Energy-Efficient Data Gathering and Aggregation for Wireless Sensor Networks

by

Yuexian Wang

B.E. (Electrical & Electronic),
Northwestern Polytechnical University, China, 2006.
M.E. (Pattern Recognition & Intelligent Systems),
Northwestern Polytechnical University, China, 2009.

Thesis submitted for the degree of

Master of Engineering Science

in

School of Electrical and Electronic Engineering
The University of Adelaide, Australia

August 2011

THE UNIVERSITY OF ADELAIDE
AUSTRALIA

Typeset in $\LaTeX\,2_\varepsilon$
Yuexian Wang

To my parents

Contents

Abstract

Wireless sensor networks, when compared with other traditional wireless communication systems, possess two unique characteristics: (i) the limited battery power supply of sensor nodes, and (ii) the redundant data which are correlated among different nodes. These two are associated with energy consumption and data traffic control. The research in this thesis aims at designing an energy efficient routing scheme with data aggregation in wireless sensor networks.

In this thesis, we developed an energy-efficient routing scheme consisting of the setup phase, the routing tree optimisation phase and the data gathering phase. The setup phase is to build initial routing trees by the ant colony optimisation algorithm which is executed between the base station and all sensor nodes. A key to our routing scheme is the routing tree optimisation phase. The routing tree optimisation is performed by the base station using the particle swarm optimisation algorithm. We propose a modified particle swarm optimisation algorithm that is capable of jointly exploring the data traffic and communication structure to provide the optimal strategy for data gathering. Once the routing tree optimisation has been accomplished, it comes to the data gathering phase. Data flows to the aggregator node, the aggregator node then transmits the gathering data to the base station via multi-hop in this phase of operation.

The performance of our routing scheme is evaluated by comparing with three existing routing schemes using simulations. Our scheme performs as well as the shortest path tree algorithm and saves more than 45% energy over the other two algorithms in the non-aggregation scenario. If perfect aggregation occurs, our scheme obtains about 5% energy reduction at least. When varying from non to perfect aggregation, the simulation results show that our scheme can adapt to the change of data correlation condition and achieve at least 25% energy saving on average. Since our scheme can save energy and enhance transmission efficiency, it is well suited for applications where energy and data traffic are the primary considerations.

Statement of Originality

This work contains no material that has been accepted for the award of any other degree or diploma in any university or other tertiary institution and, to the best of my knowledge and belief, contains no material previously published or written by another person, except where due reference has been made in the text.

I give consent to this copy of the thesis, when deposited in the University Library, being available for loan, photocopying and dissemination through the library digital thesis collection.

The author of this thesis acknowledges that copyright of published work contained within this thesis (as listed in the publications page) resides with the copyright holder(s) of that work.

Signed

Date

Acknowledgments

First and foremost, I must recognise my principal supervisor Associate Prof. Cheng Chew Lim for his constant and patient guidance during the period of my master study. His meticulous approach to learning and tolerance towards others will influence me for the rest of my life. Talking with him not only in professional areas but also in culture has always been enjoyable and beneficial. In addition, many thanks to him for great help on my application for half fee remission scholarship.

I would also like to wholeheartedly thank my co-supervisors Dr John Asenstorfer for providing valuable suggestions to my research.

Sincere thanks to my colleagues in the School of Electrical and Electronic Engineering for their unselfish help in extending my knowledge. They are Mr Yinhao Ding, Dr Adriel Cheng, Mr Hyukjin Lee, Dr Xiaoxi Xu, Mr Kiet To, Mr Adam Burdeniuk and Mr Hong Gunn Chew.

I am indebted for the work done by the staff in the School of Electrical and Electronic Engineering, particularly, Associate Professor Michael Liebelt and Mr Stephen Guest who managed my scholarship and research funding issues. Of course thanks to the four kind ladies in the school office.

Many thanks to my friends, specifically Zhonghao Hu, Bin Guo, Matthew Trinkle, Lifang Lu, Ruiting Yang and Yi Li in Adelaide for their constant support and encouragement during my master studies.

I am grateful to my parents who dedicate their unconditional love and encouragement to me. Since I first entered primary school, all through my educational journey of twenty years, their love has always accompanied me. Without them, I could not imagine how I could have accomplished this work.

Yuexian Wang (August 2011)

Conventions

Typesetting

This thesis is typeset using the LaTeX2e software.

The fonts used in this thesis are Times New Roman and Sans Serif.

Referencing

Referencing and citation style in this thesis are based on the Institute of Electrical and Electronics Engineers (IEEE) Transaction style [1].

For electronic references, the last accessed date is shown at the end of a reference.

Units

The units used in this thesis are based on the International System of Units (SI units) [2].

Spelling

The Australian English spelling is adopted in this thesis.

Abbreviations

ACA Ant Clustering Algorithm

ACO Ant Colony Optimisation

BS Base Station

CDMA Code Division Multiple Access

CH Cluster Head

DD Directed Diffusion

GAF Geographic Adaptive Fidelity

GA Genetic Algorithms

GEAR Geographic and Energy Aware Routing

LEACH Low Energy Adaptive Clustering Hierarchy

MAC Media Access Control

MAST Minimum Aggregation Spanning Tree

MEMS Micro Electro Mechanical System

MST Minimum Spanning Tree

OSI Open System Interconnection

PEGASIS Power-Efficient Gathering in Sensor Information Systems

PSO Particle Swarm Optimisation

QoS Quality of Service

RF Radio Frequency

SI Swarm Intelligence

SLT Shallow Light Tree

SPIN Sensor Protocols for Information via Negotiation

SPT Shortest Path Tree

TDMA Time Division Multiple Access

TSP Traveling Salesman Problem

WSNs Wireless Sensor Networks

List of Figures

Chapter 1

Introduction and Motivation

THIS chapter gives a brief introduction to routing in wireless sensor networks. The research problems and the contributions of this thesis are presented. Finally, the thesis structure is discussed and the contents of each chapter are summarised.

1.1 Research Area

Wireless sensor networks are an integration of micro electro mechanical systems (MEMS), low-power electronics and low-power radio frequency (RF) design [3] [4] [5] [6]. A wireless sensor network generally consists of three main components: (i) Numerous sensor nodes. Sensor nodes are capable of object sensing, data processing, storing, and routing activities. (ii) A base station (BS). The base station may be a fixed or mobile node which can link the sensor network to an existing communications infrastructure or to the internet with the users to disseminate the data sensed for further processing. (iii) Wireless transmission media. In a sensor network, communicating nodes are linked by a wireless medium. For different application requirements like marine applications, we must choose the corresponding transmission media.

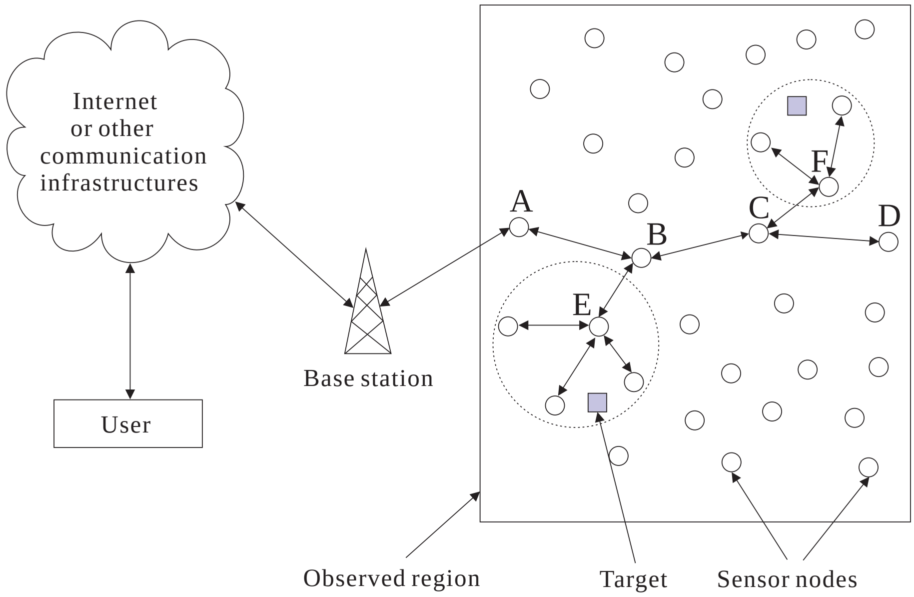


Figure 1.1. System architecture of wireless sensor networks.

Fig. 1.1 shows the system architecture of a typical WSN. First, sensor nodes which are usually scattered in an observed region sense and collect high-quality information about the environment. Data aggregation or combination of the individual nodes, which is generally defined as the use of techniques that gather several sources of raw data from multiple sources to combine new raw data, will keep surveillance more power-efficient and potentially more accurate. Then, in order to obtain and analyse the data extracted from remote objects, each sensor node must promptly collect and forward the event reports either to other sensors or back to an external BS.

The open system interconnection reference model (OSI/RM) [7] [8] established by the international standards organisation (ISO) specifies the relationship between messages transmitted in a communication network and applications programs run by the users. Hence, we can use a similar model to classify WSNs into different functions which imply communication layers. Fig. 1.2 shows the five layers of OSI/RM for WSNs.

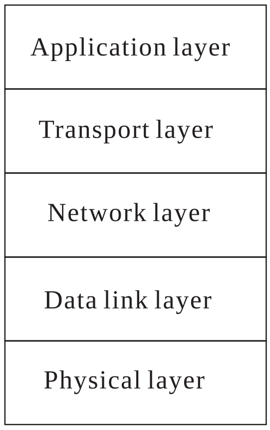


Figure 1.2. The model of WSNs layers.

OSI/RM consists of the application layer, transport layer, network layer, data link layer, and physical layer. The applications layer represents programs run by users. The transport layer maintains the flow of data and provides error detection and correction. Routing is performed in the network layer. The data link layer implements media access control (MAC) protocol to resist fading, filter noise and reduce message collision. The physical layer provides the actual hardware communication link interconnections.

The research work in this thesis focuses on routing schemes in WSNs. In network concepts, routing is the act of directing and delivering network traffic across a network from sources to a destination, whose kernel is path determination. A routing scheme specifies how routers communicate with each other and how transmission occurs across a network. The routing scheme is responsible for selecting an end-to-end routing path fulfilling the desired quality of service (QoS) characteristics.

Routing in sensor networks is very challenging due to several characteristics that distinguish them from contemporary communication and wireless ad-hoc networks. First, it is not possible to build a global addressing scheme because of large numbers of sensor nodes. Therefore, classical IP-based protocols cannot be applied to sensor networks. Second, in contrast to typical communication networks almost all applications of sensor networks require the flow of sensed data from multiple regions (sources) to a particular sink. Third, generated data traffic has significant redundancy since multiple sensors may generate the same data within the vicinity of a phenomenon. Such redundancy needs to be exploited by the routing protocols to improve energy and bandwidth utilisation. Fourth, sensor nodes are tightly constrained in terms of transmission power, on-board energy, processing capacity and storage and thus require careful resource management.

The introduction above demonstrates that routing scheme in WSNs is a complex system which involves various challenging research issues, such as network dynamics, node deployment, energy consideration and data aggregation. This thesis emphasises the data gathering and aggregation in WSNs.

More details about WSNs routing schemes will be introduced in Chapter 2.

1.2 Research Problems

The objective of this thesis is:

- To study energy-efficient routing schemes with data aggregation in a distributed wireless environment subject to limited transmission range, and periodic and aperiodic traffic patterns.

To better illustrate the research problems, we construct a routing tree depicted in Fig. 1.3. Data aggregation is the most important ingredient for a well-designed routing scheme. In sensor networks, however, data centric routing, in which routing is done based on the content of the data packets, is often adopted to promote data aggregation. It is assumed that the aggregation results of data from different nodes can be combined in a single packet. Although there are three routing options in Fig. 1.3, we cannot easily

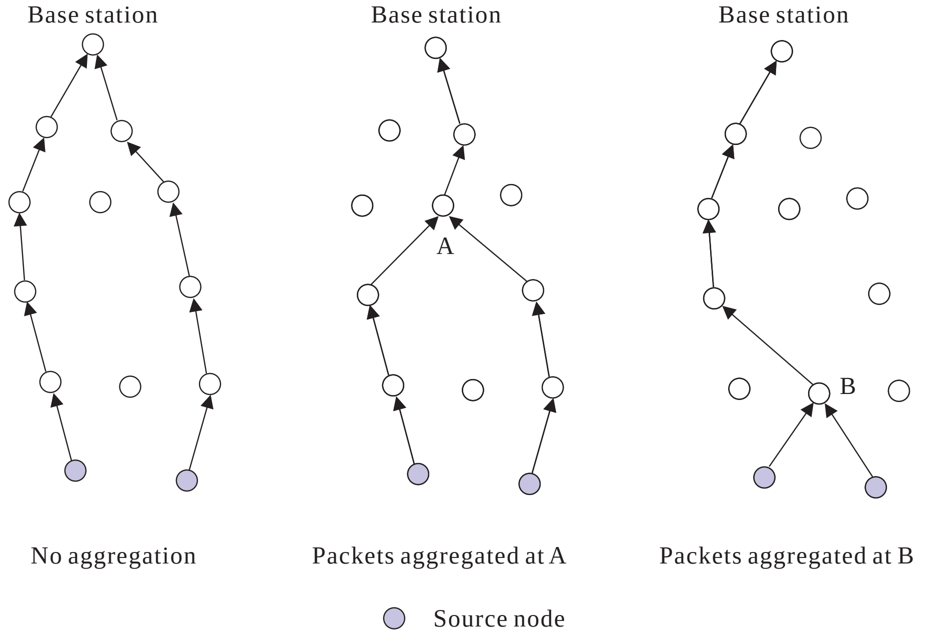


Figure 1.3. The impact of routing on data aggregation.

determine which routing scheme is optimal in the minimisation of energy dissipation. Hence, we address three research problems.

1. Designing an energy-efficient routing scheme to reduce energy consumption for data transmission.

Since the distances between nodes and the base station may be long and the transmission power of a wireless radio is proportional to distance squared or an even higher order in the presence of obstacles, traditional single-hop routing will drain energy of sensor nodes quickly.

2. Selecting a generic data aggregation model which accommodates different correlation conditions.

The data correlation coefficient is the key to data aggregation, but this coefficient is heavily dependent on the application scenarios. The existing aggregation models are not sufficient because they can only deal with some extreme conditions where data correlate each other completely.

3. Finding an optimum routing tree with data aggregation to minimise energy cost.

The existing algorithms for finding a routing tree with data aggregation generate suboptimal aggregation trees because their aggregation model is not generic to capture the various correlation conditions. By using the generic aggregation model, an optimal routing tree with data aggregation can adapt to the change of data correlation conditions and select reasonable paths to minimise total energy dissipation.

1.3 Original Contributions

Our research contributions in this thesis are summarised as follows:

- A new multi-hop routing algorithm that can reduce the energy consumption for the data transmission from the source nodes to the base station and prolong the network lifetime.
- A generic aggregation model which does not depend on any specific relationship among information supplied by sensor nodes nor on any particular model of data aggregation to adapt to a variety of applications.
- A particle swarm optimisation algorithm that finds a near optimal routing tree with data aggregation to minimise energy consumption in different data correlation conditions.

Although our routing scheme is specific for the applications of wireless sensor networks, some algorithms, such as the modified PSO algorithm, can be adopted to optimise other quality of service metrics for other wireless networks. In addition, the multi-hop routing algorithm in our scheme can be applied to other wireless network routing (like ad-hoc network routing).

The following paper presented as part of the work in this thesis:

- Y. Wang and C. C. Lim, "Gathering correlated data in wireless sensor networks using a heuristic algorithm," in *Proc. 2011 International Conference on Opto-Electronics Engineering and Information Science*, vol. 1, Dec. 2011, pp. 417-421.

1.4 Thesis Structure

The thesis is presented in five chapters:

Chapter 1 provides a brief introduction to wireless sensor networks. In addition, the research problems for doing the work, the contributions to knowledge provided by the thesis, and the structure of the thesis are also discussed.

Chapter 2 presents the background including the approach used for the routing scheme, aggregation models and data gathering algorithms which are utilised to connect sensor nodes via suboptimal aggregation trees under certain conditions.

Chapter 3 introduces one solution for constructing a routing tree with data aggregation in wireless sensor networks. The fundamental system models are discussed and established. A modification of the particle swarm optimisation algorithm is proposed. Compared with three other existing algorithms, the performance of an aggregation tree using the PSO algorithm is analysed as well.

Chapter 4 sets up network simulation and gives a set of simulation results to evaluate the performance of our routing scheme with data aggregation by comparing with other routing schemes in terms of consumed energy and energy efficiency.

Chapter 5 reviews and concludes the thesis. In addition, some recommendations for future work are given. Finally, the original contributions to knowledge are re-summarised.

Chapter 2

Background

THIS chapter contains details of data-centric routing schemes in wireless sensor networks. The classification of routing schemes in this chapter provide a clear view of routing background and its working mechanism. In addition, data gathering algorithms are introduced to represent three suboptimal schemes for generating data aggregation trees.

2.1 Introduction

Routing is an essential part in the network layer. A routing scheme defines the way of disseminating information, and it enables the information to select applicable routes between any two nodes on a communication network. There are two main components to consider in routing scheme design: communication structure and data control. Before addressing some particular aspects in network layer of wireless sensor networks, it is significant to understand some background of routing approaches, data reduction and data gathering algorithms. Hence, the issues of routing schemes are introduced in this chapter. The introduction provides options and explanations in choosing a suitable method for a particular application. Moreover, for each method in routing scheme the operations and regulations establish boundaries on how the technology may be applied.

2.2 Approach for Routing Schemes

Adopting efficient routing approaches is important for many routing schemes in wireless sensor networks. Approaches, such as multi-hop routing and clustering, are available to improve the performance of schemes in terms of energy efficiency and network organisation.

2.2.1 Multi-hop Routing

In wireless sensor networks, if source nodes are far away from the base station, they will drain energy quickly because of direct transmission. Hence, it is sensible to apply multi-hop routing to reduce the energy consumption of delivering data from the sources to the base station. In the multi-hop routing, one or more intermediate nodes forward data in relay if the energy consumption for transmission can be decreased [9] [10] [11].

Among many energy-aware multi-hop routing approaches, minimum transmission energy (MTE) routing is the most popular one for wireless networks [4] [12] [13]. In

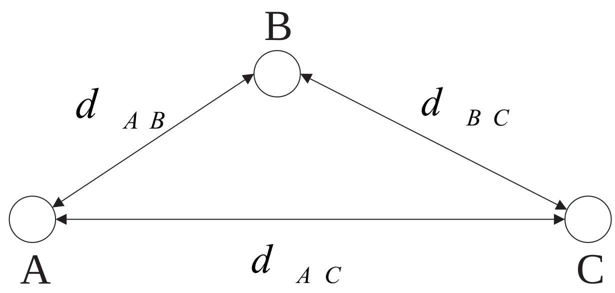


Figure 2.1. Minimum transmission energy (MTE) routing.

MTE, sensor nodes do not only transmit their own sensing data, but also serve as routers for other sensor nodes to deliver data packets from the source node to a given destination so as to minimise the total transmission energy. Assuming power dissipation is proportional to the distance between the transmitting node and the receiving node, consider the situation shown in Fig. 2.1, node A will transmit to the node C through node B if:

$$d_{AB}^2 + d_{BC}^2 < d_{AC}^2. \tag{2.1}$$

This multi-hop routing can be employed in wireless sensor networks where all the nodes must send their data to the base station. Each relay node uses a particular routing scheme to the determine its next hop until data are collected by the base station. The routes maintenance requires nodes to update the paths periodically so as to keep connectivity with the base station. For long-distance transmissions, multi-hop routing routing can dramatically reduce transmission power compared to direct communication.

2.2.2 Clustering

Clustering is an efficient approach that has been proposed to support scalable media access control (MAC) and routing protocols in wireless networks [14] [15] [16] [17] [18]. For large-scale sensor networks, it can be divided into small groups (cluster) using clustering. In each cluster sensor nodes are elected as two roles which are a cluster head and numerous cluster members. In this case, the cluster head collects data from member nodes and then forwards them to the base station.

In addition, the cluster head facilitates in-network data aggregation, which is significant for prolonging the lifetime of network, and effectively manages the network topology. The election of the cluster heads usually involves (i) the determination of a set of parameters for each node either to the whole network or to group of nodes and (ii) the comparison of these parameters in order to choose the best nodes as cluster heads.

There are two typical algorithms to represent how the clustering approach works.

- **WCA**
  The on-demand weighted clustering algorithm (WCA) [19] algorithm proposed by Chatterjee et al. takes a set of parameters, such as the node degree, transmission power, battery power and the speed of the nodes into consideration and calculates an objective function with the four factors to selects a cluster head. It formulates the clustering problem to an optimisation problem for achieving more reasonable maintenance of the network stability.

- **Multi-layer clustering algorithms**
  Multi-layer clustering algorithms have been proposed for efficient routing in wireless ad hoc and sensor networks [17] [20]. The cluster heads on the lower level elect a cluster head from them for the upper level, and they also aggregate data and transmit them to the upper level accordingly. It is noteworthy that the algorithm assumes that network topology changes slowly and infrequently.

Although clustering performs well as mentioned above, there are some drawbacks in it. First, the cluster heads election should know based global information of network, which incurs high overhead and slow reaction to topology changes. Second, it is complex to construct and maintain the desired hierarchical structure in the multi-layer clustering algorithms.

2.3 Data Reduction

In WSNs, it is not feasible to assign internet protocol (IP) address to each node due to the sheer number of nodes deployed. Moreover, in the sensor network, data produced by different neighbour nodes may be highly correlated and redundant. These characteristics shifts the research focus from the traditional address-centric routing, which is to find short routes between pairs of addressable terminals, to a more data-centric routing, which aims at finding routes from multiple sources to a single destination with in-network combination of redundant data.

Unlike the traditional ad-hoc networks, energy resources in WSNs are usually scarce due to the cost and size constraints of sensor nodes. The energy usage for data transmitting and receiving can be defined as follows:

$$E_{Tx} = f(k_{t_{ij}}, d_{i,j}), \tag{2.2}$$

$$E_{Rx} = g(k_{r_{hi}}). \tag{2.3}$$

where E_{Tx} is energy expended for data transmitting, $k_{t_{ij}}$ is amount of transmitted data, $d_{i,j}$ is euclidian distance between node i and j, E_{Rx} is energy expended for data receiving, and $k_{r_{hi}}$ is the number of bits received by relay node i from node h. Since the most important constraint imposed on total energy consumption is the amount of transmitted and received data, minimising the number of transmissions is vital in data-centric applications. In order to minimise the amount of data exchanged among nodes, we can apply data reduction techniques which includes two ideas: aggregation and approximation.

2.3.1 Data Aggregation

In the sensor network, data generated from different neighbour nodes may be highly correlated and redundant. As a result, similar packets from multiple nodes can be aggregated instead of delivering the raw data packets. Since data aggregation is incorporated in routing schemes to reduce the amount of transmissions while still distributing information about the events of interests, it is deemed as an effective approach to balance the communication needs and energy constraints.

Perfect aggregation [21] is the most common aggregation scheme. With perfect aggregation, a sensor node combines data packets received by different neighbours into a single data packet, and then transmits it to the next hop, where minimum, maximum, average or count operations is executed as a perfect aggregation function [22] [23]. Because it is assumed that data from different sources are complete correlated, such an operation can dramatically reduces the traffic load and conserve energy of the sensors [24] [25] [26]. In this sense perfect aggregation is quite efficient and has been assumed in most studies [27] [28] [21] [29]. However, it is not generic and limits the application of wireless sensor networks.

In-network aggregation is more suitable for exploratory, continuous queries that need to obtain a live estimate of some (aggregated) quantity, but it should be chosen carefully according to applications. Because aggregator nodes send aggregated data to the base station, they may be attacked by malicious attacker. If security cannot be guaranteed, the base station cannot ensure the validity of the aggregated data.

2.3.2 Data Approximation

Approximation techniques [30] [31] [32] are lossy data compression approach which mainly occurs in source nodes and the base station. The data values that the sensor transmits to the base station are split into two parts which are base signal and compressed sensor data updates. Each sensor allocates a small capacity of memory to maintain the base signal. Source nodes extract the values of prominent features from the recorded data and treat them as base signal in the approximate representation that is transmitted to the base station. Compared with the base signal, measurement values are processed to be compressed sensor data updates. When the base station receives compressed sensor data updates, it appends the latest updates to a log file. Thus, the obtained approximation approximates the original measurement value collected by sensors.

Whenever the source nodes collect enough data to transmit, the base signal should be checked. It will be updated if sensors capture new features from sensed data. When such updates occur, the base signal updates are transmitted to the base station. After that, the base station appends the base signal updates in a special log file that is unique for each sensor. With this method, the approximation received by the base station adapts to time-varying data.

In such cases, sensor nodes are mostly preserving energy and periodically process and transmit large batches of their measurements to the monitoring station for further processing and archiving, and approximation techniques are thus more suitable for the collection of historical data through long-term queries.

2.4 Data Gathering Algorithms

Energy consumption, scalability, and load balancing are important requirements for many data gathering sensor network applications. Therefore, data gathering algorithms are proposed for achieving development in this area and categorised into cluster-based algorithm, the chain-based algorithm and the tree-based algorithm for power saving in term of transmission structure.

2.4.1 Cluster-based Algorithms

Clustering schemes have been proposed in order to meet the energy efficiency and scalability requirement of the WSNs. The main issue is forming subnetwork clusters, encouraging multi-hop transmission and enabling data aggregation. In addition, tasks for the sensor nodes with different characteristics are also performed. Some cluster-based algorithms have been developed based on hierarchical structures.

- **LEACH**
  Low energy adaptive clustering hierarchy (LEACH) protocol [33] is one of the most popular clustering algorithms with distributed cluster formation for WSNs. The algorithm randomly selects cluster heads and periodically rotates the role so that the energy loads can be shared among the sensors. Time division multiple access (TDMA) or code division multiple access (CDMA) schedules are also created for each cluster node in order to avoid excessive contentions of the channel

and allow sensors to effectively turn their radios off when not actively transmitting. It forms clusters based on the received signal strength and uses the cluster head (CH) nodes as routers to the base station. One way to reduce the amount of communication is to incorporate data aggregation on the cluster heads before sending out packets to the base station.

LEACH forms clusters by using a distributed algorithm, where nodes make autonomous decisions without any centralised control. However, this approach does not take actual energy consumption of the cluster heads into account. Heinzelman et al. also assume that all nodes are time synchronised, nodes have homogenous energy levels initially, use one-hop clustering, and that they can communicate directly with the BS. Researchers have extended LEACH to LEACH-C [34] that adds a central control algorithm considering energy to form the clusters. LEACH-C produces better clusters by dispersing the cluster head nodes throughout the network.

- **TEEN**
  In 2001, A. Manjeshwar and D. P. Agarwal [35] proposed threshold sensitive energy efficient sensor network (TEEN) protocol. TEEN is also LEACH-like for cluster head election, but is more threshold-sensitive hierarchical. In this protocol, the cluster head sends its members two threshold values. One is hard threshold which is minimum possible value of the sensed attribute to trigger a sensor node. Hard threshold allows nodes to transmit the event only when the sensed attribute is in the range of interest, thus reducing the number of transmissions significantly. The other is soft threshold which is a small change in the value of the sensed attribute. Once the hard threshold is satisfied, the node is triggered to transmit data only when the current value of the sensed attribute differs from sensed value by an amount which ia equal to or greater than the soft threshold. Consequently, the soft threshold further prevents from the redundant data transmission.

Since the protocol will be reactive if there are sudden changes in the sensed attribute, it is suitable for time-critical applications. However, TEEN is not applicable for periodic reports because the user may not get any data at all if the thresholds are not reached.

- **HEED**

  More recently, HEED [9] is proposed by Younis and Fahmy in 2004. The protocol extends LEACH by incorporating communication range limits and cost information. In HEED, the initial probability for each sensor to become a cluster head is dependent on its residual energy. In references [36] [37] [38], the suitable cluster head is selected by combining load balancing, topology and energy information, but without considering data like in TEEN, and these algorithms are a little complex. Some signal strength-based clustering algorithms, such as references [39] [40], can be used in application of target tracking, but they do not achieve energy optimisation.

  HEED uses the clustering approach by considering the residual energy and has a constant iteration number, so it can prolong the network time and suit for large network. However, since HEED enables every node to independently and probabilistically decide on its role in the clustered network, it cannot guarantee optimal elected set of cluster heads. And the one-hop routing approach in HEED only perform well in a case when the cluster head close to the base station. When a cluster head locates far from the base station, it may consume more energy to forward data to the base station via one hop.

2.4.2 Chain-based Algorithms

In cluster-based sensor networks, sensors transmit data to the cluster head where data aggregation is performed. However, if the cluster head is far away from the sensors, they might expend excessive energy in communication. Further improvements in energy efficiency can be obtained if sensors transmit only to close neighbours. The key idea behind chain based data aggregation is that each sensor transmits only to its closest neighbour.

- **PEGASIS**

  Power-efficient gathering in sensor information systems (PEGASIS) protocol [25] is a LEACH-inspired protocol. PEGASIS is not exactly a cluster-based protocol, as nodes are not explicitly grouped into clusters. Instead of forming clusters, PEGASIS is based on forming chains of sensor nodes. In the chain, each node only aggregates the collected data with its own data, then passes the aggregated data to a close neighbour. Only one sensor node is elected from the chain to communicate with the base station. The nodes then transmit until the data reach the base station, in this way the amount of energy spent per round is significantly reduced. This approach distributes the energy load evenly among the sensor nodes in the network.

  The difference from LEACH is to employ multi-hop transmission and selecting only one node to transmit to the sink or base station. Since the overhead caused by dynamic cluster formation is eliminated, multi hop transmission and data aggregation is employed, PEGASIS outperforms the LEACH. However, excessive delay is introduced for distant nodes, especially for large networks and single leader can be a bottleneck.

- **COSEN**

  In contrast to PEGASIS, chain oriented sensor network for efficient data collection (COSEN) [41] is a two-layer hierarchical chain-based routing scheme. In the scheme, sensor nodes are divided into several low-level chains according to geographical location. For the low-level chain leader election, energy efficiency is a major concern, so the chain leader is the sensor node with the maximum residual energy. Furthermore, the low-level leaders are regarded as members of a high-level chain, and the high-level chain leader will be elected accordingly. During the phase of data communication, all member nodes execute a similar procedure as that in PEGASIS to send their aggregated data, via their respective low-level leaders and the high-level leader, toward the base station.

  COSEN, compared with PEGASIS, can decrease the transmission delay and energy consumption, but it still gives rise to a considerable amount of redundant transmission paths, especially for those nodes which are near to the base station but have to detour their aggregated data toward the chain leaders.

2.4.3 Tree-based Data Gathering Algorithm

In order to avoid drawbacks of long one hop transmission in cluster-based algorithm and excessive delay in cluster-based algorithm, tree-based data gathering algorithm is proposed. In a tree based network, sensor nodes are organised into a tree structure where data aggregation is performed at intermediate nodes along the tree and a concise representation of the data is transmitted to the root node. Tree based data aggregation is suitable for applications which involve in-network data aggregation. One of the main aspects of tree-based networks is the construction of an energy efficient data aggregation tree.

- **Shortest path tree algorithm**

  Shortest path tree (SPT) based on Dijkstra [42] algorithm is to find a set of edges connecting all nodes such that the sum of the edge lengths from the source to each node is minimised. Since each node in the network has some data to transmit to the base station, it can be viewed as a network with sources and one base station. If the redundancy among the information gathered by different sensors is fairly low, then it can be interpreted that the entire data sensed by each node travels to the base station. Let the energy cost for each packet be the sum of energy consumed by every node involved in its transmission from the source to the base station, it can be concluded that shorter the path traveled by the packet and less the number of hops, lower is the energy expended. SPT constructs trees based on this notion.

  In order to minimise the total path lengths, the path from the root to each node must be a shortest path connecting them. Since, weight on the edges in the graph represents energy for communication, the SPT algorithm also accounts for the number of hops between a node and the base station. Therefore, SPT-based approach is used for tree construction when there is no redundancy among the information gathered by different sensors. While a class of SPT-based routing algorithms has been developed in [43] [44] [45] assuming statistically independent information, the more realistic case of correlated data has also been considered in [46] [47] [48] [49] [50].

- **Minimum spanning tree algorithm**

  Minimum spanning tree utilises prim algorithm [51] to enable each node to select the nearest neighbour node to be its parent node, and incrementally add nodes to construct a routing tree. Consider the data between different sensors are complete correlated. In this case, a packet transmitted by any node will be eliminated after one hop. To spend least energy in transmitting a packet, a node should transmit it to its closest (by weight) neighbour towards the base station.

  Since sensor nodes only know local network topology, this kind of greedy algorithm can play a part in scalability of network and equalisation of energy consumption to some extent. As a consequence, MST-based approach is applicable for tree construction when the gathered data are identical for every sensor.

- **Shallow light tree algorithm**

  Shallow light tree (SLT) algorithm [52] [53] is to find a spanning tree that simultaneously approximates a shortest-path tree and a minimum spanning tree. The basic idea of the algorithm is to regard the MST algorithm as an initial tree structure, traverse the current tree, and check each vertex whether the distance requirement for that vertex is met in the current tree. If it is not met, the edges of the shortest path between the vertex and the root are substituted for current edges so as to maintain a tree structure, and the weight of other the vertices which connects to the new edges is update accordingly. After all vertices have been checked and paths have been added as necessary, the remaining tree is the desired SLT.

2.5 Conclusion

This chapter has introduced how the routing schemes of wireless sensor networks are classified, developed, operated and regulated. The tree-based data gathering algorithms have been introduced since the correlated data gathering and aggregation are the major concern of the work in this thesis. The contents of this chapter provide a wireless routing background and serve as complementary materials for understanding the rest of the thesis.

Chapter 3

Optimisation on Data Gathering and Aggregation

PARTICLE swarm optimisation (PSO) algorithm for optimising data gathering and aggregation is described in this chapter. A modification on PSO algorithm which maps elements of algorithm to practical problem is discussed. Following this discussion, a novel data gathering algorithm with aggregation for minimising energy consumption is proposed, analysed and verified.

3.1 Introduction

Data aggregation is one significant issue in network routing of energy-constrained wireless sensor networks since routing with aggregation can consume less energy than delivering data to base station directly. The energy consumption is determined by traffic control and data transmission. Hence it is necessary to find a transmission structure that cooperates with data aggregation in order to minimise energy consumption. In such cases, routing tree with data aggregation is a reasonable routing scheme [23].

Our research problem is to find a routing tree with data aggregation to achieve minimum energy expense. This problem is an NP-complete problem [48] [54] because it can be reducible to weighted set cover problem in graph theory, which has been shown to be NP complete [55]. Since solving the NP-complete problem requires intense computation cost that is super-polynomial in the input size, it may be enough to find a near optimal solution to get a satisfactory result in polynomial time instead of an exact solution. Heuristic algorithms are applied to either give nearly the optimal answer fast and easily or provide a solution not for all instances of the problem.

This chapter consists of ten sections. Section 3.2 introduces network model, data correlation model and energy model which are used to set up network environment in our study. Section 3.3 formulates the research problem and describes the optimisation objective. In order to solve the problem, Section 3.4 and Section 3.5 introduce three heuristic algorithms to provide possible solutions. Section 3.6 and Section 3.7 provide a novel routing scheme which is aiming at achieve minimum energy expense by optimising aggregation tree on data traffic and communication structure. The performances of our proposed routing scheme compared with some existing routing algorithms are analysed in Section 3.9. Finally, we provide our conclusion in Section 3.10.

3.2 System Model

3.2.1 Network Model

A set of sensors are assumed to be dispersed in a field and quasi-stationary. All communication is over wireless links. Wireless links are considered bidirectional and symmetric so that any two nodes can communicate using the same transmission power levels only if they are in range of each other. Sensor nodes are homogenous and arbitrarily allocated with equal initial energy. Nodes are not location-aware, i.e., not equipped with GPS-capable antenna, and nodes are left unattended after deployment. Therefore, battery recharge is not possible. Efficient, energy-aware sensor network routing schemes are thus required for energy conservation.

From a viewpoint of graph theory, a wireless sensor network can be represented by an undirected graph $G(V, E)$, where V represents the set of all sensors in the network, and $E \subset V \times V$ represents the set of communication links between a pair of nodes. Graph $G(V, E)$ contains $n = \mid V \mid$ nodes and $l = \mid E \mid$ links. It is defined that every node is $v_i \in V$ $(1 \leq i \leq n)$, the set of source nodes is $S \subset V$, $d \in V$ is the base station in wireless sensor networks, and every link is $e_{ij} = (v_i, v_j) \in E$ $(1 \leq i \neq j \leq n)$.

All traffic generated by sensors are destined for the base station, composing a routing tree. The problem on routing is to find out an optimal routing tree when transmitting data from a number of sensing nodes to the base station so that the base station can promptly detecting or tracking information of observed region and conduct corresponding processing.

3.2.2 Correlation of Data

In this thesis, we consider a multi-hop wireless sensor network with one base station and n sensors distributed uniformly in a sensor field. The base station sends a query and k $(k \leq n)$ sensors respond to that query. We consider the problem of efficiently aggregating the information sent by the k sensors to the base station.

Specifically, the goal is to optimise the message complexity for sending data generated by the k sensors to the base station. It is assumed that there is correlation among the data generated by the k sensors. To possibly accommodate a wide range of scenarios, we abstract data redundancy among two sensor nodes using a single value ρ, termed correlation coefficient. ρ will determine the amount of data reduction due to aggregation. Assume that data amount after aggregation is not less than any of its inputs and not more than the sum of all inputs, $\rho \in [0, 1]$.

We consider the case when there is a reasonable degree of correlation between the information collected from different sensors. For example, let R_{v_i} and R_{v_j} be the amount of data generated by two sensors in response to a query. Without loss of generality, if the size of data sent by the two sensor is not the same, we have $R_{v_i} > R_{v_j}$. Hence, the output from the aggregator node is:

$$R(v_i, v_j) = R_{v_i} + (1 - \rho)R_{v_j} \tag{3.1}$$

where $R(v_i, v_j)$ is the amount of data after aggregating R_{v_i} and R_{v_j}. For this case, when the information from the two sensors are perfectly correlated ($\rho = 1$), we see that the message size after aggregation is the same as the amount of data generated by one source (R_{v_i}). On the other hand, if there is no correlation ($\rho = 0$) between the two sensed data, the message size after aggregation is $R_{v_i} + R_{v_j}$.

As the data of nodes are correlated, the amount of data traffic depend on the transmission structure, so for an arbitrary ρ the optimal solution for data gathering is not fixed and unique. For example, as shown in Figure 3.1, assume $R_{v_i} = R_{v_j} = R$, according to the aggregation rule mentioned above, if $\rho \leq \frac{1}{2}$ then the cost of case (a) is less than the cost of case (b), otherwise case (b) has a cost less than case (a). Therefore data traffic depend on the structure of communication, and hence optimising the cost function is difficult.

3.2.3 Energy Model

The free space model is used to predict the received signal power of each packet. The free space propagation model assumes the ideal propagation condition that there is only one clear line-of-sight path between the transmitter and the receiver. The following equation calculates the received signal power in free space at distance d from the

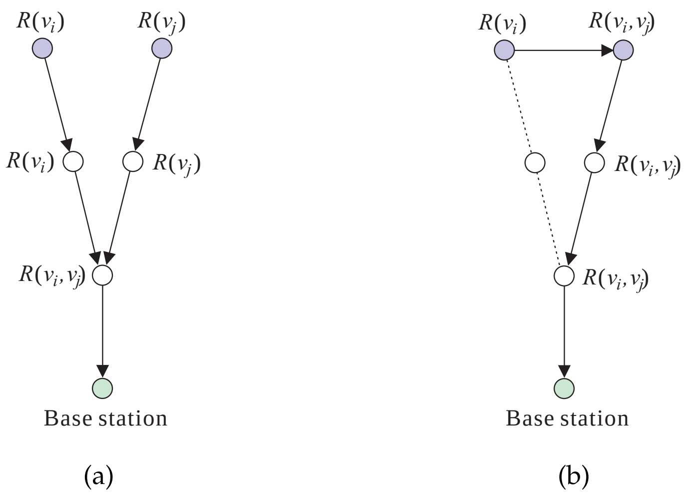


Figure 3.1. Two transmission structures with aggregation.

transmitter.

$$P_r(d) = \frac{P_t G_r G_t \lambda^2}{(4\pi)^2 d^2 L} \tag{3.2}$$

where P_t is the transmitted signal power, $P_r(d)$ is the received power which is a function of the distance between the transmitter and the receiver, G_t and G_r are the antenna gains of the transmitter and the receiver respectively, L is the system loss factor, and λ is the wavelength of the transmitted signal.

In this thesis, by assuming that the gains between a pair of transmitter-receiver are the same in both directions, nodes use (3.2) to compute the distance to their neighbours and the base station.

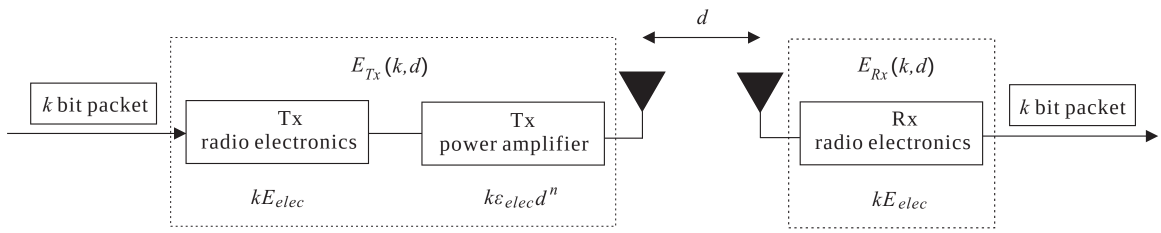


Figure 3.2. Radio energy dissipation model.

Furthermore, according to the radio energy dissipation model illustrated in Fig. 3.2, we adopt both the free-space propagation model and the two-ray ground propagation model to approximate path loss sustained because of wireless channel transmission [56] [57] [58]. The unit energy consumption for transmitter or receiver electronics is defined as E_{elec}, which a radio dissipates to run the transmitter or receiver circuits. Both the free space (d^2 power loss) and the multi-path fading (d^4 power loss) channel models are used in this model, depending on the distance between the transmitter and the receiver. The energy expanded by the radio for transmitting a k bit message over a distance d, is given by:

$$E_{Tx}(k,d) = \begin{cases} kE_{elec} + k\epsilon_{fs}d^2, & d < d_0; \\ kE_{elec} + k\epsilon_{tg}d^4, & d \geq d_0; \end{cases} \tag{3.3}$$

$$d_0 = \sqrt{\frac{\epsilon_{fs}}{\epsilon_{tg}}} \tag{3.4}$$

$$E_{Rx}(k) = kE_{elec} \tag{3.5}$$

where E_{Tx} is the energy dissipated in the transmitter, E_{elec} is the per bit energy dissipation for running the transceiver circuitry, ϵ_{fs} and ϵ_{tg} are both parameters but for different propagation model, d_0 is the cross-over distance, and E_{Rx} is the radio expends of receiver.

It is assumed that the radio channel is symmetric, which indicates that the energy cost of transmitting a packet from v_i to v_j is the same as the cost of transmitting a packet from v_j to v_i.

In our network model, a father node can always aggregate the data gathered from its children into a single length-fixed packet, and therefore, it also consumes E_{da} amount of energy for data aggregation. We use the same parameters as in [58]: E_{elec} = 50nJ/bit, d_0 = 87m, ϵ_{fs} = 10pJ/bit/m^2, and ϵ_{tg} = 0.0013pJ/bit/m^4.

3.3 Description of Optimisation Objective

It has been proved that the tasks performed by the sensor nodes that are related with communications (transmitting and receiving data) spend much more energy than those related with data processing and memory management [59] [60]. Since one of the main concerns in wireless sensor networks is to save as much energy as possible, it would be preferable that the routing algorithm can perform as much data processing as possible in the network nodes, than transmitting all data through multi-hop to the base station to be processed there. Therefore, the key optimisation objective is considered: the data traffic and transmission structure.

Given the set of source node S and the base station d in wireless sensor networks $G(V, E)$, our objective is to find a connected subgraph $G'(V', E') \subseteq G$, where $S \subset V'$, and $d \in V'$, in order to minimise the energy consumption for transmitting data from all source nodes to the base station. The objective function is formulated as follows:

$$\min \sum_{v_i \in V'} R_{v_i} w(v_i, d) \tag{3.6}$$

subject to:

$$\sum_{i,j \in V', i \neq j} c_{ij} \geq 1 \tag{3.7}$$

$$\mid V' \mid = p \tag{3.8}$$

$$p \leq n \tag{3.9}$$

where $w(v_i, d)$ is the total weight of edges on path from node i to base station on constructed aggregation tree. In wireless networks, the weight of each edge can be considered as the energy expended for per unit data communication and data aggregation. R_{v_i} is the amount of data generated by node v_i. $c_{ij} \in \{0, 1\}$ indicates the link state between nodes i and j. $c_{ij} = 1$ means the link is active and $c_{ij} = 0$ otherwise. As the connectivity constraint, in (3.7) means the sensor node can have one communication link with its neigbour at least. Constraints (3.8) and (3.9) guarantee that the number of active node (all source nodes, intermediate nodes and the base station) p cannot exceed the total number of sensors n in wireless sensor networks.

Assume each source (node) is a discrete random variable with specific amount of data to transmit. It should be noted that generated amount of data R_{v_i} in each relay node depends on incoming flow to that node. Because of correlation between data sources, each relay node can combine available data of children nodes into new data. Clearly each relay node reduces the correlation of received data from other nodes. Therefore, R_{v_i} is a decreasing function of the amount of incoming flow because data aggregation always reduces amount of information.

Assume that relay nodes can perform data aggregation on data gathered from different sensors, not only forwards them. In such correlated data network, the amount of transmitted data for each node depends on network aggregation structure, so changing the structure of network tree affects both the amount of transmitted data of each node (R_{v_i}) and the path cost ($w(v_i, d)$). Therefore, we have a joint treatment of data aggregation and transmission structure to find an optimal routing tree for minimising (3.6). Because finding an aggregation tree for minimising cost function (3.6) is NP-complete, using a heuristic algorithm is sensible.

3.4 Swarm Intelligence

Swarm intelligence (SI) [61] [62] constructs stochastic optimisation algorithms by imitating group behaviour in nature. Since it emerged in 80's of 20th century, this raised multiple disciplines researchers' attention on this field. As a result, SI has become a hot and frontier of artificial intelligence and interdisciplinary involved in economy, society and creatures. SI exploits group advantages to provide a novel perspective on searching solution for complex problems under the premise that non-centralised control and no global model exist.

Some scholars extended definition on swarm intelligence and regarded social behaviour of some creatures, such as fish school, bird flock and ant colony, as swarm intelligence behaviour, but there are following two common understandings. One is collective intelligence expressed by a set of simple intelligence agents, which is exemplified in ant colony optimisation (ACO) and ant clustering algorithm (ACA). The other regards members of group as particles instead of simple intelligence agents, which is exemplified in particle swarm optimisation (PSO).

3.4.1 Ant Colony Optimisation

Ant colony optimisation algorithm is one of the essential branches of swarm intelligence, and it is a stochastic search algorithm which can find the optimal solution by evolving the population of candidate solutions. The algorithm inspired by the behaviour of a real ant colony was proposed by Dorigo in 1992 [63]. After substantial observation and careful research, biologists found that ants communicate with each other via a substance called pheromone.

As a real ant moves, it deposits and senses pheromone on the ground. When an ant reaches a point that has more than one outgoing branch, the probability of selecting a certain branch is dependent on the amount of pheromone deposited on each branch. More ants passing along a path results in more pheromone being left on that path resulting in a higher probability of the path being selected. The pheromone on the shortest path will grow faster than that on other branches. Hence, individual ants are able to follow this path and find food. On the other hand, with time flying, the pheromone will dissipate in a certain proportion, so pheromone on the paths which fewer ants passed will disappear gradually.

The ant colony optimisation algorithm is jointly completed by artificial ants, and it imitates the collaboration of real ant colony. Each artificial ant independently searches solutions in the space of candidate solutions, and deposits a certain amount of information. Better performance of the solution results in more information being left on that solution resulting in a higher probability of the solution being selected. At the initial stage, the amount of information allotted to solutions is identical. With the run of algorithms the amount of information deposited on the optimal solution will increase, and the algorithm will gradually converge.

Because of the advantages of the ACO algorithm, it has been employed in a wide range of applications. The ACO algorithm was first successfully used to solve traveling salesman problem (TSP) [64], then it was applied to solve scheduling problem accompanied by in-depth research. In telecommunication networks, the ACO algorithm can be exploited to solve capacity balancing problem as well. However, the ACO algorithm still has some disadvantages. For example, it needs long time to execute calculation and stagnation often happens.

There are many improved versions for the ACO algorithm so far, but they substantially stem from the standard ACO algorithm. In order to illustrate the standard ACO algorithm well, we take symmetrical TSP of n cities as an example to briefly introduce how the ACO algorithm finds the shortest path. At the initial stage, we assume that there are m ants, and each path has the same amount of pheromone $\tau_{ij}(0) = C$ (C is constant). During the motion, the kth ($k = 1, 2, ..., m$) ant decides the transition direction according to the amount of pheromone on each path. At the moment t the transition probability from city i to city j is

$$p_{ij}^{k}(t) = \begin{cases} \frac{[\tau_{ij}(t)]^{\alpha}[\eta_{ij}(t)]^{\beta}}{\sum\limits_{s \in allowed_k} [\tau_{is}(t)]^{\alpha}[\eta_{is}(t)]^{\beta}}, & j \in allowed_k \\ 0, & otherwise \end{cases} \tag{3.10}$$

where $allowed_k$=$\{\{0, 1, ..., n-1\} - tabu_k\}$ is a set of cities which can be selected in next step. Unlike the actual ant colony, the artificial ant colony has memory function. $tabu_k$ ($k = 1, 2, ..., m$) is used to record the cities passed by ant k. The set $tabu_k$ adjusts dynamically with evolution. α is heuristic information factor, and it demonstrates the effect of accumulative information in the course of of ants movement. The bigger α is, the more the ant is inclined to choose the path passed by other ants and the stronger collaboration among ants. β is heuristic expectation factor, and it indicates the effect of heuristic information in the course of of ants movement. $\eta_{ij}(t)$ represents the heuristic information from city i to city j, normally $\eta_{ij}(t)$ is defined as follows:

$$\eta_{ij}(t) = \frac{1}{d_{ij}} \tag{3.11}$$

where d_{ij} is the distance between city i and city j. As for ants, the smaller d_{ij} is, the bigger $\eta_{ij}(t)$ is and the bigger $p_{ij}^{k}(t)$ will be. Obviously, the heuristic function shows the expectation degree of transfer from city i to city j. When ant completes a cycle after n moments, the amount of pheromone on each path has to modify according to the following formula:

$$\tau_{ij}(t+n) = (1-\rho)\tau_{ij}(t) + \rho\Delta\tau_{ij}(t) \tag{3.12}$$

where the parameter ρ demonstrates pheromone evaporation rate, while $1-\rho$ indicated a pheromone residue factor. $\Delta\tau_{ij}$ is the increment of the amount of pheromone on path (i, j) in this cycle.

3.4.2 Particle Swarm Optimisation

Particle swarm optimisation (PSO) algorithm stems from the simulation of a simplified society model, and it was proposed as a typical swarm optimisation algorithm inspired by the behaviour of bird flocking [65]. Scholars found that during the flight birds often make a sudden change of direction, disperse or get together. Although their behaviours are usually unpredictable, the whole population always maintains consistent, and the individual keeps the most appropriate distance between each other as well.

Through the research on the behaviour of similar creature population, the scholars found that there exists a sharing mechanism of social information in the population of creature, and it brings an advantage of population evolution and forms the basis of the PSO algorithm. Besides, human beings usually take advantage of their and others experience for decision-making, and this behaviour also composes a basic concept of the PSO algorithm. Based on these ideas, Eberhart and Kennedy first proposed the particle swarm optimisation algorithm in 1995.

When using the PSO algorithm to solve optimisation problems, the solutions corresponds to the position of a bird in searching space, and the birds are named as "particle" or "agent". Each particle has its own position, velocity and fitness value. In addition, each particle records the position of current optimal particle and searches towards this position in the solution space. Each iteration is not completely stochastic because the next solutions will be searched on this basis of the existing best solution.

The PSO algorithm first conducts population initialisition for stochastic particles, then in each iteration particles update themselves by following two extreme values. The one is the best solution found by the particle itself, which is named individual extreme point (represented by p_{ibest}). The other is the current best solution found in the population, which is represented by p_{gbest}. The PSO algorithm finds out the optimal solution by means of collaboration among particles. It exploits the idea that information sharing can generate evolutionary advantage in biological populations, while genetic algorithm (GA) is based on Darwin's theory on evolution, namely survival of the fittest.

The positions of particles are generated randomly in the searching space, and the initial velocity of each particle is randomly given as well. The concept of the PSO algorithm is based upon population collaboration and information exchange accelerating the motion of each particle to the best positions.

Assume that in a D-dimensional searching space m particles compose a swarm and fly with a certain velocity, where the particle i denotes a D-dimensional vector $x_i = \{x_{i1}, x_{i2}, ..., x_{id}\}$, $i = 1, 2, ..., m$. The performance of x_i is assessed by its fitness value which is calculated through substituting x_i into the objective function. The motion of particles changes according to the following equations:

$$v_{id}^{k+1} = \omega v_{id}^{k} + c_1 rand()(p_{ibest}^{k} - x_{id}^{k}) + c_2 rand()(p_{gbest}^{k} - x_{id}^{k}) \tag{3.13}$$

$$x_{id}^{k+1} = x_{id}^{k} + v_{id}^{k+1} \tag{3.14}$$

where k is the iteration number, v_{id} denotes the velocity of the particle i in the dth-dimensional ($1 \leq d \leq D$) particle swarm space, x_{id} represents the particle i current position, p_{ibest} is its best previous position, and p_{gbest} is the best position among all particles in the population. Function $rand()$ is a random function with a range [0, 1]. Constants c_1 and c_2 are learning factors used to accelerate the motion speed of particles. The inertia weight, ω is a user-specified parameter that controls, with c_1 and c_2, the impact of previous historical values of particle velocities on its current one. A large inertia weight facilitates global exploration while a small inertia weight facilitates fine-tuning local search. In each dimension, the velocity of particle will be limited in a upper limit V_{max} ($V_{max} > 0$) to maintain v_{id}^{k+1} within a reasonable range.

In (3.13) the first term is the velocity of particle in the last iteration. The second term is the cognition part, and it indicates that the particle flies to the best position which has been found by itself. The third term is the society part, and it demonstrates that information sharing and collaboration guide the particles towards the current best position of population. (3.14) is employed to calculate the new coordinates of particle's position. (3.13) and (3.14) determine the next motion and location of the particle jointly. The motion tracks can be illustrated in the Figure 3.3.

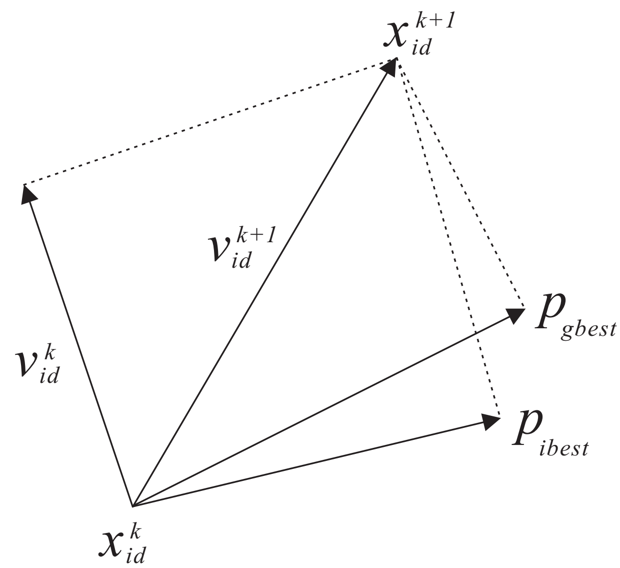


Figure 3.3. Motion tracks of a particle.

3.5 Genetic Algorithm

Genetic algorithm (GA) is a stochastic search method which is inspired by evolution laws of nature [66]. The main property of the GA is that the algorithm can operate directly on the objective without constrained by derivation or continuity of function. The procedure of the GA is as follows:

Step 1. Encoding: Because the GA cannot process data of solution space directly, so we express them as genotype string data of genetic space through encoding. Generally, the variable is expressed by binary encoding. Because binary encoding requires transform between real numbers, decimal system is sometimes used to encode naturally.

Step 2. Population initialisation: In the GA a string data is called an individual, and N individuals compose a population. First, system generates N string data randomly. The iteration of the GA then starts with an initial point which refers to the N string data.

Step 3. Fitness evaluation: Substitute initial population into pre-defined fitness function, and use the fitness values to determine the merits of individuals.

Step 4. Selection: According to the fitness values computed by **Step 3**, select the fittest individuals as parents to reproduce offsprings.

Step 5. Reproduction: In this step, generate new individuals from the mating pool by crossover and mutation operations.
a.) Crossover: The crossover operation takes a pair of parents from the mating pool randomly, and gives a pair of offsprings chromosomes by exchanging substrings of the two parent chromosomes. The new offsprings inherit the parents' features which represent an exchange and combination of information. After the crossover operation, we replace the parents in the mating pool by their offsprings. The mating pool has therefore been modified, but still maintains the same number of elements.
b.) Mutation: The mutation is an important method for preserving the diversity of the individuals by introducing small and random changes in to them. In order to achieve that, chromosomes are taken from the mating pool randomly, and the value of a gene is modified with a given probability. The mutation supplies an opportunity for generating new individuals in a larger range.

Step 6. Stopping criteria check: If the stopping criteria is satisfied, the GA ends, otherwise return to **Step 3**.

3.6 Proposed Solution to the Energy-Efficient Problem

The central theme of the problem mentioned in Section 3.3 is to find out a transmission structure which can minimise energy consumption in a large wireless sensor network where the data are correlated. For this optimisation problem, the swarm intelligence can efficiently approximate to the optimal solution.

This thesis proposes a novel routing scheme for data gathering and aggregation. In the routing scheme, source nodes first release control packets and enable them to move randomly from the source nodes to the base station. When the control packets arrive at the base station, the paths passed by the packets are then encoded to be individuals, and initial population is formed accordingly. By means of selection, crossover and mutation among the individuals, we can find a near optimal routing tree for data gathering and aggregation.

This scheme based on the swarm intelligence consists of

- an ACO algorithm which is exploited to provide candidate solutions,
- a PSO algorithm that aims at finding out a near optimal aggregation tree which minimises energy consumption of routing.

3.7 PSO Modified by GA for routing with aggregation

For a routing tree with data aggregation, the existing algorithms introduced in Section 3.4 and Section 3.5 are insufficient to implement optimisation. Solely using the ACO algorithm needs a considerable number of control packets, and the overhead will increase accordingly. The genetic algorithm can evolve into a satisfactory result, but its convergence is slow. Optimisation on an aggregation tree relates to path selection, and path selection belongs to a discrete problem. The standard PSO algorithm is the real valued PSO, and it cannot operate addition or subtraction directly on the path. Hence, the standard PSO algorithm should be extended to deal with the discrete optimisation problems which require the ordering or arranging of discrete elements, eg. the routing tree with aggregation problem.

A popular solution is to keep the velocity update equation (3.13) unchanged, but the actual new position components are changes to be 1 or 0 with a probability [67]. Another method views the position vectors of the particles as probabilities and employing roulette wheel selection for discretisation [68]. The two techniques both extend the real valued PSO to its binary version, namely Binary PSO (BPSO), by a simple discretisation of the values of velocities or positions of particles. A more general discrete PSO is proposed in [69] to solve the traveling salesman problem. The method redefines the six basic mathematical objects and operations in the discrete space.

Since the standard PSO cannot conduct optimisation directly on the discrete problem, it needs an equivalent form for velocity and displacement equation. In velocity equation, subtraction between the optimal position and the current position embodies the trend that particles close to the optimal position. In such cases, the subtraction, which can be understood as the process of information exchange, is similar to the crossover operator of the GA. Multiplication between inertia weight and velocity can be interpreted as extending search range, and it is similar to the mutation operator of the GA. As a consequence, we consider to propose a PSO algorithm which is modified by the GA in order to address the discrete nature of our optimisation problem. Hence, (3.13) and (3.14) are replaced by (3.15) and (3.16).

$$v_{id}^{k+1} = \omega v_{id}^{k} \oplus (p_{ibest}^{k} \otimes x_{id}^{k}) \oplus (p_{gbest}^{k} \otimes x_{id}^{k}) \tag{3.15}$$

$$x_{id}^{k+1} = x_{id}^{k} \oplus v_{id}^{k+1} \tag{3.16}$$

where k is the iteration number, v_{id} denotes the velocity of particle i in the dth-dimensional ($1 \leq d \leq D$) particle swarm space, x_{id} represents the particle i current position, p_{ibest} is its best previous position, p_{gbest} is the best position among all particles in the population, the operator $\oplus$ represents a composition, and the operator $\otimes$ indicates information exchange which is implemented by crossover.

Through the study of the distributed properties of the ACO algorithm, we exploit a relatively small number of control packets to generate some candidate paths, collect and forward the information about the network to the base station. In the base station a set of routing trees is encoded as the individuals, and employ the PSO algorithm to find out the near optimal transmission structure. The base station then sends the determined routing information to each source node, and enables the source nodes and relay nodes to update their routing tables. Fig. 3.4 shows the conceptual flow of particle swarm motion process.

In the following sections, we define our particle representation, specify the initial population, describe the fitness function and selection, and the strategies for crossover and mutation.

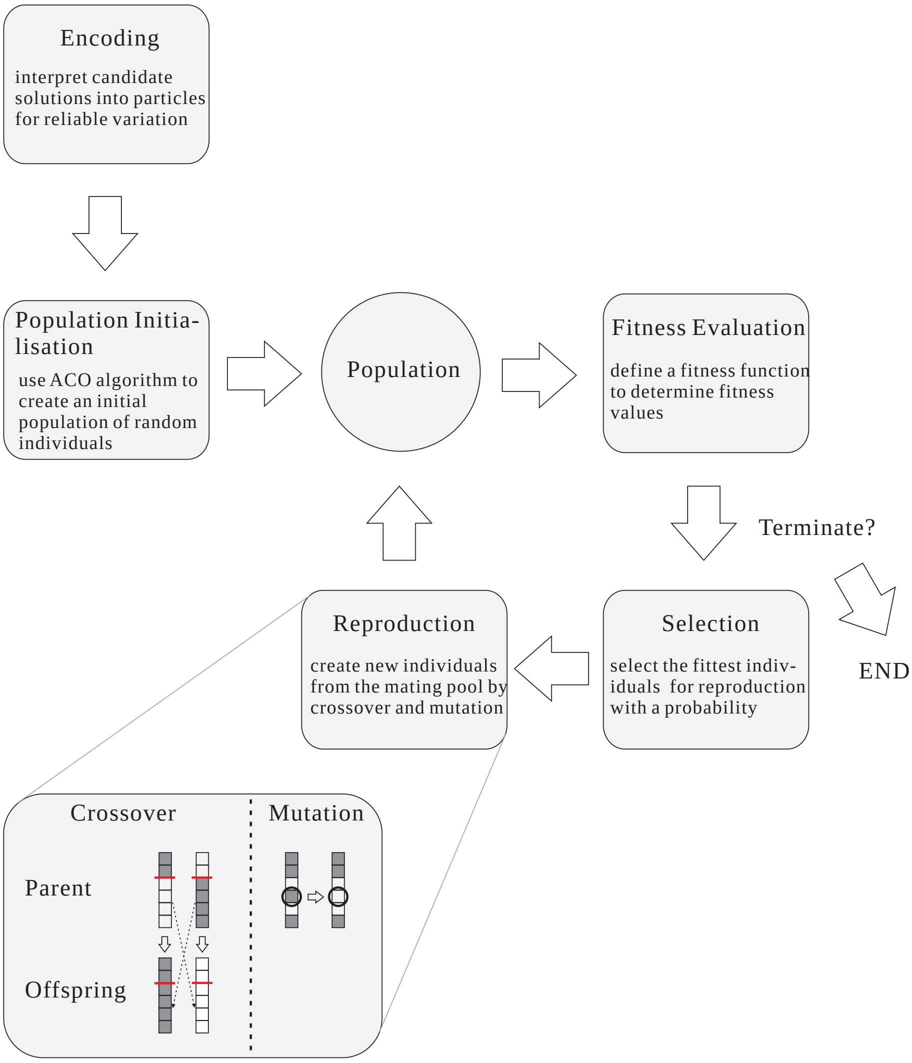
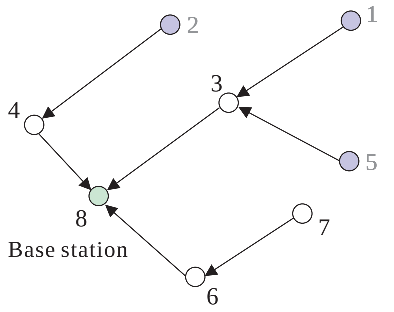


Figure 3.4. Discrete PSO algorithm conceptual flow.

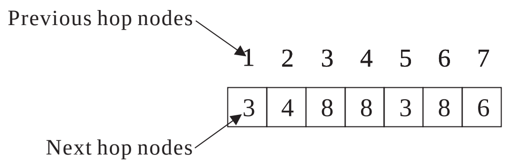
3.7.1 Encoding

Encoding involves coding paths serial into a feasible solution (or a position) in the search space. In the case of encoding, different schemes which are applied to corresponding problems will affect the design of selection, crossover or mutation directly, thereby influencing the convergence and complexity of the PSO algorithm. Usually Prüfer number [70] is used for tree encoding, but this manner has complex encoding-decoding process and bad locality that small variations in the representation may cause big adjustments in the network structure. Also it has poor heritability and if the graph is incomplete then this method represents infeasible solutions. Gen et al. [71] utilises priority-based encoding scheme, but this scheme needs to know neighbour nodes' priority in advance and is non-applicable for solving our problem. Ahn and Ramakrishna [72] makes use of variable-length encoding scheme which can narrow search space. However, this design will increase complexity of crossover or selection as result of variable-length of string data. Encoding approach proposed by Munetomo et al. [73] is similar to the encoding method of the TSP [74] which is based on path, but it is difficult to do crossover. Therefore, we adopt fixed-length position encoding to perceive routing tree directly.

We represent the individual, for a specific aggregation tree, as a string of node numbers. The length of each individual is always equal to the number of relay nodes. A routing scheme for a network with 7 relay nodes, and one base station, is shown in Fig. 3.5(a) and the corresponding particle is shown in Fig. 3.5(b). In this example, the value of the gene in position 1 is 3, indicating that node 1 transmits to node 3. Similarly, the value in position 3 is 8, indicating that node 3 transmits to node 8 (the base station).



(a) Routing tree



(b) Individual (Solution representation)

Figure 3.5. Encoding a routing tree as an individual. For (b), the number serial at the upper string indicates the previous hop nodes while the number serial at the lower string represents the next hop nodes.

3.7.2 Population Initialisation

In general, there are two ways to generate the initial population, heuristic initialisation and random initialisation. For most large scale problems, such as network communication design, random initialisation has better effect on global optimal solutions. The ACO algorithm mentioned in Section 3.8.1 is used to generate initial population that consists of random trees. The ACO algorithm is an algorithm that finds paths for connected graphs. It starts from source nodes and selects next hop nodes in accordance with pheromone. If pheromone is updated according to transition probability functions, the next hop nodes are randomly selected to form the paths with the ACO algorithm. As a consequence, we have an initial population with random trees. Note that all initial particles are valid and no repair function is needed.

We improve the performance of particle swarm search by determining a proper initial population. The main idea of our work is that the correspondence of nodes behaviour through local optimisation (versus global optimisation) is greater than the correspondence of nodes behaviour in random initialisation. The SPT algorithm and the MST algorithm have good results in initial population, when correlation coefficient is equal to 0 and 1 respectively. For instance, when correlation coefficient approaches 0, if in the SPT algorithm, node v_i connects to v_j, then in the global optimum the probability for v_i to connect to v_j is greater than the probability of v_i to connect to another node. Hence the probability of similarity between corresponding elements of the local optimum and the global optimum is greater than corresponding elements of random vector and global optimum.

We can say that the probability of mapping the values of optimised nodes (by The SPT algorithm or the MST algorithm) to values of nodes in the global optimum is greater than the probability of mapping the values of random implementation to the global optimum. By using this idea, the performance of the PSO algorithm is improved when the search space increases.

3.7.3 Fitness Function

After generating each new individual, we need to evaluate its fitness value. We define the fitness value as the energy consumption of the network. We calculate the total transmit energy, $E_{Tx}(k_{t_{ij}}, d_{ij})$, dissipated in a round by each relay node i, $1 \leq i \leq n$, to transmit $k_{t_{ij}}$ bits data to another node (either a relay node or the base station) j,

$1 \leq i \leq n+1$, using the following first order radio model [58]:

$$E_{Tx}(k_{t_{ij}}, d_{ij}) = \begin{cases} k_{t_{ij}} E_{elec} + k_{t_{ij}} \epsilon_{fs} d^2, & d < d_0; \\ k_{t_{ij}} E_{elec} + k_{t_{ij}} \epsilon_{tg} d^4, & d \geq d_0; \end{cases} \tag{3.17}$$

$$d_0 = \sqrt{\frac{\epsilon_{fs}}{\epsilon_{tg}}} \tag{3.18}$$

where d_{ij} is the euclidian distance between node i and j, E_{elec} is the per bit energy dissipation for running the transceiver circuitry, ϵ_{fs} and ϵ_{tg} are both parameters but for different propagation model, d_0 is the cross-over distance. Similarly, the receive energy, $E_{Rx}(k_{r_i})$, dissipated in a round by each relay node i, $1 \leq i \leq n$, is defined as

$$E_{Rx}(k_{r_{hi}}) = k_{r_{hi}} E_{elec} \tag{3.19}$$

where $k_{r_{hi}}$ is the number of bits received by relay node i from node h in a round. Hence, we compute E_o, the total energy dissipated by each relay node i for one round of data gathering as

$$E_o = \sum_{h \in R_i} E_{Rx}(k_{r_{hi}}) + E_{da} \sum_{h \in R_i} k_{r_{hi}} + \sum_{j \in T_i} E_{Tx}(k_{t_{ij}}, d_{ij}) \tag{3.20}$$

where R_i is the set made up of the nodes that transmit $k_{r_{hi}}$ bits message to node i in a round. T_i is the set made up of the nodes that receive $k_{t_{ij}}$ bit message from node i in the same round. E_{da} is the per bit energy dissipation for aggregating data which are from different sources. Obviously, our metric for energy dissipation takes into consideration including the transmit energy, the receive energy and aggregation energy. Therefore, the fitness function is defined as E_o.

3.7.4 Selection

The function of selection operator is to select individuals which have relative large fitness values with relative large probabilities from their parent generation. The PSO algorithm then puts the chosen individuals in the offspring generation and waits for further evolution implemented by crossover and mutation. There are many different selection schemes such as the roulette wheel selection, the liner ranking selection, and the tournament selection. The principle of these selection schemes is consistent, which is to select individuals "randomly" from an old population to generate a novel population. In fact, the "random" selection is not completely random. It copies an old individual and adds it into new population with a certain probability which is proportional to the ratio of its fitness value to the sum of all individuals' fitness.

Among various selection schemes, the roulette wheel selection [75] is the most frequently used, but this selection may have problems when the fitness values differ very much. In order to circumvent the problems of fitness proportionate selection methods, we adopt the ranking selection [76] [77] [78] as a part of implementation of the modified PSO algorithm in this thesis.

In the ranking selection, the probability of an individual to be selected is assigned according to its rank which is based on the objective function values in the sorted list of all individuals in the population. Using the rank smoothes out larger differences of the fitness values and emphasises small ones. Nevertheless, because the ranking selection is also based on probability to come into effect, on the one hand, the virtue of this method is that individuals with low fitness value are given the opportunity to be selected so as to keep diversity of the population. On the other hand, individuals with high fitness value can be eliminated, making evolution a temporary retrogression. In order to compensate for this deficiency, the optimal individual in the parent generation is handed down to the offspring generation unconditionally according to the PSO algorithm. Consider n individuals in the population, this selection scheme performs as follows:

Step 1. Rank individuals according to the fitness values and determine their positions in the population. For instance, the ranking of the least fit individual is the 1st, and the ranking of the fittest individual is the nth.

Step 2. Calculate the probability p_i to be selected for the ith individual as:

$$p_i = \frac{i}{\sum\limits_{j=1}^{n} j} \tag{3.21}$$

Step 3. For the ith individual, calculate the accumulative probability g_i from the 1th to the ith individual:

$$g_i = \sum_{j=1}^{i} p_j \tag{3.22}$$

Step 4. Generate a random number r which is uniformly distributed in $[0, 1]$. If the condition satisfies $g_{i-1} < r \leq g_i$, the ith individual is selected.

Step 5. Repeat **Step 4** until the amount of generated individuals is equal to the size of population.

3.7.5 Crossover

The offspring generation obtained by crossover has to represent routing trees from source nodes to the base station, otherwise they are illegal solutions. Consequently, we propose the following crossover method for the above defined individuals.

In the velocity formula of the PSO algorithm, the operator $\otimes$ in (3.15) indicates information exchange which is similar to the operation of crossing in the GA. In the GA, the crossover replaces the corresponding position of ordinary genes with a segment of the optimal genes. So we use the crossover as the equivalent form of the $\otimes$ operation in (3.13).

The term $p^k_{gbest} \otimes x^k_{id}$ indicates that a particle tends to the global optimum, so it is equivalent to the crossover operator of dealing with the generic individuals and the *gbest*. Because of the requirement of encoding, source nodes are known at the stage of crossover. First, we randomly select a gene in the locus of the same source node between the global optimal individual and a generic individual and put it into the same locus of the offspring. We then make the same selection at the locus of the relay node which is indicated by the previous chosen gene. This process is repeated until the base station is found. For the rest nodes which are not used, genes from the same locus between the global optimal individual and a generic individual are selected randomly. Fig. 3.6 is an example for crossover between the global optimal aggregation tree and a generic aggregation tree and corresponding interpretation in aggregation tree structure.

The term $p_{ibest}^k \otimes x_{id}^k$ indicates that a particle tends to the local optimum, so it is equivalent to the crossover operator between the generic aggregation trees and the *ibest*. Fig. 3.7 is an example for crossover between the local optimal aggregation tree and a generic aggregation tree and corresponding interpretation in network structure.

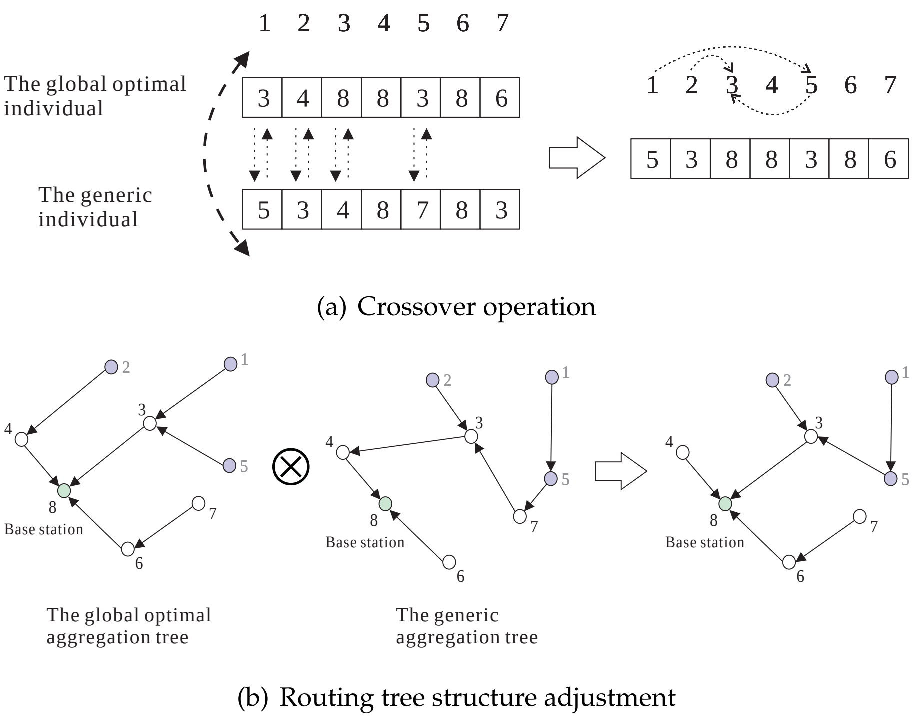


Figure 3.6. Crossover between the global optimal individual and the generic individual and corresponding interpretation. For (a), the number serial at the upper string indicates the previous hop nodes while the number serial at the lower string represents the next hop nodes.

3.7.6 Mutation

In the velocity formula of the PSO algorithm, multiplying ω by v_{id}^k indicates that a particle searches toward a new space. This moving tendency is similar to the operation of mutation in the GA. In the GA, the mutation operator makes ordinary genes have large changeable scales and search toward wider space, so we use the mutation as the equivalent form of the multiplying items ωv_{id}^k. The mutation operator also has many forms for corresponding encoding schemes. The design of the mutation is to prevent

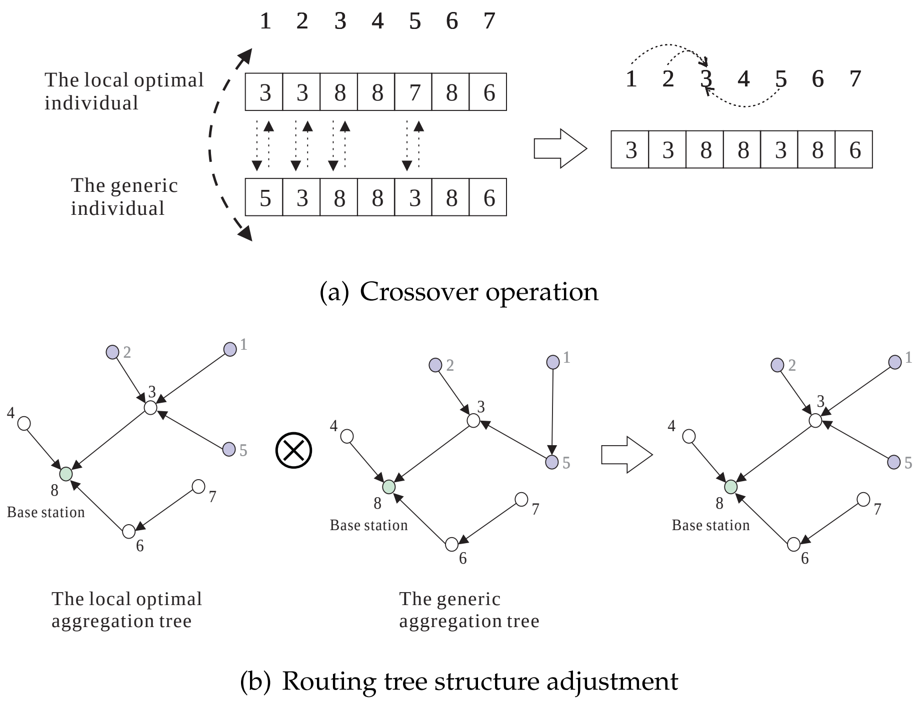


Figure 3.7. Crossover between the local optimal individual and the generic individual and corresponding interpretation. For (a), the number serial at the upper string indicates the previous hop nodes while the number serial at the lower string represents the next hop nodes.

algorithm from falling into the local optimal solution. It can make the population tend to diversify and guarantee that there are better individuals in the offspring generation.

Because of the requirement of encoding, source nodes are known at the stage of mutation. If a certain individual is selected to conduct mutation, the routing tree represented by the individual will be altered. We select a node i to substitute for node j from set Ω where the node i has the same previous hop node and next hop node as node j. Hence, we can use node i to replace node j, thereby generating a new routing tree. This process is illustrated by Fig. 3.8.

3.7.7 Addition

The addition operation indicates the sum of several processing steps. Through orderly executing the equivalent subitems in the PSO algorithm, we can obtain the effect of addition. In the equivalence of the addition, the outputs of previous steps are the

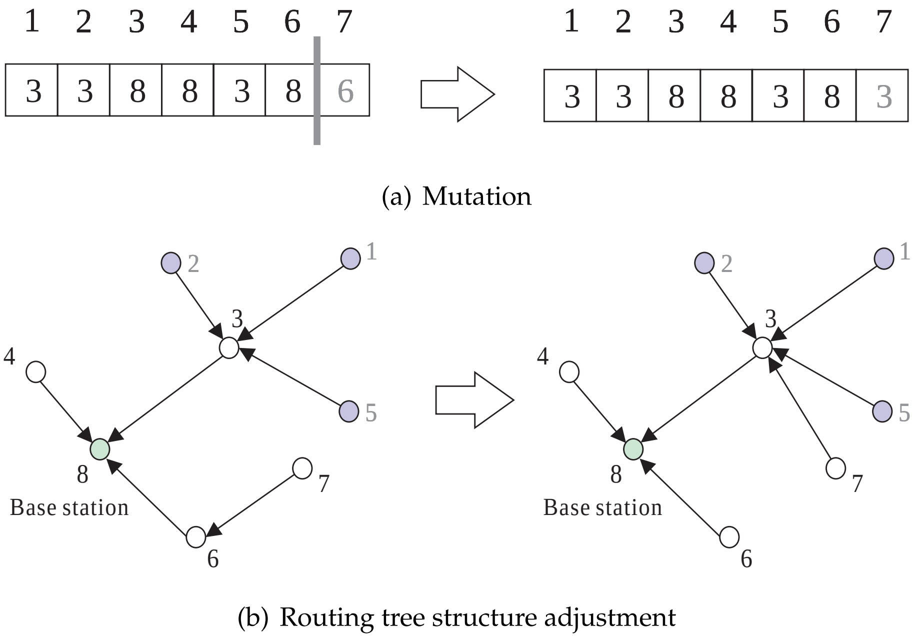


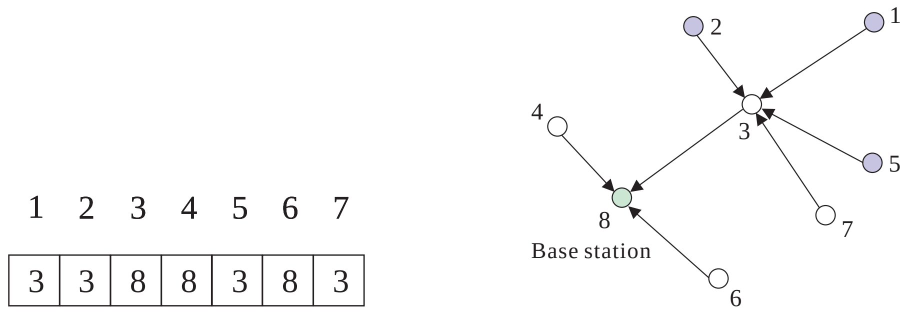
Figure 3.8. Examples of the mutation operation. For (a), the number serial at the upper string indicates the previous hop nodes while the number serial at the lower string represents the next hop nodes. The gray number after the bar is selected to execute the mutation.

inputs of latter steps. According to (3.15) and (3.16), we can deduce that

$$\begin{aligned} x_{id}^{k+1} &= x_{id}^{k} + v_{id}^{k+1} \\ &= x_{id}^{k} \oplus \omega v_{id}^{k} \oplus (p_{ibest}^{k} \otimes x_{id}^{k}) \oplus (p_{gbest}^{k} \otimes x_{id}^{k}) \end{aligned} \tag{3.23}$$

The computation sequence is from right to left. We still take the network shown in Fig. 3.5(a) as an example. Applying two crossover operations to a generic aggregation tree shown in Fig. 3.6 and Fig. 3.7 and one mutation operation shown in Fig. 3.8, we can acquire the final result as Fig. 3.9.

By means of the equivalence of the crossover, mutation and addition, the PSO algorithm can be used to solve the routing tree with data aggregation problem. In addition, in order to preserve information of the optimal solution, the optimal individual in a particular iteration is not updated. After this iteration, if the effect of another individual is superior to the optimal individual, the individual which has better effect will become the optimal individual. The previous optimal particle is regarded as an generic particle and will be updated in the next iteration.



(a) Individual (b) Routing tree

Figure 3.9. Decoding of result output by PSO algorithm. For (a), the number serial at the upper string indicates the previous hop nodes while the number serial at the lower string represents the next hop nodes.

3.8 Routing Scheme in Detail

The implementation of the reliable routing scheme, which is broken up into rounds, can be divided into three broad phases: setup phase, routing tree optimisation phase, and data gathering phase. In the following subsections, we will discuss each of these in detail.

3.8.1 Setup Phase

After the deployment of the base station and all nodes, the fast setup phase will operate immediately. This phase is made up of three steps:

Step 1. The base station discovery: The base station broadcasts a "hello" message to all the nodes at a certain power level. In this way, each node can compute the approximate distance to the base station and select the appropriate power level to communicate with the base station when acting as the root of the tree.

Step 2. Neighbour discovery: Each node broadcasts its information of node identity (ID) and current transmitting power to the nodes in its coverage area. When node i receives this message from node j, it will add node j to its neighbour table and record node j's ID, energy and distance, which are computed using (3.2).

Step 3. Initial trees construction: First, the base station broadcasts a query to all the nodes in the network. After receiving this packet, k of the n sensors will compute the distance to the base station and respond to that query. We assign identical pheromone τ_0 = 0.5 to each node in network. When the source node wants to send data, it will comply with the ACO algorithm to select the next hop node using transition probability formula (3.10).

Because optimisation on a routing tree is mainly implemented by the PSO algorithm, we carry out the local pheromone updating to decrease computation and complexity of the ACO algorithm. The function of the local pheromone updating is to enable the later control packets, which are influenced by pheromone of the visited nodes, to have a strong ability to explore the nodes which have not been visited. As the control packet moves between nodes i and j, it updates the amount of pheromone on the link (i, j) according to following equation:

$$\tau_{ij}(t+n) = max\{\tau_{min}, min\{\tau_{max}, (1-\rho)\tau_{ij}(t) + \rho\Delta\tau_{ij}(t)\}\} \tag{3.24}$$

$$\Delta\tau_{ij}(t) = \sum_{k=1}^{m} \Delta\tau_{ij}^{k}(t) \tag{3.25}$$

$$\Delta\tau_{ij}^{k}(t) = \frac{1 + d_i - d_j}{d_j} \tag{3.26}$$

where τ_{min} and τ_{max} are minimum and maximum of the amount of pheromone the pheromone respectively, ρ $(0 < \rho < 1)$ is pheromone evaporation rate, d_i is the distance between nodes i and base station, d_j is the distance between nodes j and base station, and Q is a constant. We set τ_{min} = 0.3, τ_{max} = 0.7, ρ = 0.2, α = 1, and β = 2.

After a certain number of control packets arrive at the base station, the process of initial trees construction ends. Each track of control packet constitutes a path, and the path from a certain source node to the base station is not unique. Therefore, the ACO algorithm provides the PSO algorithm with various routing trees which are the candidate solutions of the PSO algorithm.

3.8.2 Routing Tree Optimisation Phase

This phase is an interim phase before data gathering and aggregation is operated. In this phase, the routing tree will be optimised to tradeoff between data aggregation and transmission structure. As mentioned in Section 3.1, optimal aggregation tree problem is NP-complete, and hence PSO algorithm can be applied to solve it with a near optimal solution.

The PSO algorithm is an iteration-based optimisation tool, and it has wide applicability to all sorts of continuous combinatorial optimisation problems. The kernel of the PSO algorithm is to make use of three pieces of information, including generic particles, local extremum of particles and global extremum of population, to guide the next iteration of particles' velocity and position. For the matter of optimal aggregation tree problem, the different routes corresponding to current positions are independent, and this situation does not match continuous variables optimisation in the PSO algorithm. Besides that, addition and subtraction operations between paths and the rate of path change are difficult to express directly by using the velocity and displacement formulas of the PSO algorithm. Hence, we need to make further improvements in the PSO algorithm so that it can be used to optimise paths. The specific process of optimisation is explained in Section 3.7.

3.8.3 Data Gathering Phase

The data gathering phase occupies the main network lifetime, which is longer than the topology adjustment phase. In this phase, data flows to the aggregator node which is elected by the PSO algorithm in the end of the routing tree optimisation phase. The aggregator node then aggregates the gathering data to a new fixed length packet and transmits it to the base station via multi-hop.

3.9 Performance Analysis

In this section, we perform a comparison between the PSO algorithm for aggregation tree and other algorithms such as the SPT algorithm, the MST algorithm, and the SLT algorithm. As mentioned before, the SLT algorithm is a routing algorithm proposed in [53], aimed at simultaneously approaching to both the MST algorithm and the SPT algorithm for given nodes. The SLT algorithm is employed in [49] as an approximation solution to solve the aggregation tree problem. From the comparison, we will conclude that PSO surpasses other three algorithms with two typical correlation coefficients.

3.9.1 Scenario

Consider a sensor network where nodes are scattered on an $N \times N$ square grid, where only N nodes in the left column are sources. The base station is located at the rightmost bottom corner. We assume that each source generates unit data I_0 which is responded to the query of the base station. Data packets will be aggregated at a relay node on their paths to the base station when the paths overlap at the node. The aggregation cost at the base station can be negligible from the total routing cost since a base station usually is not subject to the restrictions of energy.

Nodes in the grid can only communicate with their neighbours. The energy dissipation for delivering one bit of data between adjacent nodes is assumed to be c_0. Let q_0 be the cost for aggregating per bit of multi data packets. In addition, variable ρ indicates different correlation coefficients for I_0. Under this setup, we compare four routing schemes, namely the SPT algorithm, the MST algorithm, the SLT algorithm, and the PSO algorithm for aggregation tree. According to (3.1), data traffic after aggregation is mainly determined by ρ. Therefore, ρ is a key parameter which can significantly affect routing decisions when involving data aggregation. For instance, a low ρ may enable a node to employ shortest path transmission strategy, especially when the data amount cannot be significantly reduced. We consider two typical scenarios to demonstrate their performance differences with the change of ρ:

- In the first scenario, ρ approaches 1. In other words, the data among different sensors are highly correlated.
- In the second scenario, ρ approaches 0. That is to say, the redundancy among the information gathered by different sensors is fairly low.

3.9.2 Near Perfect Aggregation

When ρ approaches 1, the routing tree established by four algorithms are depicted in Fig. 3.10. Without loss of generality, at each relay node, two correlated data packets I_0 are aggregated to output another $I_0 + I_0(1-\rho)$ packet. In the SPT algorithm, the distance from each source node to the base station is $N-1$ hops. In the MST algorithm, the farthest source is $2(N-1)$ hops from the base station. Since $2(N-1) < (1+\sqrt{2})(N-1)$, according to [53], the SLT algorithm will degrade into the MST algorithm for this scenario. In the following, we will examine the cost of the MST (SLT) algorithm, the SPT algorithm, and the PSO algorithm for this network.

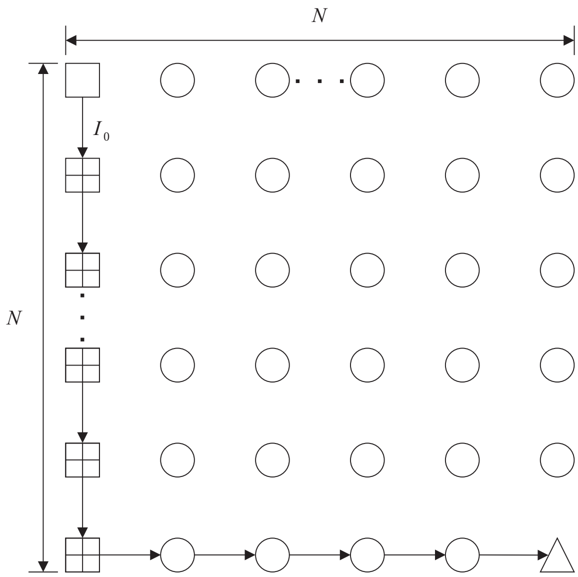
The cost for the MST algorithm, C_{MST}, can be derived as

$$\begin{aligned} C_{MST} &= \sum_{i=1}^{N-1}[I_0 + (i-1)(1-\rho)I_0]c_0 + \sum_{i=1}^{N-1}[I_0 + (N-1)(1-\rho)I_0]c_0 \\ &\quad + \sum_{i=1}^{N-1}[I_0 + I_0 + (i-1)(1-\rho)I_0]q_0 \\ &= 2(N-1)I_0c_0 + \frac{(N-1)(3N-4)}{2}(1-\rho)I_0c_0 \\ &\quad + [2(N-1) + \frac{(N-1)(N-2)}{2}(1-\rho)]I_0q_0 \end{aligned} \tag{3.27}$$

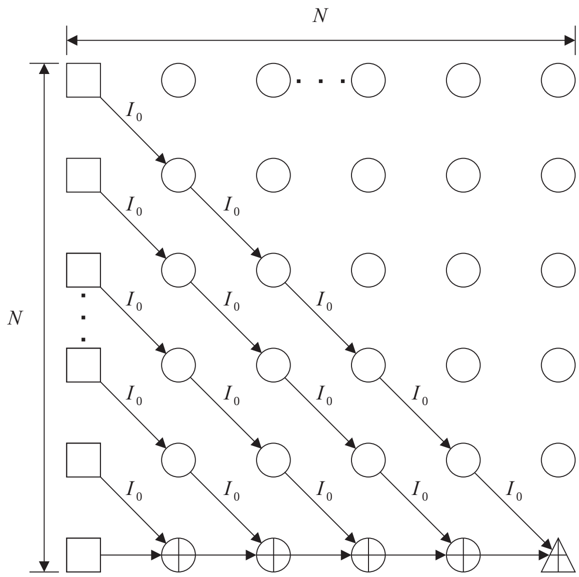
The cost for the SPT algorithm is

$$\begin{aligned} C_{SPT} &= \sum_{i=1}^{N-1} iI_0c_0 + \sum_{i=1}^{N-1}[I_0 + (i-1)(1-\rho)I_0]c_0 \\ &\quad + \sum_{i=1}^{N-2}[I_0 + I_0 + (i-1)(1-\rho)I_0]q_0 \\ &= \frac{N(N-1)}{2}I_0c_0 + (N-1)I_0c_0 + \frac{(N-1)(N-2)}{2}(1-\rho)I_0c_0 \\ &\quad + [2(N-1) + \frac{(N-2)(N-3)}{2}(1-\rho)]I_0q_0 \end{aligned} \tag{3.28}$$

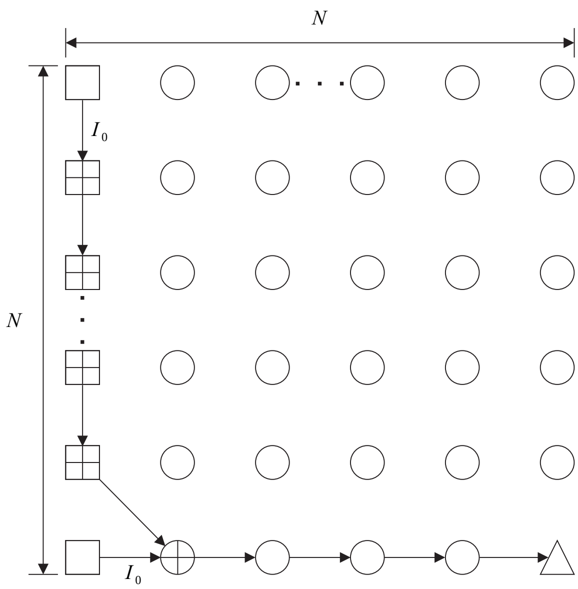
In this case, it is easy to verify that the performance of the MST algorithm is the better than the SPT algorithm. As mentioned in Section 3.7.2, the SPT algorithm and the MST algorithm have good results in initial population, respectively when ρ is equal to 0 and 1. Hence, we select the SPT structure and the MST structure as individuals in the PSO algorithm to execute variation. An apparent offspring we can get is that all sources



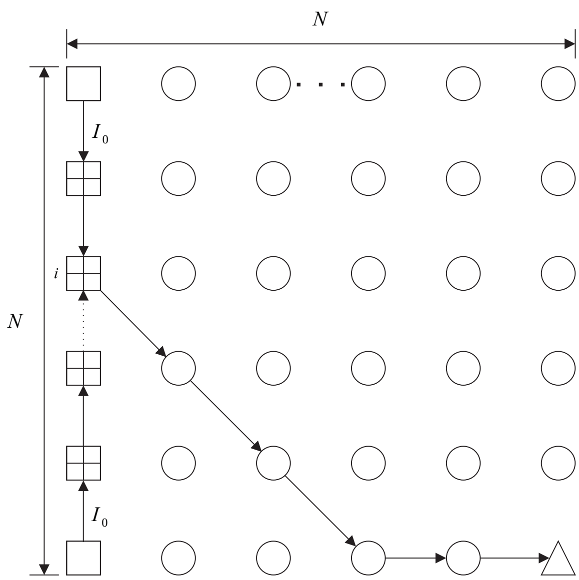
(a) MST (SLT)



(b) SPT



(c) PSO ($N = 3, 4$)



(d) PSO ($N \geq 5$)

Figure 3.10. Data aggregation trees for the MST algorithm, the SPT algorithm, the SLT algorithm, and the PSO algorithm when ρ approaches 1.

except node 1 connect to one node which in turn will connect to the base station via the shortest path as shown in Fig. 3.10(c). The cost for this aggregation tree is

$$\begin{aligned} C_{PSO_1} &= \sum_{i=1}^{N-1} [I_0 + (i-1)(1-\rho)I_0]c_0 + I_0 c_0 + \sum_{i=1}^{N-2} [I_0 + (N-1)(1-\rho)I_0]c_0 \\ &+ \sum_{i=1}^{N-1} [I_0 + I_0 + (i-1)(1-\rho)I_0]q_0 \\ &= 2(N-1)I_0 c_0 + \frac{3(N-1)(N-2)}{2}(1-\rho)I_0 c_0 \\ &+ [2(N-1) + \frac{(N-1)(N-2)}{2}(1-\rho)]I_0 q_0 \end{aligned} \tag{3.29}$$

The cost is the lessen than the MST algorithm when $N = 3, 4$. When $N \geq 5$, we use the MST, the SPT and the routing tree shown in Fig. 3.10(c) as individuals to continue the optimisation by the PSO algorithm. After several iterations, source nodes will be connected to a centre in the left column, and the centre node is connected to the base station via the shortest path as shown in Fig. 3.10(d). Since the amount of source nodes is a randomisation number, the location of the centre node will vary accordingly, which will induce different paths and consequently have different total costs. When $N = 2k, (k = 1, 2, ..., m)$,

$$\begin{aligned} C_{PSO_2} &= \sum_{i=1}^{\frac{N}{2}-1} [I_0 + (i-1)(1-\rho)I_0]c_0 + \sum_{i=1}^{\frac{N}{2}} [I_0 + (i-1)(1-\rho)I_0]c_0 \\ &+ \sum_{i=1}^{N-1} [I_0 + (\frac{N}{2}-1)(1-\rho)I_0 + (1-\rho)I_0 + (\frac{N}{2}-2)(1-\rho)^2 I_0]c_0 \\ &+ 2\sum_{i=1}^{\frac{N}{2}-1} [I_0 + I_0 + (i-1)(1-\rho)I_0]q_0 \\ &+ [I_0 + (\frac{N}{2}-2)(1-\rho)I_0 + I_0 + (\frac{N}{2}-1)(1-\rho)I_0]q_0 \end{aligned} \tag{3.30}$$

The first and second terms of (3.30) represent the transmission costs from the left top and left bottom sources to the centre node, the third term summarises the transmission costs from centre nodes to the base station, and the fourth and fifth terms captures the aggregation costs on the left line. The above equation can be simplified as

$$\begin{aligned} C_{PSO_2} &= [2(N-1) + \frac{(N-2)^2 + 2N(N-1)}{4}(1-\rho) + \frac{(N-1)(N-4)}{2}(1-\rho)^2]I_0 c_0 \\ &+ [2(N-1) + (\frac{N}{2}-1)(\frac{N}{2}-2)(1-\rho) + (N-3)(1-\rho)]I_0 q_0 \end{aligned} \tag{3.31}$$

When $N \geq 5$, the difference of cost in (3.29) and (3.31) is demonstrated in the equation below:

$$\begin{aligned}
C_{PSO_1} - C_{PSO_2} &= [\frac{3N^2 - 12N + 8}{4}(1-\rho) - \frac{(N-1)(N-4)}{2}(1-\rho)^2]I_0c_0 \\
&\quad + (\frac{N^2}{4} - N + 2)(1-\rho)I_0q_0 \\
&> [\frac{3N^2 - 12N + 8}{4} - \frac{(N-1)(N-4)}{2}](1-\rho)I_0c_0 \\
&\quad + (\frac{N^2}{4} - N + 2)(1-\rho)I_0q_0 \\
&> \frac{N(N-2)}{4}(1-\rho)I_0c_0 + (\frac{(N-2)^2}{4} + 1)(1-\rho)I_0q_0 \\
&> 0
\end{aligned} \tag{3.32}$$

Since $C_{PSO_1} < C_{MST}$, we have $C_{PSO_2} < C_{MST}$. Evidently, provided that ρ approaches 1, the PSO algorithm always outperforms the MST algorithm. Simultaneously, it is always better than the SPT algorithm. When $N = 2k + 1, (k = 1, 2, ..., m)$, the same conclusions can still be drawn by following a similar analysis.

3.9.3 Near Non-aggregation

In order to make aggregation meaningful, the aggregation cost should be smaller than the transmission cost. Otherwise, relay nodes will prefer forwarding data directly instead of doing aggregation for energy saving. Without loss of generality, at each relay node, aggregation between any two data packets will occur on condition that

$$\begin{aligned}
[I_0 + (1-\rho)I_0]c_0 + 2I_0q_0 &< 2I_0c_0 \\
q_0 &< \frac{\rho c_0}{2}
\end{aligned} \tag{3.33}$$

Considering q_0 is a constant which is larger than 0, (3.33) cannot be satisfied when ρ approaches 0. In other words, it is not necessary to perform data aggregation during routing. Hence, the PSO algorithm takes the SPT algorithm as the optimal solution. Evidently, the PSO algorithm always outperforms the MST algorithm as well.

The above analysis concludes that the PSO algorithm can actually approximate to the optimal solution and outperform the MST (SLT) algorithm and the SPT algorithm in different scenarios while the SPT algorithm and the MST algorithm can only perform well in certain extreme cases. Indeed, the data correlation coefficient is usually between 0 and 1. In the next chapter, we will give extensive simulation results to illustrate the outperforming of the PSO algorithm for an aggregation tree under more general system setups.

3.9.4 Time complexity analysis

In this subsection, we will analyse the time complexity of PSO algorithm. Assume that there are n nodes in wireless sensor networks, M (M is a constant) initial particles and t evolution iterations. In the first step, it will cost $O(1)$ times for selection. During the second step, it first takes $O((M-2)n)$ times to perform crossover between generic particles and the global optimum. Then, the PSO algorithm performs crossover between generic particles and the local optimum. Thus, its time complexity is $O((M-2)n)$ as well. Next, the algorithm will perform mutation among M particles. The time complexity is $O(1)$. After that, the decoding time in this method is $O(n \log n)$. Therefore, the total complexity of PSO algorithm is $t(O(1) + O((M-2)n) + O((M-2)n) + O(1) + O(n \log n)) = O(tn \log n)$.

3.9.5 Message complexity analysis

Next, we analyse the message complexity of the ACO algorithm. Assume that there are n nodes in wireless sensor networks, K source nodes. In the first phase, each node will broadcast its local information once, such as node identity (ID) and current transmitting power, so the message complexity is $O(n)$. In the second phase, each source node sends a control data to base station via a path. The transition probability determined by pheromone levels influences the next hop selection, which helps to increase the diversity of routing tree. It takes $O(Kn)$ messages at most for a routing tree construction. In the third phase, since the PSO algorithm requests M initial routing trees. Therefore, the total message complexity of the ACO algorithm is $O(n) + O(MKn) = O(n)$.

3.10 Conclusion

In this chapter, we have formulated the research problem and described the optimisation objective. Concerning optimisation on data gathering with aggregation, three heuristic algorithms were introduced to provide possible solutions to the problem. P-SO algorithm modified by the GA was discussed to address the discrete nature of our optimisation problem. Following this discussion, a novel data gathering algorithm with aggregation for minimising energy consumption was proposed. The mathematical analysis shows that the aggregation tree using the PSO algorithm outperforms other three existing routing algorithms.

Chapter 4

Performance Evaluation by Simulation

THIS chapter provides a set of simulation results to evaluate the performance of our proposed routing scheme under different environments and conditions. The performance of the particle swarm optimisation (PSO) algorithm for an aggregation tree is compared with three other existing algorithms in terms of energy consumption.

4.1 Introduction

In this chapter, we present an extensive set of simulations to evaluate the performance of our proposed routing scheme. For sensor nodes randomly deployed in a 2D field, we compare the performance of the PSO algorithm with other routing algorithms based on the shortest path tree (SPT) algorithm, the minimum spanning tree (MST) algorithm, and the shallow light tree (SLT) algorithm. The impact of network connectivity and the correlation coefficient on different algorithms is studied. Concurring with our design goal and analysis of the the PSO algorithm, our key finding of the experiments is that the PSO algorithm can adapt itself to a wide range of network connectivity and data correlation. While other algorithms may achieve better performance in some extreme cases, they suffer from varying conditions and hence perform poorly in general scenarios.

4.2 Simulation Set-up

A square region of size 50m$\times$50m is uniformly divided into 100 grids, and each sensor is deployed in one grid. Since the coordinate of each sensor is determined by a random function which can generate uniformly distributed random numbers within the interval [0,50], the sensor nodes are randomly distributed. We assume that each source node produces one unit of data and sends it to the base station located at the bottom-right corner as shown in Fig. 4.1. A few sensors act as sources and all sensors can be acting as routers.

The data packet size is 4000 bits. This means that each node transmits a 4000 bits data packet to the base station once. The calculation of energy consumption for data transmission is based on the energy model presented in Section 3.2.3. The correlation model introduced in Section 3.2.2 is employed here to estimate the packet size after data aggregation. The parameters utilised in the simulations are summarised in Table 4.1. The simulation results presented in this chapter are the averages of 30 simulation runs.

4.3 Performance Evaluation

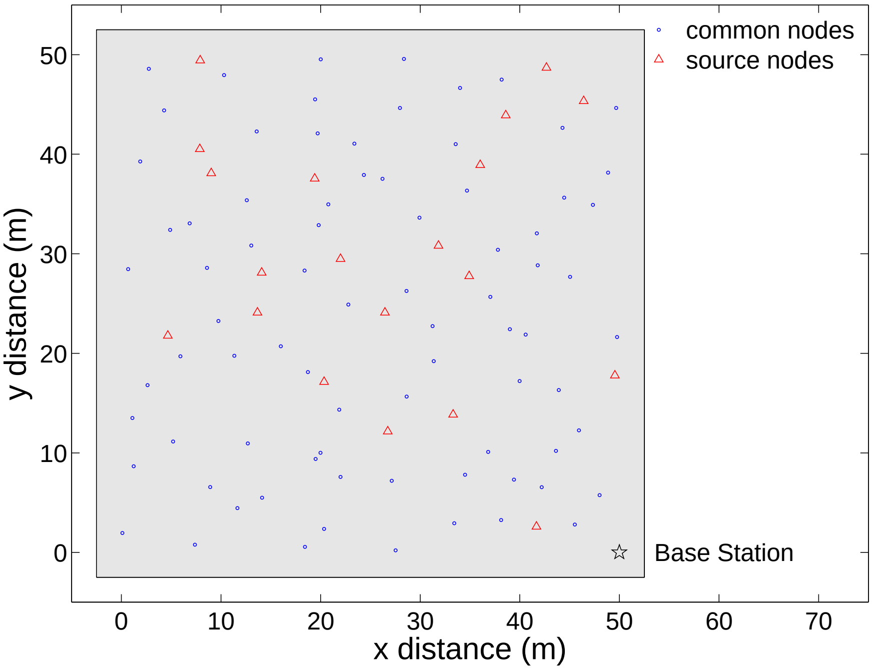


Figure 4.1. Network structure for simulation.

Table 4.1. Parameters of simulation.

Parameter	Value
Number of nodes N	100
Network size	50m$\times$50m
Base station location	(50m, 50m)
Initial energy of sensor nodes	1J
Data packet size	4000bits
Radio electronics energy E_{elec}	50nJ/bit
Cross-over distance d_0	87m
Amplifier parameter of free space model ϵ_{fs}	10pJ/bit/m^2
Amplifier parameter of two-ray ground model ϵ_{tg}	0.0013pJ/bit/m^4
Data aggregation energy E_{da}	15nJ/bit/signal

4.3.1 Compared objectives

For the simulations described in this section, we implemented the SPT algorithm, the MST algorithm, the SLT algorithm, and the PSO algorithm to compare the performance of our routing scheme. We briefly summarise these four algorithms here.

We define a source nodes set as V', T is the set of nodes existing in the routing tree and the weight from the nodes of T to root node s has been identified, and L is the set of relay nodes.

(a) The SPT algorithm

The SPT algorithm is an interactive process that works through a graph or a set of vertices and paths to calculate any source node to the base station in the set. It goes through these steps:

Step 1. Select the base station to be s: $T = \{s\}$.

Step 2. If $x \in L$ links to s, update the weight $W_{s,x}$ between s and x, otherwise $W_{s,x} = \infty$.

Step 3. Find out $t_i \in V'$, subject to $W_{s,t_i} = min\{W_{s,x}|x \in L\}$

Step 4. Record the relay nodes from s to t_i, and update T and V': $T = T \cup \{t_1\} \cup \{t_2\} \cup ... \cup \{t_i\}$, $V' = V' - \{t_i\}$. If $V' = \emptyset$, the algorithm ends, otherwise return to **Step 2**.

These steps are shown in Fig. 4.2 as a flow chart.

(b) The MST algorithm

The MST algorithm enables each source node to link to the nearest source node as its parent node, and incrementally add nodes to construct a routing tree. It can be implemented according to following steps:

Step 1. Select the base station to be s: $T = \{s\}$.

Step 2. If $x \in L$ links to s, update the weight $W_{s,x}$ between s and x, otherwise $W_{s,x} = \infty$.

Step 3. Find out $t \in V'$, which subjects to $W_{p,t} = min\{W_{s,x}|s \in T, x \in L\}$.

Step 4. Record the relay nodes from s to t_i, and update T, V' and s: $T = T \cup \{t_1\} \cup \{t_2\} \cup ... \cup \{t_i\}$, $V' = V' - \{t_i\}$, and $s = t_i$. If $V' = \emptyset$, the algorithm ends, otherwise return to **Step 2**.

These steps are shown in Fig. 4.3 as a flow chart.

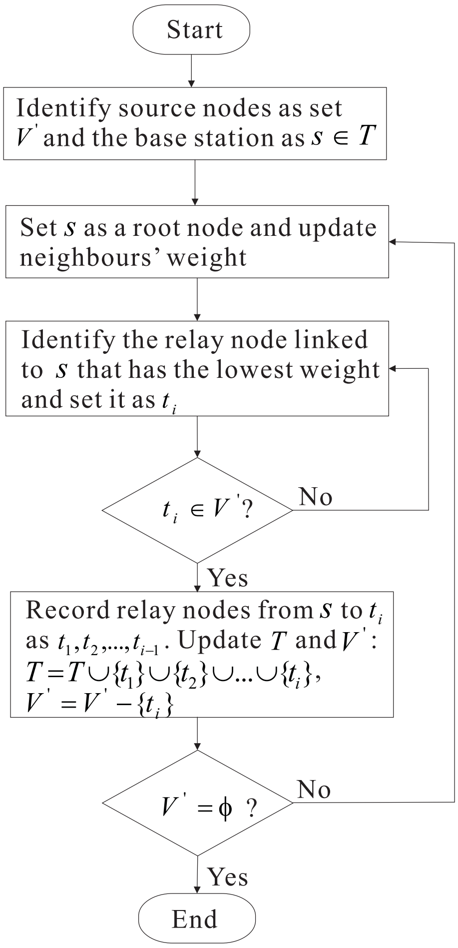


Figure 4.2. The flow chart of the SPT algorithm.

(c) The SLT algorithm

The SLT algorithm is designed to find a spanning tree that simultaneously approximates a shortest-path tree and a minimum spanning tree. It works as the following procedure:

Step 1. Select the minimum spanning tree to be T, the base station to be s.

Step 2. Find out the farthest node $t \in V'$. If the weight of t to the base station is larger than $1 + \sqrt{2}$ times of that in the shortest path tree, the node will use the shortest path to the base station to substitute the original path, and update T and the weight of node x which connects to the node t in the current tree as $W_{s,x} = min\{W_{s,x}, W_{s,t} + W_{t,x}\}$.

Step 3. Update V': $V' = V' - \{t_i\}$. If $V' = \emptyset$, the algorithm ends, otherwise return to **Step 2**.

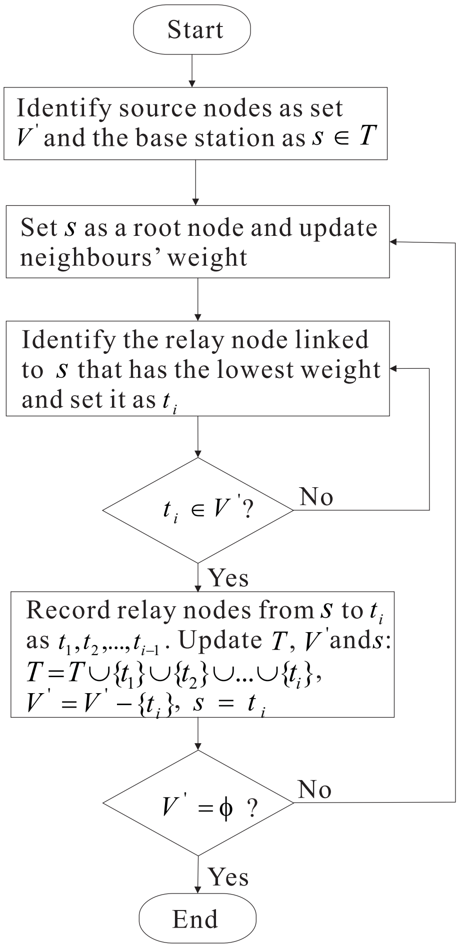


Figure 4.3. The flow chart of the MST algorithm.

These steps are shown in Fig. 4.4 as a flow chart.

(d) The PSO algorithm

The PSO algorithm utilises a heuristic method to find a near optimal routing tree. It performs as follows:

Step 1. Build initial population (routing tree) $T_1, T_2, ..., T_n$.

Step 2. Set a counter $t = 0$.

Step 3. Define a fitness function to evaluate the fitness of each particle. The fitness function is defined in Section 3.7.3.

Step 4. Select fittest individuals for crossover and mutation with a probability. The selection scheme is introduced in Section 3.7.4.

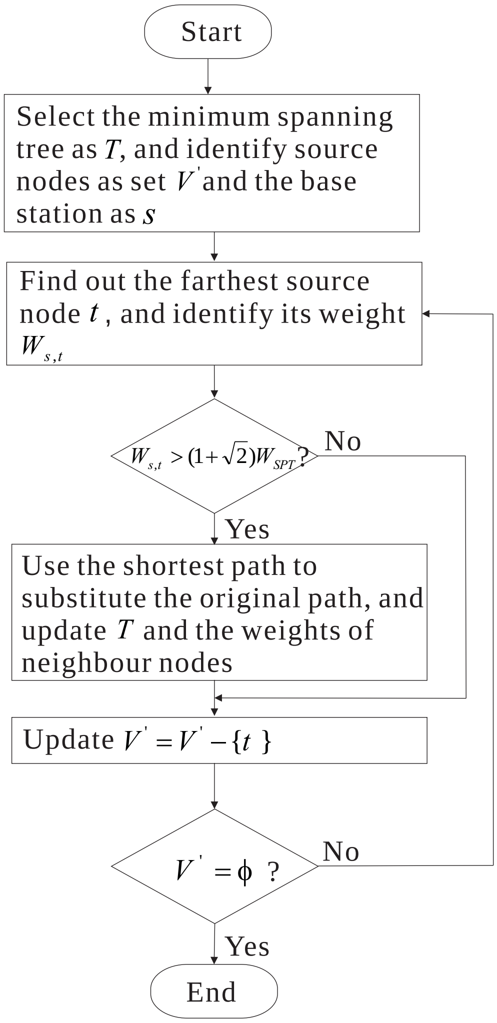


Figure 4.4. The flow chart of the SLT algorithm.

Step 5. Perform two crossover operations to a generic particle T_i. One is executed between T_i and global best particle T_{gbest}. The other is executed between T_i and local best particle T_{lbest}. The crossover scheme is specified in Section 3.7.5.

Step 6. Perform mutation to T_i. The mutation scheme is discussed in Section 3.7.6.

Step 7. If $t > N$, the algorithm ends, otherwise update t: $t = t + 1$ and return to **Step 2.**

These steps are shown in Fig. 4.5 as a flow chart.

4.3.2 Simulation Results

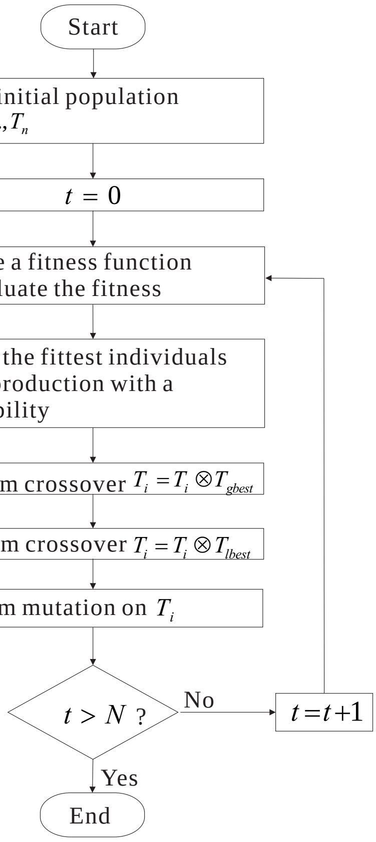


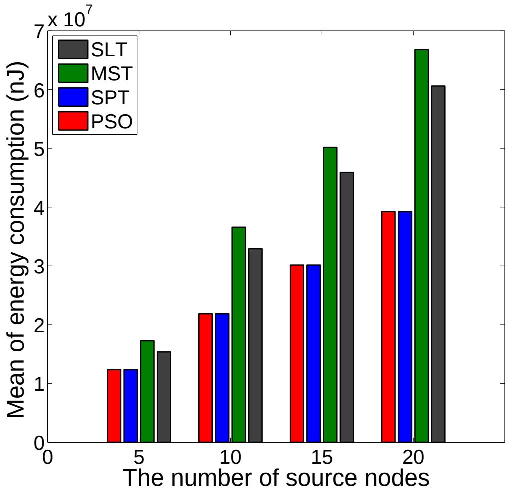
Figure 4.5. The flow chart of the PSO algorithm.

(a) Impact of Network Connectivity

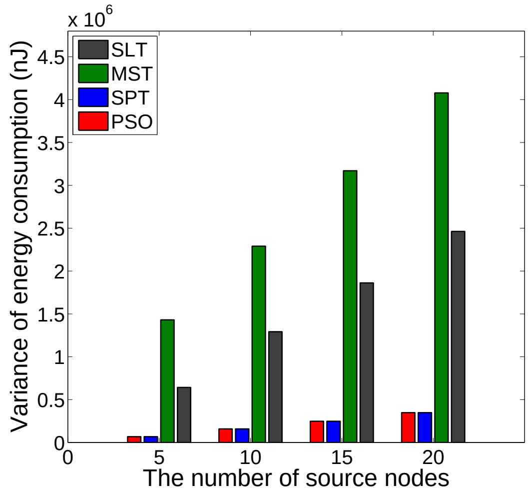
Since R denotes the transmission range of a node and k represents the number of source nodes, by varying R and k, we can control the connectivity of the network. Naturally, different connectivity will affect the behaviour of different routing algorithms.

The first set of simulations is carried out to investigate the total energy consumption with ρ approaches 0 and R=10m, 15m, and 20m. The simulation results are depicted in Fig. 4.6, Fig. 4.7, and Fig. 4.8.

Fig. 4.6(a), Fig. 4.7(a), and Fig. 4.8(a) show the mean of energy consumption per round versus the number of source nodes using an aggregation tree with the PSO algorithm, the MST algorithm, the SPT algorithm, and the SLT algorithm. Clearly, when ρ approaches 0 the PSO algorithm performs as well as the SPT algorithm and reduces energy consumption significantly compared with the MST algorithm and the SLT algorithm. A reduction of average energy consumption of about 40% and 50% can be

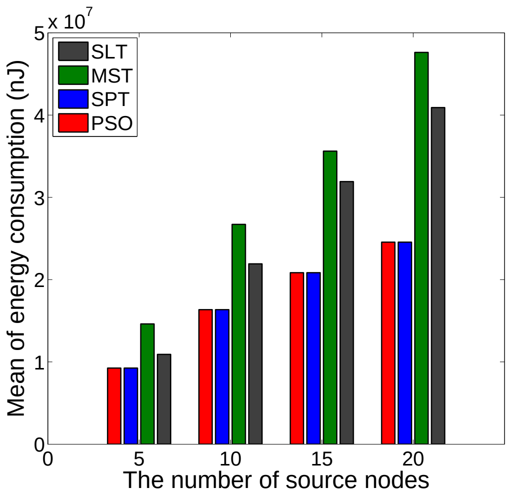


(a) Mean of energy consumption versus the number of source nodes

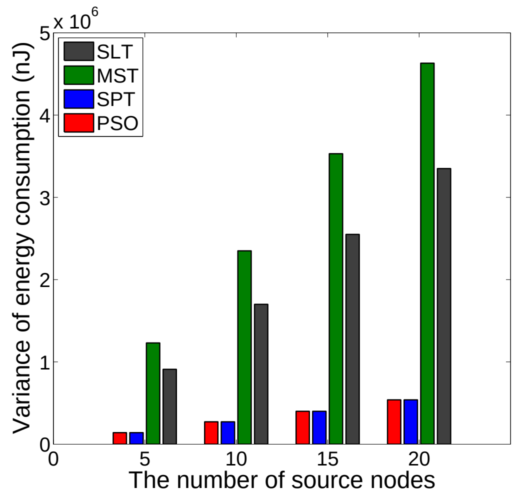


(b) Variance of energy consumption versus the number of source nodes

Figure 4.6. Energy consumption versus the number of source nodes when R=10m and ρ approaches 0.



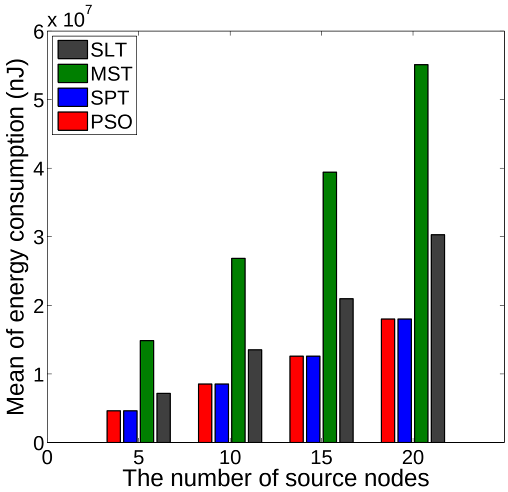
(a) Mean of energy consumption versus the number of source nodes



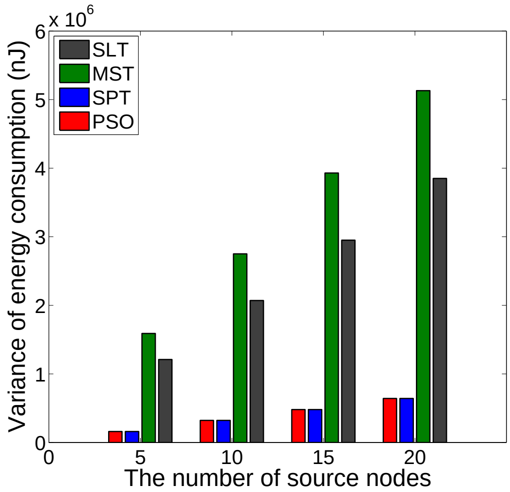
(b) Variance of energy consumption versus the number of source nodes

Figure 4.7. Energy consumption versus the number of source nodes when R=15m and ρ approaches 0.

obtained by the PSO algorithm as well as the SPT algorithm over the SLT algorithm and the MST algorithm, respectively.



(a) Mean of energy consumption versus the number of source nodes



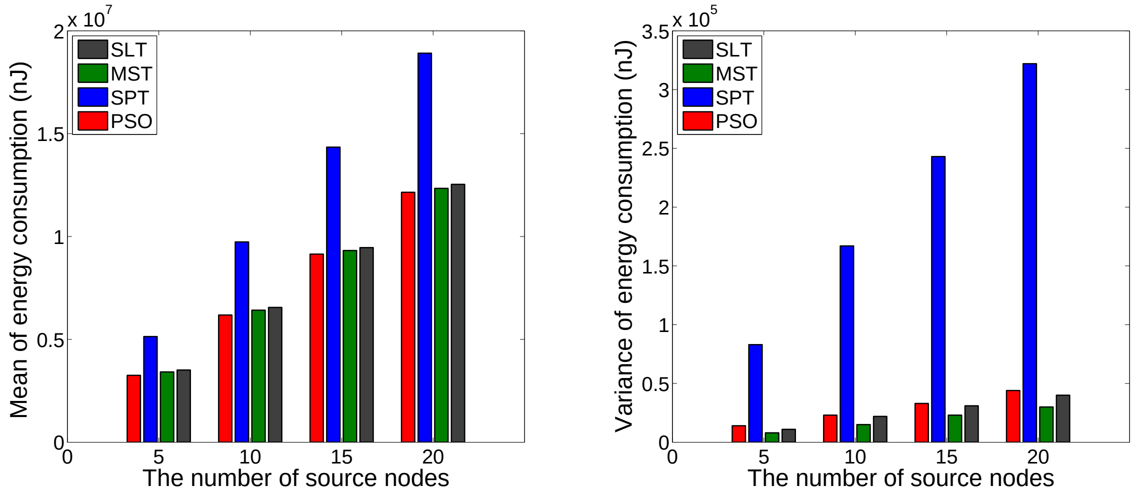
(b) Variance of energy consumption versus the number of source nodes

Figure 4.8. Energy consumption versus the number of source nodes when R=20m and ρ approaches 0.

This can be explained as follows: In a network with poor correlation, the PSO algorithm reduces to the SPT algorithm, and data should be transmitted directly to the base station via the shortest paths instead of aggregator nodes by detouring, since there is no redundancy among the information gathered by different sensors and aggregation at any relay node is not efficient in reducing the data amount. As the PSO algorithm explicitly considers correlation condition, this phenomenon can be captured and exploited. On the contrary, the SLT results in a fixed routing structure according to network topology and a fixed approximation ratio to the MST algorithm and the SPT algorithm and hence cannot dynamically adapt to the change of data correlation. Therefore, when ρ approaches 0, the SLT algorithm cannot recognise the advantage of transmitting over direct links and results in poor performance.

Fig. 4.6(b), Fig. 4.7(b), and Fig. 4.8(b) show the variance of energy consumption with the number of source nodes changing in the four algorithms. From these graphs, when ρ approaches 0 it can be easily observed that the average energy consumption of the MST algorithm varies much more than the three other algorithms. Since the MST algorithm has the most transmission tasks, the energy consumption dramatically changes from scenario to scenario with different network topologies. When k increases from

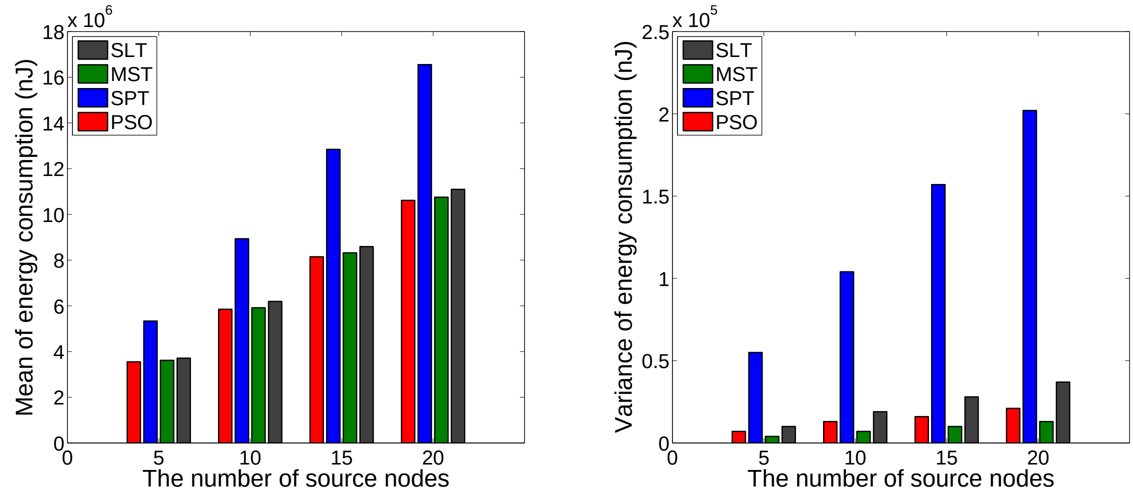
5 to 20, there is a rise in the variance of energy consumption for the four algorithms accordingly. The reason is that the increase in the number of source nodes raises the load of the transmission task.



(a) Mean of energy consumption versus the number of source nodes

(b) Variance of energy consumption versus the number of source nodes

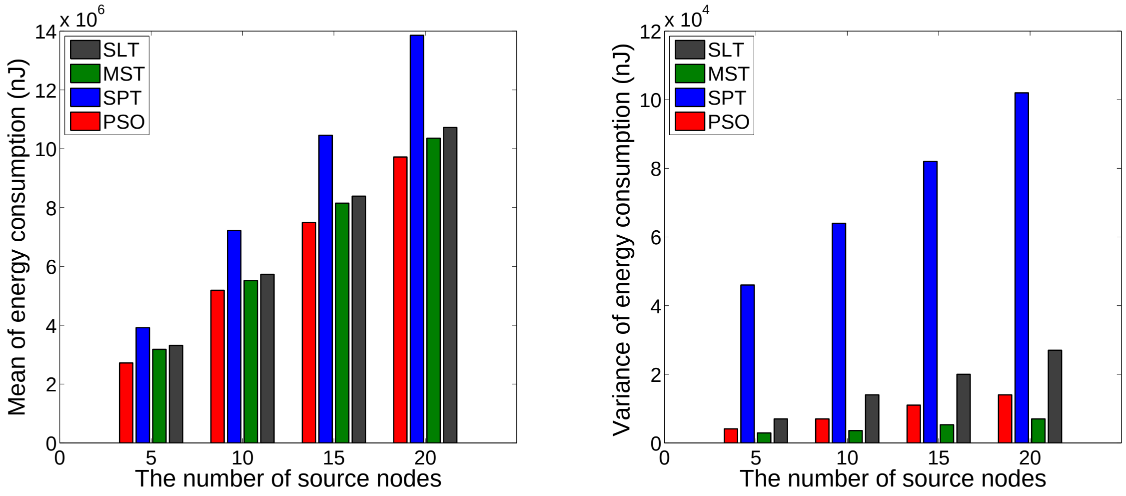
Figure 4.9. Energy consumption versus the number of source nodes when R=10m and ρ approaches 1.



(a) Mean of energy consumption versus the number of source nodes

(b) Variance of energy consumption versus the number of source nodes

Figure 4.10. Energy consumption versus the number of source nodes when R=15m and ρ approaches 1.



(a) Mean of energy consumption versus the number of source nodes

(b) Variance of energy consumption versus the number of source nodes

Figure 4.11. Energy consumption versus the number of source nodes when R=20m and ρ approaches 1.

The second set of simulations was carried out to measure the total energy consumption with ρ approaches 1 and R=10, 15, and 20. The simulation results are depicted in Fig. 4.9, Fig. 4.10, and Fig. 4.11.

When ρ approaches 1 as illustrated in Fig. 4.9(a), Fig. 4.10(a), and Fig. 4.11(a), the PSO algorithm performs better than all other algorithms with regard to the mean of energy consumption. A reduction of average energy consumption of about 5%, 5%, and 40% can be obtained by the PSO algorithm over the MST algorithm, the SLT algorithm, and the SPT algorithm respectively.

The performance of the MST algorithm is the closest to that of the PSO algorithm. However, since some nodes within one hop to the base station would waste energy for detouring to aggregator nodes rather than transmitting directly to the base station, the MST algorithm cannot achieve the optimal performance. Because the nodes can not sufficiently take advantage of data aggregation to reduce the transmission task, the SPT algorithm expends the most energy compared with the three other algorithms. The SLT algorithm can balance between data aggregation and direct transmission, but it gets the benefit implicitly. Hence, we observed that the energy consumption of the

PSO algorithm increases more slowly than the SLT algorithm with the increase in k. Longer transmission range and thus better network connectivity of the network is in favour of the PSO algorithm as it can employ more direct shortest paths to prevent unnecessary aggregation cost at the node near the base station.

Fig. 4.9(b), Fig. 4.10(b), and Fig. 4.11(b) show the variance of energy consumption with the number of source nodes changing in the four algorithms when ρ approaches 1. From these graphs, it can be easily observed that the average energy consumption of the SPT algorithm varies much more than the three other algorithms. Since the SPT algorithm cannot sufficiently exploit data aggregation to improve energy efficiency, its transmission task is much greater than the three other algorithms, and the energy consumption varies in a wider range accordingly. When k increases from 5 to 20, there is a rise in the variance of energy consumption for the four algorithms accordingly since the increase in k raises the load of the transmission task.

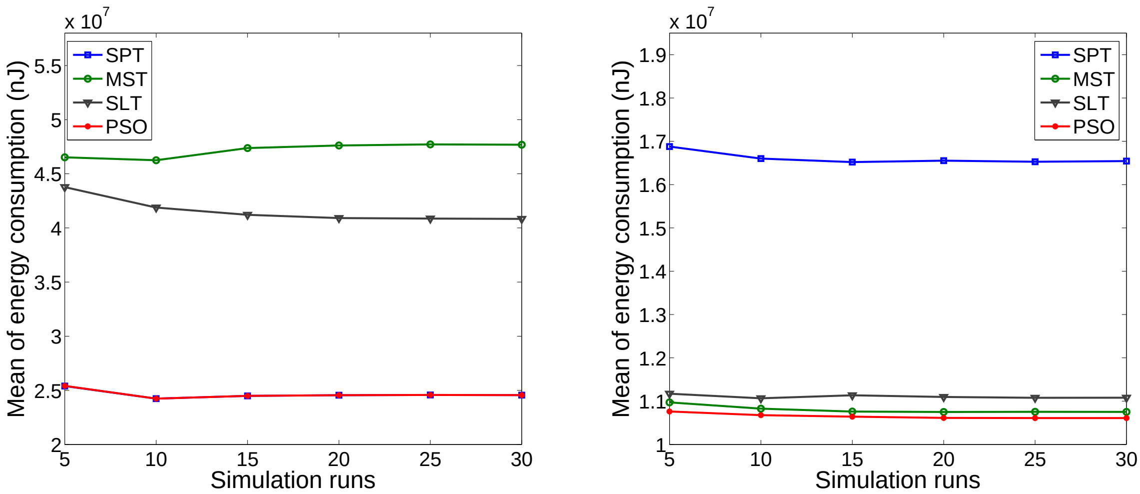
(b) Impact of Simulation Runs

The third set of simulations was carried out to measure the average energy consumption impacted by simulation runs with R=15m and k=20.The simulation results are depicted in Fig. 4.13.

Fig. 4.12(a) and Fig. 4.12(b) show the results by varying the number of runs in different data correlation scenarios. Although the degree of data aggregation differs markedly between them, these two figures demonstrate a similar trend that with the increase in the number of runs the average energy consumption of the four algorithms fluctuates and gradually respectively stabilises after performing 30 runs. Hence, 30 runs are reasonable for statistical significance in our simulation.

(c) Impact of Correlation Coefficient

The correlation coefficient ρ determines data reduction ratio after data aggregation, and it is heavily dependent on the application scenarios. For instance, for the purpose of sensing environment temperature, each node only sends out one temperature value packet after data aggregation and hence ρ for this scenario is 1. On the other hand,



(a) Mean of energy consumption versus simulation runs when ρ approaches 0

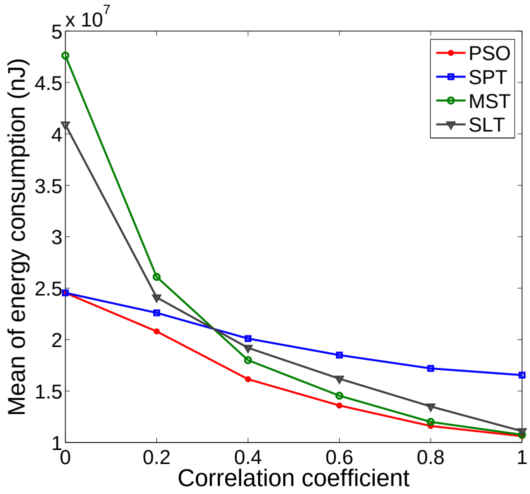
(b) Mean of energy consumption versus simulation runs when ρ approaches 1

Figure 4.12. Energy consumption versus simulation runs when R=15m and k=20.

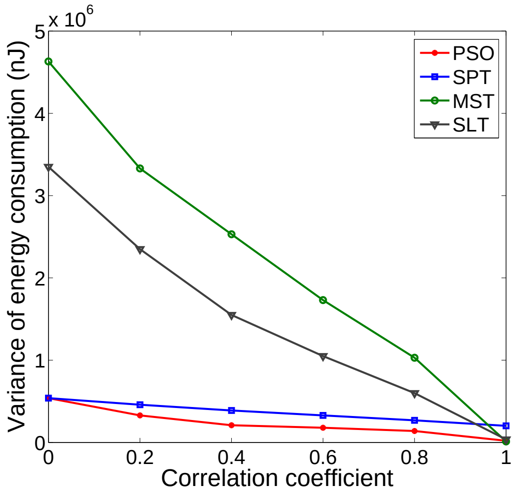
in a multimedia sensor network, images collected by different sensor nodes may have different redundancy because of overlapping fields of view. In this scenario, ρ may vary from 0 to 1. Therefore, it is necessary to examine the performance of the four algorithms under different correlation coefficients.

The fourth set of simulations was carried out to measure the impact of correlation coefficient by comparing our proposed PSO algorithm and the three other algorithms with R=15m and k=20. The simulation results are depicted in Fig. 4.13.

We vary ρ from 0 to 1 with a step size of 0.2. Fig. 4.13(a) illustrates the average energy consumption of the four algorithms. The costs of all algorithms decrease with the increase in ρ, the correlation coefficient. This exemplifies that data aggregation in sensor networks can greatly benefit the routing performance by reducing redundancy among correlated data. In addition, the PSO algorithm evidently outperforms other algorithms in the energy consumption with the increase in the correlation coefficient. A reduction of average energy consumption of about 25%, 30%, and 35% can be obtained by the PSO algorithm over the SLT algorithm, the MST algorithm, and the SPT algorithm, respectively.



(a) Mean of energy consumption versus correlation coefficient



(b) Variance of energy consumption versus correlation coefficient

Figure 4.13. Energy consumption versus correlation coefficient when R=15 and k=20.

When ρ is small, the SPT algorithm performs well. However, it does not benefit from the increase in ρ as it cannot efficiently perform data aggregation to eliminate redundancy among data. In the contrast, the MST algorithm gets the worst performance when ρ is small since it pursues data aggregation but data are actually low correlated. Although the SLT algorithm is more balanced than the MST algorithm and the SPT algorithm, it is still constrained by the fixed routing structure and a fixed approximation ratio to the MST algorithm and the SPT algorithm. We observed the PSO algorithm performs much better than the SLT algorithm. The main reason is that the PSO algorithm recalculates total energy dissipation in every stage to get perfect matching and, thus, can adapt to the correlation among nodes.

Fig. 4.13(b) shows the variance of energy consumption in the four algorithms when ρ changes from 0 to 1. From this graph, it can be easily observed that the average energy consumption of the MST algorithm and the SLT algorithm varies much more than the other two algorithms. The main reason is that the MST algorithm and the SLT algorithm prefer to detour to pursue data aggregation. Although the transmission task decreases with the rise in correlation coefficient, it is still more than the other two algorithms for a wide range of correlation coefficient.

The fifth set of simulations was carried out to measure the maximum run (the first node dies) by comparing our proposed algorithm and the three other methods with R=15m and k=20.

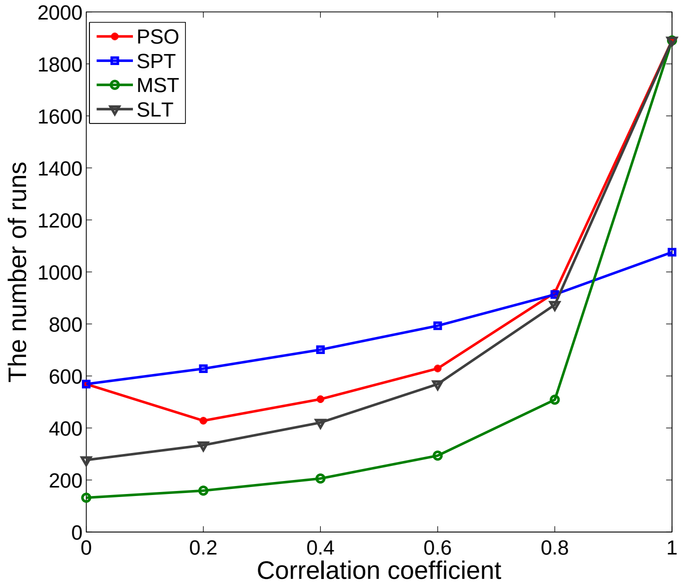


Figure 4.14. The number of runs versus correlation coefficient when R=15m and k=20.

We can see from the figure that the maximum run of all algorithms grows with the increase in ρ. The main reason is that data aggregation can significantly reduce traffic by eliminating redundancy among correlated data and hence consuming less energy. Since aggregator nodes receive data from different nodes and perform data aggregation, they always die earlier than others due to the heavy load. When $\rho < 0.8$, the SPT algorithm outperforms the three other algorithms markedly in terms of maximum run because data from different sources are opportunistically aggregated and the amount of data for aggregation is not large. Since the SPT algorithm cannot sufficiently exploit data aggregation to reduce energy cost, it does not benefit as well as the three other algorithms when ρ increases from 0.8 to 1. We observed that the PSO algorithm performs much better than the SLT algorithm and the MST algorithm. Because the MST algorithm and the SLT algorithm prefer to detour to pursue data aggregation, large amount of data may be aggregated on nodes near the sources, and these aggregator nodes will exhaust energy earlier. In addition, it can be clearly seen from Fig. 4.14 that the maximum run of the PSO algorithms drops dramatically when ρ varies from 0 to 0.2. When $\rho = 0$, there is no redundancy among data and thus, the PSO algorithm transmits data via shortest path. When ρ changes from 0 to 0.2, the PSO algorithm can find more aggregator nodes to share the same paths and the load of aggregator nodes become heavier. As a result, the maximum run of the PSO algorithm declines.

4.4 Conclusion

The performance of our proposed routing scheme was evaluated under different environments and conditions by simulation. Simulation results showed that our scheme performs as well as the shortest path tree algorithm and saves more than 45% of energy over the other two algorithms in the non-aggregation scenario. If perfect aggregation occurs, our scheme obtains about 5% energy reduction at least. When varying from non to perfect aggregation, the simulation results show that our scheme can adapt to the change of data correlation condition and achieve at least a 25% energy saving on average. Since our scheme can save energy and enhance transmission efficiency, it is well suited for applications where energy and data traffic are the primary considerations. It is shown that the PSO algorithm can perform well in data gathering and aggregation for wireless sensor networks.

Chapter 5

Conclusions and Future Work

THIS chapter concludes the thesis by reviewing the work done, re-summarising the original contributions, and recommending future work that could be undertaken by others.

5.1 Review of and Conclusions from the Work in This Thesis

We developed an energy-efficient data gathering and aggregation routing scheme that caters for the energy challenges in wireless sensor networks. The ant colony optimisation algorithm in the scheme collects information about the network and builds initial routing trees. This can avoid the broadcast storm which renders the network unable to transport normal traffic and provide population diversity for routing tree optimisation. The particle swarm optimisation (PSO) algorithm which is performed by the base station is utilised in the scheme to optimise a joint objective between data traffic and transmission structure. The number of traffic tasks can be decreased by executing data aggregation in aggregator nodes. Using this heuristic algorithm, more reasonable aggregator nodes in terms of energy-efficiency can be selected. In order to reduce the energy consumption for data transmission from aggregator nodes to the base station, the particle swarm optimisation algorithm optimises transmission structure as well. This algorithm conducts a multi-hop routing in terms of energy consumption. Significant energy dissipation of data transmission can be reduced by using this algorithm.

The performance of the developed protocol is evaluated by simulations. Energy consumption and network lifetime are defined as the performance metrics for comparing the proposed algorithm with three existing routing algorithms. Simulation results show that the proposed algorithm outperformed existing algorithms in terms of energy consumption and that our scheme can adapt to the change of network connectivity and data correlation condition. Furthermore, these results verified the theoretical analysis of our algorithm regarding optimisation on data traffic and transmission structure. It is shown that the particle swarm optimisation algorithm can perform well in the routing scheme for wireless sensor networks.

Because routing tree optimisation is conducted in the base station, there are small time latency and communication overhead problems when using the proposed algorithm as the routing scheme for wireless sensor networks. In addition, using the multi-hop routing for data transmission from cluster heads to the base station needs a bit more time than direct transmission. Since our scheme can save energy and enhance transmission efficiency, it is well suited for applications where energy and data traffic are the primary considerations.

5.2 Recommendations on Future Work

The energy-efficient data gathering and aggregation based on the PSO algorithm proposed in this thesis offers good performance in data-gathering and aggregation applications of wireless sensor networks. The analysis and simulation results show that the routing algorithm outperforms the three other existing algorithms. However, some aspects of the protocol still need to be improved.

While the aggregation tree with the PSO algorithm has been shown to outperform other routing algorithms, including the SPT algorithm, the MST algorithm, and the SLT algorithm in various system settings, it assumes that aggregation is performed at the intersection nodes whenever data streams are encountered. However, such a strategy may introduce unnecessary energy consumption since it cannot adaptively adjusts aggregation decisions for sensor nodes regarding whether aggregation should be performed.

The correlation model employed in this thesis is not an approximated spatial model, where the correlation coefficient decreases with the distance between two nodes provided that they are within a correlation range. If two nodes are more than a certain distance apart, the correlation coefficient is simply 0. Otherwise, the correlation coefficient is influenced by the distance between the nodes. By varying the correlation range, we can control the average correlation coefficient of the network. Therefore, the data correlation model should be modified further.

5.3 Conclusion

This chapter summarises the research carried out in the duration of the master by research study. The research done in this thesis contributes to knowledge in data gathering and aggregation for wireless sensor networks. The thesis provides a general method for (a) gathering information by an ant colony optimisation algorithm which is an approach to provide particles for the PSO algorithm and (b) aggregation tree optimisation using a modified PSO algorithm. The contributions in this thesis could be used by other researchers in their own studies and applications. The work in this thesis and the recommendations on future work in Section 5.2 will create more research possibilities for improving wireless sensor networks systems.

Appendix A

THIS appendix contains work of the published paper as part of the contributions in this thesis.

Wang, Y. and Lim, C.C. (2010) Gathering correlated data in wireless sensor networks using a heuristic algorithm IN Proceedings of 2011 International Conference on Opto-Electronics Engineering and Information Science (ICOEIS 2011), December 23-25, Xi'an, China

Bibliography

[1] M. Shell. How to Use the IEEEtran LaTeXClass. [Online]. Available: http://www.ngi2009.eu/IEEEtran_HOWTO.pdf [29 July 2010].

[2] Bureau International des Poids et Mesures. The international system of units (SI). [Online]. Available: http://www.bipm.org/utils/common/pdf/si_brochure_8_en.pdf [29 July 2010].

[3] W. R. Heunzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in *Hawaii International Conference on System Sciences*, Jan. 2000, pp. 1–10.

[4] J. Zhao and A. T. Erdogan, "A novel self-organizing hybrid network protocol for wireless sensor networks," in *First NASA/ESA Conference on Adaptive Hardware and Systems*, Istanbul, Turkey, Jun. 2006, pp. 412–419.

[5] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Network*, vol. 38, no. 4, pp. 393–422, 2002.

[6] C. F. Garca-Hernndez, P. H. Ibargengoytia-Gonzlez, J. Garca-Hernndez, and J. A. Prez-Daz, "Wireless sensor networks and applications: a survey," *International Journal of Computer Science and Network Security*, vol. 7, no. 3, pp. 264–273, Mar. 2007.

[7] A. S. Tanenbaum, *Computer Networks*, 3rd ed. Prentice-Hall International, 1996.

[8] C. E. Perkins, *AD HOC Networking*, 1st ed. Boston: Addison-Wesley, 2000.

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

[10] T. Huynh and C. Hong, "An energy*delay efficient multi-hop routing scheme for wireless sensor networks," *IEICE*, vol. E89-D, no. 5, pp. 1654–1661, May 2006.

[11] V. Mhatre and C. Rosenberg, "Design guidelines for wireless sensor networks: communication, clustering and aggregation," *Ad Hoc Networks*, vol. 2, no. 1, pp. 45–63, 2004.

[12] W. Ye, J. Heidemann, and D. Estrin, "System capacity, latency, and power consumption in multihop-routed SS-CDMA wireless network," in *Radio and Wireless Conference (RAWCON)*, Aug. 1998, pp. 55–58.

[13] T. J. Shepard, "A channel access scheme for large dense packet radio networks," in *ACM SIGCOMM*, Stanford, Aug. 1996, pp. 219–230.

[14] C. Li, M. Ye, G. Chen, and J. Wu, "An energy-efficient unequal clustering mechanism for wireless sensor networks," in *2nd IEEE International Conference on Mobile Ad-hoc and Sensor Systems*, Washington, DC, Nov. 2005, pp. 8–15.

[15] M. Ye, C. Li, G. Chen, and J. Wu, "EECS: An energy efficient clustering scheme in wireless sensor networks," in *Performance, Computing, and Communications Conference*, Phoenix, Apr. 2005, pp. 535–540.

[16] C. P. Low, C. Fang, J. M. Ng, and Y. H. Ang, "Efficient load-balanced clustering algorithms for wireless sensor networks," *Computer Communications*, vol. 31, pp. 750–759, 2008.

[17] S. Bandyopadhyay and E. J. Coyle, "An energy efficient hierarchical clustering algorithm for wireless sensor networks," in *IEEE INFOCOM*, Apr. 2003, pp. 1713–1723.

[18] A. A. Abbasi and M. Younis, "A survey on clustering algorithms for wireless sensor networks," *Computer Communications*, vol. 30, pp. 2826–2841, Jun. 2007.

[19] M. Chatterjee, S. K. Das, and D. Turgut, "WCA: A weighted clustering algorithm for mobile ad hoc networks," *Journal of Cluster Computing*, vol. 5, no. 2, pp. 193–204, Apr. 2002.

[20] S. Banerjee and S. Khuller, "A clustering scheme for hierarchical control in multi-hop wireless networks," in *INFOCOM*. Los Alamitos, CA: IEEE Computer Society Press, 2001, p. 1028C1037.

[21] C. Intanagonwiwat, D. Estrin, R. Govindan, and J. Heidemann, "Impact of density on data aggregation in wireless sensor networks," in *Proc. 22nd International Conference on Distributed Computing Systems*, Nov. 2002, pp. 457–458.

[22] J. Zhao, R. Govindan, and D. Estrin, "Computing aggregates for monitoring wireless sensor networks," in *Proc. IEEE International Workshop on Sensor Network Protocols and Applications*, May 2003, pp. 139–148.

[23] B. Krishnamachari, D. Estrin, and S. Wicker, "Modelling data-centric routing in wireless sensor networks," in *Proc. IEEE INFOCOM*, New York, NY, Jun. 2002.

[24] C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed diffusion: A scalable and robust communication paradigm for sensor networks," in *Proc. of the 6th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'00)*, Boston, MA, Aug. 2000, pp. 56–67.

[25] S. Lindsey and C. S. Lindsey, "PEGASIS: Power-efficient gathering in sensor information systems," in *Proc. IEEE aerospace conf.*, 2002, pp. 1125–1130.

[26] Y. Yao and J. Gehrke, "The cougar approach to in-network query processing in sensor networks," *SIGMOD Record*, vol. 31, no. 3, pp. 9–18, Sept. 2002.

[27] K. Akkaya, M. Younis, and M. Youssef, "Efficient aggregation of delay-constrained data in wireless sensor networks," in *Proc. Internet compatible QoS in Ad Hoc Wireless Networks 2005*, Jan. 2005, pp. 904–909.

[28] M. Enachescu, A. Goel, R. Govindan, and R. Motwani, “Scale free aggregation in sensor networks,” in *Proc. First International Workshop on Algorithmic Aspects of Wireless Sensor Networks*, Jul. 2004, pp. 71–84.

[29] M. Khan, G. Pandurangan, and B. Bhargava, “Energy-efficient routing schemes for wireless sensor networks,” Department of Computer Science, Purdue University,” Technical Report CSD TR 03-013, Jul. 2003.

[30] A. Deligiannakis and Y. Kotidis, *Exploiting spatio-temporal correlations fordata processing insensor networks*. GeoSensor Networks, 2008, ch. Data Acquisition and Processing, pp. 45–65.

[31] A. Deligiannakis and Y. Kotidis, “Data reduction techniques in sensor networks,” *IEEE Data Eng. Bull.*, pp. 19–25, 2005.

[32] A. Deligiannakis, Y. Kotidis, and N. Roussopoulos, “Compressing historical information in sensor networks,” in *2004 ACM SIGMOD International Conference on Management of Data*, 2004, pp. 527–538.

[33] W. R. Heunzelman, A. P. Chandrakasan, and H. Balakrishnan, “Energy-efficient communication protocol for wireless micro-sensor networks,” in *Proc. the Hawaii International Conf. on System Sciences*, Maui, 2000, pp. 3005–3014.

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

[35] A. Manjeshwar and D. P. Manjeshwar, “TEEN: a routing protocol for enhanced efficiency in wireless sensor networks,” in *Proc. 15th international workshop on parallel and distributed computing issues in wireless networks and mobile computing*, San Francisco, 2001, pp. 2009–2015.

[36] S. Ghiasi, A. Srivastava, X. Yang, and M. Sarrafzadeh, “Optimal energy aware clustering in sensor networks,” *Sensors*, vol. 2, no. 7, pp. 258–269, 2002.

[37] A. Jing and D. Turgut, “A cluster-based energy balancing scheme in heterogeneous wireless sensor networks,” *Lecture notes in computer science*, vol. 3420, pp. 467–474, 2005.

[38] M. Qin and R. Zimmermann, “An energy-efficient voting-based clustering algorithm for sensor networks,” in *Proc. 1st ACIS International Workshop on Self-Assembling Wireless Networks (SAWN)*, May 2005, pp. 444–451.

[39] Q. Fang, F. Zhao, and L. Guibas, “Lightweight sensing and communication protocols for target enumeration and aggregation,” in *Proc. 4th ACM international symposium on Mobile ad hoc networking and computing*, Annapolis, 2003, pp. 165–176.

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

[41] N. Tabassum, Q. E. K. Mamun, and Q. Urano, "Cosen: A chain oriented sensor network for efficient data collection," in *Proc. the Global Telecommunications Conference*, vol. 6, 2003, pp. 3525–3530.

[42] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematik*, vol. 1, pp. 269–271, 1959.

[43] W. Heinzelman, J. Kulik, and H. Balakrishnan, "Adaptive protocols for information dissemination in wireless sensor networks," in *Proc. ACM MobiCom*, Aug. 1999, pp. 174–185.

[44] A. Ahmed, H. Shi, and Y. Shang, "A survey on network protocols for wireless sensor networks," in *Proc. IEEE International Conf. Information Technology: Research and Education*, Aug. 2003, pp. 301–305.

[45] F. Ye, H. Luo, J. Cheng, S. Lu, and L. Zhang, "A two-tier data dissemination model for large-scale wireless sensor networks," in *Proc. ACM MobiCom*, Sept. 2002, pp. 148–159.

[46] S. Pattem, B. Krishnamachari, and R. Govindan, "The impact of spatial correlation on routing with compression in wireless sensor networks," in *Proc. Third International Symp. Information Processing in Sensor Networks*, Apr. 2004, pp. 28–35.

[47] W. Zhang and G. Cao, "Optimizing tree reconfiguration for mobile target tracking in sensor networks," in *Proc. IEEE INFOCOM*, vol. 4, Mar. 2004, pp. 2434–2445.

[48] R. Cristescu, B. Beferull-Lozano, and M. Vetterli, "On network correlated data gathering," in *Proc. IEEE INFOCOM*, vol. 4, Mar. 2004, pp. 2571–2582.

[49] P. Rickenbach and R. Wattenhofer, "Gathering correlated data in sensor networks," in *Proc. ACM Workshop Foundations of Mobile Computing*, Oct. 2004, pp. 60–66.

[50] Y. Yu, B. Krishnamachari, and V. Prasanna, "Energy-latency tradeoff for data gathering in wireless sensor networks," in *Proc. IEEE INFOCOM*, vol. 1, Mar. 2004, pp. 244–255.

[51] R. C. Prim, "Shortest connection networks and some generalization," *Bell System Technical*, vol. 36, no. 6, pp. 1389–1401, 1957.

[52] A. Goel and K. Munagala, "Balancing steiner trees and shortest path trees online," in *Proc. 11th ACM-SIAM Symp. on Discrete Algorithms*, Jan. 2000, pp. 562–563.

[53] B. Raghavachari, S. Khuller, and N. Young, "Balancing minimum spanning and shortest path trees," in *Proc. Fourth ACM-SIAM Symp. on Discrete Algorithms*, 1993, pp. 243–250.

[54] R. Cristescu, B. Beferull-Lozano, M. Vetterli, and R. Wattenhofer, "Network correlated data gathering with explicit communication: NP-completeness and algorithms," *IEEE/ACM Transactions on Networking*, vol. 14, no. 1, pp. 41–54, Feb. 2006.

[55] M. R. Garey and D. Johnson, *Computers and intractability: A guide to the theory of NP-Cpmpleteness*. W. H. Freeman, 1979.

[56] H. Xia, H. L. Bertoni, and L. R. Maciel, "Radio propagation characteristics for line-of-sight microcellular and personal communications," *Antenna and Propagation*, vol. 41, no. 10, pp. 1439–1447, Oct. 1993.

[57] T. S. Rappaport, *Wireless Communications: Principles and Practice*, 2nd ed. New Jersey: Prentice Hall, PTR, 2001.

[58] W. B. Heinzelman, "Application-specific protocol architectures for wireless networks," PhD. dissertation, Massachusetts Institute of Technology, 2000.

[59] M. Z. Zuniga and B. Krishnamachari, "Integrating future large-scale wireless sensor networks with the internet," Department of Electrical Engineering, University of Southern California," Technical Report CS 03-792, 2003.

[60] J. Alonso, A. Dunkels, and T. Voigt, "Bounds on the energy consumption of routings in wireless sensor networks," in *2nd WiOpt, Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks*, Cambridge, UK, March 2004.

[61] J. Kennedy, R. C. Eberhart, and Y. H. Shi, *Swarm intelligence*. San Francisco: Morgan Kaufmann, 2001.

[62] M. Clerc and J. Kennedy, "The particle swarm-explosion, stability, and convergence in a multidimensional complex space," *IEEE Trans. on Evolutionary ComPutation*, vol. 6, no. 1, pp. 59–73, 2002.

[63] M. Dorigo, "Optimization, Learning and Natural Algorithms," PhD thesis, Politecnico di Milano, Italy, 1992.

[64] D. L. Applegate, R. E. Bixby, V. Chvátal, and W. J. Cook, *The traveling salesman problem: a computational study*.

[65] J. Kennedy and R. C. Eberhart, "Particle Swarm Optimization," in *the IEEE Conf. on Neural Networks, IV*, 1995, pp. 1942–1948.

[66] Z. Michalewicz, *Genetic Algorithms + Data Structures = Evolution Programs*, 3rd ed. Springer-Verlag, 1996.

[67] J. Kennedy and R. C. Eberhart, "A discrete binary version of the particle swarm algorithm," in *Proc. IEEE International Conference on Systems, Man, and Cybernetics*, vol. 5, New Jersey, USA, Oct. 1997, pp. 4104–4109.

[68] D. K. Agrafiotis and W. Cedeño, "Feature selection for QSAR and QSAR using binary particle swarms," *J. Med. Chem.*, vol. 45, pp. 1098–1107, 2002.

[69] M. Clerc, "Discrete particle swarm optimization illustrated by the traveling salesman problem," *New optimization techniques in engeering*, pp. 219–239, 2004.

[70] B. A. Julstrom, "A genetic algorithm for the rectilinear Steiner problem," in *Proc. 5th International Conference on Genetic Algorithms*. CA: Morgan Kaufmann, 1993, pp. 474–480.

[71] M. Gen, R. Cheng, and D. Wang, "Genetic algorithms for solving shortest path problems," in *IEEE Intemational Conference on Evolutionary ComPutation*, Indianapolis, IN , USA, 1997, pp. 401–406.

[72] C. W. Ahn and R. S. Ramakrishna, "Genetic algorithm for shortest path routing problem and the sizing of populations," *IEEE transactionson evolutionary computation*, vol. 6, no. 6, pp. 566–579, 2002.

[73] M. Munetomo, Y. Takai, and Y. Sato, "A migration scheme for genetic adaptive algorithm," in *Proc. IEEE Intemational Conf. Systems, Man, and Cyberneties*, 1998, pp. 2774–2779.

[74] D. B. Fogel, "Applying evolutionary programming to seleted travelling salesman problems," *Cybemetics and Systems*, vol. 24, pp. 27–36, 1993.

[75] K. A. D. Jong, "An analysis of the behavior of a class of genetic adaptive systems," PhD thesis, Computer and Communication Sciences, University of Michigan, Aug. 1975.

[76] J. E. Baker, "Adaptive selection methods for genetic algorithms," in *Proc. the 1st International Conference on Genetic Algorithms*, 1985, pp. 101–111.

[77] L. D. Whitley, "The GENITOR algorithm and selective pressure: Why rank-based allocation of reproductive trials is best," in *Proc. the 3rd International Conference on Genetic Algorithms*, 1989, pp. 116–121.

[78] J. J. Greffenstette and J. E. Baker, "How genetic algorithms work: A critical look at implicit parallelism," in *Proc. the 3rd International Conference on Genetic Algorithms*, 1989, pp. 20–27.