announcements from every cluster head in order to select the nearest cluster head to engage with. Finally, each node has to register to its cluster head in order to secure a time slot from the cluster head. All these communications would increase the energy consumption in both the cluster heads and the cluster members.

PEGASIS gives a better performance at the beginning. However, because of the chain network property, the separations among nodes increase gradually as the number of operating nodes decreases. Sensor nodes are then forced to do longer communication in order to maintain the chain structure, which increase the energy consumption of the whole network. Moreover, in order to form a chain network, PEGASIS requires all nodes to maintain a complete database of location and energy status of all the nodes in the network. These requirements impose extra burdens to the network and shorten the network lifetime. At the time when the first node ran out of energy, the number of data aggregation processes that can be done by networks with PEGASIS is almost 4 times to that of networks with LEACH. Such performance gap is slightly different from that obtained in [29]. This is because the energy consumed in cluster formation and database update, which has not been considered previously, is now included in the simulations.

PEDAP gives similar performance as PEGASIS. Due to the definition of its cost function [30], a node closer to the base station is more likely to become a cluster head. Such configuration can greatly reduce the energy used in long distance communications. However, such strategy will also shorten the lifetimes of those nodes that are close to the base station, This situation is common in flat multi-hop network structures. When all nodes near the base station have run out of energy, the communication distances for the remaining nodes to the base station will increase drastically, causing a rise in energy consumption. A sharp drop in the number of operating nodes is observed in systems operated with PEDAP.

The proposed algorithm makes use of a multi-layer hierarchy in which nodes in the middle layers can perform data fusion on the data received. Such arrangement can greatly reduce the burdens of cluster heads. Moreover, cluster heads are elected randomly with a priority given to nodes with higher residual energy. The work load of cluster heads is evenly distributed throughout the network. Nevertheless, most communications are performed locally and local communication radii are optimized according to the procedures in Section 2.5. With the proposed algorithm, the energy consumption of the whole network is greatly reduced, thus giving a much longer network lifetime.

2.7.2 Network Coverage

Sensors embedded in wireless sensor nodes are capable of collecting data within a limited distance r_{sen} m. The circular area with radius r_{sen} m around a wireless sensor node is called the *sensible dish*. A sensing terrain is regarded as 100% covered if its area is fully covered by the sensible dishes of wireless sensor nodes. If the non-operating nodes are evenly distributed, the remaining nodes may still manage to cover the whole sensing terrain. The sensing ability of the network is maintained. On the other hand, if the nonoperating nodes are crowded in certain areas, these areas will soon become blind spots to the system. The distribution of non-operating nodes can be indicated by comparing the number of operating nodes to the sensing terrain coverage. Figure 2.6 shows the



(b)

Figure 2.6: Sensing coverage of networks with (a) $\rho = 0.12 \text{ nodes/m}^2$; (b) $\rho = 0.2 \text{ nodes/m}^2$. Results shown are averaged values taken over 100 independent simulations.

sensing terrain coverage of each algorithm with the r_{sen} arbitrarily set to 3 m. Results show that the network formed by the proposed algorithm has the highest sensing terrain coverage among the four algorithms under test. As observed previously, performances of all algorithms under test are degraded when ρ is increased from 0.12 to 0.2.

LEACH tries to distribute the loading of the cluster heads to all nodes in the network by switching the cluster heads from time to time. Due to the 2-hop structure of the network, a node far from a cluster head will have to consume more energy than a node close to a cluster head. This introduces an uneven distribution of energy dissipation among cluster members.

PEGASIS tries to minimize the energy consumption in local communications by organizing nodes into a single chain. The loading of the cluster head is distributed to all nodes in the network again by switching the cluster head from time to time. Unfortunately, the greedy algorithm used in the chain formation cannot maintain a constant separation of the adjacent nodes. Some nodes, especially for those located at the end of the chain, will have communication distances much longer than the others.

PEDAP attempts to arrange the sensor network in a minimum spanning tree structure. Communication distances among wireless sensor nodes are therefore minimized. However, a minimum spanning tree has no control on the number of child nodes of each parent node. Some parent nodes can be heavily loaded and run out of energy sooner than the others.

In the proposed algorithm, a cluster is arranged into multiple layers. Separations between layers are carefully adjusted such that the number of child nodes for each parent node is almost a constant. Compared to a 2-hop network, nodes with direct connection to cluster heads are greatly reduced. The multi-layer structure can also reduce communication distances.

2.7.3 Network Robustness

A network has better robustness if its network lifetime remains unaffected or only mildly affected when the distribution of the sensor nodes is different. We consider the standard deviation of the results obtained from the 100 individual simulations in Section 2.7.1 as an indicator of network robustness. As shown in Figure 2.7, the lifetimes of networks formed by the proposed algorithm have much smaller standard deviations than those of PEGASIS and PEDAP, which are comparable to LEACH. The performance of the network formed by the proposed algorithm is thus shown to be robust to the distribution of sensor nodes. Robustness of all algorithms under test are degraded when ρ is increased from 0.12 to 0.2. PEGASIS gives very high standard deviations of network lifetime due to the sensitivity of its performance to the chain length. Moreover, as the chain formation method in PEGASIS cannot guarantee a shortest chain at all times, a large fluctuation in the network lifetime can be resulted.

In networks employing PEDAP, the formation of the minimum spanning tree is based on the location of the base station and the distribution of sensor nodes. Since a sensor node close to the base station is more likely to become a cluster head, a high density of sensor nodes close to the base station will definitely enhance the performance and vice versa. With the random distribution of sensor nodes, a large variation in network lifetime can be expected.



(b)

Figure 2.7: Lifetime standard deviations across 100 simulations of networks with (a) $\rho = 0.12$ nodes/m²; (b) $\rho = 0.2$ nodes/m².

2.7.4 Averaged Data Collecting Time

Limited by the cost and size, sensor nodes are usually equipped with simple transceivers and most of these transceivers can only handle one transmission at a time. If one time slot is used to measure the duration of one data packet transmission, a sensor node of node degree k (k - 1 incoming connections and 1 outgoing connection) will take k - 1time slots to collect all data packets from its incoming connections. Therefore, different network topologies will have different impact on the data collection efficiency. The number of time slots used by the base station to collect data packets from all wireless sensor nodes is recorded and used to evaluate the data collection efficiency of each algorithm. Results are shown in Figure 2.8. Since a control packet is much smaller than a data packet, the time consumed in transmitting the control packets is neglected in all clustering algorithms under test. Performance of all algorithms under test are degraded when ρ is increased from 0.12 to 0.2. Results show that networks formed by the proposed algorithm can provide the shortest data collecting time among the three algorithms under test. The reason is that the proposed algorithm can form networks in multiple layers which makes interleaving possible during data collection. Networks formed by LEACH show periodical variations in data collecting time. This is due to the cluster head selection function being dependent upon the number of data aggregation processes. Since the cluster head election mechanism in LEACH is a function of the number of completed data aggregation processes [28], the number of clusters varies periodically as the simulation moves on. The peaks in Figure 2.8 are moments when the number of clusters is very low. For systems using PEGASIS, data at one end of the chain must travel for more than half of the chain before it can reach the cluster head. Interleaving is still



(b)

Figure 2.8: Averaged data collecting time of networks with (a) $\rho = 0.12 \text{ nodes/m}^2$; (b) $\rho = 0.2 \text{ nodes/m}^2$. Results shown are averaged values taken over 100 independent simulations.

possible, however with limited effect. This explains why the averaged data collecting time of networks formed by PEGASIS is much longer than that of networks formed by the proposed algorithm. PEDAP forms connections by only considering the distance between 2 nodes and the residual energy of the transmitting node. It has no control on the number of connections per node and may result in an extreme distribution of the node degree. Nodes with a high degree will thus become the bottlenecks in the data collecting process. In the proposed algorithm, separations among layers are optimized such that the connection degrees for all the nodes in the network can be maintained at a relatively constant level. The time taken in the data collecting process is therefore shortened.

2.8 Chapter Summary

In this chapter, a clustering algorithm based on social insect colonies has been proposed. Inspired by the structural organization of social insect colonies, an algorithm has been derived for forming and maintaining clusters in a wireless sensor network. It has been shown that the proposed decentralized clustering algorithm can improve network lifetime significantly over three well studied clustering algorithms. The performance of the proposed algorithm is shown to be robust to the distribution of sensor nodes. Nevertheless, the proposed algorithm can greatly shorten the data collecting time of the system.

Chapter 3

Scheduling

3.1 Introduction

Like many engineering problems, the design of sensor networks requires technical expertise from a variety of disciplines including signal processing, power management, communications, computational algorithms and circuit design. Because of the distributive nature of sensor networks, optimization of a number of performance parameters and operating cost poses a challenging problem for circuits and systems design. In particular, in contrast to conventional sensing systems, sensor networks make use of a huge volume of low-cost wireless sensor nodes to perform close-range sensing. This high redundancy of sensing power can greatly enhance the sensing resolution and make sensor networks robust to the rapidly changing environment.

The use of excessive sensing power, however, may bring undesirable consequences to sensor networks. The first consequence is reduction of the network lifetime. Another consequence is that the huge amount of sensor nodes will cause severe interference, especially in regions where sensor nodes are densely populated. Both consequences can be relieved by employing soft deployment techniques. Soft deployment, also known as scheduling, in sensor networks is a process of maintaining a balance among energy saving, bandwidth efficiency and sensing quality by putting a portion of sensor nodes into a sleep state.

Soft deployment can be classified into four main categories. They are the "always on" (AO), the "random on-off" (ROF), the "selective on-off" (SOF), and the "periodic on-off" (POF) deployment schemes [49]. Soft deployment in sensor networks has been addressed previously by other researchers [52, 53, 54, 55, 56], with different specific objectives and constraints. In this chapter, an energy efficient soft deployment scheme for wireless sensor networks is proposed. The proposed scheme is a kind of adaptive POF soft deployment scheme which uses only local information for making deployment decisions. The main objective of this chapter is to design an energy efficient soft deployment scheme for sensor networks to detect rare targets with low mobility. The rest of the chapter is organized as follows. Section 3.2 defines the soft deployment problem being tackled. The biological phenomenon giving inspiration to this work is described in Section 3.3. Section 3.4 briefly explains the operation of the proposed scheme. Probabilistic formulations of the proposed scheme and two other generic soft deployment schemes are given in Section 3.5. The proposed scheme is then compared and evaluated against these two generic soft deployment schemes in Section 3.6. Effects of varying each parameter of the proposed scheme will be discussed in Section 3.7. Finally, Section 3.8 concludes the chapter.

3.2 Problem Formulation

The sensing field is a two-dimensional plane of $w \times d m^2$. All the soft deployment schemes described in this chapter operate in discrete time steps called rounds, and each round lasts for time t_{step} . The target of interest considered in this chapter is a kind of point source phenomenon with an effective radius of r_{eff} m. The target is moving at a constant velocity of s m/round horizontally across the sensing field. At any particular instance, only one target will appear in the sensing field. A sensor network of N sensor nodes is deployed randomly onto the sensing field. Each sensor node will have a sensing radius of r_{sen} m. An active target can be detected if the distance between the centroid of the target and that of an active sensor is within $R_d = r_{sen} + r_{eff}$ m. Parameter R_d is regarded as the *detectable radius*. The dish with radius R_d and with a target located at its center is defined as the *detectable area*.

In this chapter, it is assumed that the energy of a sensor node is mainly consumed by its transceiver module, amplifier module, and sensor module. A simple first-order radio model [28] is used to calculate the energy dissipation of each sensor node in wireless communications. In the model, a sensor node will dissipate $E_{\text{TX/RX}}$ J/bit to run its transceiver module and E_{AMP} J/bit/m² to run its amplifier module. In addition to the energy dissipated in wireless communications, a sensor node will dissipate E_{SEN} J/bit to run its sensor module. Thus, a sensor node will dissipate E_{SEN} J of energy to capture every single bit of information from its surrounding.

For ease of evaluating system performance, three performance indicators are defined. Their definitions are as follows.

3.2.0.1 Target 3-Coverage Hit-Rate (THR)

A target is having 3-coverage (3Cov) when it is detected by 3 active sensor nodes simultaneously. In most applications, sensor networks are used to tell the locations of interested targets. To obtain the location of a target in a two-dimensional plane, 3coverage is essential. THR is defined as the ratio of the total time (round) a target is having 3-coverage to the total time (round) a target is active inside the sensing field.

3.2.0.2 Detection Delay (DD)

We define DD in this chapter as the time when a target emerged up to the time when it is having a 3-coverage for the first time. A system with a small DD can provide newly emerged targets' location information more rapidly and hence can facilitate the allocation of resources to their surroundings.

3.2.0.3 Energy Consumption Per Successful Detection (ECSD)

ECSD can reflect the power efficiency of the system. In this chapter, a successful target detection refers to a target having 3-coverage. Resources under-utilizing, resources over-utilizing and communication overhead will all increase the energy consumption per successful target detection.

In this chapter, the primary objective of an energy efficient soft deployment scheme is defined as to reduce ECSD as much as possible while keeping THR at a high value and DD at a low value. Networks under test are assumed to be synchronized by some synchronization mechanisms such as the one described in [50], which is beyond the scope of this chapter. Inter-communications among sensor nodes are assumed to be collision free and with zero packet loss. Since our focus here is the design of the soft deployment scheme that can provide a sufficient number of active sensor nodes, data collection processes such as routing [57, 58, 59], clusters formation [60, 61, 62] and data aggregation [63, 64, 65] are not considered in this chapter. Nonetheless, with the sufficient number of active sensors, all these data collection processes can be built on top of the proposed soft deployment scheme.

3.3 Task Switching in Ant Colonies

Same as in Chapter 2, social insect colonies are chosen as the source of inspiration in our study. In social insect colonies, individuals are assigned to different task groups to perform different tasks such as foraging, patrolling and midden work. Biologists have found that the allocation of individuals among different tasks in social insect colonies basically involves two mechanisms, namely the interactions among individuals from the same or different task groups and the task performance evaluations of individuals [66].

In social insect colonies, the probability for an individual to carry out task A rather than task B depends on the ratio of the number of task A workers it has encountered recently to that of task B workers [67]. For an individual doing task A, a high encounter rate with other workers doing the same task will increase the probability for the individual to switch task. Initially when the number of foragers is small, the encounter rate for a midden worker to other midden workers is much higher than that to foragers. In this case, midden workers will have higher probabilities of switching into the foraging group. As the number of foragers increases, the encounter rate for a midden worker to a forager increases at the same time. It makes midden workers less willing to go foraging. It is a kind of feedback system for controlling the size of different task groups which makes social insect colonies adaptive to environmental changes.

As food sites started to deplete, more foragers return to their nest without food. When they are examined by the others, they are regarded as unsuccessful foragers. Encountering with unsuccessful foragers will stop non-foraging workers from switching to foragers [68]. Unsuccessful foragers will become inactive and idle in the nest. This mechanism forms a negative feedback in controlling an oversized task group. A task group should shrink when its consumption outruns its contribution to the colonies.

As the population density of a social insect colony increases, encounter rate among individuals increases drastically. An individual may oscillate among tasks rapidly when its encounter rate is too high, which can be fatal to both the individual and its colony. To avoid such oscillation from happening, individuals try to limit their encounter rates by changing their encounter pattern as their population density increases [69, 70]. This measure can effectively reduce unnecessary communications among individuals.

3.4 Operation of the Proposed Soft Deployment Scheme

In the proposed algorithm, each sensor node in a wireless sensor network is analogous to an individual in a social insect colony. Different tasks in an ant colony are mapped to different operating states of sensor nodes. The interactions among ants are represented by the information exchange among sensor nodes. Sensor nodes will make scheduling decision by examining information obtained from their neighbors. The proposed scheme tries to coordinate the operation of each sensor node by using this local information.

Sensor nodes in the proposed scheme can be in either a *sleep state*, a *listen state* or an *active state*, as shown in Figure 3.1. As defined in Section 3.2, the proposed scheme



Figure 3.1: A generalized state diagram of sensor nodes in the proposed soft deployment scheme.

runs in rounds, and each round lasts for time t_{step} . Sensor nodes in the proposed scheme are only allowed to switch their states at the end of each round. Sensor nodes in the sleep state are turned off completely. Sensor nodes in the active state will perform sensing and broadcast their evaluation results to their neighbors. Sensor nodes in the listen state will preserve energy by switching off their sensor modules and amplifier modules, but their transceiver modules will remain active to eavesdrop communications nearby. Thus, the energy consumption of a sensor node in the active state is the highest, followed by the listen state, and the energy consumption in the sleep state is the lowest.

To initialize the scheme, all sensor nodes will be put into the sleep state for a random sleep duration t_s bounded by the system maximum sleep time a, where a is a multiple

of t_{step} . A timer will be used for the countdown of the sleep duration. At the end of the sleep duration, sensor nodes will put themselves into the listen state.

Sensor nodes in the active state will broadcast their evaluation results to their neighbors who are in the listen state using local control packets (LCP). An evaluation result can indicate whether an active node can detect a target (successful) or not (unsuccessful). The mechanism will be explained in the later part of this section. Sensor nodes in the listen state will "listen" to the communication channel. By listening to the communication channel, sensor nodes in the listen state can collect evaluation results of their active neighbors. Any sensor node in the listen state will use the evaluation results collected to give an evaluation function $v(n_{sa})$, which is given by

$$v(n_{\rm sa}) = \kappa^{n_{\rm sa}} \tag{3.1}$$

where n_{sa} is the number of successful active neighbors and κ is a constant between 0 and 1. Before the end of the current round, sensor nodes in the listen state will generate a random number between 0 and 1, and compare it with $v(n_{sa})$. If the random number is smaller than $v(n_{sa})$, the sensor node will become active in the next round. Otherwise, the sensor node will switch back to the sleep state in the next round. The sleep duration t_s , proportional to the complement of $v(n_{sa})$, is given by

$$t_s = \left\lceil a(1 - v(n_{\rm sa})) \right\rceil \tag{3.2}$$

where $\lceil u \rceil$ denotes the smallest integer equal to or greater than u. With (3.1) and (3.2), a sensor node will have a longer sleep duration if the performance of its neighbors are better, and vice versa.

A sensor node j in the active state will sense the surrounding environment for time t_{sen} , where $t_{\text{sen}} \leq t_{\text{step}}/2$. If an interested target is captured within time t_{sen} , sensor

node j will mark its evaluation result e_j as 1 (successful). Otherwise, the evaluation result will be marked as 0 (unsuccessful). Upon obtaining its own evaluation results, sensor node j will broadcast its evaluation results toward its surroundings using LCP. At the end of the round, an unsuccessful sensor node will switch to the sleep state, while a successful sensor node will remain active for another round. Nodes that switch from the active state to the sleep state will sleep for a random duration which is again bounded by the system's maximum sleep time a.

To maintain the inter-communications at a constant level independent of network density, communication radius $r_{\rm com}$ of a sensor node is adjusted to keep its number of neighbors approximately at a threshold *b*. This can be achieved by the following algorithm:

- 1. Initialize the sensor node with the shortest $r_{\rm com}$. In the proposed scheme, the shortest $r_{\rm com}$ is 0.5(b-1) m.
- 2. Broadcast a density probing packet (DPP) to its surroundings. The DPP contains the identity of the issuing node.
- 3. After receiving the DPP, nodes within $r_{\rm com}$ will reply with a reply packet (RP). The RP contains both the identities of the issuing node and the replying node.
- By counting the number of RP received, a node can estimate the number of neighbors in its surroundings. If the number of neighbors is below the threshold b, increase r_{com} by a step of 0.05b m and repeat step 2.

The above algorithm will only be carried out once. Therefore it will not introduce large overheads to the network.

3.5 Probabilistic Formulation

A generic ROF and a generic SOF soft deployment schemes are evaluated together with the proposed soft deployment scheme for comparison purposes. Probabilistic models for all the three soft deployment schemes on the performance indicators defined previously are constructed and shown as follows.

3.5.1 Probabilistic Model of the ROF Soft Deployment Scheme

In the ROF soft deployment scheme, a sensor node can be in either a sleep state or an active state. Sensor nodes in the sleep state are turned off completely while sensor nodes in the active state will use their sensor modules to monitor their surrounding environment. A sensor node will become active in the coming round with a probability of p, where $p \in [0, 1]$. Conversely, a sensor node in the ROF soft deployment scheme will turn into the sleep state in the coming round with a probability of 1 - p.

Initially, all sensor nodes in the ROF soft deployment scheme are put into the sleep state. A sensor node will turn active in the coming round with a probability of p. For a sensor network of N sensor nodes, the expected number of sensor nodes $E(n_{det})$ located inside a detectable area is expressed as

$$\mathbf{E}(n_{\rm det}) = \lfloor \frac{N\pi R_{\rm d}^2}{wd} \rceil$$
(3.3)

where $\lfloor u \rceil$ denotes the integer nearest to u. By assuming that the sensor nodes are evenly distributed, the probability for a successful 3-coverage detection Pr(3Cov) is expressed as

$$\Pr(3\text{Cov}) = 1 - \sum_{i=0}^{2} C_i^{\mathrm{E}(n_{\mathrm{det}})}(p)^i (1-p)^{\mathrm{E}(n_{\mathrm{det}})-i}$$
(3.4)

With (3.4), the expected detection delay E(DD) is expressed as

$$E(DD) = \sum_{i=0}^{\infty} i(1 - \Pr(3Cov))^i (\Pr(3Cov))$$
(3.5)

The expected energy consumption per successful detection is expressed as

$$E(ECSD) = \frac{E(n_{det})\varrho E_{SEN}}{Pr(3Cov)}$$
(3.6)

where ρ is the number of bits collected by the sensor module in each round.

3.5.2 Probabilistic Model of the SOF Soft Deployment Scheme

In a SOF soft deployment scheme, a sensor node can be in either a listen state or an active state. Sensor nodes in the listen state will monitor the communication channel by using their transceiver modules. Sensor nodes in the active state will use their sensor modules to monitor the environment. In the meantime, sensor nodes in the active state will also use their transceiver modules and amplifier modules to broadcast messages to their neighbors.

In a SOF soft deployment scheme, an active sensor node will broadcast a message using LCP to its neighbors within r_{bcast} m. This message is used to suppress the nearby sensor nodes in the listen state from being active in the coming round. A sensor node in the listen state will become active in the coming round if such a message is not received in the current round. Once a sensor node becomes active, it will remain active until its power drains out. The process carries on until every sensor node is either active or become suppressed. If each active sensor node is regarded as a dish of radius r_{bcast} m, the system can be viewed as having a *random-loose-packed* (RLP) configuration. Finding the separation among neighboring active nodes (dishes) in a RLP configuration analytically is a non-trivial task, so does the total number of active nodes (dishes) [71].

n	c_1	c_2		
300	1.359	1.217		
500	1.384	0.7534		
700	1.394	0.5324		

Table 3.1: Value of constants c_1 and c_2 used in (3.7).

Table 3.2: Value of constants c_3 , c_4 , c_5 , and c_6 used in (3.8).

n	<i>C</i> 3	c_4	C_5	<i>c</i> ₆
300	391	-0.4827	22.72	-0.02122
500	623.6	-0.7284	108.8	-0.1794
700	907.7	-0.8955	160.8	-0.2216

Therefore, both parameters are obtained empirically using computer simulations. Averaged separation \bar{r}_{sep} for an active node to 6 other nearest active nodes can be described as follows:

$$\bar{r}_{\rm sep} = c_1 r_{\rm bcast} + c_2 \tag{3.7}$$

where c_1 and c_2 are constants and their values are presented in Table 3.1. The relation between r_{bcast} and the total number of active nodes n_{act} in the system can be expressed as

$$n_{\rm act} = c_3 \exp\left(c_4 r_{\rm bcast}\right) + c_5 \exp\left(c_6 r_{\rm bcast}\right) \tag{3.8}$$

where c_3 , c_4 , c_5 , and c_6 are constants and their values are presented in Table 3.2.

With these parameters, one can always triangulate a sensor network into a graph G of triangles by regarding each active sensor node as a vertex and connect the 6 nearest

vertices with edges. The probability of a 3-coverage detection can then be regarded as the probability for enclosing a triangle completely by placing a circle of radius $R_{\rm d} = r_{\rm sen} + r_{\rm eff}$ m randomly onto the graph G. If the sensor nodes are evenly distributed and the number of sensor nodes N is large enough to fully cover the sensing terrain, all triangles in graph G can be assumed as equilateral triangles with side length equal to $\bar{r}_{\rm sep}$ m. The situation is illustrated in Figure 3.2.



Figure 3.2: Translating a coverage problem into a clean tile problem.

In Figure 3.2, the big equilateral triangles represent the triangles formed by connecting the neighboring active sensor nodes together. An active sensor node is located at each vertex. The 2 circles are concentric circles. The outer circle represents the detectable area. The inner circle is defined as the *imaginary circle*. The imaginary circle has a radius of r_{img} m. If the side of the equilateral triangles is greater than $\sqrt{3}R_{\rm d}$ (i.e. $\bar{r}_{\rm sep} > \sqrt{3}R_{\rm d}$), a triangle can never be covered by the detectable area completely (i.e. $\Pr(3\text{Cov}) = 0$). On the other hand, if the side of the equilateral triangles is less than $R_{\rm d}$ such that at least one triangle can be covered by the detectable area at any particular instance, $\Pr(3\text{Cov}) = 1$. For $R_{\rm d} \leq \bar{r}_{\rm sep} \leq \sqrt{3}R_{\rm d}$, $\Pr(3\text{Cov})$ can be calculated as follows.

Suppose the target has moved to a location such that 2 vertices of $\triangle ABC$ touch the inside of the detectable area (Figure 3.2). Adjust the radius r_{img} of the imaginary circle until the imaginary circle is just touching two sides of $\triangle ABC$. The radius r_{img} can be expressed as

$$r_{\rm img} = \frac{\sqrt{\bar{r}_{\rm sep}^2 - (\frac{1}{2}\bar{r}_{\rm sep})^2} - \sqrt{R_{\rm d}^2 - (\frac{1}{2}\bar{r}_{\rm sep})^2}}{2}$$
(3.9)

The problem can now be regarded as a *clean tile problem* [72]. The probability of a 3-coverage detection is equal to the probability of having a clean tile of the imaginary circle inside an equilateral triangle. A clean tile is possible if the center of the imaginary circle is located within the small triangle (\triangle XYZ) in Figure 3.2. The probability is therefore expressed as the area of \triangle XYZ divided by that of \triangle ABC.

$$\Pr(3\text{Cov}) = (1 - \frac{2\sqrt{3}r_{\text{img}}}{\bar{r}_{\text{sep}}})^2$$
(3.10)

With (3.5) and (3.10), the corresponding E(DD) can be calculated. The expected energy consumption per successful detection is written as

$$E(ECSD) = \frac{n_{act}(\varrho E_{SEN} + \varphi(E_{TX/RX} + E_{AMP} r_{bcast}^2))}{\Pr(3Cov)} + \frac{(N - n_{det})(\varphi E_{TX/RX})}{\Pr(3Cov)}$$
(3.11)

where ρ is the number of bits collected by the sensor module in each round and φ is the length of an LCP.

3.5.3 Probabilistic Model of the Proposed Soft Deployment Scheme

In this section, the proposed soft deployment scheme will be expressed analytically in terms of Markov chains. Basically, any distribution of a stochastic process for time t that depends only on what happened at time t - 1 can be described in terms of a Markov process. A Markov process can be expressed using a Markov chain.

The operation of each sensor node can be described by a time sequence of random variables X_t , i.e., $X_0, X_1, X_2 \cdots$. The values of these random variables are selected from a set of states D. This set of states consists of three main subsets S_t , L_t and A_t , which represent the sleep state, the listen state and the active state at round t, respectively.

The state subset S_t consists of a + 1 states (denoted as $S_{t,0}, \dots, S_{t,a}$). Each of these states represents a sleep state with a different value of remaining sleeping duration. The state subset L_t consists of b + 1 states (denoted as $L_{t,0}, \dots, L_{t,b}$). Each of these states represents a listen state with a different number of positive evaluation packets received from a node's neighbors. Finally, the state subset A_t consists of a state sub-subset $A_{t,s}$ and a state $A_{t,u}$. The state sub-subset $A_{t,s}$ represents active states corresponding to the successful capture of a target at round t. Note that an active node succeeds to capture a target will remain active until it fails to do so. This means that an active node may stay active with a time duration between 1 and $\frac{2R_d}{s}$ rounds depending on their orientations relative to the moving target. Therefore, the state sub-subset $A_{t,s}$ will consist of $c = \frac{2R_d}{s}$ states (denoted as $A_{t,s,1}, \dots, A_{t,s,c}$). $A_{t,u}$ represents an active state corresponding to the unsuccessful capture of a target in round t.

Therefore, a sensor node with a maximum sleeping duration of a rounds, b neighbors and an active duration bounded by 0 and c will have a state set of size |D| = a+b+c+3. For sensor nodes in the sleep state, the probabilities of the intra-subset transitions can be written as

$$\Pr(\mathbf{S}_{t,h}|\mathbf{S}_{t-1,g}) = \begin{cases} 1, & h = g - 1\\ 0, & \text{otherwise} \end{cases}$$
(3.12)

where $a \ge g \ge 1$.

A sensor node in the sleep state will always switch to one of the listen states in the coming round when its timer counts to zero. The transition probability for a sensor node in $S_{t-1.0}$ to switch to one of the listen states depends on $Pr(A_{t-1.s})$. Probability $Pr(A_{t-1.s})$ can be expressed as

$$\Pr(A_{t-1.s}) = \sum_{i=1}^{c} \Pr(A_{t-1.s.i})$$
(3.13)

where $Pr(A_{t-1,s_i})$ can be obtained from the state distribution vector w_{t-1} , which will be defined later. The transition probabilities $Pr(L_{t_j}|S_{t-1,0})$ is therefore expressed as

$$\Pr(\mathcal{L}_{t_{-j}}|\mathcal{S}_{t-1.0}) = \mathcal{C}_{j}^{b}(1 - \Pr(\mathcal{A}_{t-1.s}))^{b-j}(\Pr(\mathcal{A}_{t-1.s}))^{j}$$
(3.14)

where $j = 0, 1, 2, \dots, b$. A node in the listen state can switch to either the sleep state or the active state in the coming round. The transition probability $Pr(S_{t_h}|L_{t-1_j})$ is given by

$$\Pr(\mathbf{S}_{t_h}|\mathbf{L}_{t-1_j}) = \begin{cases} 1 - \kappa^j, & h = \lfloor (1 - \kappa^j) \times a \rceil \\ 0, & \text{otherwise} \end{cases}$$
(3.15)

where $j = 0, 1, 2, \dots, b$.

The transition probability for a node in the listen state to switch to the active state without successfully capturing a target is represented by

$$\Pr(\mathbf{A}_{t_u}|\mathbf{L}_{t-1_j}) = \kappa^j \left(1 - \frac{\pi R_{\mathrm{d}}^2}{wd}\right)$$
(3.16)

where $j = 0, 1, 2, \dots, b$. As mentioned above, a successful active node may stay active for a period of time depending on its orientation relative to the target. For a target with velocity *s* m/round, the detectable area will move at the same velocity. Therefore, the transition probability $Pr(A_{t.s.i}|L_{t-1.j})$ should depend on the overlapping area of the detectable areas in different rounds. Let $f(\Delta t)$ be the overlapping area of the detectable areas with time difference Δt , i.e.,

$$f(\Delta t) = 2\left(\arccos\left(\frac{s\Delta t}{2R_{\rm d}}\right)\right)R_{\rm d}^2 - \left(\sqrt{R_{\rm d}^2 - \left(\frac{s\Delta t}{2}\right)^2}\right)s\Delta t \tag{3.17}$$

Therefore, the probability for a node in the listen state to switch to the active state and to successfully capture a target is

$$\Pr(\mathbf{A}_{t.s.q}|\mathbf{L}_{t-1.j}) = \kappa^{j} \left(\frac{\pi R_{d}^{2}}{wd}\right) \\ \times \frac{(\pi R_{d}^{2} - f(q)) - (\pi R_{d}^{2} - f(q-1))}{\pi R_{d}^{2}} \\ = \kappa^{j} \left(\frac{f(q-1) - f(q)}{wd}\right)$$
(3.18)

where $q = 1, 2, \dots, c$. A sensor node in $A_{t.s.1}$ will become unsuccessful in the coming round. A sensor node in the subset $A_{t-1.s.q}$ will always switch to the subset $A_{t.s.q-1}$ in the next round, where $q = 2, 3, \dots, c$. Thus,

$$\Pr(A_{t_u}|A_{t-1_s_1}) = 1$$
(3.19)

$$\Pr(\mathbf{A}_{t_s_p}|\mathbf{A}_{t-1_s_q}) = \begin{cases} 1, & p = q - 1, \\ 0, & \text{otherwise.} \end{cases}$$
(3.20)

Finally, a sensor node in the subset A_{t-1} will switch to one of the states S_{t-h} randomly, where $h = 1, 2, \dots, a$, i.e.,

$$\Pr(S_{t_{-}h}|A_{t-1_{-}u}) = \frac{1}{a}$$
(3.21)

where $h = 1, 2, \dots, a$. All other transition probabilities not defined above are zero. With all these transition probabilities, a transition matrix Q_{t-1} can be constructed. The subscript t - 1 indicates that the transition matrix includes the changing parameter $Pr(A_{t-1,s})$. The transition matrix Q_{t-1} is a $|D| \times |D|$ square matrix. Each entry is the transition probability for a node in a particular state at time t - 1 to switch to another state at time t.

The performance of the proposed soft deployment scheme can be estimated with the help of a state distribution vector. The state distribution vector w_t is a |D|-dim vector. Each of the entries represents the probability for a sensor node to fall in a particular state. The summation of all entries in w_t will always be 1. According to the proposed soft deployment scheme, all sensor nodes will start in the sleep state with random sleeping durations. Thus, for $\Lambda \in D$, entries of w_0 are expressed as

$$\Pr(\Lambda) = \begin{cases} \frac{1}{a}, & \Lambda = \mathcal{S}_{0_m} \text{ and } m = 1, 2, \cdots, a \\ 0, & \text{otherwise.} \end{cases}$$
(3.22)

The state distribution during the operation of the proposed scheme can be calculated by the following iterative equation.

$$w_t = Q_{t-1}w_{t-1}, \quad t = 1, 2, \cdots$$
 (3.23)

At each iteration, $Pr(A_{t,u})$, $Pr(A_{t,s})$ and $Pr(L_t)$ are extracted from w_t to estimate the performance of the proposed scheme, while $Pr(A_{t,s})$ is used to calculate Q_t .

For a sensor network of N sensor nodes, the probability for a target to be detected by at least ϑ sensor nodes simultaneously at time t is equal to

$$\Pr(N,\vartheta,t) = 1 - \sum_{\gamma=0}^{\vartheta} \left(C_{\gamma}^{N} (1 - \Pr(\mathbf{A}_{t.s}))^{N-\gamma} (\Pr(\mathbf{A}_{t.s}))^{\gamma} \right)$$
(3.24)

With the above equation, E(THR) of a sensor network of N sensor nodes can be calculated by

$$E(THR) = \frac{\sum_{t=1}^{t_{d}} Pr(N, 3, t)}{t_{d}}$$
(3.25)

where $t_{\rm d}$ is the number of rounds needed by a target to move across the sensing field.

Similarly, E(DD) can be expressed as

$$E(DD) = \sum_{t=2}^{t_{d}} \left\{ \left(\prod_{\tau=0}^{t-2} (1 - \Pr(N, 3, \tau)) \right) \times (t - 1) \right. \\ \left. \times \left(\sum_{\nu=0}^{2} \Pr(N - \nu, 3 - \nu, t) \right. \\ \left. \times \left(C_{\nu}^{N} (1 - \Pr(A_{t-1.s}^{*}))^{N-\nu} (\Pr(A_{t-1.s}^{*}))^{\nu} \right) \right) \right\}$$
(3.26)

Note that a sensor node in $A_{t-1.s.1}$ at time t-1 has no effect on DD at time t. Therefore in (3.26), the term $Pr(A_{t-1.s}^*)$ is expressed as

$$\Pr(\mathbf{A}_{t-1.s}^{*}) = \sum_{i=2}^{c} \Pr(\mathbf{A}_{t-1.s.i})$$
(3.27)

E(ECSD) can be expressed in terms of the total number of sensor nodes n, the state distribution vector and the amount of energy dissipated at each state. Before the proposed soft deployment scheme begins, sensor nodes are required to adjust their communication radii adaptively. Each sensor node will begin with a communication radius of 0.5(b-1) m which will be increased with a step size of 0.05b. Suppose the system has n sensor nodes and the sensing terrain is $w \times d$ m². On average, a sensor node with approximately b neighbors will have an averaged communication radius of \bar{r}_{com} m, where \bar{r}_{com} equals

$$\bar{r}_{\rm com} = \sqrt{\frac{(b+1)wd}{N\pi}} \tag{3.28}$$

For each sensor node, a portion of the energy is consumed in broadcasting the DPP to and receiving the RP from its neighbors. At the same time, another portion of the

energy is consumed in receiving the DPP from and transmitting the RP to its neighbors. Assuming that both a DPP and a RP are having the same length as an LCP (i.e. φ bits) and that the sensor nodes are evenly distributed, the total energy consumed in the density probing process per sensor node ($E_{\rm DP}$) can be written as

$$E_{\rm DP} = \sum_{u=1}^{\sigma} \left\{ \varphi(\iota_u + 1)(0.5(b-1)) + 0.05(b-1)u \right\}^2 E_{\rm AMP} + 2\iota_u \varphi E_{\rm TX/RX} \right\}$$
(3.29)

where $\sigma = \lfloor \frac{\bar{r}_{\rm com} - 0.5(b-1)}{0.05b} \rceil$ and the variable ι_u is equal to

$$\iota_u = \left\lceil \frac{N}{wd} \times \pi \times (0.5(b-1) + 0.05bu)^2 \right\rceil$$
(3.30)

Assume that data collected by the sensor module in each round are ρ bits long. When the proposed soft deployment scheme begins, a sensor node in the active state will dissipate $E_{\rm A} = \rho \times E_{\rm SEN} + \varphi \times \bar{r}_{\rm com}^2 \times E_{\rm AMP}$ J per round, while a sensor node in the listen state with j active neighbors will dissipate $E_{\rm Lj} = j \times \varphi \times E_{\rm TX/RX}$ J per time step.

The expected energy consumption per successful detection E(ECSD) can be described using the following equation.

$$E(ECSD) = \frac{N}{E(THR)} \times \left\{ E_{DP} + \sum_{t=1}^{t_{d}} \frac{E_{A}Pr(A_{t}) + \sum_{j=0}^{b} E_{L_{j}}Pr(L_{t,j})}{t_{d}} \right\}$$
(3.31)

In (3.31), the term $Pr(A_t)$ is expressed as

$$\Pr(\mathbf{A}_{t}) = \Pr(\mathbf{A}_{t_u}) + \sum_{i=1}^{c} \Pr(\mathbf{A}_{t_s_i})$$
(3.32)

3.6 Simulations and Evaluations

Simulations are carried out using MATLAB. In each simulation, a sensor network of either 300, 500 or 700 sensor nodes will be distributed randomly in a sensing field of



Figure 3.3: Target hit-rate of the random on-off soft deployment scheme.

 $50 \times 50 \text{ m}^2$. Initially, each sensor node is given 5 J of energy. When the residual energy drops below 0 J, the node is regarded as depleted and non-functioning. Sensing radius of each sensor node is set to $r_{\text{sen}} = 3 \text{ m}$. Parameter κ is fixed to 0.7. Control packets, including LCP, DPP, and RP, are all having the same length of $\varphi = 200$ bits. Data collected by the sensor module in each round is assumed to be 10 times larger than a control packet such that $\rho = 2000$ bits. $E_{\text{TX/RX}}$ and E_{AMP} are equal to 50×10^{-9} J/bit and 100×10^{-12} J/bit/m², respectively. The energy consumed in capturing one bit of data from the target is approximately equal to the energy consumed in receiving one bit of data from a neighbor node [46] and therefore $E_{\text{SEN}} = 50 \times 10^{-9}$ J/bit.

The moving target is having an effective radius of $r_{\text{eff}} = 5$ m and a constant velocity of s = 0.5 m/round. To eliminate the effect of the boundaries, a target will emerge at



Figure 3.4: Detection delays of the random on-off soft deployment scheme.

 $x = R_{\rm d} = 8$ m of the sensing field with its vanish place located at $x = 50 - R_{\rm d} = 42$ m. The centroid of the target is only allowed to move within the region where $8 \le x \le 42$ m and $8 \le y \le 42$ m. As a result, $t_{\rm d} = \frac{42-8}{0.5} = 68$. At any time, there will be one and only one target moving in the sensing field.

The ROF and SOF soft deployment schemes described in Section 3.5 are evaluated together with the proposed soft deployment scheme for comparison purposes. The performance indicators of each soft deployment scheme are affected by changing the parameter(s) of each scheme. The parameters for the ROF and SOF soft deployment schemes are p and r_{bcast} , respectively. The proposed scheme has three parameters, which are a, b, and κ . To simplify the tuning process, only a is adjustable. Parameters b and κ are fixed at 3 and 0.7, respectively.



Figure 3.5: Energy consumption per successful detection of the random on-off soft deployment scheme.

Each soft deployment scheme will undergo simulations to investigate the relations among its parameter(s) and the performance indicators. Each single simulation will last for 1000 rounds. For different network sizes, 100 simulations will be used to obtain the averaged values of THR, DD, and ECSD. The probabilistic model of each soft deployment scheme is used to calculate the expected values of the performance indicators in each simulation setting. Results are shown in Figs. 3.3–3.11.

For the ROF soft deployment scheme, an increase in p will turn more sensor nodes into the active state. As shown in Figure 3.3, THR increases exponentially as p increases. The rise in THR causes a fall in DD. The ECSD of the ROF soft deployment scheme is in linear relation with the parameter p. For the SOF soft deployment scheme, an increase



Figure 3.6: Target hit-rate of the selective on-off soft deployment scheme.

in r_{bcast} will put more sensor nodes into the listen state. THR descends when r_{bcast} extends. On the contrary, DD goes up when THR goes down. Increasing r_{bcast} will show a positive effect to the reduction of ECSD at the beginning. However, over lengthening r_{bcast} will reduce Pr(3Cov) and cause ECSD to increase dramatically. For the proposed soft deployment scheme, an increase in *a* will put more sensor nodes into the sleep state. All three performance indicators of the proposed soft deployment scheme show similar behaviors to those of SOF as parameter *a* increases. Nevertheless, the changes are more gradual. As shown in the figures, the simulation results match closely with the probabilistic model for all three soft deployment schemes.

According to the results obtained, one can easily notice that the ROF soft deployment scheme can have a THR close to 100% and a DD of almost 0 time slots by increasing *p*.



Figure 3.7: Detection delays of the selective on-off soft deployment scheme.

Similar results can be obtained for the SOF soft deployment scheme by reducing r_{com} . However, both adjustments will dramatically increase ECSD which is not suggested in wireless sensor networks especially when the interested targets are having low velocity and rarely appear. Therefore, it is more reasonable to consider situations in which a short DD is allowed and a slight reduction in THR is acceptable.

In the following simulations, each of the soft deployment schemes under evaluation is tuned to minimize ECSD, provided that a THR of at least 98% and a DD of at most 1 time slot is maintained. Again, each single simulation will last for 1000 rounds. For different network size, 100 simulations will be used to obtain the averaged values of THR, DD, and ECSD. Simulation results are given in Figure 3.12.

When comparing with the SOF and the ROF soft deployment schemes, the proposed



Figure 3.8: Energy consumption per successful detection of the selective on-off soft deployment scheme. Note that in the simulations, ECSD becomes infinite when r_{bcast} is beyond 7.5 m (where n=300,500) and 8 m (where n=700).

scheme can reduce ECSD from a minimum of 42.17% to a maximum of 57.65%. Such performance improvement is possible because the proposed scheme is able to *adaptively* perform periodic on-off soft deployment which only wakes up the necessary number of nodes while keeping most other nodes in the sleep state. The inter-communications among sensor nodes are kept to minimum by controlling the number of neighbors per sensor node, thus ECSD is only mildly affected by network size n. If the requirements on THR and DD can be further relaxed such that a soft deployment scheme is only required to maintain a THR of at least 96% and a DD of at most 2 time slots, the energy saving performance of the proposed scheme is even more promising when comparing



Figure 3.9: Target hit-rate of the proposed soft deployment scheme.

with the SOF and ROF soft deployment schemes. Under such requirement relaxation, the proposed scheme can reduce the ECSD by a minimum of 59.4% to a maximum of 72.59% as shown in Figure 3.13.

3.7 Discussions of Results

In this section, the proposed soft deployment scheme will undergo simulations to investigate the effect of changing the network size and varying each parameter on the three performance indicators. The number of sensor nodes n will be varied from 300 to 700 with a step size of 200. Parameter b will be varied from 1 to 5 with a step size of 1. The maximum sleep time a will be varied from 16 rounds to 160 rounds with a step size


Figure 3.10: Detection delays of the proposed soft deployment scheme.

of 16 rounds. Parameter κ will be varied from 0.5 to 0.9 with a step size of 0.2. The effects of these parameters to the three performance indicators will be evaluated using the Pearson's correlation coefficient. The significant levels of the correlation coefficients are determined by using the *p*-values at 95% confidence intervals.

3.7.1 Expected Target 3-Coverage Hit-Rate

3.7.1.1 Effect of *n*

A positive correlation between parameter n and THR is observed. The correlation is significant only when r_{eff} is small (1–3 m). This is due to the negative feedback mechanism of the evaluation packets which controls the sensing power to an essential level without



Figure 3.11: Energy consumption per successful detection of the proposed soft deployment scheme.

over monitoring the environment.

3.7.1.2 Effect of *a*

A significant negative correlation between parameter a and THR is observed for all cases. When a increases, more sensor nodes will be in the sleep state. This will greatly affect the tracking ability of a sensor network and lower the 3-coverage hit-rate.

3.7.1.3 Effect of *b*

The correlation between b and THR is significantly positive when $r_{\rm eff}$, a and κ are relatively small ($r_{\rm eff} = 1-3$ m, $a \le 48$, $\kappa = 0.5-0.7$). This is because when the sensing



Figure 3.12: The minimum ECSD of the proposed, the generic SOF, and the generic ROF soft deployment schemes with THR≥98% and DD≤1.

ability of the system is low, increasing the number of neighbors can cause more nodes to become active and compensate for the shortage of sensing power.

3.7.1.4 Effect of κ

A positive correlation between parameter κ and THR is observed under all combination of parameters. However, the relation is not significant in most cases. Parameter κ is suggested to be used as an auxiliary tuning parameter.



Figure 3.13: The minimum ECSD of the proposed, the generic SOF, and the generic ROF soft deployment schemes with THR≥96% and DD≤2.

3.7.2 Expected Detection Delay

3.7.2.1 Effect of *n*

A negative correlation is observed between n and the expected detection delay. However, the correlation is not considered as significant under all conditions. This phenomenon shares the same explanation as given in Section 3.7.1.1.

3.7.2.2 Effect of *a*

A significant positive correlation between parameter a and DD is observed all the time. When a increases, sensor nodes will sleep for a longer time, hence reducing the chance for an event to have 3-coverage. The correlation is most significant when parameters n and $r_{\rm eff}$ are relatively large (n = 700, $r_{\rm eff} = 5$ m). Parameters b and κ have mild effects on this correlation.

3.7.2.3 Effect of *b*

A significant negative correlation between parameter b and DD can only be observed when parameters n is small (n = 300-500), a is low ($a \le 48$) and r_{eff} is small ($r_{\text{eff}} =$ 1 m). The explanation is similar to the one given in Section 3.7.1.3.

3.7.2.4 Effect of κ

A negative correlation between parameter κ and DD is observed under all combinations of parameters. Similar to the observations made in Section 3.7.1.4, the correlation is not significant in most cases. Therefore, it is again suggested to use κ as an auxiliary tuning parameter.

3.7.3 Expected Energy Consumption Per Successful Target Detection

3.7.3.1 Effect of *n*

The correlation between n and ECSD changes from positive to negative with increasing magnitudes as parameters r_{eff} and κ increases. Therefore, to reduce energy consumption, κ should be reduced when more nodes are decided to be deployed. Parameter b shows little effect on the results.

3.7.3.2 Effect of *a*

The correlation between a and ECSD is found to be significant and positive when r_{eff} is small (1 m). This positive correlation is the strongest when parameter n is also small.

When the effective radius of the event is large (3–5 m), a transition from a strong positive to a strong negative correlation is observed between a and ECSD. In region where the correlation is negative, systems with number of neighbors b equal to 1 show the strongest correlation. A large value of κ ($\kappa = 0.9$) will further increase the magnitude of the correlation when the correlation between a and ECSD is in the negative region.

3.7.3.3 Effect of b

Under all conditions, parameter *b* shows a positive correlation with ECSD. By increasing the number of neighbors per node, each sensor node will have to broadcast its evaluation results for a bigger region and consumed more energy.

3.7.3.4 Effect of κ

The correlation between κ and ECSD shows a positive value in most cases. The values become significant when *n* is large (n = 500-700), r_{eff} is large ($r_{\text{eff}} = 3-5$) and *a* is small. This concurs with the result found in Section 3.7.3.1 that parameter κ should be reduced for systems with a large number of sensor nodes.

3.8 Chapter Summary

An energy efficient soft deployment scheme is proposed. Simulation results show that, when comparing with other generic soft deployment schemes, the proposed scheme can reduce the energy consumption per successful target detection by a minimum of 42.17% to a maximum of 72.59% while keeping the target 3-coverage hit-rate and detection delay at acceptable values. The proposed scheme is very suitable for rare and slow target

tracking applications where energy saving is crucial. The performance of the proposed scheme is only mildly affected by the network size. The analysis of parameters show that for specific applications, the performance can be further biased to either having a higher target detection capability or a lower energy consumption by tuning some parameters.

Chapter 4

Network Structure

4.1 Introduction

Adaptability, comprehensive sensing coverage, and high fault tolerance are some of the unique advantages of wireless sensor networks. Wireless sensor networks consist of large amounts of wireless sensor nodes, which are compact, light-weighted, and battery-powered devices that can be used in virtually any environment. Because of these special characteristics, sensor nodes are usually deployed near the targets of interest in order to do close-range sensing. The data collected will undergo in-network processes and then return to the user who is usually located in a remote site. Most of the time, the targets of interest are located in extreme environments, which are too hostile for the maintenance of the sensor nodes. Sensor nodes must conserve their scarce energy by all means. Failure to do so may reduce the sensing coverage or even cause the shutting down of the whole network. Thus, energy conservation is often an essential design consideration of sensor networks.

As mentioned in Chapter 2, much prior work has focused on conserving energy by clustering. A network with clustering is divided into several clusters. Within each cluster, one of the sensor nodes is elected as the *cluster head* (CH), the rest being the *cluster members* (CM). The cluster head will collect data from its cluster members directly or in a multi-hop manner. To facilitate parallel data transmissions and alleviate interference among wireless sensor nodes, code division multiple access (CDMA) is usually employed in cluster-based wireless sensor networks [28, 29, 30]. By organizing wireless sensor nodes into clusters, energy dissipation is reduced by decreasing the number of nodes involved in long distance transmission [25]. The number of data transmissions and energy consumption can be further reduced by performing data/decision fusion on nodes along the data aggregation path. Clustering provides a significant improvement in energy saving. Due to the distributed nature of wireless sensor networks, asynchronous half-duplex direct-sequence CDMA is usually used in CDMA-based wireless sensor networks. To avoid data collision, cluster heads will collect data from their cluster members one by one. Let T be the average transmission delay among nodes. Assume that a node is always capable of fusing all incoming packets into a single packet by means of data/decision fusion techniques and the size of an aggregated packet is the same as an incoming packet. Referring to the situation shown in Figure 4.1 (a), a base station will take $4 \times T$ to collect a complete set of data from the network. Apart from requiring a longer delay in data collection, cluster members will need larger buffers to handle the incoming data while waiting for the belonging cluster head to become available. As a result, a considerable amount of energy is wasted while waiting. The bottleneck problem mentioned above can be alleviated by modifying the network structure. By transforming the original network into a multi-hop network as shown in Figure 4.1 (b), it can be shown



Figure 4.1: (a) Data collection in a single cluster 2-hop network and (b) data collection in an improved single cluster multi-hop network. Circles with CM represent the cluster members. Circles with CH represent the cluster heads. Filled circles with BS represent the base stations. Dashed arrows show the direction of data flow.

that the time needed by the base station to collect a full set of data from the network can be reduced to $3 \times T$.

The aim of this chapter is to investigate the characteristics of a delay-aware data collecting network structure in CDMA-based wireless sensor networks. Two algorithms for forming such a network structure are proposed for different scenarios. The algorithms will form networks with minimum delays in the data collecting process. At the same time, the algorithms will try to keep the transmission distance among wireless sensor nodes at low values which is minimizing the energy consumption of the network. The rest of the chapter is organized as follows. Section 4.2 briefly reviews related work. Section 4.3 defines the proposed network structure. Section 4.4 explains the algorithm for forming the proposed network structure in different scenarios. A numerical analysis is given in Section 4.5 to show how different network structures perform in terms of delays in data collecting processes. Simulation results and their analysis will be given in Sections 4.6 and 4.7. Finally, Section 4.8 summarizes the chapter.

4.2 Related Work

Due to the energy constraint of individual sensor nodes, energy conservation becomes one of the major issues in sensor networks. In wireless sensor networks, a large portion of the energy in a node is consumed in wireless communications. The amount of energy consumed in a transmission is proportional to the corresponding communication distance. Therefore, long distance communications between nodes and the base station are usually not encouraged. One way to reduce energy consumption in sensor networks is to adopt some clustering algorithms [25]. A clustering algorithm tries to organize sensor nodes into clusters. Within each cluster, one node is elected as the cluster head. The cluster head is responsible for 1) collecting data from its cluster members, 2) fusing the data by means of data/decision fusion techniques, and 3) reporting the fused data to the remote base station. In each cluster, the cluster head is the only node involved in long distance communications. Energy consumption of the whole network is therefore reduced.

Intensive research [28, 29, 30, 73] has been conducted on reducing energy consumption by forming clusters with appropriate network structures. Recently, Heinzelman, Chandrakasan, and Balakrishnan proposed a clustering algorithm called LEACH [28]. In networks using LEACH, sensor nodes are organized in multiple-cluster 2-hop (MC2H) networks (i.e., cluster members \rightarrow cluster head \rightarrow base station). Using the idea of clustering, the amount of long distance transmissions can be greatly reduced. Lindsey and Raghavendra proposed another clustering algorithm called PEGASIS [29], which is a completely different idea by organizing sensor nodes into a single-chain (SC) network. In such networks, sensor nodes are organized in a single chain and a single node on the chain is selected as the cluster head. By minimizing the number of cluster heads, the energy consumed in long distance transmission is further minimized. Tan and Körpeoglu developed PEDAP [30], which is based on the idea of minimum spanning tree (MST). Besides minimizing the amount of long distance transmission, the communication distances among sensor nodes are minimized. Fonseca et. al. proposed the collection tree protocol (CTP) [73]. The CTP is a kind of gradient based routing protocol which uses expected transmissions (ETX) as its routing gradient. ETX is the number of expected transmissions of a packet necessary for it to be received without error [74]. Paths with low ETX are expected to have high throughput. Nodes in a network using CTP will always pick a route with the lowest ETX. In practical situations, the ETX of a path is proportional to the corresponding path length [75]. Thus, CTP can greatly reduce the communication distances among sensor nodes. All these algorithms show promising results in energy saving. However, a network formed by an energy efficient clustering algorithm may not necessarily be desirable for data collection. An analysis on how these network structures perform in terms of data collection efficiency will be given in Section 4.5.

The focus of this chapter is on investigating the data collection efficiency of networks formed by different clustering algorithms. Therefore, event triggering algorithms such as TEEN [33] and APTEEN [34] will not be considered in this chapter. A related work on data collection efficiency was done by Florens, Franceschetti, and McEliece [76]. In their work, lower bounds on data collection time are derived for various network structures. However, the effect of data fusion, which is believed as one of the major features of sensor networks, was ignored. Wang *et. al.* [77] proposed link scheduling algorithms for wireless sensor networks which can raise network throughput considerably. In their work, however, it is assumed that data links among wireless sensor nodes are predefined. In contrast, the objective of this chapter is to form data links among wireless sensor nodes and thus to shorten the delays in the data collection processes. Another related work was done by Solis and Obraczka [78] who studied the impact of timing in data aggregation for sensor networks.

4.3 The Proposed Network Structure

The proposed network structure is a tree structure, where the degree distribution of the cluster members follows an inverse exponential base-2 function, as shown in Table 4.1. A node with degree k will collect data from k - 1 nodes of degree ranging from $1, 2, \cdots$ up to k - 1. At the same time, it will forward the fused data to a node of degree higher than its own. The cluster head will be considered as a special case. The cluster head is the one with the highest degree in the network. Instead of communicating with a node of higher degree, the cluster head will forward the fused data directly to the base station. An example of the proposed network with N = 16 is shown in Figure 4.2. In this example, it takes $5 \times T$ for the base station to collect all data from 16 nodes. By dividing the time domain into time slots of durations T, the above process will last for 5 time slots. It will be proven in Section CH5:numerical that in order to deliver the maximum efficiency of

Table 4.1: Cluster members degree distribution of the proposed network structure with network size $N = 2^q$, where $q = 1, 2, \cdots$.

Degree12
$$\cdots$$
 $\log_2 N - 1$ $\log_2 N$ Frequency $\frac{N}{2^1}$ $\frac{N}{2^2}$ \cdots $\frac{N}{2^{(\log_2 N-1)}}$ $\frac{N}{2^{(\log_2 N)}}$



Figure 4.2: Proposed network structure with network size N = 16. Circles with CM represent the cluster members. Circle with CH represents the cluster head. Filled circle with BS represents the base station. Degree of each node is represented by the variable k. Dashed arrows show the direction of data flow.

the proposed network structure, the number of nodes N has to be restricted to $N = 2^q$, where $q = 1, 2, \cdots$. It will also be shown in a later part that such restriction can be relaxed by giving up some performance.

Lemma 1: Assume $N = 2^q$, where $q = 1, 2, \cdots$. Under the proposed network

structure, a node *i* of connection degree $k \ge 2$ (where $k \in \mathbb{Z}$) requires k - 1 time slots to collect data from all its child nodes.

Proof: Assume $N = 2^q$, where $q = 1, 2, \cdots$. For a node of connection degree k = 2, the time slots required for it to collect data from all its child nodes is equal to the number of child nodes it has, which is 1. Thus the case for k = 2 is true. Now let us assume that any node of connection k = n requires n - 1 time slots to collect all data from its child nodes. For node *i* of connection degree k = n + 1, it has *n* directly connected child nodes. Each of these directly connected child nodes has a connection degree uniquely ranging from 0 to n - 1. Thus, they need 0 to n - 1 time slots to collect data from all their sub-child nodes plus one extra time slot to report their aggregated data to node *i*. Therefore, the maximum time slots required for node *i* to collect data from all their sub-child nodes plus one extra time slot to report their aggregated data to node *i*. Therefore, the maximum time slots required for node *i* to collect data from all their sub-child nodes plus one extra time slot to report their aggregated data to node *i*. Therefore, the maximum time slots required for node *i* to collect data from all their sub-child nodes plus one extra time slot to report their aggregated data to node *i*. Therefore, the maximum time slots required for node *i* to collect data from all their slots to collect data from all the nodes is k - 1 = n. By induction, the Lemma is proved.

Theorem 1: Assume $N = 2^q$, where $q = 1, 2, \cdots$. Under the proposed network structure, the number of time slots t(N) required for the base station to collect data from the whole network is given by

$$t(N) = \log_2 N + 1 \tag{4.1}$$

Proof: Assume $N = 2^q$, where $q = 1, 2, \cdots$. Under the proposed network structure, for a network of size N, the cluster head is the only node with the maximum connection degree which is

$$k_{\max} = \log_2 N + 1$$

From Lemma 1, the number of time slots required for a cluster head (of connection degree k_{max}) to collect data from all its child nodes is

$$k_{\rm max} - 1 = \log_2 N$$

Thus, the number of time slots t(N) required for the base station to collect data from the whole network is the time slots required by the cluster head to collect data from all its child nodes plus one, i.e.,

$$t(N) = \log_2 N + 1$$

4.4 Network Formation Algorithm

The data collection efficiency of a sensor network can be greatly improved by adopting the proposed network structure. Nevertheless, the energy consumption of a wireless sensor node is proportional to the transmission distance. To reduce the energy consumption of the network, proper network formation algorithms are essential to construct the desired network structure while keeping the communication distances of the network at low values. In this section, two network formation algorithms are proposed. Both algorithms can organize a network into the proposed structure while keeping the communication distances short.

4.4.1 Top-Down Approach

The top-down approach is a kind of centralized control algorithm. In this approach, the base station is assumed to have the coordinates of all sensor nodes in the network. The

whole approach is going to be carried out at the base station. At the end of the optimization process, the base station will instruct the sensor nodes to establish the appropriate structure.

The network structures for $N = 2^0$ and $N = 2^1$ are trivial. For networks with $N = 2^q$ nodes, where $q = 2, 3 \cdots$, the proposed network structure can be constructed according to the following algorithm.

- 1. The algorithm starts with considering the whole network as a fully connected network. In this chapter, the term *connected* refers to the existence of a data link between two wireless sensor nodes which is used to transmit data packets in the data collecting processes. Two wireless sensor nodes are defined as *disconnected* from each other if there does not exist any direct data link between them. The connection degree of a wireless sensor node is telling the number of data links associated with such node. A node with connection degree of 3 implies that such a node has formed three data links with three other nodes. For a network of $N = 2^q$ nodes, where $q = 2, 3, \dots$, each node will begin with degree equal to N - 1. The nodes will form the set $\hat{H}_{s=1}$. Set $\xi = \frac{N}{2}$.
- Select ξ nodes from set Ĥ_s such that the total communication distances within the ξ-subgraph is maximized. These ξ nodes will form set H_{s+1}. The rest of the nodes from Ĥ_s will form set Ĥ_{s+1}. The algorithm will then remove all connections (data links) among nodes within H_s. Set iterators s←s + 1 and ξ ← ξ/2.
- 3. Repeat step 2 until $\xi < 2$. Set r = 2.
- 4. Nodes with degree N r form set L. Nodes with degree greater than N r form set U such that set L and set U are of the same size. Connections among nodes



Figure 4.3: Network formation of the proposed network structure using centralized top-down approach ($N \ge 4$).

in the two sets are reduced until each node in set L is only connected to a single node in set U. The total communication distances of the outcome can be further optimized. Details of the optimization method are given later in this section. After reducing the number of connections, set $r \leftarrow r \times 2$.

5. Repeat step 4 until r = N.

A flow chart of the network formation algorithm is given in Figure 4.3. The algorithm will produce 2 nodes with degree $\log_2 N$. Among these 2 nodes, the one which is located closer to the base station will be selected as the cluster head and be connected directly to the base station. Therefore the cluster head will have a degree of $\log_2 N + 1$

	А	В	С	D
А	0	13	2	3
В	13	0	7	10
С	2	7	0	5
D	3	10	5	0

Table 4.2: The edge weight matrix of graph G in Example 1

which is the highest within the cluster.

The procedure in step 2 is in fact the *heaviest k-subgraph* problem defined in [79]. The heaviest *k*-subgraph problem is to find the *k*-vertex subgraph out of a given graph, such that the total weight of edges among these *k* vertexes is maximized. In this chapter, since the secondary objective is to minimize the transmission distances among wireless sensor nodes, the weight of an edge (data link) connecting any two vertexes (wireless sensor nodes) is defined as a variable proportional to the geographical distance between them. The proposed algorithm can always be modified to accommodate different cost metrics by redefining the edge weights. The problem is a kind of combinatorial problem which is defined as NP-complete. Dynamic programming is commonly used to solve combinatorial problem [80] and thus it is used to solve the heaviest *k*-subgraph problem in this chapter.

Example 1: This example is to illustrate how to use dynamic programming to solve a typical heaviest k-subgraph problem. Given a fully connected graph G with 4 vertexes, A, B, C, and D. The weights of edges among vertexes are expressed by the edge weight matrix shown in Table 4.2. Suppose the heaviest 3-subgraph is required to be found from



Figure 4.4: Using dynamic programming to solve the heaviest *k*-subgraph problem in Example 1 the graph *G*. The solution can be obtained by carrying out the following procedures.

• Among all the vertexes, arbitrarily select one vertex as the first vertex of the locally heaviest 3-subgraph. The subgraph is labeled as *locally heaviest* because the search space of the dynamic programming is confined by the selection of the first vertex. The solution is therefore a local optimum solution. This local optimum problem will be tackled later. In this example, vertex A is selected as the first vertex. Vertex A can then join with either vertex B, C, or D to form three different 2-subgraphs. Their total edge weight are calculated and shown in Figure 4.4(a).

- Suppose vertex B is selected as the third vertex of the locally heaviest 3-subgraph. According to Figure 4.4(b), the other two vertexes can either be A-C or A-D. The total edge weight of the subgraphs A-C-B and A-D-B are 22 and 26, respectively. Therefore, if vertex A is selected as the first vertex and vertex B is selected as the third vertex, the 3-subgraph formed with the vertexes A, B, and D will have the heaviest total edge weight. If vertex C is selected as the third vertex, the 3-subgraph formed with the vertexes A, B, and C will have the heaviest total edge weight (Figure 4.4(c)). Similarly, if vertex D is selected as the third vertex, vertexes A, B, and D will form a subgraph with the heaviest total edge weight (Figure 4.4(d)).
- By comparing the total edge weight of all the subgraphs obtained above, it can be shown that the 3-subgraph with vertexes A, B, and D can provide the heaviest total edge weight. Note that as mentioned before, the search space is confined by the selection of the first vertex. To avoid being trapped in local optimum points, it is necessary to repeat the above procedures with different vertexes selected as the first vertex. All results are then compared against each other to obtain the globally heaviest 3-subgraph.

The nodes involved in the optimization in step 4 will form a distance matrix with each entry storing the distance between two nodes. The x-axis is representing the nodes from set L while the y-axis is representing the nodes from set U. The two sets of nodes therefore form a bipartite graph and the optimization problem becomes a *weighted matching problem*. This problem can be optimized by applying matching techniques such as Hungarian Method [81, 82] or Munkres' Assignment Algorithm [83]. In this chapter, Munkres' Assignment Algorithm is used to solve the weighted matching problem.

For networks with number of nodes other than $N = 2^k$, where $k = 1, 2 \cdots$, dummy nodes are virtually added in the calculation process to expand the network in order to fulfill the network size requirement of the algorithm. These dummy nodes will have infinite separations with the real nodes and have infinite separations among themselves. The number of dummy nodes will always be smaller than N/2. At the end of the optimization process, these dummy nodes will all have degree of 1 which can be ignored and removed without partitioning the network. Since the condition $N = 2^q$, where $q = 1, 2 \cdots$ is fulfilled during the network formation, *Lemma 1* and *Theorem 1* still applied. Thus, the time slots required for complete data collection will still be governed by equation (4.1), provided that the number of nodes N in equation (4.1) is replaced by the number of real nodes N_r plus the number of dummy nodes N_d . In general, equation (4.1) can be written as

$$t(N) = \log_2 (N_r + N_d) + 1$$

= $\lceil \log_2 N_r \rceil + 1$ (4.2)

where $\lfloor u \rfloor$ denotes the smallest integer equal to or greater than u.

4.4.2 Bottom-Up Approach

Comparing with the top-down approach, the bottom-up approach is more scalable and can be implemented in either centralized or decentralized fashion. Specifically, a decentralized bottom-up approach can be described as follows.

1. In the network construction process below, wireless sensor nodes will operate in

contention based random multiple access control protocol. However, once the network construction process is completed, wireless sensor nodes will communicate in CDMA control protocol. Each node is labeled with a unique identity and marked as level ϵ . The unique identity will only serve as an identification which has no relation with sensor nodes' locations and connections. Here, ϵ is a function of the number of nodes in a cluster. For a cluster of *i* nodes, ϵ is equal to $\log_2 i$. Since nodes are disconnected initially (i.e. no data link exists among wireless sensor nodes), these *N* nodes can be considered as *N* single-node clusters. Each node will therefore be assigned as level 0. Within each cluster, one node will be elected as the sub-cluster head. We denote SCH(ϵ) as a sub-cluster head of a level ϵ cluster. In the bottom-up approach, a SCH can only make connection (i.e. form a data link) with another SCH of the same level. Because there is only 1 node in each cluster, all nodes begin as SCH with $\epsilon = 0$. The dimensions of the terrain (t_x, t_y) are provided to the sensor nodes before deployment.

2. Each SCH performs random back-off and then broadcasts a *density probing packet* (DPP) to its neighboring SCHs which are within a distance of r_{dp} m. Note that the size of a DPP is much smaller than that of a data packet. Energy consumed in transmitting a DPP is negligible to that of a data packet. A SCH can use the number of received DPP, together with the dimensions of the terrain, to estimate the total number of nodes (N_{est}) in the network. A SCH will use the N_{est} to adjust its communication distance r_{com} , which is derived as

$$r_{\rm com} = \frac{\sqrt{t_x^2 + t_y^2}}{\alpha - \beta - \epsilon}, \quad \beta + \epsilon < \alpha \tag{4.3}$$

where $\alpha = \lceil \log_2(N_{est}) \rceil + 1$, β is a constant which is set to 0 initially. The SCH

will use $N_{\rm est}$ to estimate the node density ρ of the network.

3. Each SCH will do a random back off and then broadcast an *invitation packet* (IP) to its neighbors within $r_{\rm com}$ m. The IP contains the level ϵ and the identity of the issuing SCH. Again, an IP is much smaller than a data packet. Energy consumed in transmitting an IP is negligible to that of a data packet. A SCH will estimate the distances to its neighboring SCHs using the received signal strength of the IPs received. The information will be stored in the local database of a SCH. Normally, the expected number of IPs received \aleph is derived as

$$\aleph = \lceil \frac{\rho \pi r_{\rm com}^2 - 1}{\epsilon} \rceil \tag{4.4}$$

A SCH will count the number of IPs received. If the number of IPs received has exceeded \aleph or a maximum duration has been reached, a SCH will search its database for the nearest neighbor with the same level. The SCH will then send a *connection request* (CR) to this nearest neighbor. If both SCHs are the nearest neighbor of each other, a connection will be formed between these 2 SCHs.

4. Once they are connected, the two SCHs and their belonging clusters will form a composite cluster. One of the two involved SCHs will become the chief SCH of the composite cluster and increment its level *ε* by 1. In this thesis, the SCH with higher residual energy will become the chief SCH of the composite cluster. The nodes in the composite cluster will listen to the communication channel and reply any CR from lower levels with a *rejecting packet* (RP). When no more CR from lower levels can be heard, the chief SCH will start to make connection with other SCHs of the same level.

- 5. If a RP is received, a SCH will send a CR to its next nearest neighbor in its database. If such neighbor does not exist, the SCH will adjust its r_{com} by setting β ← β + 1. The SCH will then broadcast an *urgent connection request* (UCP) using the new r_{com}. Upon receiving the UCP, a SCH of the same level will grant the request if it is still waiting for a CR.
- 6. If no connection can be made within a period of time, either all neighbors of the same level are unavailable or all CRs have been rejected, the SCH will increment its β and broadcast the UCP again. This process repeats as long as β + ε < α. If β + ε ≥ α, the SCH will make connection with the base station directly.</p>
- 7. The above processes continue until no more connection can be formed.

When being implemented in a decentralized control manner, the above algorithm may end up with multiple composite clusters if the number of nodes is not equal to 2^q , where $q = 1, 2 \cdots$. SCHs of these composite clusters will communicate with the base station directly. By virtue of pairing up composite clusters of same sizes, the algorithm will end up with composite clusters of completely different sizes. Considering the base station as the root of the network, the number of time slots required by the base station to collect data from all sensor nodes is

$$t(N) = \left\lceil \log_2 \left(N + 1 \right) \right\rceil \tag{4.5}$$

In contrast, the above algorithm can also be carried out at the base station as a centralized control algorithm. The base station is again assumed to have the coordinates of all sensor nodes in the network. When the number of nodes is not equal to 2^q , where $q = 1, 2 \cdots$, dummy nodes can be virtually added in the calculation process, depending

on the application. If a single cluster is required, dummy nodes should be virtually added to fulfill the requirement of $N = 2^q$, where $q = 1, 2 \cdots$. However, if multiple clusters can be formed, dummy nodes are not essential. When dummy nodes are virtually added, these dummy nodes will have infinite separations with the real nodes and with themselves. Note that whenever a real SCH is connecting with a dummy SCH, the real SCH will always be the chief SCH of the composite cluster. This is to ensure the removal of dummy nodes at the end of the calculation process will not partition the network. The number of time slots required by the base station to collect data from all sensor nodes will be governed by equations (4.2) (for single cluster) and (4.5) (for multiple clusters).

4.5 Numerical Analysis

Theorem 2: Assume that each sensor node can only communicate with one sensor node at a time, and that data fusion is applicable. For a single cluster network of $N = 2^q$ nodes, where $q = 1, 2, \dots$, the minimum number of time slots required by the base station to collect data from N sensor nodes is

$$t(N)_{\min} = \log_2 N + 1. \tag{4.6}$$

Proof: In a period of t time slots, a parent node v can collect data from at most t directly connected child nodes, provided that these t child nodes are using different time slots to communicate. Within these t child nodes, the u^{th} node will report data at time slot u, which implies the u^{th} node can collect data from at most u - 1 directly connected child nodes of itself before it has to report data to its parent node. Therefore, for a period of t time slots, a parent node v can receive data from at most 2^t nodes (including itself). On the other hand, the minimum number of time slots required for a parent node

to collect data from N nodes (including itself) is $\log_2 N$. Thus, the minimum number of time slots required for a base station to collect data from N nodes is $t(N)_{\min} = \log_2 N + 1$.

For a single cluster network with N nodes, where $N > 0 | N \in \mathbb{Z}$, the minimum number of time slots required for a base station to collect data from N nodes is

$$t(N)_{\min} = \lceil \log_2 N \rceil + 1 \tag{4.7}$$

From Theorem 1 and equation (4.2), it can be shown that the proposed network structure is an optimum network structure in terms of data collection efficiency provided that

- 1. each sensor node can only communicate with one sensor node at a time,
- 2. data fusion can be carried out at every sensor node, and
- 3. sensor nodes are connected to a single cluster with a single cluster head.

The same idea can be applied to multiple-cluster network by considering the base station as the root of the network structure. Therefore, in multiple-cluster networks, the minimum number of time slots required for a base station to collect data from N nodes is

$$t(N)_{\min} = \lceil \log_2(N+1) \rceil \tag{4.8}$$

Using equation (4.5), it can be shown that the proposed network structure is again an optimum network structure in terms of data collection efficiency provided that

- 1. each sensor node can only communicate with one sensor node at a time,
- 2. data fusion can be carried out at every sensor node, and
- 3. sensor nodes are connected into multiple clusters.

In a MC2H network with N nodes organized in g clusters, where $N \ge \sum_{m=1}^{g} m$, the time slots required by the base station to collect data from all sensor nodes is minimized when all clusters have different numbers of nodes. Therefore, each cluster can communicate with the base station interleavingly. Meanwhile, the number of nodes in the largest cluster should be minimized such that the total number of time slots required by the base station is also minimized. An example for g = 2 is shown below.

Example 2: For a MC2H network of N nodes organized in 2 clusters (where $N \ge$ 3), in order to achieve the maximum data collection efficiency, the number of nodes in these 2 clusters should be equal to

$$\frac{N+1}{2} \text{ and } \frac{N-1}{2}, \text{ for } N \text{ odd}$$

$$\frac{N}{2} + 1 \text{ and } \frac{N}{2} - 1, \text{ for } N \text{ even}$$
(4.9)

The minimum number of time slots $t(N)_{min}$ required by the base station to collect data from all sensor nodes is equal to the number of nodes in the largest cluster. Therefore,

$$t(N)_{\min} = \begin{cases} \frac{N+1}{2}, & N \text{ is odd} \\ \frac{N}{2} + 1, & N \text{ is even} \end{cases}$$
(4.10)

In general, for a MC2H network of N nodes organized in g clusters, where $N \geq \sum_{m=1}^{g} m$, the number of nodes in the j^{th} cluster can be written as

$$\lfloor \frac{N - S_g + (j-1)(g+1)}{g} \rfloor + 1, \quad j = 1, 2, 3, \cdots, g$$
(4.11)

where $\lfloor u \rfloor$ denotes the largest integer equal to or smaller than u and $S_g = \sum_{m=1}^g m = g(g+1)/2$. Thus, the minimum number of time slots $t(N)_{\min}$ required by the base station to collect data from all sensor nodes is equal to

$$t(N)_{\min} = \lfloor \frac{N - S_g + (g - 1)(g + 1)}{g} \rfloor + 1.$$
(4.12)

Based on (4.11), the optimum number of clusters g_{opt} for a MC2H network in terms of data collection efficiency can be obtained from the following inequality:

$$\frac{(1+g)g}{2} \ge N$$

$$\Rightarrow \quad g \ge \frac{-1+\sqrt{1^2+8N}}{2}$$
(4.13)

2

$$\Rightarrow \quad g_{\text{opt}} = \lceil \frac{-1 + \sqrt{1^2 + 8N}}{2} \rceil \tag{4.14}$$

where N is the number of nodes in the network, and g is the number of clusters.

Theorem 3: For a MC2H network of N nodes organized in g clusters, where $g \le N < S_g$, where $S_g = \sum_{m=1}^g m$. The minimum number of time slots $t(N)_{\min}$ required by the base station to collect data from all sensor nodes is equal to the number of clusters in the system, i.e., g.

Proof: Consider a MC2H network of N nodes organized in g clusters, where $N = S_g$ and $S_g = \sum_{m=1}^g m$. To achieve the maximum data collection efficiency, these g clusters will all have different numbers of nodes ranging from 1 to g. The minimum number of time slots $t(N)_{\min}$ required by the base station to collect data from all sensor nodes is equal to the number of nodes in the largest cluster, which is g. Suppose one node has to be removed from the network such that N is reduced to N - 1. To maintain the number of clusters in the network, this particular node must be removed from one of the clusters except the one with a single node. Removing this node from the cluster, if not the largest one, will have no effect on the time slots required by the base station to collect data from all sensor nodes. However, removing this node from the largest cluster will cause two clusters to have the same number of nodes, which is g - 1. An extra time slot is needed for these 2 largest clusters to do interleaving. Therefore, the minimum number of time slots $t(N)_{\min}$ required by the base station is always equal to the number

of clusters in the system.

In contrast, for a MC2H network with N nodes organized in g clusters, where $N \ge g$, the number of time slots required by the base station to collect data from all sensor nodes is maximized when N - (g - 1) nodes belong to the same cluster. The remaining g - 1clusters will all have a cluster size of 1, and we have

$$t(N)_{\max} = \begin{cases} N - (g - 1), & N > 2(g - 1) \\ N - (g - 1) + 1, & N = 2(g - 1) \\ g, & \text{otherwise} \end{cases}$$
(4.15)

In a SC network, the number of time slots required by the base station to collect data from all sensor nodes is minimized when the cluster head is at the middle of the chain, i.e.,

$$t(N)_{\min} = \begin{cases} \frac{N+1}{2} + 1, & N \text{ is odd} \\ \frac{N}{2}, & N \text{ is even} \end{cases}$$
(4.16)

where N is the number of nodes in the network.

On the contrary, in a SC network, the number of time slots required by the base station to collect data from all sensor nodes is maximized when the cluster head is at the end of the chain, i.e.,

$$t(N)_{\max} = N \tag{4.17}$$

where N is the number of nodes in the network.

In networks using MST and CTP, the number of time slots required by the base station to collect data from all sensor nodes is lower bounded by equations (4.6) and (4.7). On the other hand, the number of time slots required by the base station to collect data from all sensor nodes is maximized when the resultant networks of MST and CTP are in single cluster 2-hop structure which is upper bounded by $t(N)_{max} = N$.



Figure 4.5: Averaged data collecting time of different single tree structures. Note that results obtained from the proposed algorithm using the top-down approach agree perfectly with those obtained from the bottom-up approach.

4.6 Simulations

In this section, the proposed network structure will be compared with the MC2H network, the SC network, the MST network, and the CTP network. Networks with nodes N varying from 4 to 64, with a step size of 4, will be distributed randomly and evenly on a sensing field of $50 \times 50 \ m^2$. The center of the sensing field is located at (x, y) = (25 m, 25 m). In the simulations, it is assumed that a node can either receive or transmit at any time. It is also assumed that a node is always capable of fusing all incoming packets into a single packet by means of data/decision fusion techniques and the size of an aggregated packet is the same as an incoming packet. The channel condition is assumed to be



Figure 4.6: Averaged ψ of different single tree structures.

perfect. For each network, the averaged data collecting time (DCT) will be used to indicate its data collection efficiency. Besides being efficient in the data collecting process, the energy consumption of the network should always be kept at low values. Assuming that a data packet is much larger than a control packet and the network construction processes are rarely triggered, most of the energy in wireless sensor nodes are consumed by their transmitting units during the data collecting processes. If the transmission distance is d m long and the path loss exponent is equal to h, to make sure data can be received at the receiving end, the energy required by the transmitting unit is proportional to d^h . Therefore, the energy consumption of a network can be estimated using the communication distances among wireless sensor nodes. In this chapter, the energy consumption of a network is estimated by the following function:

$$\psi = \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} c_{ij} d^{h}_{ij}$$
(4.18)

where c_{ij} is an indicator to indicate the present $(c_{ij}=1)$ or absent $(c_{ij}=0)$ of a connection between node *i* and node *j*. Here, d_{ij} is the distance between node *i* and node *j*. In the simulations, the path loss exponent *h* is assumed to be 2. All results presented in this chapter are taken from the average of 50 simulations.

For convenience, the network structures under test are classified into two types: Type I) single-cluster network structure; Type II) multiple-cluster network structure. Under this classification, all structures under test belong to Type I, whereas MC2H and the proposed network structure belong to both types.

To maintain the fairness of the comparison, different simulation settings will be used for network structures of different types. For Type I structures, to eliminate the effect of the base station location to the ψ , the distance between cluster head and the base station is not included in the calculation of ψ . For Type II structures, since the base station is one of the data aggregation points, its location cannot be ignored but assumed to be at the center of the sensing field (i.e., x = 25 m, y = 25 m).

For the proposed network structure to work as a Type I structure, either the topdown or the bottom-up approach can be applied provided that sufficient dummy nodes are added. To work as a Type II structure, the proposed network structure can only be constructed by the bottom-up approach without adding any dummy node.

The cluster number of the MC2H network is fixed to 1 when it works as a Type I structure. ψ of the MC2H network is minimized by selecting the node with minimum separations from its fellow nodes as the cluster head. To work as a Type II structure,



Figure 4.7: Averaged data collecting time of different multiple-cluster structures.

cluster heads in the MC2H networks are selected in a random manner as given in [28], while the optimum number of cluster heads is selected according to equation (4.14). In both configurations, cluster members are connected to their nearest cluster head. The SC network can only work as a Type I structure, and the chain is formed by using a greedy algorithm as given in [29]. To minimize DCT, the node closest to the middle of the chain (in terms of hops) will be selected as the cluster head.

Similar to the SC network, the MST network can only work as a Type I structure. Networks will be formed by using the Prim's algorithm as given in [30]. To minimize DCT, the node with the smallest separation (in terms of hops) to all leaf nodes will be selected as the cluster head. In networks using CTP, the node closest to the center of the sensing terrain is regarded as the root of the collection tree. The ETX of a path is



Figure 4.8: Averaged ψ of different multiple-cluster structures.

expressed as the squared value of the path length [74]. The root of the tree will have an ETX of 0. The ETX of an arbitrary node is the cumulated ETX from it, through its parent nodes, to the root [73]. Each node will choose its best route by selecting the path with the minimum cumulated ETX. Simulation results are shown in Figs. 4.5–4.8.

4.7 Analysis

As mentioned in Section 4.5, the DCT of networks with the proposed network structure is the lowest among Type I structures. In simulations among the six Type I structures, DCT of networks with the proposed network structure is the lowest, followed by networks with CTP. Since the aim of the MST is to minimize the total weight of edges, it does not perform well in reducing bottleneck and therefore it ranks fourth. In a SC network, it takes a very long time for data to propagate from both ends of the chain to the cluster head at the middle. This explains why SC networks have much higher DCT than networks with the proposed network structure. With the single cluster 2-hop structure, the networks with MC2H structure do not have any advantage in reducing DCT. The MC2H network is the one with the highest DCT among Type I structures.

In terms of minimizing ψ , the MST network no doubt ranks first. The ETX used in network with CTP can greatly reduce the communication distances among sensor nodes and make it ranks second. Nodes in networks with SC structure try to reduce the total communication distances by connecting to their nearest neighbors only. The strategy is effective for network with small number of nodes. However, to maintain a single-chain structure, it is unavoidable for the SC network to increase its ψ as the number of nodes increases. The SC network therefore ranks third. Using the optimization techniques employed in Section 4.4, the ψ of networks using the proposed top-down or bottom-up approach does not increase drastically as N increases and they ranks fourth and fifth. When $N \leq 56$, the performance of the top-down approach in terms of ψ is better than that of the bottom-up approach. However, as N further increases, the top-down approach is outperformed by the bottom-up approach. This is because all the optimization techniques employed in the top-down approach are carried out individually. Although all the optimization techniques will provide optimum solutions, there is lack of a global optimization method. This makes the top-down approach more effective for small-scale networks, but at the same time, it is more prone to be trapped in local optimum points as the network density increases. Therefore, the top-down and the bottom-up approaches are recommended for low density and high density networks, respectively. With all sen-
sor nodes connected to a single cluster head, the MC2H network is the structure with the highest ψ without question.

In simulations of the two Type II structures, networks formed by the proposed algorithm are shown to have lower DCT. Although the networks in MC2H structure have been tuned to give the optimum number of clusters, there is no control on the distribution of sensor nodes in each cluster. The DCT of networks in MC2H structure therefore greatly increases as N increases. In terms of ψ , both the proposed and the MC2H network structures give similar results when $N \leq 12$. For N > 12, networks formed by the proposed algorithm have lower ψ than those constructed in MC2H structure. The gap increases further as N increases. According to (4.14), a MC2H network is most efficient if there are g_{opt} clusters, where g_{opt} is proportional to N. As N increases, g_{opt} increases and thus more nodes are involved in long distance transmissions. The same thing happens to networks constructed by the proposed algorithm. Nevertheless, due to the special topology of the proposed network structure, the number of clusters increases at a more gentle rate. This explains why ψ of networks formed by the proposed algorithm is lower than those constructed in MC2H structure.

4.8 Chapter Summary

In this chapter, a delay-aware data collecting network structure and its formation algorithms are proposed. To cater for different applications, network formation can be implemented in either centralized or decentralized manner. Two network formation approaches are derived to provide optimized results for networks with different sizes. Comparing with the multiple-cluster 2-hop network, the single-chain network, the minimum spanning tree network, and the collection tree network, the proposed network structure is shown to be the most efficient in terms of data collecting time. The proposed network structure can greatly reduce the data collecting time without extending the total communication distance significantly.

Chapter 5

Conclusions

Advanced technologies have facilitated the manufacturing of inexpensive and compact wireless sensor nodes. The sensing capabilities of wireless sensor nodes have never been compromised due to their low cost and tiny size. The progresses in sensors and integrated circuit fabrication technologies have even provided wireless sensor nodes with higher sensing and processing power. In the foreseeable future, more and more applications will be developed based on wireless sensor networks.

At the same time, the rapid developments in wireless sensor networks have brought lots of challenges to system engineers. Basically, most of the challenges are associated with the massive numbers of wireless sensor nodes in networks. The interaction and coordination of these large numbers of wireless communication devices, which have never been tackled in other communication systems, make the networks and their associated problems highly complex.

5.1 **Bio-inspired Solutions**

Three problems in wireless sensor networks, namely, the clustering problem, the surplus sensing power, and the problem of delays in data collecting process, have been investigated in this thesis. Each problem has been analyzed in depth individually. In addition to the comprehensive analysis of the problems, solutions are provided. Through observing the social insect colonies, it can be shown that social insect colonies are actually sharing a number of similarities with wireless sensor networks. For example, both systems are crowded with simple individuals, each individual will make decisions mainly based on some local information, and the global information is usually propagated in a multi-hop manner. With these similarities, it can be predicted that both systems are dealing with similar kinds of fundamental problems.

In nature, only systems with high adaptability and fitness can survive. To increase their chances of survival, biological systems, including social insect colonies, will optimize themselves by means of evolutions. Living organisms have become the most optimized or nearly optimized systems by evolution and natural selection. In constructing practical engineering systems, biological phenomena, therefore, represent good sources of inspiration for achieving high efficiency and performance. A classic example is the design of an airplane, which is actually inspired by the streamlined bodies of flying creatures. By drawing up analogies between social insect colonies and wireless sensor networks, effective control mechanisms used in social insect colonies can be applied to wireless sensor networks in order to overcome similar challenges.

In this thesis, a novel clustering algorithm and an effective scheduling scheme are designed according to the rationales mentioned above. The proposed clustering algorithm



Figure 5.1: Wireless sensor networks with the proposed scheduling scheme and the proposed clustering algorithm. Filled circle represents the cluster head.

has been inspired by the bee colonies, while the proposed scheduling scheme has been inspired by the ant colonies. In computer simulations, the proposed clustering algorithm and the proposed scheduling scheme have been shown to be effective in prolonging network lifetimes and maximizing target detection capabilities, respectively. Nevertheless, performances of both solutions are highly robust to changes in the sensing terrain.

The two bio-inspired solutions mentioned above can be integrated together without any compatibility concern. The scheduling scheme is used to govern the number of active wireless sensor nodes in the network, as shown in Figure 5.1(a). The clustering algorithm operates on top of the scheduling scheme and arranges those active wireless sensor nodes into clusters, as shown in Figure 5.1(b).

In this thesis, two social insect colonies phenomena have been studied and adopted in wireless sensing systems. It is believed that more and more phenomena observed in social insect colonies and other biological systems can be converted into computational algorithms. Apart from tackling problems in wireless sensor networks, these bio-inspired computational algorithms can also be applied to solve other kinds of computational problems.

5.2 Non Bio-inspired Solutions

Generally speaking, reducing energy consumption, prolonging network lifetime and enhancing sensing coverage are the common objectives in the design of wireless sensor networks algorithms. However, in some time-critical applications, minimizing the delays in data collecting processes can be very important. Ordinary energy-aware clustering algorithms will usually form networks with heavily loaded nodes. Note that wireless sensor nodes are simple communication devices. A typical wireless sensor node can only handle one communication at a time. A node with high connection degree will have to communicate with its neighboring nodes one by one. Such high degree nodes can get congested easily and become the bottlenecks in data collecting processes.

To resolve the bottleneck problem, a delay-aware data collecting network structure for wireless sensor networks has been proposed and elaborated in Chapter 4. The proposed network structure distributes communication time slots among nodes in such a way that a child node can always transmit its data to its parent node with minimum delay. While the delays in data collecting processes are being minimized, it is unavoidable that the communication distances among wireless sensor nodes increase. Since the energy consumption of a sensor node is proportional to its communication distance, it is always desirable to minimize the communication distances among wireless sensor nodes. Two network constructing algorithms have been proposed. The two network constructing algorithms will organize wireless sensor nodes to form the proposed network structure. At the same time, the proposed algorithms will try to minimize the communication distances and thus save energy. In the computer simulations, when comparing with other network structures, the networks formed by the two proposed network constructing algorithms have been shown to be effective in shortening delays in data collecting processes. In addition, communication distances among wireless sensor nodes can be kept at low values.

Similar to the proposed clustering algorithm, the proposed network structure and its constructing algorithms can also be integrated with the proposed scheduling scheme. The proposed scheduling scheme will control the number of active nodes in the network. One of the two proposed network constructing algorithms will then be used to form the proposed delay-aware network structure using the active nodes.

The proposed delay-aware network structure has demonstrated that bio-inspired solutions are not the only way to solve wireless sensor networks' problems. More importantly, the proposed network structure has shown that energy efficiency should not be the only consideration in wireless sensor networks system design. Constraints and limitations of wireless sensor networks may vary according to different applications. Algorithms or schemes applied in wireless sensor networks should be application specific and always leave rooms for users to define their optimization priorities.

5.3 Future Work

Currently, we have only selected three problems in wireless sensor networks for investigation. Also, the problems are only analyzed theoretically and verified using computer simulations. An in-depth and comprehensive study is therefore needed for verification of the theoretical and simulation results.

5.3.1 Short Term Goals

5.3.1.1 Mobile Wireless Sensor Networks

In some applications, wireless sensor nodes are mounted on autonomous vehicles such as unmanned aerial vehicles (UAVs) and autonomous underwater vehicles (AUVs). These vehicles give mobility to wireless sensor nodes and greatly enhance the sensing coverage and the connectivity of a network. In this thesis, only stationary wireless sensor networks are considered. Therefore, one possible future work is to analyze the adaptability of the proposed algorithms in mobile wireless sensor networks.

5.3.1.2 Heterogeneous Wireless Sensor Networks

In most wireless sensor networks, wireless sensor nodes deployed are assumed to have the same physical properties and capabilities. The main reason for using homogeneous wireless sensor nodes is to reduce the design cost and the production cost of the nodes, thus minimizing the cost of the whole system. Sometimes, however, using heterogeneous wireless sensor nodes can offer extra performance to the system. For example, network lifetime can be prolonged by installing high capacity batteries to nodes far away from the base station. In the algorithms proposed in the thesis, only homogeneous wireless sensor nodes have been considered. In future, all proposed algorithms will be analyzed in networks with heterogeneous nodes.

5.3.1.3 Localization in Wireless Sensor Networks

Localization is often required for wireless sensor networks to identify the location of a target. Location information can now be easily obtained by installing global positioning systems (GPS) modules onto wireless sensor nodes. However, GPS may not be applicable in urban areas where high rise buildings are commonly found. Also, it is not economical to install a GPS module in every wireless sensor node.

Note that wireless sensor nodes are able to estimate their distances to their neighboring nodes using the received signal strength and the round trip time. If the neighboring nodes are having the location information, it is possible for a wireless sensor node to calculate its location without the use of a GPS module. Also, in urban areas, wireless sensor nodes with good GPS signal reception can help other wireless sensor nodes localize themselves.

From the example shown in Figure 5.2, it can be seen that the positions of the nodes with GPS modules are crucial in the localization processes. If the nodes with GPS modules are not evenly distributed, some nodes may not be able to localize themselves. Some further questions are:

- If the nodes with GPS modules are evenly distributed, how many nodes with GPS modules will be enough for all nodes in the network to get localized?
- 2. If the nodes with GPS modules are not evenly distributed, how many nodes with GPS modules will be enough for a certain percentage of nodes in the network to get localized?



Figure 5.2: Localization processes in wireless sensor networks with different distribution of wireless sensor nodes with GPS modules. Empty circles represent normal wireless sensor nodes. Circles with crosses represent wireless sensor nodes with GPS modules. Dashed lines indicate communications among wireless sensor nodes.

5.3.1.4 System Integration

In this thesis, three problems have been selected for in-depth study. At the same time, three solutions have been proposed to solve these problems. The future work is to integrate all three algorithms. The main objective will be to address the optimization problem involving the three solutions which are implemented in the same system at the same time. Furthermore, the proposed algorithms will be integrated with other existing algorithms to perform further investigation on the compatibility issues among algorithms.

5.3.2 Long Term Goals

As mentioned in the previous chapters, studies on the wireless sensor networks have been conducted by a number of research groups over the world. However, existing studies are mainly focused on deploying wireless sensor networks in rural areas which have very different characteristics from highly urbanized areas like Hong Kong. As part of Hong Kong's infrastructural development, a versatile wireless sensor network platform is needed. The future work of this project will be to study all the necessary requirements for wireless sensor networks to be deployed in densely populated urban area. Based on the information collected, a multipurpose wireless sensor network platform for Hong Kong will be developed. The aim of this platform is to support various monitoring activities in Hong Kong, such as disease spreading, flooding, landslips, etc.

5.3.2.1 Monitoring Clinical Signs of Avian Influenza in Poultry

In 1997, an outbreak of avian influenza H5N1 caused a depopulation of all poultry in farms and wet markets in Hong Kong. During this outbreak, 8 people in Hong Kong died from avian influenza. After the disease outbreak, the HKSAR government joined hands with the poultry industry to execute different kinds of preventive measures which included cleaning wet markets monthly, isolating poultry farms from wild birds and vaccinating poultry. All these measures have successfully made avian influenza disappear from Hong Kong for the following two years. Unfortunately, new waves of human infection cases were recorded in 1999 and 2003.

Vaccinating our poultry can be the most effective way in stopping the disease from spreading. However, study in [84] has confirmed that the H5N1 can be mutated, making the existing vaccine useless. It is always possible to develop new vaccines to tackle the mutated disease. The problem is that developing an effective vaccine can be time consuming. It may take even more time in clinical tests to prove that vaccinated poultry is safe for human consumption.



Figure 5.3: Using wireless sensor networks to monitor clinical signs of avian influenza in poultry.

According to [85], infected poultry will show various clinical signs. Examples are lack of coordination, incapability of standing up by themselves, decrease in water or food consumption, and showing abnormal locomotion. Observing abnormal behaviors in poultry can serve as a frontline detection of an outbreak of avian influenza and this can be done by using wireless sensor networks. By attaching wireless sensor nodes to poultry as shown in Figure 5.3, clinical signs of avian influenza in poultry can be monitored continuously. Any abnormal behavior can be detected at the early stage of an outbreak. Inter-relation among poultry can also be investigated using the data obtained. Diagnosis and treatment can be carried out strategically. Mortality can therefore be greatly reduced.

5.3.2.2 A Flooding Monitoring System

Hong Kong's rainy season always comes with serious flooding in low-lying area. In June 2005, 223mm of rainfall was recorded by the Hong Kong Observatory. In that month, 155 flooding reports were received by the Hong Kong government. In July 2008, heavy rain unleashed serious flooding in the New Territories. With a highway to the airport being covered by muddy water, more than 150 flights were delayed, and one inbound flight was canceled. Since 1989, the Drainage Service Department has started a series of river training projects that provide the primary drainage network which have shown significant improvement in resolving the flooding problem. However, in some densely populated and old districts, such kind of massive constructions may not be applicable and may take years to complete. The loss of money and life can be greatly reduced if the public can be informed with some real-time sewage information. What seems to be impossible in the past can now be achieved by using wireless sensor networks. Waterproof wireless sensor nodes can be installed in rain drainage underground pipes to record water level and flow rate. The wireless sensor nodes will send out warning when some threshold has been reached. A wireless sensor node can be equipped with a small scale generator to be powered from heavy rainfalls or other renewable sources.

5.3.2.3 A Man-Made Slope Monitoring System

The rainy season brings frequent landslides to Hong Kong every year, resulting in a total of more than 470 people killed in the past 50 years. Landslides are mainly triggered by water ingress into slopes and by soil erosion during heavy rain. To prevent the water ingress and soil erosion, many man-made slopes and retaining walls are built with



Figure 5.4: Using wireless sensor networks to monitor man-made slopes.

drainage provisions and protective surfacing. Although proper routine maintenance of a slope or retaining wall can greatly reduce the probability of a landslide, the risk of slope failure at times of heavy rainfall can never be entirely eliminated. This problem can be relieved by the use of wireless sensor networks. Hundreds of wireless sensor nodes can be installed in man-made slopes or planted under the protective concrete surface. The sensors equipped in each wireless sensor node can monitor the conditions of the slope such as pressure and soil humidity. Sudden or abnormal changes of conditions will trigger a sensor to release a warning to the base station. The base station can be connected to a centralized mainframe using the GSM system as shown in Figure 5.4. Emergency maintenance and civilian evacuation plans can be carried out before a landslide happens.

Bibliography

- Y. Mukaiyama, K. Mizuno, A. Horide, T. Sakakibara, S. Maruyama, I. Kamata, and A. Shinoda, "Application of fault location system in power substations," *IEEE Trans. Power Delivery*, vol. 8, no. 1, pp. 48–55, January 1993.
- [2] C.-Y. Chong and S. P. Kumar, "Sensor networks: evolution, opportunities, and challenges," *Proceedings of the IEEE*, vol. 91, no. 8, pp. 1247–1256, August 2003.
- [3] E. C. Whitman, "SOSUS: The secret weapon of undersea surveillance," Undersea Warfare, vol. 7, no. 2, 2005. [Online]. Available: http://www.navy.mil/navydata/cno/n87/usw/issue_25/sosus2.htm
- [4] I. Khemapech, I. Duncan, and A. Miller, "A survey of wireless sensor networks technology," in *Proc. 6th Annual PostGraduate Symp. Convergence of Telecommunications,Networking and Broadcasting, (PGNET 2005)*, Liverpool, UK, June 2005, pp. 26–31.
- [5] (2005, January) Semi-automatic ground environment (SAGE). The MITRE Corporation. [Online]. Available: http://www.mitre.org/about/sage.html

- [6] R. T. Lacoss, "Distributed mixed sensor aircraft tracking," in *Proc. 6th American Control Conf.*, Minneapolis, Minnesota, USA, June 1987, pp. 1827–1830.
- [7] W. D. Bryan, H. G. Nguyen, and D. W. Gage, "Man-portable networked sensor system," in SPIE Proc. 3394: Sensor Technology for Soldier Systems, 1998, pp. 15–17.
- [8] MICA2DOT wireless microsensor mote (document part number: 6020-0043-05 Rev A), Crossbow Technology Inc. [Online]. Available: http://www.xbow.com/Products/productdetails.aspx?sid=73
- [9] C.-T. Cheng, C. K. Tse, and F. C. M. Lau, "Cooperative path planner for UAVs using ACO algorithm with Gaussian distribution functions," in *Proc. IEEE Int. Symp. Circuits and Systems, (ISCAS 2009)*, Taipei, Taiwan, May 2009, pp. 173– 176.
- [10] K. Akkaya and M. Younis, "A survey on routing protocols for wireless sensor networks," *Ad Hoc Networks*, vol. 3, pp. 325–349, May 2005.
- [11] G. J. Pottie and W. J. Kaiser, "Wireless integrated network sensors," *Communications of the ACM*, vol. 43, pp. 51–58, May 2000.
- [12] M. A. M. Vieira, A. B. da Cunha, and D. C. da Silva Jr., "Designing wireless sensor nodes," in *Proc. 6th Int. Workshop Embedded Computer Systems: Architectures, Modeling, and Simulation, (SAMOS 2006)*, Samos, Greece, July 2006, pp. 99–108.
- [13] A. Lachenmann, P. J. Marrón, M. Gauger, D. Minder, O. Saukh, and K. Rothermel,"Removing the memory limitations of sensor networks with flash-based virtual

memory," in *Proc. 2nd ACM SIGOPS/EuroSys European Conf. Computer Systems,* (*EuroSys '07*), Lisbon, Portugal, March 2007, pp. 131–144.

- [14] D. Culler, D. Estrin, and M. Srivastava, "Overview of sensor networks," *Computer*, vol. 37, no. 8, pp. 41–49, August 2004.
- [15] L. Cai, W. Zhang, E. Li, Z. Liang, Z.-G. Hou, and M. Tan, "Design and implementation of a cdma-based remote monitoring and controlling system," in *Proc. SICE Annual Conference 2007*, Kagawa, Japan, September 2007, pp. 2445–2450.
- [16] I. Vasilescu, C. Detweiler, and D. Rus, "AquaNodes: an underwater sensor network," in *Proc. 2nd Workshop Underwater Networks*, (WuWNet '07), Montreal, Quebec, Canada, September 2007, pp. 85–88.
- [17] A. Kansal, J. Hsu, S. Zahedi, and M. B. Srivastava, "Power management in energy harvesting sensor networks," ACM Trans. Embedded Computing Systems, vol. 6, no. 4, pp. 32–38, September 2007.
- [18] K. Roth and J. Brodrick, "Energy harvesting for wireless sensors," *ASHRAE Jour.*, pp. 84–89, May 2008.
- [19] M. Dalbro, E. Eikeland, A. in't Veld, S. Gjessing, T. Lande, H. Riis, and O. Sorasen, "Wireless sensor networks for off-shore oil and gas installations," in *Proc.* 2nd Int. Conf. Sensor Technologies and Applications, (SENSORCOMM '08), Cap Esterel, France, August 2008, pp. 258–263.
- [20] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Anderson, "Wireless sensor networks for habitat monitoring," in *Proc. 1st ACM Int. Workshop Wireless*

sensor networks and Applications, (WSNA '02), Atlanta, Georgia, USA, September 2002, pp. 88–97.

- [21] L. Schwiebert, S. K. S. Gupta, and J. Weinmann, "Research challenges in wireless networks of biomedical sensors," in *Proc. 7th Annual Int. Conf. Mobile Computing and Networking*, (*MobiCom '01*), Rome, Italy, July 2001, pp. 151–165.
- [22] C. R. Baker, K. Armijo, S. Belka, M. Benhabib, V. Bhargava, N. Burkhart, A. D. Minassians, G. Dervisoglu, L. Gutnik, M. B. Haick, C. Ho, M. Koplow, J. Mangold, S. Robinson, M. Rosa, M. Schwartz, C. Sims, H. Stoffregen, A. Waterbury, E. S. Leland, T. Pering, and P. K. Wright, "Wireless sensor networks for home health care," in *Proc. 21st Int. Conf. Advanced Information Networking and Applications Workshops, (AINAW '07)*, vol. 2, Niagara Falls, Ontario, Canada, May 2007, pp. 832–837.
- [23] G. Amato, S. Chessa, F. Conforti, A. Macerata, and C. Marchesi, "Health care monitoring of mobile patients," *ERCIM News*, no. 60, pp. 69–70, January 2005.
- [24] D. Husemann and M. Nidd, "Pervasive patient monitoring take two at bedtime," *ERCIM News*, no. 60, pp. 70–71, January 2005.
- [25] J. N. Al-karaki and A. E. Kamal, "Routing techniques in wireless sensor networks: a survey," *IEEE Wireless Communications Mag.*, vol. 11, no. 6, pp. 6–28, December 2004.
- [26] D. Estrin, L. Girod, G. Pottie, and M. Srivastava, "Instrumenting the world with wireless sensor networks," in *Proc. Int. Conf. Acoustics, Speech and Signal Pro-*

cessing, (*ICASSP '01*), vol. 4, Salt Lake City, Utah, USA, May 2001, pp. 2033–2036.

- [27] J. Ai, D. Turgut, and L. Bölöni, "A cluster-based energy balancing scheme in heterogeneous wireless sensor networks," in *Proc. 4th Int. Conf. Networking, (ICN 2005)*, vol. 1, Reunion Is., France, April 2005, pp. 467–474.
- [28] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An applicationspecific protocol architecture for wireless microsensor networks," *IEEE Trans. Wireless Communications*, vol. 1, no. 4, pp. 660–670, October 2002.
- [29] S. Lindsey and C. S. Raghavendra, "PEGASIS: Power-efficient gathering in sensor information systems," in *Proc. IEEE Conf. Aerospace*, vol. 3, Big Sky, Montana, USA, March 2002, pp. 1125–1130.
- [30] H. Ö. Tan and Í. Körpeoĝlu, "Power efficient data gathering and aggregation in wireless sensor networks," ACM SIGMOD Record, vol. 32, no. 4, pp. 66–71, December 2003.
- [31] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: a review," ACM Computing Surveys (CSUR), vol. 31, no. 3, pp. 264–323, September 1999.
- [32] M. N. Halgamuge, S. M. Guru, and A. Jennings, "Centralised strategies for cluster formation in sensor networks," *Classification and Clustering for Knowledge Discovery*, pp. 315–334, 2005.
- [33] A. Manjeshwar and D. P. Agrawal, "TEEN: a routing protocol for enhanced efficiency in wireless sensor networks," in *Proc. 15th Int. Symp. Parallel and Dis*-

tributed Processing, (IPDPS 2001), San Francisco, California, USA, April 2001, pp. 2009–2015.

- [34] A. Manjeshwar and D. P. Agrawal, "APTEEN: a hybird protocol for efficient routing and comprehensive information retrieval in wireless sensor networks," in *Proc. 16th Int. Symp. Parallel and Distributed Processing, (IPDPS 2002)*, Fort Lauderdale, Florida, USA, April 2002, pp. 195–202.
- [35] E.-H. Jung, S.-H. Lee, J.-W. Choi, and Y.-J. Park, "A cluster head selection algorithm adopting sensing-awareness and sensor density for wireless sensor networks," *IEICE Trans. Communications*, vol. E90-B, no. 9, pp. 2472–2480, September 2007.
- [36] J. B. Free, *The Social Organization of Honeybees*. London: Edward Arnold (Publishers) Limited, 1977.
- [37] J. Huxley, Ants. London: Arrow Books Ltd, 1962.
- [38] T. Pankiw, "Cued in: honey bee pheromones as information flow and collective decision-making," *Apidologie*, vol. 35, no. 2, pp. 217–226, March 2004.
- [39] D. M. Gordon, Ants at Work: How an Insect Society is Organized. New York: The Free Press, 1999.
- [40] K. Naumann, "Grooming behaviors and the translocation of queen mandilbular gland pheromone on worker honey bees (Apis Mellifera L)," *Apidologie*, vol. 22, no. 5, pp. 523–531, 1991.

- [41] K. Naumann, M. L. Winston, K. N. Slessor, G. D. Prestwich, and B. Latli, "Intranest transmission of aromatic honey bee queen mandibular gland pheromone components: movement as a unit," *Canadian Entomologist*, vol. 124, no. 5, pp. 917– 934, 1992.
- [42] S. Wongsiri, B. P. Oldroyd, and P. Nanork, "The reproductive dilemmas of queenless red dwarf honeybee (Apis Florea) workers," *Behavioral Ecology and Sociobiology*, vol. 61, no. 1, pp. 91–97, November 2006.
- [43] O. Malka, S. Shnieor, A. Hefetz, and T. Katzav-Gozansky, "Reversible royalty in worker honeybees (Apis Mellifera) under the queen influence," *Behavioral Ecology* and Sociobiology, vol. 61, no. 3, pp. 465–473, January 2007.
- [44] M. Dorigo, V. Maniezzo, and A. Colorni, "The ant system: optimization by a colony of cooperating agents," *IEEE Trans. Systems, Man, and Cybernetics-Part B*, vol. 26, pp. 29–41, February 1996.
- [45] S. Bandyopadhyay and E. J. Coyle, "An energy efficient hierarchical clustering algorithm for wireless sensor networks," in *Proc. 22nd Annual Joint Conf. IEEE Computer and Communications Societies, (INFOCOM 2003)*, vol. 3, San Francisco, California, USA, April 2003, pp. 1713–1723.
- [46] M. Younis, M. Youssef, and K. Arisha, "Energy-aware routing in cluster-based sensor networks," in Proc. 10th IEEE Int. Symp. Modeling, Analysis and Simulation of Computer and Telecommunication Systems, (MASCOTS 02), Fort Worth, Texas, USA, October 2002, pp. 129–136.

- [47] J. Hill, R. Szewczyk, A. Woo, S. Hollar, D. E. Culler, and K. S. J. Pister, "System architecture directions for networked sensors," ACM SIGPLAN Notices, vol. 35, no. 11, pp. 93–104, November 2000.
- [48] R. Kershner, "The number of circles covering a set," *American Jour. Mathematics*, vol. 61, no. 3, pp. 665–671, July 1939.
- [49] S. Pattem, S. Poduri, and B. Krishnamachari, "Energy-quality tradeoffs for target tracking in wireless sensor networks," in *Proc. 2nd Int. Workshop Information Processing in Sensor Networks, (IPSN 03)*, Palo Alto, California, USA, April 2003, pp. 32–46.
- [50] J. Elson, L. Girod, and D. Estrin, "Fine-grained network time synchronization using reference broadcasts," in *Proc. Symp. Operating System Design and Implementation*, (OSDI '02), Boston, Massachusetts, USA, December 2002, pp. 147–163.
- [51] L. Qun and D. Rus, "Global clock synchronization in sensor networks," *IEEE Trans. Computer*, vol. 55, no. 2, pp. 214–226, February 2006.
- [52] V. Turau and C. Weyer, "Scheduling transmission of bulk data in sensor networks using a dynamic tdma protocol," in *Proc. Int. Workshop Data Intensive Sensor Networks*, (*DISN 07*), Mannheim, Germany, May 2007, pp. 321–325.
- [53] B. Hohlt, L. Doherty, and E. Brewer, "Flexible power scheduling for sensor networks," in *Proc. 3rd Int. Symp. Information Processing in Sensor Networks, (IPSN 04)*, Berkeley, California, USA, April 2004, pp. 205–214.

- [54] D. Schrage and P. G. Gonsalves, "Sensor scheduling using ant colony optimization," in *Proc. 6th Int. Conf. Information Fusion*, (*Fusion 04*), vol. 1, Cairns, Queensland, Australia, July 2003, pp. 379–385.
- [55] C. Decker, T. Riedel, E. Peev, and M. Beigl, "Adaptation of on-line scheduling strategies for sensor network platforms," in *Proc. IEEE Int. Conf. Mobile Adhoc* and Sensor Systems, (MASS 06), Vancouver, British Columbia, Canada, October 2006, pp. 534–537.
- [56] J.-F. Chamberland and V. V. Veeravalli, "The art of sleeping in wireless sensing systems," in *Proc. IEEE Workshop Statistical Signal Processing*, (SSP 03), Saint-Louis, Missouri, USA, September 2003, pp. 17–20.
- [57] J. Jang, "A study on a sequenced directed diffusion algorithm for sensor networks," in *Proc. 9th Int. Conf. Advanced Communication Technology, (ICACT 07)*, Gangwon-Do, Korea, February 2007, pp. 679–683.
- [58] S. Hao and T. Wang, "Sensor networks routing via bayesian exploration," in *Proc.* 31st IEEE Conf. Local Computer Networks, (LCN 06), Tampa, Florida, USA, November 2006, pp. 954–955.
- [59] H. Sun, J. Jiang, M. Lin, and X. Tan, "Queen-ant-aware-based algorithm for wireless sensor networks routing," in *Proc. IEEE Int. Conf. Information Acquisition*, (*ICIA 06*), Shandong, China, August 2006, pp. 622–626.
- [60] W. Dali and H. A. Chan, "Clustering ad hoc networks: schemes and classifications," in *Proc. 3rd Annual IEEE Communications Society Conf. Sensor, Mesh and*

Ad Hoc Communications and Networks, (SECON 06), vol. 3, Reston, Virginia, USA, September 2006, pp. 920–926.

- [61] M. P. Singh and M. M. Gore, "A new energy-efficient clustering protocol for wireless sensor networks," in *Proc. 2nd Int. Conf. Intelligent Sensors, Sensor Networks and Information Processing, (ISSNIP 05)*, Melbourne, Australia, December 2005, pp. 25–30.
- [62] S. M. Guru, A. Hsu, S. Halgamuge, and S. Fernando, "Clustering sensor networks using growing self-organising map," in *Proc. 1st Int. Conf. Intelligent Sensors, Sensor Networks and Information Processing, (ISSNIP 04)*, Melbourne, Australia, December 2004, pp. 91–96.
- [63] M. Lee and S. Lee, "Data dissemination for wireless sensor networks," in Proc. 10th IEEE Int. Symp. Object and Component-Oriented Real-Time Distributed Computing, (ISORC 07), Santorini Island, Greek, May 2007, pp. 172–180.
- [64] O. Goussevskaia, M. D. V. Machado, R. A. F. Mini, A. A. F. Loureiro, G. R. Mateus, and J. M. Nogueira, "Data dissemination based on the energy map," *IEEE Communications Mag.*, vol. 43, no. 7, pp. 137–143, July 2005.
- [65] A. Visvanathan, J. Youn, and J. Deogun, "Hierarchical data dissemination scheme for large scale sensor networks," in *Proc. IEEE Int. Conf. Communications, (ICC* 05), Seoul, Korea, May 2005, pp. 3030–3036.
- [66] D. M. Gordon, "The organization of work in social insect colonies," *Nature*, vol. 380, pp. 121–124, April 1996.

- [67] D. M. Gordon and N. J. Mehdiabadi, "Encounter rate and task allocation in harvester ants," *Behavioral Ecology and Sociobiology*, vol. 445, no. 5, pp. 370–377, April 1999.
- [68] D. M. Gordon, "Behavioral flexibility and the foraging ecology of seed-eating ants," *The American Naturalist*, vol. 138, no. 2, pp. 379–411, August 1991.
- [69] S. W. Pacala, D. M. Gordon, and H. C. J. Godfray, "Effects of social group size on information transfer and task allocation," *Evolutionary Ecology*, vol. 10, no. 2, pp. 127–165, March 1996.
- [70] S. C. Nicolis, G. Theraulaz, and J. L. Deneubourg, "The effect of aggregates on interaction rate in ant colonies," *Animal Behaviour*, vol. 69, no. 3, pp. 535–540, March 2005.
- [71] E. L. Hinrichsen, J. Feder, and T. Jøssang, "Random packing of disks in two dimensions," *Physical Review A*, vol. 41, no. 8, pp. 4199–4209, April 1990.
- [72] A. M. Mathai, An Introduction to Geometrical Probability: Distributional Aspects with Applications. Amsterdam : Gordon and Breach, 1999.
- [73] R. Fonseca, O. Gnawali, K. Jamieson, S. Kim, P. Levis, and A. Woo, "The collection tree protocol," TinyOS Enhancement Proposals (TEP) 123, December 2007.
- [74] D. S. J. D. Couto, "High-throughput routing for multi-hop wireless networks," Ph.D. dissertation, Dept. Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Massachusetts, USA, June 2004.

- [75] O. Tekdas, J. H. Lim, A. Terzis, and V. Isler, "Using mobile robots to harvest data from sensor fields," *IEEE Wireless Communications Mag.*, vol. 16, no. 1, pp. 22–28, February 2009.
- [76] C. Florens, M. Franceschetti, and R. J. McEliece, "Lower bounds on data collection time in sensory networks," *IEEE Jour. Selected Areas in Communications*, vol. 22, no. 6, pp. 1110–1120, August 2004.
- [77] W. Wang, Y. Wang, X.-Y. Li, W.-Z. Song, and O. Frieder, "Efficient interferenceaware TDMA link scheduling for static wireless networks," in *Proc. 12th Annual Int. Conf. Mobile Computing and Networking, (MobiCom'06)*, Los Angeles, California, USA, September 2006, pp. 262–273.
- [78] I. Solis and K. Obraczka, "The impact of timing in data aggregation for sensor networks," in *Proc. IEEE Int. Conf. Communications*, vol. 6, Paris, France, June 2004, pp. 3640–3645.
- [79] A. Billionnet, "Different formulations for solving the heaviest k-subgraph problem," *Information Systems and Operational Research*, vol. 43, no. 3, pp. 171–186, August 2005.
- [80] S. M. Roberts and B. Flores, "Solution of a combinatorial problem by dynamic programming," *Operations Research*, vol. 13, no. 1, pp. 146–157, January 1965.
- [81] H. W. Kuhn, "The Hungarian method for the assignment problem," *Naval Research Logistics*, vol. 52, no. 1, pp. 7–21, February 2005.
- [82] C. H. Papadimitriou and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity*. New York: Dover Publications INC., 1998.

- [83] J. Munkres, "Algorithms for assignment and transportation problems," *Jour. Society for Industrial and Applied Mathematics*, vol. 5, no. 1, pp. 32–38, March 1957.
- [84] (2006, March) Minor mutations in avian flu virus increase chances of human infection. ScienceDaily. Scripps Research Institute. [Online]. Available: http://www.sciencedaily.com/releases/2006/03/060318144425.htm
- [85] A. R. W. Elbers, G. Koch, and A. Bouma, "Performance of clinical signs in poultry for the detection of outbreaks during the avian influenza A (H7N7) epidemic in the Netherlands in 2003," *Avian Pathology*, vol. 34, no. 3, pp. 81–187, February 2005.