Intelligent Systems Approach to the Next Generation Communication Network Routing Design

By

Wei Ren

A thesis submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy in Engineering

School of Engineering, University of Warwick

June 2010

Contents

List of Figures

List of Tables

Acknowledgements

I would like to thank the following people from the School of Engineering, University of Warwick for their assistance during my period of research. I would like to thank my supervisors, Professor Evor L. Hines and Dr. Mark S. Leeson for their support and encouragement over the years. Secondly, I would like to extend my sincere gratitude (in no particular order) to Professor Habib F. Rashvand, Professor Roger J. Green, and all the members of the Intelligence Systems Engineering Laboratory.

I would like to thank Dr. Yousef S. Kavain from Shahid Chamran University, Iran, for the corporation and good time during producing the research work on survivable DWDM optical mesh network design.

A special thanks goes to all my friends, who have supported me through the good and bad times. Last but not least, I express my most sincere appreciation to my parents and Miss Zheng Xie for their understanding, patience and support during the entire period of my PhD studies.

Declaration

This thesis is presented in accordance with the regulations for the degree of doctor of philosophy. All work reported has been carried out by the author unless otherwise stated, including the production of this document.

Glossary of Abbreviations

AA	Artificial Ant
ABR	Associativity Based Routing protocol
ACO	Ant Colony Optimisation
AMRoute	Ad hoc Multicast Routing Protocol
ANSI	Ad hoc Networking with Swarm Intelligence
AODV	Ad hoc On-demand Distance Vector
ATM	Asynchronous Transfer Mode
CBR	Constant Bit Rate
DPP	Dedicated Path Protection
DSDV	Destination Sequenced Distance Vector
DSR	Dynamic Source Routing
DWDM	Dense Wavelength Division Multiplexing
EA	Evolutionary Algorithm
FDM	Frequency Division Multiplexing
FTTP	Fibre To The Premises network
GA	Genetic Algorithm
GP	Genetic Programming
GRPH	Multicast Group Hello message
ILP	Integer Linear Programming
IP	Internet Protocol
LISP	List Processor
MACT	Multicast route Activation message
MANET	Mobile Ad hoc Network

MANSI	The Multicast for Ad hoc Network with Swarm Intelligence
MAODV	Multicast On-demand Distance Vector routing protocol
ODMRP	The On-demand Multicast Routing Protocol
OXC	Optical Cross-Connect
PDH	Plesiochronous Digital Hierarchy
PON	Passive Optical Network
PSO	Particle Swarm Optimisation
QoS	Quality of Service
RERR	Route Error message
RREP	Route Reply message
RREQ	Route Request message
RWA	Routing and Wavelength Assignment
SDH	Synchronous Digital Hierarchy
S-expression	Symbolic expressions
SI	Swarm Intelligence
SILS	Swarm Intelligence based Link Stability routing protocol
SIMS	Swarm Intelligence based Multicasting with Stability metrics routing protocol
SPP	Shared Path Protection
SSA	Signal Stability Adaptive routing protocol
TDM	Time Division Multiplexing
WDM	Wavelength Division Multiplexing

List of Author's Publications

Book Chapters

W. Ren, E. L. Hines, M. S. Leeson, Y. S. Kavian, and D. iliescu, "Swarm intelligence based link stability routing optimisation algorithm for mobile ad hoc network," in *Intelligence Systems: T echniques a nd Applications* , E. Hines, M. Leeson, and etc., Eds. Netherlands: Shaker Publishing, 2008, pp. 379-398.

Y. S. Kavian, W. Ren, E. L. Hines, M. S. Leeson, Habib F. Rashvand, M. Naderi, "Design of Survivable DWDM Optical Mesh Transport Networks Using Genetic Algorithms," in *Intelligent Systems: T echniques and Applications* , E. Hines, M. Leeson, and etc., Eds. Netherlands: Shaker Publications, 2008, pp. 343-475.

Y. S. Kavian, W. Ren, M. S. Leeson, E. L. Hines, H. F. Rashvand, and M. Naderi, "Survivable DWDM Optical Mesh Transport Network Design via Genetic Algorithms", in *Reliability Engineering Advances*, Nova Science Publishers, 2009.

Journal Papers

W. Ren, E. L. Hines, M. S. Leeson, Daciana Iliescu, Yousef S. Kavian, "Swarm Intelligence based Multicast Routing Optimisation for Mobile Ad Hoc Network", *Ad Hoc Networks,* submitted on April 13th , 2010 (under review).

Y. S. Kavian, W. Ren, M. Naderi, M. S. Leeson, E. L. Hines, "Fault Tolerant Dense Wavelength Division Multiplexing Optical Transport Networks", *Journal of Telecommunication and Information Technology (invited), Vol. 1, pp. 63-67,* 2009.

Y. S. Kavian, W. Ren, M. Naderi, M. S. Leeson, E. L. Hines, "Survivable Wavelength-Routed Optical Network Design Using Genetic Algorithms", *European Transaction on Telecommunication,* Vol. 19, No. 3, pp. 247-255, 2008.

Y. S. Kavian, H. F. Rashvand, W. Ren, M. S. Leeson, E. L. Hines, M. Naderi, "Network Topology Effect on QoS Delivering in Survivable DWDM Optical Networks", *Journal of Telecommunication and Information Technology (invited), Vol. 1, pp. 68-71,* 2009.

Y. S. Kavian, H. F. Rashvand, W. Ren, M. Naderi, E. L. Hines, M. S. Leeson, *"*Genetic algorithm quality of service design in resilient dense wavelength division multiplexing optical networks", *IET Communication* , Vol. 2, No. 4, pp. 505–513, 2008.

Y. S. Kavian, H. F. Rashvand, W. Ren, M. S. Leeson, E. L. Hines, M. Naderi, "RWA problem for designing DWDM networks-delay against capacity optimization," *Electronics Letters,* Vol. 43, No. 16, pp. 892-893, 2007.

Conference Papers

Y. S. Kavian, H. F. Rashvand, W. Ren, M. S. Leeson, E. L. Hines, M. Naderi, "QoS-based RWA in DWDM Optical Networks using Genetic Algorithms ", *Proceeding of IEEE International Conference on Transparent Optical Networks, ICTON-MW08, Marrakech*, Morocco, pp.1-4, 2008.

M. S. Leeson, Y. S. Kavian, W. Ren, E. L. Hines, M. Naderi, "Survivable Wavelength-Routed Optical Network Design Using Genetic Algorithms", *Proceeding of IEEE International Conference on Trans parent Optical Networks, ICTON-MW07,* Sousse, Tunisia, pp.1-4, 2007.

Abstract

This thesis describes the research and development of novel intelligent systems guided routing protocols for next generation communication networks. The ultimate goal of future communication networks in general is to provide access to information when needed, where needed and in whatever format needed. To achieve this goal, wireless and optical technologies must be leveraged and converged seamlessly, giving rise to *fibre-wireless* access networks. This research project focuses on the design of survivable Dense Wavelength Division Multiplexing (DWDM) optical networks and routing optimisation for self-organised mobile ad hoc networks (MANETs). There are three main objectives. The first is to develop novel intelligent system applications for the design of survivable routing in DWDM optical networks. The second is to develop new techniques to achieve robust and efficient unicast routing for MANET networks. The third is to discover techniques to achieve robust and efficient multicast routing for MANET networks.

Genetic algorithms (GAs) are employed for the routing wavelength allocation problems (RWA). Simulation results prove GAs are a promising approach to tackle RWA problems in DWDM networks for both dedicated path protection (DPP) and shared path protection (SPP). Moreover, it is concluded that under the Pan European network model the SPP approach is able to produce a saving of 8.1% in the overall network bandwidth. In the context of the MANET routing protocol design, swarm

intelligence (SI) algorithms have been applied to both unicast and multicast routing protocols. Routing stability issues for MANETs have been considered and novel statistics based stability metrics have been designed for MANET networks. The novel unicast and multicast MANET routing protocols achieve impressive performance during designed simulations.

The principal conclusion from this thesis is that intelligent systems, such as evolutionary algorithms and swarm intelligence can be successfully applied to the design of next generation communication network protocols.

Chapter 1:

Introduction

1.1 Introduction

'Telecommunication' is a term which comes from the Greek meaning 'communication at distance' from a transmitter to a receiver through signals of varied nature. In order to achieve effective communication, the choice of a proper means of transport for the signal has played a fundamental role. In the 1870s, telephony, which was the initial voice service, was invented to provide for the first time ever 'human to

human' voice communication. Since then, telecommunication technology has evolved tremendously: from short distance to long distance communications, from point-to-point to point-to-multipoint connections and further to multipoint-to-multipoint, from stationary to mobile user, from terrestrial only to terrestrial-to-air and further to terrestrial-to-sea communications. In addition, the development of information technologies has become ripe enough to provide data communication services on a large scale for 'human-to-human' and 'human-to-machine' communications in the last forty years [1]. Communication technology is still rapidly developing; today the design of next generation networks is a very active field that enjoys a rapid pace of development. Designing 'intelligent' routing for next generation networks is the primary interest of this thesis, novel intelligent systems techniques based routing protocols will be presented and discussed in detail in the following chapters.

The aim of the first chapter is to outline the perspectives of the future communication networks and to explain the importance of routing in communication networks. It also presents the main problems that this research work is concerned with solving; illustrates the research objectives; indicates how this work makes a significant contribution to knowledge. Finally, the structure of this thesis is outlined.

1.1.1 Vision of next generation communication networks

The ultimate goal of the next generation networks is to connect any person, device and resource independently of distance, location and time, through integrated intelligent interfaces and with enriched media [1]. To achieve this goal, wireless and optical technologies play a key role. Wireless and optical access networks can be thought of as complementary. Optical fibre does not go everywhere, but where it does go, it provides a tremendous amount of available bandwidth. Wireless access

networks, on the other hand, potentially go almost everywhere but provide a highly bandwidth-constrained transmission channel susceptible to a variety of impairments. Clearly, as providers need to satisfy users with continuously increasing bandwidth demands, future broadband access networks must leverage on both technologies and converge them seamlessly, giving rise to *fibre-wireless* access networks [2].

1.1.1.1 The optical access network

Fibre optic networks are the only ones that can carry a huge amount of data at the speed of light. Therefore, fibre communication techniques have been widely deployed as long-haul backbone transport networks and metro area networks.

Since the first optical protocol came into being, SONET/SDH has been proven for robustness, bandwidth transport and fast switching to protection [1]. However, the transportable bandwidth and data capacity was soon overrun by an unsaturated bandwidth appetite and new services. Within a decade or so, this led to a new optical network that was based on an optical and photonic technology known as dense wavelength division multiplexing (DWDM). A benefit of this optical network was that it helped to solve concerning the amount of transportable traffic, although at the same time it created a bottleneck at the network edge or access. Currently, different technologies are under development, and fibre is deployed at the access using an almost passive optical network (PON) technology suitable for fibre to the premises (FTTP) networks. Typically, PONs are time-division multiplexing (TDM) single-channel systems, where the fibre infrastructure carries a single upstream wavelength channel and a single downstream wavelength channel [2]. Adding the wavelength dimension to conventional TDM PONs leads to WDM PONs. WDM PONs have several advantages such as increased network capacity and improved network

scalability. Moreover, they have the potential to become the optical backbones for future communication networks [3].

1.1.1.2 The wireless access network

Recent advances in wireless communication technology have led to significant innovations that have enabled cost-effective and flexible multi-hop wireless networks. In today's wireless communication networks, the majority of the communication devices are user terminals, e.g. mobile phones, PDAs and laptop computers. Communications among them are almost via core networks even when they are located next to each other. In the next generation network environment, new technologies enable everything (such as a table, a chair, a book etc.) to become a network node or a router with an integrated radio technology. Sensors will be embedded everywhere in the future, especially in those places where human beings have difficulty in gaining access to, e.g. in space, underground and deep in the sea, in order to collect data from nature and to be aware of changes. The wide use of wireless communication owing to the scarcity of radio resources, which suggests that next generation networks should facilitate the realization of the generic access concept, i.e. to construct the wireless network in an ad hoc way.

By seamlessly converging optical and wireless access technologies, hybrid *Fibre-Wireless* access networks hold great promise to change the way we live and work in the future [2].

1.1.2 Routing in communication networks

In real communication networks, not all nodes are directly connected with each other, this is the reason why routing is necessary in a communication network. Routing is a mechanism that allows information transmitted over a network to be routed from a

source to a destination through a sequence of intermediate switching/buffering stations or nodes [4]. A routing algorithm selects paths that meet the objectives and constraints set by the user traffic and the network topology and therefore determines which network resources are traversed by which user traffic [5-7].

The problem to be solved by any routing algorithm is to direct traffic from sources to destinations maximizing network performance while minimizing costs. There are many possible routing algorithms differing mainly in terms of the characteristics of the network and of the traffic. One classification of such numerous routing algorithms can be done via the type of cast property, that is, unicast or multicast routing algorithms [8]. Unicast is still the predominant form of routing protocol within the current communication networks where a piece of information is established and maintained from one point to another point. In other words, there is just one sender, and one receiver for unicast routing algorithms. Multicasting is the networking technique for delivering the same packet simultaneously to a group of clients. One example of an application which may use multicast is a video server sending out networked TV channels. This could effectively reduce the sustained high bandwidth requirement of transmitting via the unicast routing algorithms.

In real networks, traffic conditions are constantly changing and the structure of the network itself may fluctuate. Because there are usually many possible pathways for one message to be transmitted from a given node to anther node, it is possible, in principle, to make routing algorithms adaptive enough to overcome local congestion. If there is a sudden burst of activity, or if one node becomes the destination or the emitter of a large number of calls, rerouting becomes crucial.

1.2 Introduction to the problem

Routing plays an important role in communication networks. A carefully designed routing mechanism is an essential factor to provide efficient and robust network communication. The design or optimisation of the routing for a network is considered to be an NP-hard problem, which involves huge computation. Thus, intelligent system applications are necessary in order to improve routing qualities in communication networks. *Fibre-Wireless* access networks are highly likely to become the next generation communication networks [1, 2, 9]. *Fibre-Wireless* access networks are formed by two different kinds of networks, optical access networks (such as FTTP, PONs) and the wireless access networks (such as MANETs, cellular networks). During the period of this work, it was not possible to cover the routing optimisation for all the above mentioned networks. Instead, the research and design of the survivable DWDM optical network and the routing optimisation for the self-organised mobile ad hoc networks (MANETs) are main focus of this thesis.

The ability to provide potentially unlimited transmission capacity is the most obvious advantage of DWDM technology. However, on the other hand, such networks are intolerant to link breakage at any time due to the unaffordable cost of losing tremendous amounts of data. Protection against link failures is thus very important for DWDM backbone networks and the design of survivable and efficient DWDM optical mesh networks becomes the major concern within the optical access network domain for this research work.

In the context of the wireless access network domain, MANET networks can provide users with freedom and scalability. However, routing for such dynamic and infrastructureless networks is a challenging task. Currently, the majority of the

MANET routing designs are based on improving the Quality of Service (QoS) metrics that often appear in the wired network optimisation literature, namely routing distance, end to end delay and available bandwidth. The routing stability issue is often overlooked in this process. Frequent link breakage which causing 'flooding' in MANET network, wastes the limited MANET resources, such as bandwidth, computation overhead, battery power. In an attempt to address this research gap, the topics of implementing link stability metrics and designing more robust and efficient routing protocol for MANET have been raised.

1.3 Research objective

The primary objective of this work is to investigate the potential application of intelligent systems to the design of next generation communication network routing protocols. Additional sub-objectives will be outlined later in this section.

Kasabov describes the term 'intelligent systems' as comprising methods, tools, and systems for solving problems that normally require the intelligence of humans [10]. A more detailed exposition of intelligent systems is provided in *Chapter 2* and *Chapter 4*. For the purposes of this chapter it is sufficient to describe intelligent systems as being systems which can emulate the biologically inspired process, can perform the creative improvement process and can help to improve the routing quality of simulated networks.

Given the overarching objective here, namely, to investigate the potential of intelligent systems for assisting researchers in the design of routing protocols for future communication networks, this can now be refined in terms of several sub-objectives in a modular sense. The key ones are to research and develop:

1. Novel intelligent system applications for designing survivable routings of DWDM optical networks.
2. New techniques to achieve robust and efficient unicast routing for MANET networks.
3. New techniques to achieve robust and efficient multicast routings for MANET networks.

1.4 Contribution to knowledge

In the context of the objectives outlined in *Section 1.3*, this thesis aims to demonstrate contributions to scientific knowledge via the design of future communication network routing protocols using intelligent system techniques. In particular, it shows that the survivable routing of the DWDM optical networks can be effectively planned using GAs; and demonstrates that with the aid of Swarm Intelligence (SI) algorithms more efficient unicast and multicast routing for MANETs can be achieved. The modifications and techniques embedded in the proposed GAs and SI applications will provide other researchers with a novel way of thinking in the domain of network routing design. This body of work is thus presented as a 'significant and original contribution to knowledge'.

1.5 Thesis outline

According to the different network characteristics and the various intelligent systems techniques applied to the network, the contents of this thesis are divided into two main parts. *Part I* of the thesis, which contains *Chapter 2* and *Chapter 3*, aims to outline the design of survivable DWDM optical mesh transport networks by applying GAs. *Chapter 4*, *Chapter 5* and *Chapter 6*, which comprise *Part II* of this thesis,

present novel protocols for highly dynamic MANET networks. The outline of each chapter is provided as follows.

In *Chapter 2*, an introduction to evolutionary algorithms is presented. The advantages of GA comparing to conventional optimisation algorithms are discussed, including the pros and cons of applying GAs and genetic programming (GP) to the research problem in hand. The way in which GP extends GA to enhance the problem representation is of particular interest in this research work. The idea helps in the finding of a way of encoding optical network routing into GA chromosomes without increasing the computational complexity.

Chapter 3 provides the essential background knowledge to the survival of DWDM optical networks, describing the purpose of survival routing design within the optical network domain, and indicating why GAs are particularly useful to be employed to solve this problem. The process of designing the survivable DWDM optical network is illustrated. The solutions generated by the GA under different protection schemes are provided and compared.

Chapter 4 starts *Part II* of this thesis. The swarm intelligence algorithms are explained in this chapter. Two major sub-areas of SI, namely, particle swarm optimisation (PSO) and ant colony optimisation (ACO) are introduced, and a detailed framework is implemented for each algorithm provided.

In *Chapter 5* a novel unicast MANET routing protocol, the swarm intelligence based link stability routing protocol (SILS) is presented. SILS is an ACO based protocol particularly designed to deal with scenarios with a mixture of static and dynamic network environments. At the beginning of this chapter, the traditional and existing swarm intelligence based routing protocols for MANETs are reviewed, the

importance of the stability factor in MANET routing efficiency is discussed and the complete description of the SILS implementations are provided. The SILS protocol is simulated together with two other benchmark protocols within the Simulator for Network Algorithms (Sinalgo) environment. The simulation results of these protocols are compared under different mobility scenarios.

Chapter 6 is concerned with multicast routing for MANETs, and extends the mobility scenarios to more dynamic cases. At the beginning of this chapter, the various multicast routing structures for MANET are reviewed. The swarm intelligence based multicasting with stability metric routing protocol (SIMS) is then presented. In a similar fashion to the procedure for the SILS protocol, the SIMS and another two benchmark protocols are simulated in the *Sinalgo* environment. The simulation results and the performance comparison are provided at the end of this chapter.

Chapter 7 summarises the key results of the research presented in *Chapter 3*, *Chapter 5* and *Chapter 6*. The relative merits and limitations of the approaches are discussed, and suggestions for improvements are provided. The chapter provides the principal conclusions and the main findings in this work. In addition, further areas of work are suggested.

References

[1] J. L. Salina, & Salina, P. , *Next Generation Networks: Perspectives and potentials*. Chichester, England: J. Wiley & Sons, 2007.

[2] N. Ghazisaidi, M. Maier, and C. M. Assi, "Fiber-Wireless (FiWi) access networks: A survey," *Communications Magazine, IEEE,* vol. 47, pp. 160-167, 2009.

[3] T. Koonen, "Fiber to the home/Fiber to the premises: what, where, and when," *Proc. IEEE,* vol. 94, pp. 911-934, May 2006.

[4] E. Bonabeau, M. Dorigo, and G. Theraulaz, "Swarm Intelligence: From natural to artificial systems " New York: Oxford University Press, 1999.

[5] D. Bertsekas and R. Gallager, *Data networks*. Englewood Cliffs, NJ: Prentice Hall, 1992.

[6] F. P. Kelly, "Modelling communication networks, present and future," *Phil. Trans. ,* pp. 437-463, 1995.

[7] M. Steenstrup, *Routing in communication netw orks*. Englewood Cliffs, NJ: Prentice Hall, 1995.

[8] S. K. Sarkar, T. G. Basavaraju, and C. Puttamadappa, "Ad hoc mobile wireless networks principles, protocols, and applications," New York: Auerbach Publications, 2008.

[9] M. Maier, N. Ghazisaidi, and M. Reisslein, "The audacity of Fiber-Wireless (FiWi) Networks (Invited Paper)," in *Proc. ICST Int'l. Conf. Access N et.* Las Vegas, NV, Oct. 2008.

[10] N. K. Kasabov, *Foundations of neural networks, fuzzy systems, and knowledge engineering*. Cambridge, MA: MIT Press, 1996.

Part I

SURVIVABLE DWDM OPTICAL MESH NETWORK DESIGN

Chapter 2:

Evolutionary Algorithms

2.1 Introduction

From our everyday experience we know that evidence of intelligent behaviour is easily observed in humans. The fact is we are products of evolution, and thus by modelling the process of evolution, we might expect to create intelligent behaviour. Evolutionary algorithms (EAs) simulate evolution on a computer. The result of such simulation is a series of optimisation algorithms, usually based on a simple set of

rules. Optimisation iteratively improves the quality of solution until an optimal, or at least feasible, solution is found.

In the first part of this thesis, the implementation of an EA for designing a survivable DWDM optical mesh network will be shown. Before going in to the actual designing process, the essential literature background of the EAs is provided in this chapter.

This starts with an introduction of Darwinism in *Section 2.2*, which is the origin of EAs. Two of the main sub areas of EAs, Genetic Algorithms (GAs) and Genetic Programming (GP), will be presented in *Section 2.3* and *Section 2.4* respectively.

2.2 Origin of evolutionary algorithms: Darwinism

On July 1st 1858, a British naturalist, Charles Darwin (1809-1882) presented his theory of evolution before the Linnean Society of London. This day marks the beginning of a revolution in biology, and the theory is considered to be one of the most significant theories in the world. In the theory, Darwin states that variation within species occurs randomly and the survival or extinction of each organism is determined by that organism's ability to adapt to its environment.

In Darwin's book "Origin of Species", he made the following observations about the natural process of evolution [1, 2].

- A huge number of species of organisms are living on the earth. Each species has an enormous number of individuals – the populations.
- Resource in a given environment is limited and so only a limited number of organisms can be accommodated, leading to competition for survival – the selection process.

- Surviving organisms multiply by asexual or sexual reproduction, during which random mutations often occur, and most of the characteristics of the parent(s) are inherited – inheritance with modification.

Algorithm 2.1 Structure of typical EAs

```
    // Start with an initial time
1:  t = 0
    // Initialize a random population of individuals
2:  Initialize P(0)  // P(0) means population pool at time 0
    // Evaluate fitness of all initial individuals of population
3:  Evaluate  P(0)
    // Evolution cycle
4:  REPEAT
5:          t = t + 1  // increase the time counter
            // Select a sub-population for offspring production
6:          P'(t)  ← Selected from  P(t)
7:          P''(t) ← Reproduced from P'(t)
8:          P(t + 1) ← Replace with P''(t) or P(t) ∪ P''(t)
9:  UNTIL termination criteria P(T)
10: RETURN best_solution(s)
```

EAs are techniques inspired by the process of natural evolution, more specifically, by Darwin's theory of evolution by natural selection, including the population pool as inheritance based; natural selection as the key principle of evolution; recombination (crossover); mutation functions as tools for generating the following generation. Briefly, EAs can be summarised using the pseudo code in *Algorithm 2.1*.

2.3 Genetic algorithms

2.3.1 History

Genetic algorithms (GAs) are a class of stochastic search algorithms based on biological evolution [3]. The beginning of GAs can be traced back to the 1950s when several biologists, such as N. A. Barricelli and A.S. Fraser, used computers to simulate biological systems [4]. However, GAs in their usual form are a development of John Holland, a computer scientist and psychologist at the University of Michigan. He summarized his work on adaptive and reproductive plans in 1975 in a book that serves as the starting point of nearly all known applications and implementations of GAs [3]. Holland tried to simulate natural evolution on computers. As a computer scientist, Holland was concerned with algorithms that manipulate strings of binary digits. He viewed these algorithms as an abstract form of natural evolution. Holland's GA can be represented by a sequence of procedural steps for moving from one population of artificial 'chromosomes' to a new population. It uses 'natural' selection and genetics-inspired techniques known as crossover and mutation.

2.3.2 Framework of a basic GA

Nature has an ability to adapt and learn without being told what to do. In other words, nature finds good chromosomes blindly and GAs do the same. GAs employ two mechanisms – encoding and evaluation, to link themselves with the problem they are solving. In a basic GA, encoding is carried out by representing a set of chromosomes for a specific optimisation problem. Each chromosome consists of a number of 'genes', each of which has the value 0 or 1. An evaluation or *fitness* function is used to measure the chromosome's performance, or fitness, for the optimisation problem, and has to be formulated accordingly based on the particular optimisation problem.

The GA uses a measure of fitness of individual chromosomes to carry out reproduction. As reproduction takes place, the crossover operator exchanges parts of two single chromosomes, and the mutation operator changes the gene value in some randomly chosen location of the chromosome. As a result, after a number of successive reproductions, the less fit chromosomes become extinct, while those best able to survive gradually come to dominate the population.

The main framework of a basic GA can be represented as in *Figure 2.1* and applies the following major steps [5-7]:

Step 1: Represent the problem variable domain as a chromosome of a fixed length, choose the size of a chromosome population (N), the crossover probability (ρ_c) and the mutation probability (ρ_m).

Step 2: Define a fitness function to measure the performance, or fitness, of an individual chromosome in the problem domain. The fitness function establishes the basis for selecting chromosomes that will be mated during reproduction.

Step 3: Randomly generate an initial population of chromosomes of size N:

$$x_1, x_2, \cdots, x_N$$

Step 4: Calculate the fitness of each individual chromosome:

$$f(x_1), f(x_2), \cdots, f(x_N)$$

Step 5: Select a pair of chromosomes for mating from the current population. Parent chromosomes are selected with a probability related to their fitness. High fitness chromosomes have a higher probability of being selected for mating than less fit chromosomes.

Step 6: Create a pair of offspring chromosomes by applying the genetic operators – crossover and mutation.

Step 7: Place the created offspring chromosomes in the new population.

Step 8: Repeat *Step 5* until the size of the new chromosome population becomes equal to the size of the initial population, N.

Step 9: Replace the initial (parent) chromosome population with the new (offspring) population.

Step 10: Go to *Step 4*, and repeat the process until the termination criterion is satisfied.

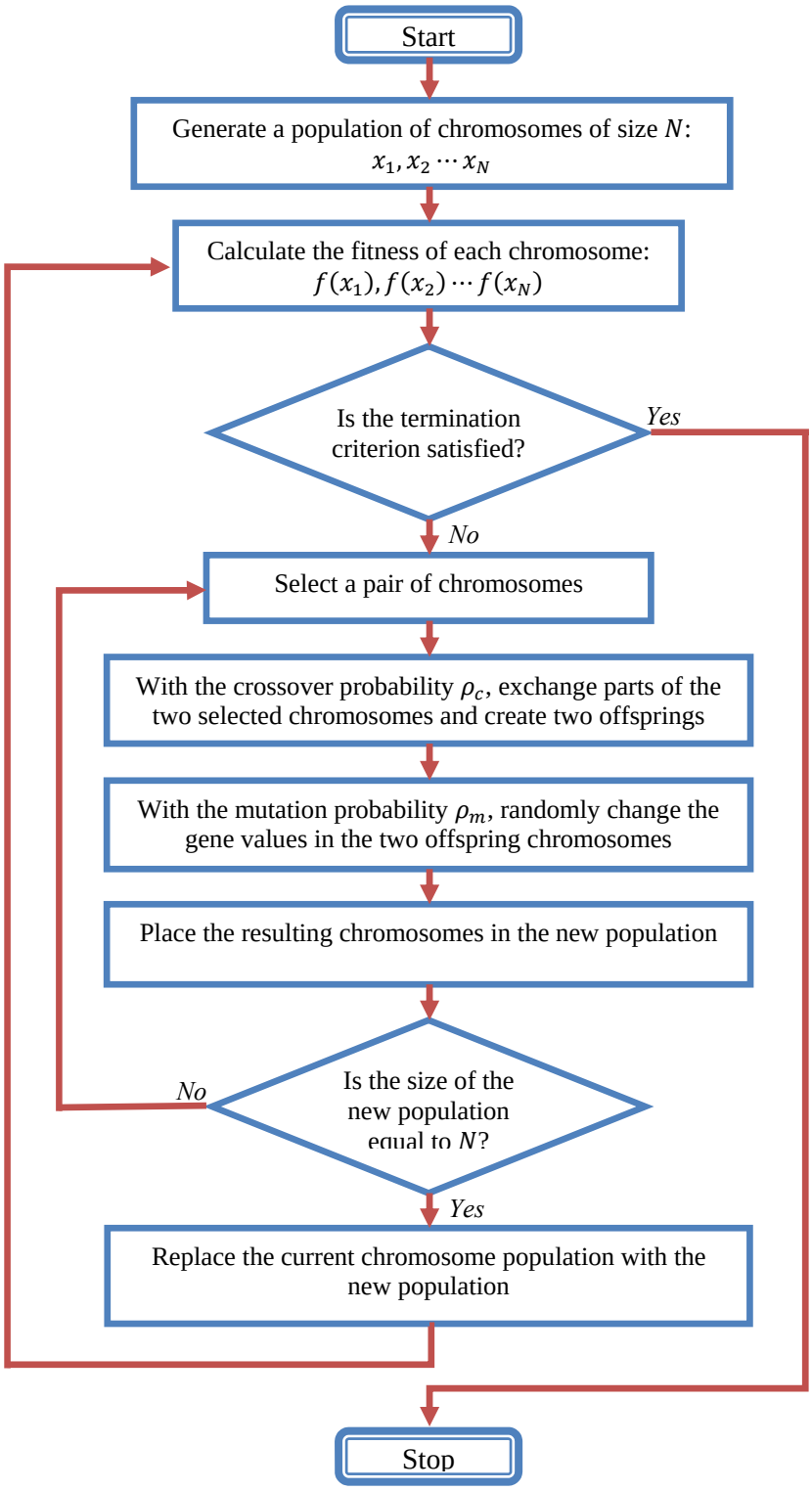

Figure 2.1 Genetic algorithm flowchart

GAs represent an iterative process with each iteration known as a generation. The entire set of generations is called a run. At the end of a run, it is expected that one or more highly fit chromosomes will be found.

2.3.3 Genetic operators

A simple GA that yields good results in many practical problems is composed of three types of operators: selection, crossover and mutation [6]. In this section, these operators are introduced in turn.

2.3.3.1 Selection

Selection is an operator used to direct the search process towards better regions of the search space by giving preference to individuals of higher fitness for crossover and reproduction [8]. Over the years, many selection methods have been proposed, examined, and compared. Here the most widely used selection schemes are considered.

1. Elitist selection [9, 10]: The fittest members of each generation are guaranteed to be selected. However, most GAs do not use pure elitist selection scheme, but instead use a modified form where a single best or a few of the best, individuals from the parent generation are copied to the offspring generation, to avoid the situation of not achieving better individuals.
2. Roulette wheel selection [7]: This scheme is the most commonly used chromosome selection techniques. In this method, each chromosome is given a slice of a circular roulette wheel. The area of the slice within the wheel is equal to the chromosome fitness ratio. To select a chromosome for mating, a random number is generated, and the chromosome whose segment spans the random number is selected.
3. Rank selection [8]: the selection probability of an individual does not depend on its absolute fitness, but only on its relative fitness in comparison with the other population members; namely, its rank when all individuals are ordered in

increasing order of fitness values. The benefit of this method is that it can prevent very fit individuals from gaining dominance early.

4. Tournament selection [11]: A small subset of individuals is picked randomly from the population pool, and the individual with the lowest cost in this subset becomes a parent. The tournament repeats for every parent needed. Tournament selection works best for larger population sizes as sorting becomes time-consuming for large populations.

Each of the selection schemes results in a different set of parents. As such, the composition of the next generation is different for each selection scheme. Roulette wheel and tournament selection are standard for most GAs [11]. It is very difficult to give advice on which selection scheme works best. In this research work the roulette wheel selection procedure is utilised.

2.3.3.2 Crossover

Crossover is the creation of one or more offspring from the parents selected in the pairing process, the most common form of crossover involves two parents that produce two offspring. There are in general three techniques for implementing the crossover operator, *single point crossover*, *multi-points crossover* , and *uniform crossover*. (see *Figure 2.2*)

1. Single point crossover
    a. Select two parents, $PARENT\#1$ and $PARENT\#2$, and then randomly choose a crossover point between the first and last bits of the parents' chromosomes.

b. Spilt the parents' chromosomes at the selected crossover point, $PARENT\#1 = PARENT\#1^A + PARENT\#1^B$ and $PARENT\#2 = PARENT\#2^A + PARENT\#2^B$.

c. Exchange the gene information between the parents to produce two offspring. E.g. $OFFSPRING\#1 = PARENT\#1^A + PARENT\#2^B$ and $OFFSPRING\#2 = PARENT\#2^A + PARENT\#1^B$.

2. Multi-point crossover

   a. Select two parents and randomly choose N crossover points between the first and last bits of the parents' chromosome.

   b. Split the parents' chromosomes at the selected crossover points, like the single point crossover does.

   c. Offspring are produced by exchanging the gene information between parents.

3. Uniform crossover

   a. Select two parents, $PARENT\#1$ and $PARENT\#2$. For each gene of the parents' chromosomes, randomly allocate a binary number, $Random$.

   b. If $Random = 1$, copy current gene information from $PARENT\#1$ to $OFFSPRING\#1$; $PARENT\#2$ to $OFFSPRING\#2$.

   c. Otherwise if $Random = 0$, copy current gene information from $PARENT\#1$ to $OFFSPRING\#2$; $PARENT\#2$ to $OFFSPRING\#1$.

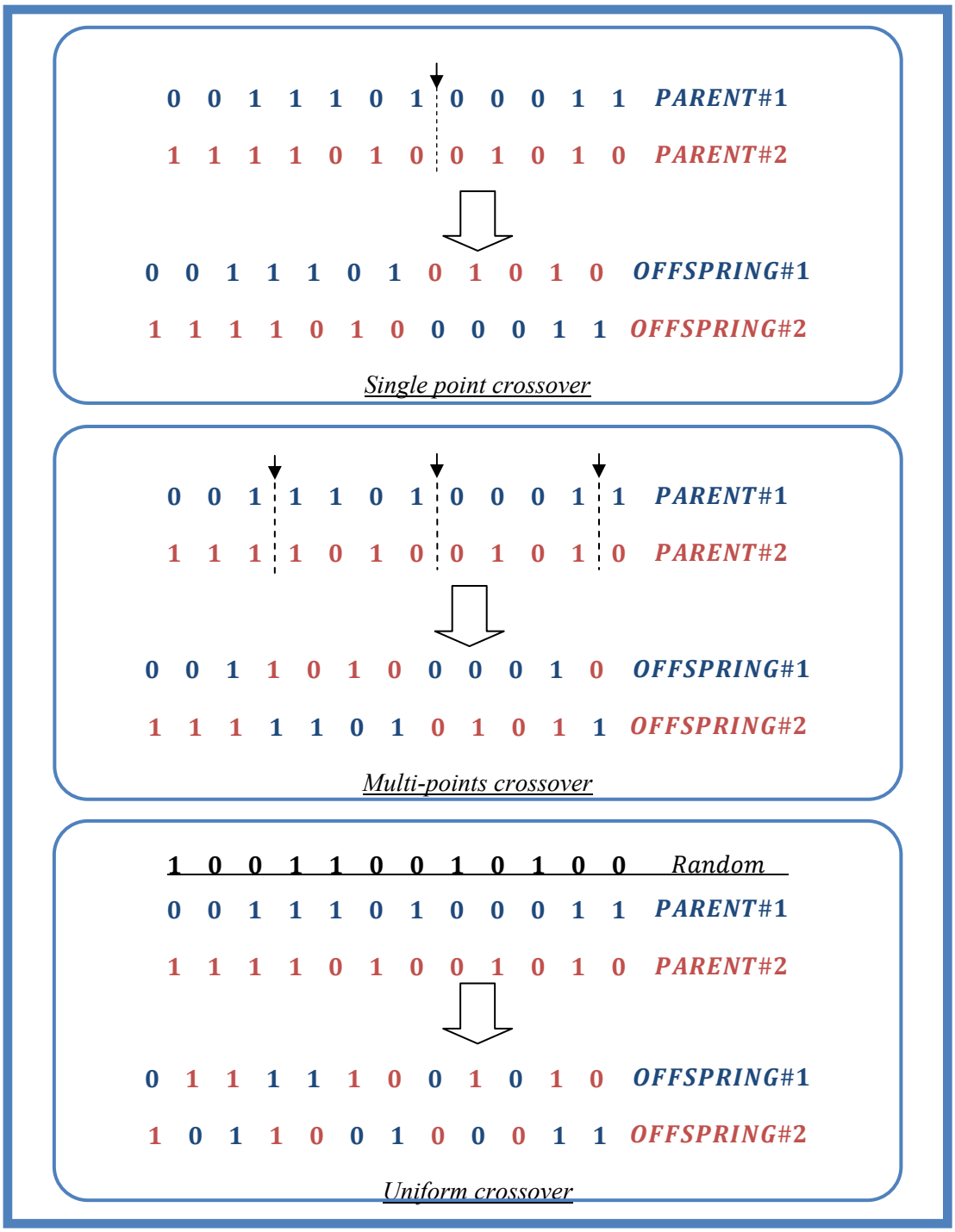

Figure 2.2 Illustration of crossover operation

2.3.3.3 Mutation

Random mutations alter a certain percentage of the bits in the list of chromosomes. Mutation is the second way a GA explores a cost surface. It can introduce traits not in the original population and keeps the GA from converging too fast before sampling the entire cost surface [11]. Mutation can be either single gene mutation or multi gene mutation as shown in *Figure 2.3*. Mutation points are randomly selected from the $N_{pop} \times N_{bits}$ total number of bits in the population matrix. At each mutation point, the mutation operation changes a 1 to a 0, and vice versa.

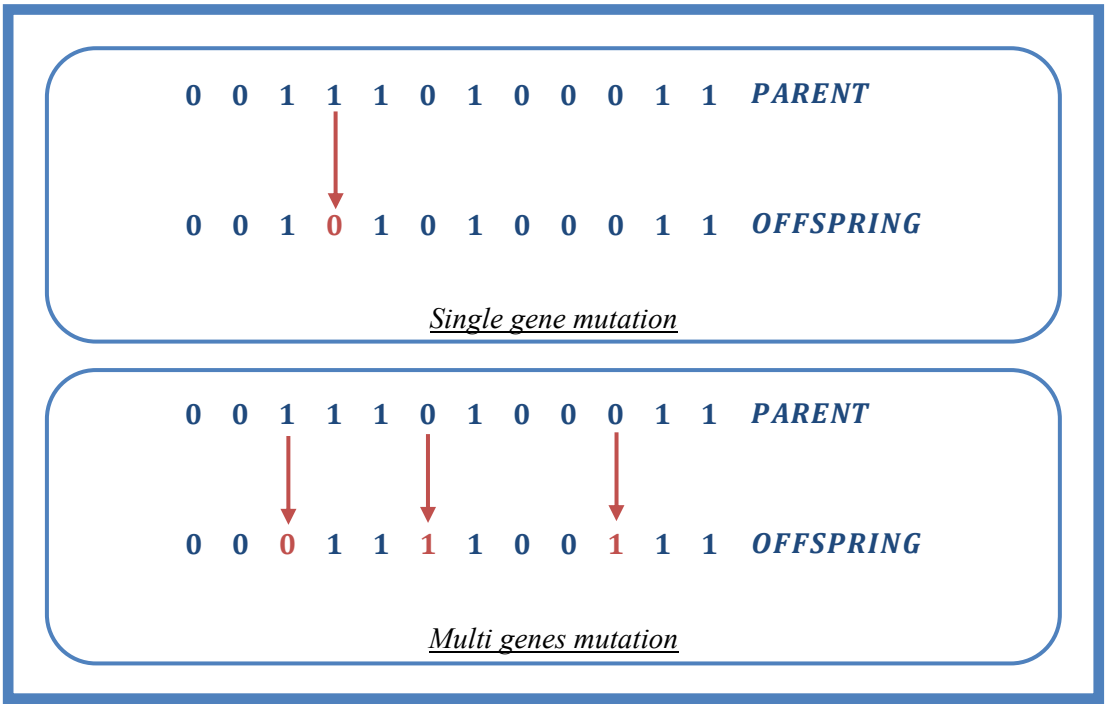

Figure 2.3 Illustration of mutation operation

The settings of the GA parameters are usually necessary to be adjusted upon optimisation problems. These settings affect the performance of the GA optimisation. Researchers have, however, suggested some typical values for GAs, an implementation guideline, which has been summarised by Rahmat-Samii et al [12, 13], is shown in *Figure 2.4*. These suggested GA settings are followed in this research work presented in *Chapter 3*.

GA Optimisation Implementation Guideline

Genetic algorithm (GA) optimisers generally require the user to select among a number of options before beginning the optimisation. The options allow the user to tailor the optimisation to better fit the optimisation job at hand. Aside from the choices of things such as selection type, encoding/decoding scheme, and details of the fitness function, the GA optimizer user must choose the population size, the probability of crossover, and the probability of mutation. If a binary encoding scheme is used, the user must decide how many bits a given parameter is allotted in the chromosome. These choices are summarized below, with some guidelines and comments.

Population Size: Typically 30-100

Comments: Larger populations provide more genetic diversity, enabling faster convergence. Smaller populations yield faster execution, especially when dealing with complicated fitness functions.

Probability of Crossover: Typically 0.6-0.9

Comment: A probability of crossover around 0.7 has been found to be optimal for a wide variety of problems. Generally speaking, crossover is the primary way a GA optimizer searches for new, better solutions. High crossover probability ensures rapid searching.

Probability of Mutation: Typically 0.01-0.1

Comments: The probability of mutation should generally be low. Mutation introduces new genetic material into the search process, but also tends to push the population's average fitness away from the optimal value. Some workers report the probability of mutation at the bit or element level, while others report it at the chromosome level, as is usually done for the probability of crossover.

Figure 2.4 GAs optimisation implementation guidelines

2.3.4 Advantage of GA

The main differences between GAs and conventional optimisation techniques are summarised by Goldberg [4] as follows:

1. GAs work with a coding of the solution set, not the solutions themselves.
2. GAs search from a population of solutions, not a single solution.
3. GAs use payoff information (fitness function), not derivatives or other auxiliary knowledge.

4. GAs use probabilistic transition rules, not deterministic rules.

According to the above differences, the strengths of GAs can be further extended [9]:

1. GAs are intrinsically parallel.

   Most other algorithms are serial, and can only explore the solution space to a problem at a time. Once the solution they discover is suboptimal, there is nothing to do but to abandon all the previous work and start all over again. However, since GAs have multiple offspring (chromosomes in a population pool) that can explore the solution space in multiple directions simultaneously. If one path turns out to be dead end, the system can easily eliminate it and continue to work on others.
2. GAs perform well on problems for which the fitness landscape is complex.

   Most practical problems have a vast solution space, which is impossible to search exhaustively. The problem that then arises is how to avoid local optima. Many search algorithms can become trapped by local optima: if they reach the top of a hill on the fitness landscape, they will discover that no better solutions exist nearby, and conclude that they have reached the best solution, even though higher peaks exist somewhere else on the map.
3. GAs can manipulate many parameters simultaneously [14].

   Many real world problems cannot be determined in terms of a minimised or maximised single value. Most of the time, they must be expressed in terms of multiple objectives, usually with tradeoffs involved. GAs are good at solving such problems: in particular, their use of parallelism enables them to produce multiple, equally-good solutions.

4. GAs can be deployed to solve problems in the environment of knowing nothing.

   Instead of using previously-known information to guide each step and making changes with a specific rule towards improvement, GAs make random changes to their individual solution, and then use the fitness function to determine whether those changes produce an improvement.

2.4 Genetic programming

GAs use fixed-length strings and this representation sometimes becomes the bottleneck of particular types of problems. To overcome the limitation of GAs, another branch of EAs, Genetic Programming (GP) is also studied here. The aim is to develop a novel version of a GA that combines advantages of both conventional GAs and GP. It may be said that GP is in fact an extension of the conventional GA, also following Darwin's principle of differential natural selection. The major distinction of GP is that it controls active components such as symbolic expressions (S-expressions) and is able to develop its own representation of a problem by allowing variable complexity of its individuals [15].

GP was greatly stimulated by John Koza in the 1990s. In Koza's book [16], GP is described as: 'The individual structures that undergo adaptation in genetic programming are hierarchically structured computer programs; the size, the shape, and the contents of these computer programs can dynamically change during the evolution process.'

2.4.1 Framework of GP

GP searches the space of possible computer programs for a program that is highly fit for solving the optimisation problem at hand [16]. The main programming language for GP is *List Processor* or (LISP), as it has the symbol-oriented structure that allows GP to manipulate by applying the genetic operators. In LISP, all data and all programs are called symbolic expressions or S-expression, and can be depicted as a rooted point-labelled tree with ordered branches. *Figure 2.5* is an example of the tree structure corresponding to the S-expression $(-(*\,A\,B)C)$. This tree has five components, each of which represents either a function or a terminal. The two internal 'pink' components of the tree are functions with labels $(-)$ and $(*)$, and the three external 'blue' components of the tree, also called leaves, are terminals with the label A, B and C.

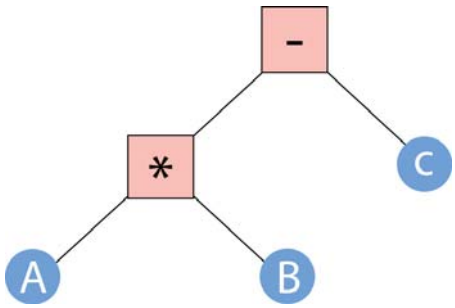

Figure 2.5 Graphical representation of the LISP S-expression $(-(*\,\boldsymbol{AB})\boldsymbol{C})$

Before applying GP to a problem, the following five preparatory steps have to be accomplished [7, 17].

1. Determine the set of terminals.
2. Select the set of primitive functions.
3. Define the fitness function.
4. Decide on the parameters for controlling the run.

5. Choose the method for designating a result of the run.

Once these five steps are complete, a run can be made. The GP run starts with a random generation of an initial population of computer programs, GP then creates computer programs by executing the following steps [7, 17]:

Step 1: Assign the maximum number of generations to be run and probabilities for cloning (ρ_r), crossover (ρ_c) and mutation (ρ_m) with $\rho_r + \rho_c + \rho_m = 1$.

Step 2: Generate an initial population of computer programs of size N by combining randomly selected functions and terminals.

Step 3: Execute each computer program in the population and calculate its fitness with an appropriate fitness function. Designate the best-so-far individual as the result of the run.

Step 4: With the assigned probabilities, select a genetic operator to perform cloning, crossover or mutation.

Step 5: If the cloning operation is chosen, select one computer program from the current population of programs and copy it into a new population.

If the crossover operator is chosen, select a pair of computer programs from the current population, create a pair of offspring programs and place them into the new population.

If the mutation operator is chosen, select one computer program from the current population, perform mutation and place the mutant into the new population.

All programs are selected with a probability based on their fitness. i.e., the higher the fitness, the more likely the program is to be selected.

Step 6: Repeat *Step 4* until the size of the new population of computer programs becomes equal to the size of the initial population, N.

Step 7: Replace the current (parent) population with the new (offspring) population.

Step 8: Go to *Step 3* and repeat the process until the termination criterion is satisfied.

Figure 2.6 [7] is a flowchart representing the above GP steps.

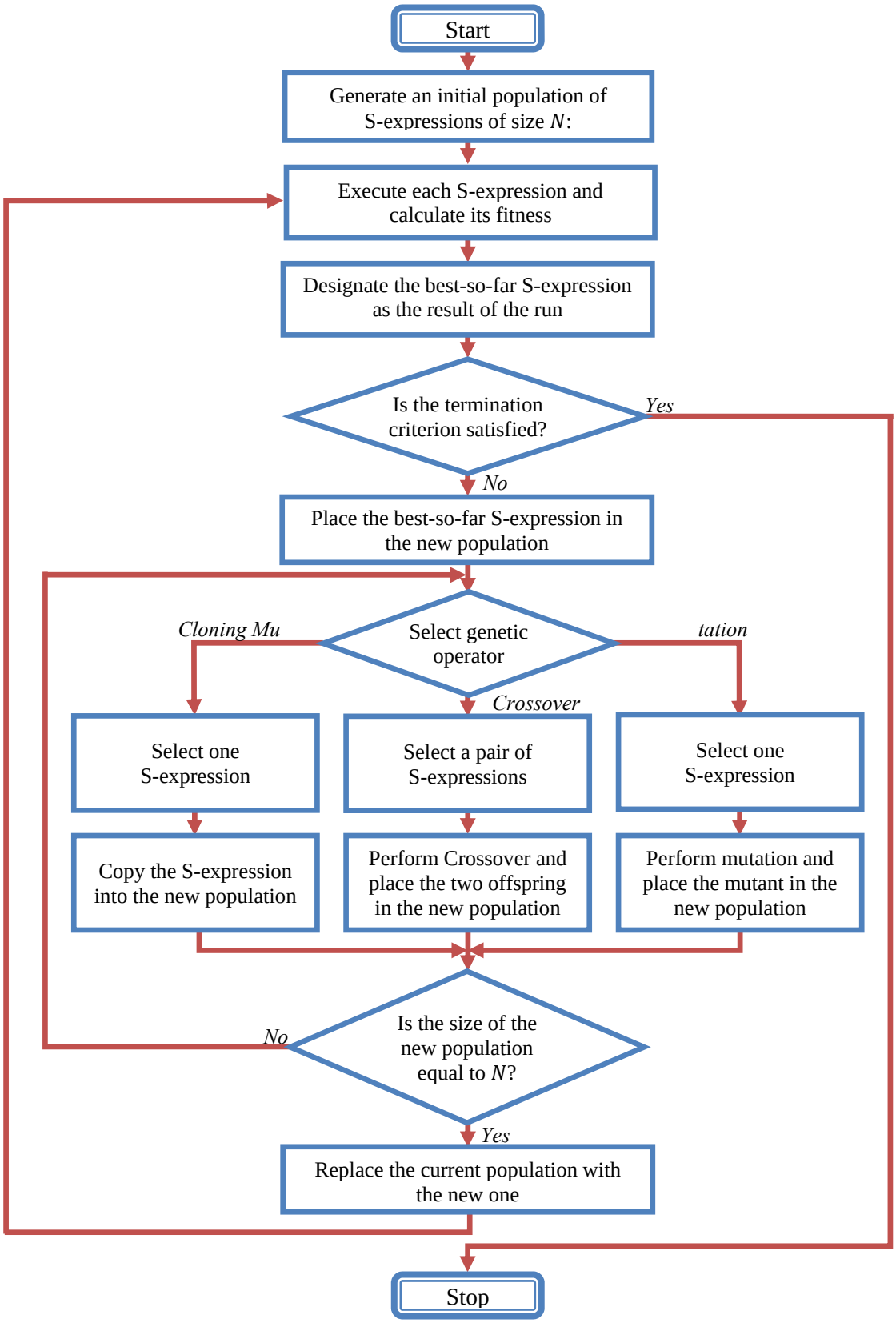

Figure 2.6 Flowchart for genetic programming

2.4.2 Genetic operators

GP inherited most of the features of GAs, including the selection, crossover and mutation operators. Compared to GAs, a slight difference is that GP uses tree structured LISP S-expressions instead of bit strings chromosomes in representing individual solutions. *Figure 2.7* and *Figure 2.8* show illustrations of the operations of GP crossover and mutation.

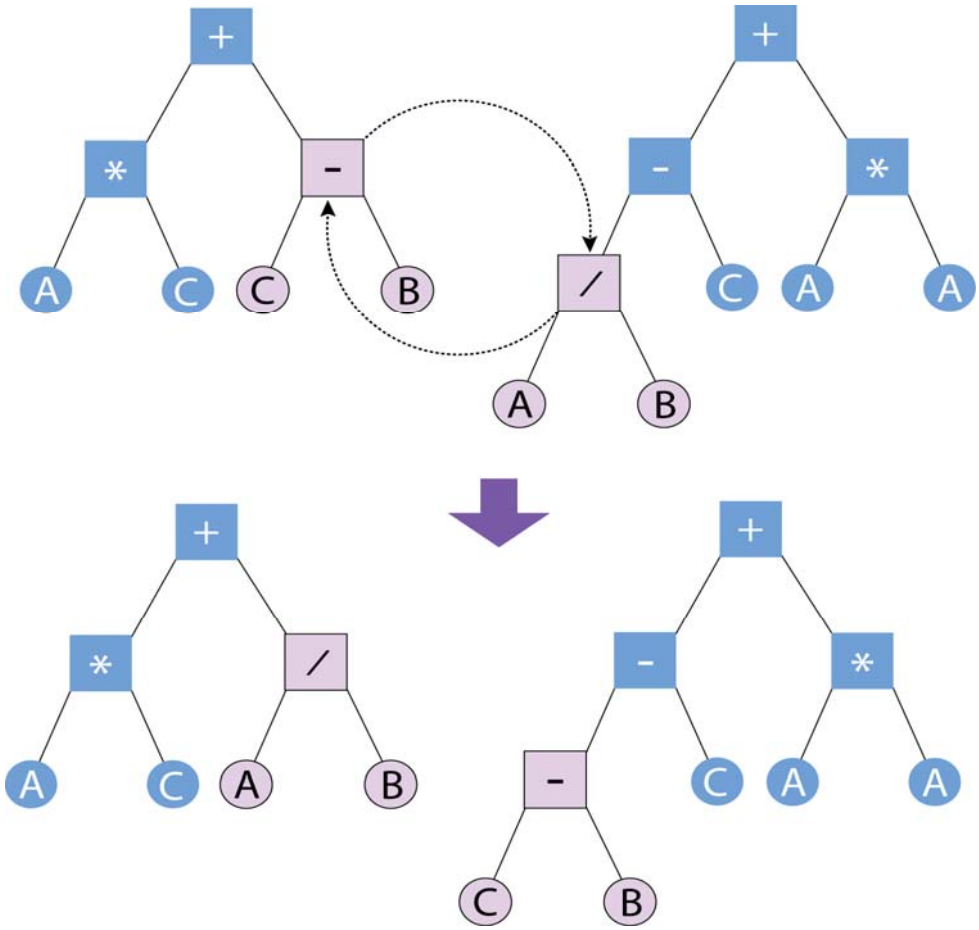

Figure 2.7 Crossover in genetic programming

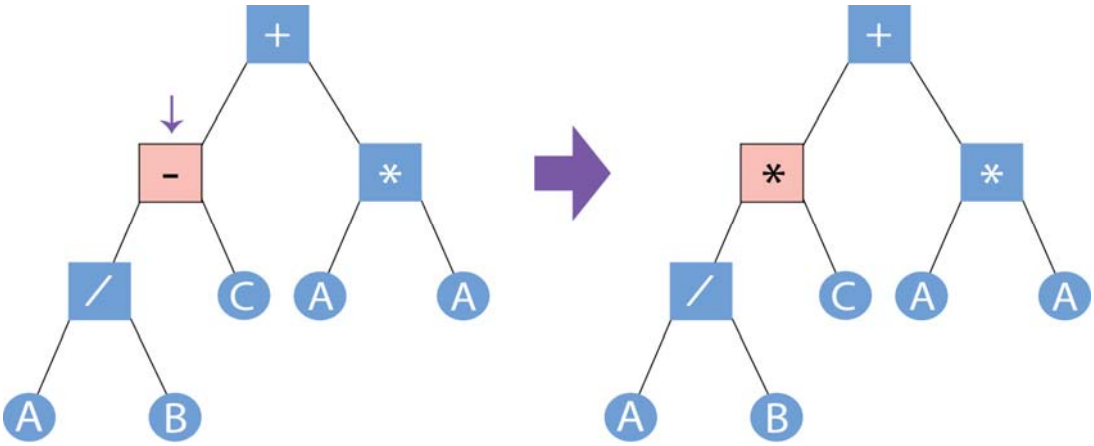

Figure 2.8 Mutation in genetic programming

2.4.3 Comparison between GP and GA

The conventional GA represents the problem solutions by encoding fixed-length bit strings. Using fixed-length coding, problem representation becomes a fundamental difficulty of GAs. A poor representation limits the power of GAs, and may lead to a false solution. Moreover, fixed-length coding often causes considerable redundancy and reduces the efficiency of genetic search. In contrast, GP uses high-level building blocks of variable length. Their size and complexity can change during breeding. GP works well in a large number of different cases [17] and has many potential applications. However, on the other hand, the structure of GP is often more complex than GA, and thus extensive computer run times may often be needed.

In this research work, the advantage of algorithm simplicity in conventional GAs and the power of problem representing in GP are mixed to produce a GA version to aid in the routing design of survivable DWDM optical mesh networks. Considering the characteristic of our problem, it is not necessary to represent optical network communication routing paths by using a tree structure, as this representation will make the algorithm much more complex and computational inefficient. Instead, a set

of two bits binary strings to represent routing solutions are designed and chromosomes formed which can be manipulated easily by GAs. In addition, the crossover and mutation operators that are applied to the chromosomes representation produce valid offspring regardless of the choice of crossover or mutation points.

2.5 Summary

In this chapter, an overview of evolutionary algorithms has been presented. Two major sub-areas of EAs were considered, namely GAs and GPs. The main steps in developing a GA were introduced, with discussion of the different implementation of the genetic operators and the advantages of GAs compared to the conventional optimisation techniques. The basic concepts of GP were provided. The use of GP overcomes the GA limitation in fixed-length coding, which results in a wider representation of different problems. However, extensive computer run times may be needed due to its complex data structures.

After providing an overview of the working process and theory of GAs and GP, original work on survivable DWDM optical mesh network design will be presented. In this work, a novel GA was developed with advantages extracted from both GA and GP. The GA is particularly aimed at the problem of survivable routing design in DWDM optical networks. The detailed description of the problem and the research work will be presented in the next chapter.

References

[1] J. Hu, "Sustainable evolutionary algorithms and scalable evolutionary synthesis of dynamic systems," in *Department of Computer Science and Engineering*: Michigan State University, 2004.

[2] C. Darwin, *The origin of species*. London: John Murray, 1859.

[3] J. H. Holland, *Adaptation in Natural and Artificial Systems* . Ann Arbor, MI: University of Michigan Press, 1975.

[4] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*: Addison-Wesley publishing company, 1989.

[5] L. Davis, *Handbook of Genetic Algorithms* . New York: Van Nostrand Reinhold, 1991.

[6] M. Mitchell, *An Introduction to Genetic Algorithms* . Cambridge, MA: MIT Press, 1996.

[7] M. Negnevitsky, *Artificial Intelligence: A Guide to Intelligent Systems* , 2nd ed.: Addison-Wesley, 2005.

[8] T. Back, D. B. Fogel, and T. Michalewicz, *Evolutionary Computation 1: Basic Algorithms and Operators*: IOP Publishing Ltd, 2000.

[9] L. Sun, "Application of evolutionary algorithms to microwave lens shape design," in *School of Engineering*. vol. PhD Coventry: University of Warwick, September 2006.

[10] D. Thierens and D. E. Goldberg, "Convergence models of genetic algorithm selection sehemes," in *Proceedings of the International Conference on Evolutionary Computation*, 1994.

[11] R. L. Haupt and S. E. Haupt, *Practical Genetic Algorithms*, Second ed. Hoboken, New Jersey: John Wiley & Sons, Inc., 2004.

[12] Y. Rahmat-Samii and E. Michielssen, *Electromagnetic optimization by genetic algorithms*: John Wiley & Sons, 1999.

[13] J. M. Johnson and Y. Rahmat-Samii, "Genetic algorithms in Engineering Electromagnetic," *IEEE Antennas and Propagation Magazine,* vol. 39, pp. 7-21, 1997.

[14] S. Forrest, "Genetic algorithms: principles of natural selection applied to comptation," *Science, New series,* vol. 261, pp. 872-878, Aug13, 1993.

[15] W. Banzhaf and J. Miller, "The challenge of complexity," in *Frontiers of evolutionary computation*, A. Menon, Ed.: Kluwer Academic Publishers, 2004.

[16] J. R. Koza, *Genetic Programming: On the programming of Computers by Means of Natural Selection*. Cambridge, MA: MIT Press, 1992.

[17] J. R. Koza, *Genetic Programming II: Automatic Discov ery of Reusable Programs*. Cambridge, MA: MIT Press, 1994.

Chapter 3:

Design of Survivable DWDM Optical Mesh Transport Networks

3.1 Introduction

Optical networking is arguably one of the hottest current topics in the telecommunications world. With this technology, huge amounts of data can be transported around the world within a single strand of glass fibre, thinner than a human hair. In the 1980's it was possible to transmit 140 Mbit/s with optical plesiochronous digital hierarchy (PDH) systems. Synchronous digital hierarchy (SDH)

technology in the 1990's has improved this capacity. SDH can transmit the capacity of 16 times 140 Mbit/s or 155 Mbit/s (16 × STM1=STM16, 2.5 Gbit/s) or up to 64 times 140 Mbit/s or 155 Mbit/s (64 × STM1=STM64, 10 Gbit/s) and is on its way to 40 Gbit/s transmission [1]. Nevertheless, even this capacity does not seem adequate for today's needs sing growing data traffic, especially the Internet is an unprecedented driving factor. To meet this demand, new technologies had to be developed. The one getting the most attention is Dense Wavelength Division Multiplexing (DWDM). This multi-wavelength technique opens up new horizons in terms of transmission capacity, as well as providing some more advantages at the same time. With DWDM systems, it is possible to transmit between about 32 and 160 times 10 Gbit/s over very large distances. This is equal to a total capacity of 1.6Tbit/s; in the laboratory it is possible to transmit 10 Tbit/s and more e.g. in 40 Gbit/s channels [1].

Survivability of optical connections is a major issue in the design of next generation DWDM networks. This is because the interruption of such high-speed optical connections, even for few seconds, results in a very large loss of information. Failures in an optical network can be distinguished depending whether they damage links or switching devices. In the past, failures were manually solved by temporarily re-routing the broken connections and sending teams of workers to repair the damaged equipment in place. Nowadays, optical networks that still require manual re-routing can be considered as unprotected. The outage periods due to traffic recovery based on the human intervention are unacceptable. At present no optical network operator is willing to accept unprotected facilities: survivability must be always guaranteed by adopting efficient automatic recovery from failures, that is to say re-routing broken connections automatically [2].

In this chapter a GA based approach for survivable DWDM optical mesh transport network design is presented. This approach has been evaluated on the '*Dedicated Path Protection*' (DPP) and the '*Shared Path Protection*' (SPP) schemes.

The chapter is organised as follows. In *Section 3.2* , the fundamental literature background on the optical networks is reviewed. This includes a brief introduction to DWDM optical mesh transport networks, an explanation of the data protection survivability architectures, and an overview of two specific optical network survivability schemes, namely the DPP scheme and the SPP scheme. One of the biggest challenges in designing survivable optical networks is the routing and wavelength assignment (RWA) problem. In *Section 3.3* , the RWA problems are passed to a variable-length GA solver for the optimal solutions. The implementation process will be explained in detail in this section. In *Section 3.4* and *Section 3.5* , the design of experiments, parameter settings and the simulation results will be presented and discussed. The chapter ends with summaries of the whole chapter and conclusions.

3.2 Basics of survivable DWDM optical mesh networks

The ever increasing demand for bandwidth is posing new challenges for transport network providers. A viable solution to meet this challenge is to use optical networks based on WDM technology.

Optical fibre offers much higher bandwidth than copper cables and is less susceptible to various kinds of electromagnetic interferences and other undesirable effects. As a result, it is considered to be an almost perfect transmission medium and has been firmly established as the medium of choice for broadband, wired networking for almost all network sizes. Optical glass fibres based on the principle of *total internal*

reflection, which was well known in the 1850s, were developed for endoscopes early in the 1900s. The use of fibre glass for communication was first proposed in 1966. In the late 1980s or so, a variety of optical networks came into existence replacing the traditional copper cable to achieve higher speeds [3].

The objective of this section is to present a broad overview of the optical network, provide the essential background knowledge in developing survivable routing schemes for the next generation DWDM optical networks, and also to review the recent works within this research area.

3.2.1 Optical transmission system

An optical transmission system has three basic components – transmitter, transmission medium, and receiver (as shown in *Figure 3.1*). The transmitter consists of a light source that can be modulated according to an electrical input signal to produce a beam of light which is transmitted into the optical fibre – the transmission medium. Typically the binary information sequence is converted into a sequence of on/off light pulses which are then transmitted into the optical fibre medium. At the receiver, the on/off light pulses are converted back to electrical signals by an optical detector. Thus this forms a unidirectional transmission system which accepts an electrical signal, converts and transmits it using light pulses through the medium, and then reconverts the light pulses to an electrical signal at the receiving end [3]. During the signal propagation, problems such as signal attenuation, sometimes occurs. However, these problems belong to optical physical layer problems and are out of the scope of discussion in this research work.

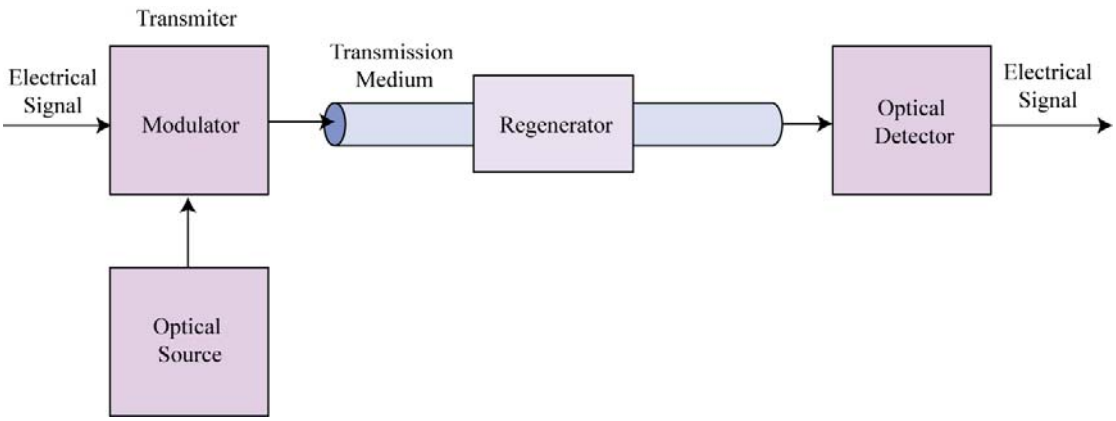

Figure 3.1 Optical transmission system

3.2.2 Basis of DWDM

Multiplexing is a process where multiple analogue message signals or digital data streams are combined into one signal over a shared medium. The need for multiplexing is driven by the fact that it is much more economical to transmit data at higher rates over a single fibre than it is to transmit at lower rates over multiple fibres, in most applications. There are fundamentally two ways of increasing the transmission capacity of a fibre, time division multiplexing (TDM) and frequency division multiplexing (FDM).

With TDM each user is assigned to a certain time slot thus making the transmission time shared by all the users. That was done by, for example transmitting the first bit of the first channel, then the first bit of the second channel, the first bit of the third and so on until the process starts all over again with the second bit of the first channel. This technique is widely used in both PDH and SDH transmission systems. With FDM each user is assigned to a certain frequency slot transmitting only with the corresponding carrier, in this way sharing the available bandwidth. Wavelength division multiplexing (WDM) is essentially the same as FDM, except WDM is used in the context of optical communication. The wavelength and frequency are related by

$\lambda = c/f$, where c denotes the speed of light in free space, which is 2.997245×10^6m/s.

However, in the modern context, if the term 'WDM' is used it usually means DWDM. Although DWDM and WDM are in principle the same thing, the distinction between them is the wavelength spacing. With old fashioned WDM, the spacing between channels can be relatively large, in the range of several tens of nanometres, whereas in DWDM, the frequency spacing between channels can be narrowed to as small as ~50 GHz, corresponding to just 0.4 nm. DWDM technology increases the overall spectral density of the transmitted signal [1].

3.2.3 Network survivability

Network survivability is the ability of a telecommunication network to continue to provide service in the event of failures. Integration of survivability into the network core at the optical layer to design fault tolerant networks is an extremely complicated problem which is known to be NP-hard [4]. Before going into the detailed DWDM network survivability routing design, a short introduction to network survivability techniques is provided in this section.

A fibre-optic based telecommunication network is constructed as the multi-layered node architecture as shown in *Figure 3.2* [5]. With this architecture, different layers of equipment are required to handle electrical and optical domain multiplexing, and the conversion between the electrical and optical domains. The fibre link between two cross-connects (OXC) comprises multiple elements such as multiplexers, amplifiers, and regenerators. A physical fibre cut may affect different equipment at the same time and may trigger near simultaneous restoration efforts at multiple layers. Although higher protocol layers, such as asynchronous transfer mode (ATM) and Internet

protocol (IP), have recovery procedures to recover from link failures, the recovery time is still significantly large (on the order of seconds), whereas it is expected that restoration times at the optical layer will be on the order of a few milliseconds to minimize data losses [6]. Furthermore, it is beneficial to consider restoration mechanisms in the optical layer due to reasons such as shorter restoration time, efficient resource utilisation, and protocol transparency, over that at the higher layer protocols.

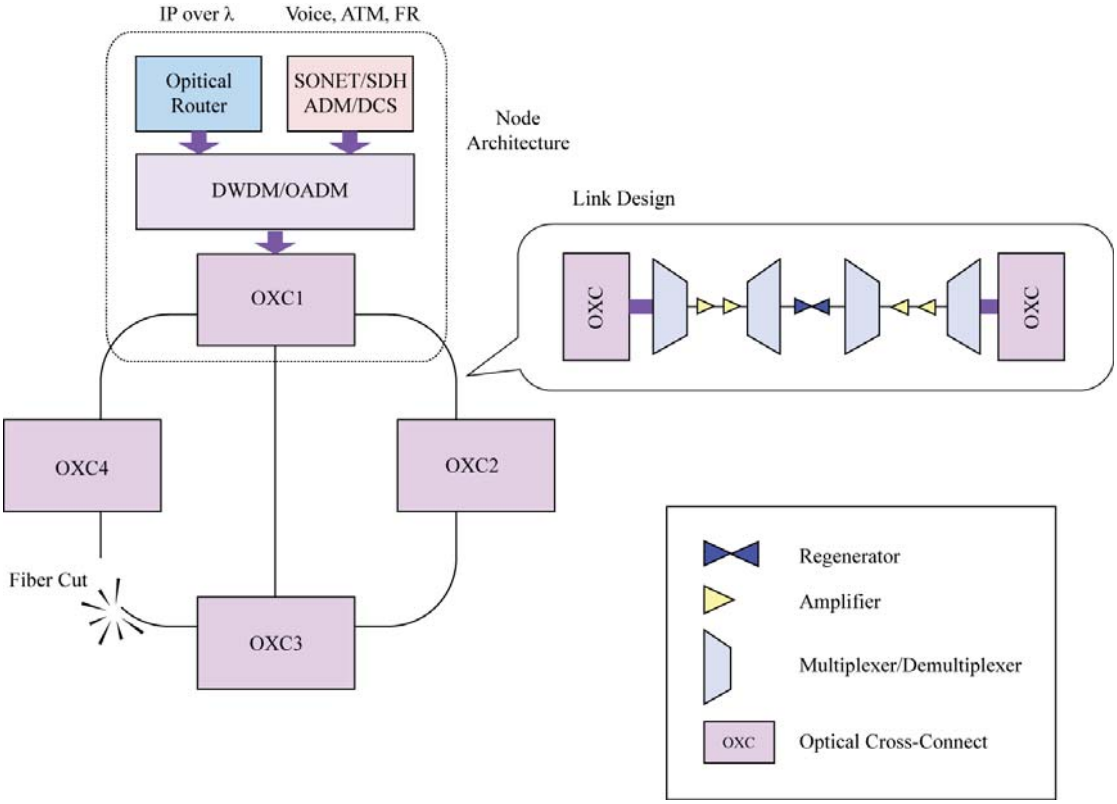

Figure 3.2 Multi-layered architecture for optical telecommunication network

Two architecture options provided the basis of the different approaches to the design of survivable optical mesh networks [4]. If backup resources are pre-computed and reserved in advance, it is called a *protection* scheme. Otherwise, when a failure occurs, if another route has to be discovered dynamically for each interrupted connection, it is referred to as a *restoration* scheme. Generally, dynamic restoration schemes are more efficient in utilizing network capacity because they do not allocate spare capacity in

advance; but protection schemes have faster recovery time and can guarantee recovery from service disruptions against which they are designed to protect [7].

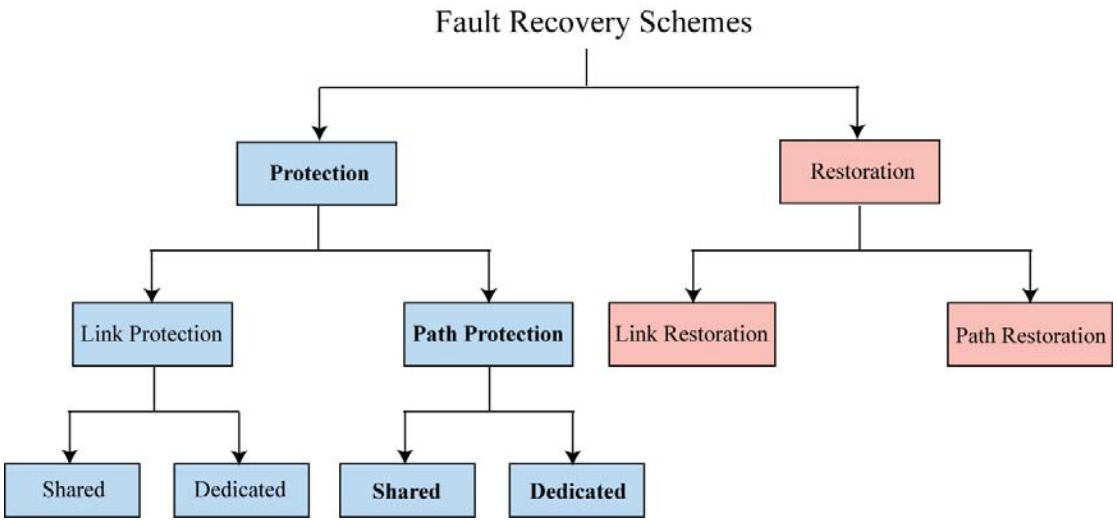

Figure 3.3 Different protection and restoration schemes in DWDM mesh networks

The fault recovery schemes can be further classified as shown in *Figure 3.3*. In *path protection* scheme, the backup lightpaths for every link failure are reserved at connection setup, and have to be disjoint with the failed link. Upon link failure, the wavelength paths reserved for this failure scenario are activated. In the *link survivability* scheme, the traffic is re-routed only around the failed link. While *path protection* leads to efficient utilisation of backup resources and lower end-to-end propagation delay for the recovered route, *link protection* provides shorter protection switching time [7]. Link and path protection schemes can be *dedicated* or *shared*. In dedicated protection there is no sharing between backup resources, while in the shared protection backup wavelengths can be shared on some links as long as their protected segments (links, paths) are mutually diverse [8]. Optical cross connects (OXCs) on backup paths cannot be configured until the failure occurs if shared protection is used. Thus, recovery time in shared protection is longer but its resource utilisation is better than dedicated protection [7].

3.2.4 Recent research

The design of fault tolerant optical core networks with protection and restoration schemes has received considerable attention in recent years. In [9], the design principles and state of the art of survivable WDM networks have been presented. The basic concepts, architectures, models and mechanisms of fault management in survivable optical networks were addressed by Park [10].

The integer linear programming (ILP) approach has been extensively employed for modelling different survivability schemes. A survey of this approach has been provided by Kennington *et al* [5]. The ILP approach works well for problems with linear objective functions and constraints [4]. However, in real life situations, when constraints or objective functions are not linear, this technique cannot be used.

The limitations of ILP models have led to the introduction of diverse heuristic algorithms. Zang *et al* addressed the RWA problem in a network with path protection under duct-layer constraints [11]. Liu *et al* focused on the routing of backup paths and spare capacity allocation in the network to guarantee seamless communication services under failure [12]. Yang *et al* presented resource efficient provisioning solutions in WDM mesh networks [13], which achieved the objective of maximizing resource sharing. Shenai *et al* presented hybrid survivability approaches for optical WDM mesh networks [14]. Zhang *et al* presented a heuristic algorithm for the design of survivable WDM networks [15]. The problem of survivable lightpath provisioning using shared path protection in optical mesh networks employing WDM was investigated by Ou *et al* [16]. In addition, Ho *et al* has considered mesh WDM optical networks with partial wavelength conversion capability [8, 17].

3.3 GA based survivable DWDM optical network design

In this section, the design of a survivable DWDM optical network using a GA will be described. The reasons and benefits for employing GA to this problem can be summarized as follows,

1. Design of survivable DWDM optical transport networks is an extremely complicated process that is considered to be an NP-hard problem.
2. In this particular optimisation problem, it is only necessary to perform offline optimisation before the network starts to operate. Moreover, the network topology is static and the full map of the network can be accessed by GA. These characteristics minimize the side effects of GA, such as high cost in computation power and time.
3. The GA can overcome the limitation that the traditional ILP approach experiences. GAs can solve the problem under nonlinear constraints and objective functions, and the quality of service (QoS) criteria can also be included in designing the survivable DWDM optical transport networks.
4. A number of successful applications that employ GA for designing communication networks [18-24] and solving NP-hard problems [25] give enhanced confidence in applying GA to our research.

Furthermore the performance of the descent algorithms, such as tabu search, simulated annealing, and GA, have been compared by Drenzer et al. in a network design problem [26]. Under the conditions of realistic network size, the GA outperformed all other algorithms in finding the best quality solution [27].

3.3.1 Mathematical representation

The RWA problem is to determine the wavelength routed working and spare paths for a given demand matrix, while minimising the number of wavelength channels used. It is assumed that the network topology is represented as a directed graph $G(N, L)$, where $N = \{n_1, n_2, \cdots, n_N\}$ is the set of nodes and $L = \{l_1, l_2, \cdots, l_M\}$ is the set of connecting links in the network. The demand matrix $D\left[d_{(o,d)}\right]_{M\times N}$ aggregates the demand between origin and destination node pairs (o, d). It is given in terms of the number of wavelengths, taken for all links from the set $\Psi = \{\lambda_1, \lambda_2, \cdots, \lambda_W\}$, required for each connection, $d_{(o,d)}$. The sets of eligible working paths, $K^w_{(o,d)}$, and spare paths, $K^s_{(o,d)}$, between each node pair before and after the event of failure are pre-computed using the K-shortest paths algorithm given in *Section 3.3.1.3*.

3.3.1.1 Objective function

The amount of primary capacity (number of wavelengths required) allocated to working (spare) lightpaths is denoted by $f_l^w(f_l^s)$ for a link l. The minimisation of the wavelengths utilized by working and spare lightpaths to service a given demand matrix may be written as

$$f_C = min\left\{\sum_{l=1}^{M}(f_l^w + f_l^s)\right\} \qquad \textit{Formula 3.1}$$

$$f_l^w = \sum_{(o,d)}\sum_{P_l^{w,k}}\sum_{w} w_w^{k,od}, \forall l \in L, \forall D \qquad \textit{Formula 3.2}$$

$$f_l^s = \sum_{(o,d)}\sum_{P_l^{s,k}}\sum_{w} s_w^{k,od}, \forall l \in L, \forall D \qquad \textit{Formula 3.3}$$

link l is traversed by a set of kth working (spare) paths $P_l^{w,k}(P_l^{s,k})$. The indicator $w_w^{k,od}(s_w^{k,od})$ is set to 1 if the kth working (spare) path between node pair (o,d) uses wavelength w, and to 0 otherwise.

3.3.1.2 Constraints

1. *The link-capacity constraint.* The total number of occupied wavelengths, working and spare, on each link is bounded by number of wavelengths per link W and the number of wavelengths per channel, w_{ch}.

$$f_l^w + f_l^s < W, \forall l \in L \qquad \textit{Formula 3.4}$$

2. *The satisfaction constraint.* Each link of the primary (working) and secondary (spare) paths that is assigned for a connection request between each node pair (o,d) must satisfy the demand between that node pair.

$$\sum_{w=1}^{W} w_w^{k,od} = d_{(o,d)}, \forall l \in L, \forall (o,d) \in D \qquad \textit{Formula 3.5}$$

$$\sum_{w=1}^{W} s_w^{k,od} = d_{(o,d)}, \forall l \in L, \forall (o,d) \in D \qquad \textit{Formula 3.6}$$

3. *The wavelength utilisation constraint.* Each wavelength can be utilised only by working paths or by spare paths.

$$w_w^{k,od} + s_w^{k,od} \leq 1 \qquad \textit{Formula 3.7}$$

4. *Disjointness.* The working path and the spare path, $\left(P_{(o,d)}^w, P_{(o,d)}^s\right)$, between each node pair (o,d) must be link disjoint to cover single link failure. This constraint guarantees that in the event of failure only one of the working paths and spare paths will fail, and the failure will be covered by spare paths.

$$P_{(o,d)}^w \cap P_{(o,d)}^s = \emptyset, \forall (o,d) \in D \qquad \textit{Formula 3.8}$$

This constraint is satisfied only if $K_{od}^{w} \cap K_{od}^{s} = \emptyset, \forall (o,d) \in D$.

3.3.1.3 K-shortest paths algorithm

The K-shortest paths algorithm is needed in our work to determine a set of shortest paths between a given (o,d) node pair of nodes. The algorithm in this work is an extension of the *Dijkstra* shortest path algorithm, and was initially proposed by Mittal et al. [28]. In the pseudo code illustrated in *Algorithm 3.1*, K_{od} represents the final set of K shortest paths found between node pair (o,d). P is the set of available paths, and for each path p of P, $L(p)$ is the set of links that forms the path p, and $L(p)$ is organised in an increasing cost order.

Algorithm 3.1 The K-shortest paths algorithm

```
PARAMETER:
K_od ← ∅
P ← ∅
FOR each node o ∈ N
    FOR each node d ∈ N − {o}
        Find the shortest path SP between o and d using
        Dijkstra Algorithm
        P = P ∪ {SP}
        n = 1
        WHILE (P ≠ ∅ and n < K)
            Take the first path p of P
            P = P − {p}
            K_od = K_od ∪ {p}
            Search L(p)
            WHILE (L(p) ≠ ∅ and n < K)
                Take the lowest cost link l of L(p)
                L(p) = L(p) − {l}
                Remove the link l and search the new shortest
path                SP′ between o and d (Dijkstra Algorithm)
                IF SP′ is found
```

```
                    P = P − {SP'}
                    K_od = K_od ∪ {SP'}
                    n = n + 1
                END IF
                Re-insert l in the network
            END WHILE
        END WHILE
    END FOR
END FOR
```

3.3.2 Optical lightpath encoding

Encoding is an important procedure in GA to map the decision variables of the optimisation problem into chromosomes that could be dealt with GA. In the problem of survivable DWDM optical mesh network design, the ultimate outcome is a set of working and spare lightpaths between each (o, d) to satisfy a demand request, considering the objective function (*Formula 3.1*), and the constraints (*Formula 3.4 - 3.8*). The chromosome is defined by assigning the integer l to each link with corresponding wavelength set $\{\lambda_{l,1}, \lambda_{l,2}, \cdots, \lambda_{l,W}\}$. Then each path of the K-shortest paths between each node pair (o, d) is assigned a binary code and is encoded to a string, $STR_{(o,d)} = \{P_1, P_2, \cdots, P_K\}$. The chromosome is a binary string formed by concatenation of the strings so that the length of each chromosome is a function of K and the number of requests, Q, in the demand matrix.

Figure 3.4 demonstrates the encoding process. On the left hand side is a set of K-shortest paths between a node pair (o, d), where $(o, d) \in D$ and $g_{od} \subset G(N, L)$. When encoding the lightpath, the best available paths of sub-graph g_{od} are assigned a smaller value binary code, and vice versa. In this way, g_{od} can be encoded into a

string, $STR_{(o,d)}$. A chromosome is formed by concatenation of the strings like $\left\{P^1_{(o,d)}, P^2_{(o',d')}, \cdots, P^Q_{(o'',d'')}\right\}$ which is an instance of the set $\left\{STR^1_{(o,d)}, STR^2_{(o',d')}, \cdots, STR^Q_{(o'',d'')}\right\}$, where $P^1_{(o,d)} \in STR^1_{(o,d)}$, $P^2_{(o',d')} \in STR^2_{(o',d')}$, $\cdots$, $P^Q_{(o'',d'')} \in STR^Q_{(o'',d'')}$ respectively.

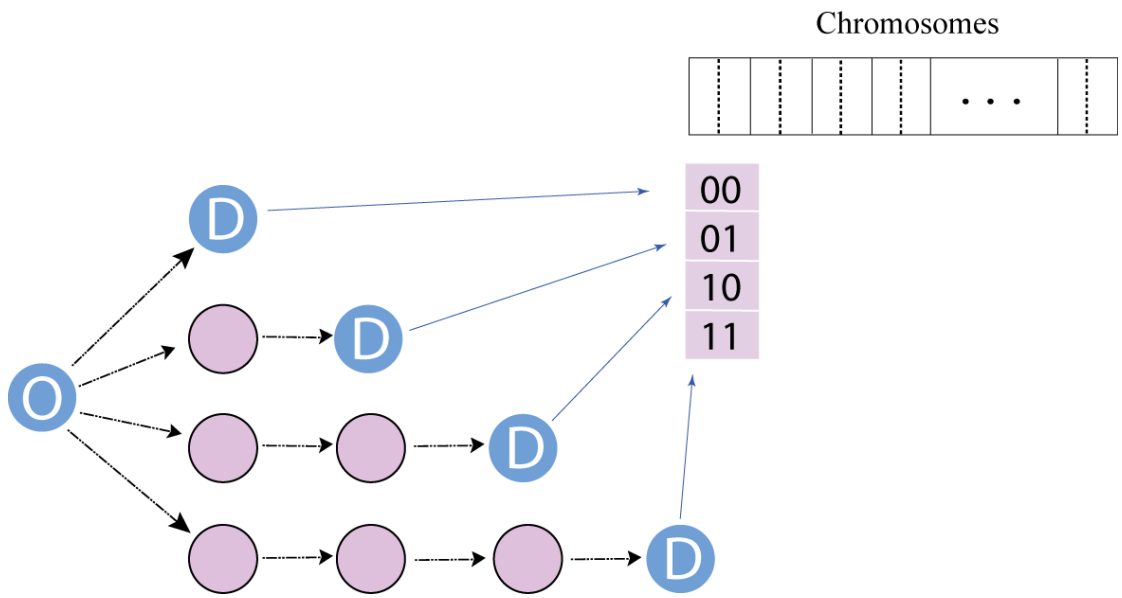

Figure 3.4 Encoding lightpaths into binary chromosomes

3.3.3 Evaluation function

An evaluation (fitness) function is a function which measures the relative performance of individuals in GAs. In this problem, the fitness function is given in *Formula 3.9*, which is formulated as a weighted sum, with weights w_w and w_s for the primary and redundant capacity respectively. f_l^w and f_l^s are the amount of primary capacity allocated to working and spare lightpaths for link l, and they have been pre-defined in *Formula 3.2* and *Formula 3.3* respectively.

$$fitness(w) = w_w \cdot \sum_{l=1}^{M} f_l^w + w_s \cdot \sum_{l=1}^{M} f_l^s \qquad \textit{Formula 3.9}$$

3.3.4 Initial population

The size and the way of creating the initial population affect the speed and the accuracy of the GA in its convergence to solutions [19, 29]. GAs with larger population sizes provide more genetic diversity, hence generally produce better solutions. However, on the other hand, large population sized GAs are usually quite costly in terms of time and memory spent. Hence it is essential to carefully consider how the initial population is created.

There are two possible approaches to create the initial population when designing the survivable DWDM optical network, namely, to initialise the population randomly or to initialise the population with a heuristic function.

The random approach of initialising the population means that the paths of each (o, d) node pair in the demand matrix D are chosen randomly from their K-shortest paths set, and could be represented by the *Formula 3.10*.

$$P^{i}_{(o,d)} = P_J, 1 \leq J \leq K, \forall (o, d) \in D \qquad \textit{Formula 3.10}$$

Alternatively, the other approach can be used, where the initial paths are chosen to be the best available path from the K-shortest paths set for all the requests in the demand matrix D, as shown in *Formula 3.11*.

$$P^{i}_{(o,d)} = P_1, \forall (o, d) \in D \qquad \textit{Formula 3.11}$$

The two approaches for creating the initial population in a sample DWDM network environment were compared with the simulation results are provided in *Figure 3.5*. It can be clearly seen from the figure that the heuristic approach has faster convergence speed, and generates less errors, thus it is chosen as the initialisation scheme for this work to design survivable DWDM mesh networks.

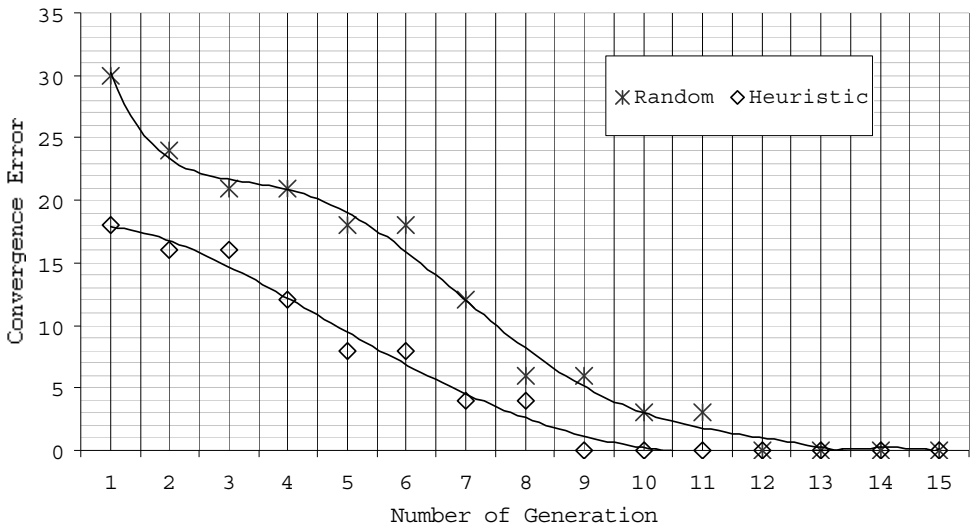

Figure 3.5 Simulation of random and heuristics initialisation

3.3.5 Reproduction

GAs employ what are called the genetic operators, namely, selection, crossover, and mutation, to search or produce healthier individuals in their population pool. In this section the implementation of these genetic operators and the creation of the new generation of individuals are described.

3.3.5.1 Selection

A particular individual is selected for reproduction based on its fitness value. In this work, a virtual roulette wheel [30] is employed to select fitter parent chromosomes. Each chromosome in the population is associated with a sector in this wheel and the area of each sector is proportional to the fitness value of its chromosome, increasing

the probability that the fitter chromosomes are selected. The steps for implementing the roulette wheel are considered as follows [31].

Step 1: Evaluate the fitness, f_i, of each individual in the population pool.

Step 2: Compute the probability (slot size), p_i, of selecting each member of the population.

$$p_i = f_i / \sum_{k=1}^{n} f_k$$

where n is the population size.

Step 3: Calculate the cumulative probability, q_i , for each individual.

$$q_i = \sum_{j=1}^{i} p_j$$

Step 4: Generate a uniform random number, $Random \in (0,1]$.

Step 5: If $Random < q_1$then select the first chromosome, x_1; else select the individual x_i such that $q_{i-1} \leq Random \leq q_i$.

Step 6: Repeat step 4 and 5 N times to create N candidates in the mating pool.

3.3.5.2 Crossover

This operation produces new, fitter chromosomes having some parts of the genetic material from both parents. In this work, path crossover involves the exchange of K-shortest lightpaths between (o, d) node pairs to traverse demand traffic. *Figure 3.6* is an example of the crossover operation. In this sample network, the demand request is between (A, E) and (D, E). If the two lightpaths, $\{[A - B - C - E], [D - E]\}$ and $\{[A - B - E], [D - A - E]\}$, are selected to be the parent individuals, after the crossover operation two offspring $\{[A - B - E], [D - E]\}$ and $\{[A - B - C - E], [D - A - E]\}$, will be produced as the new generation.

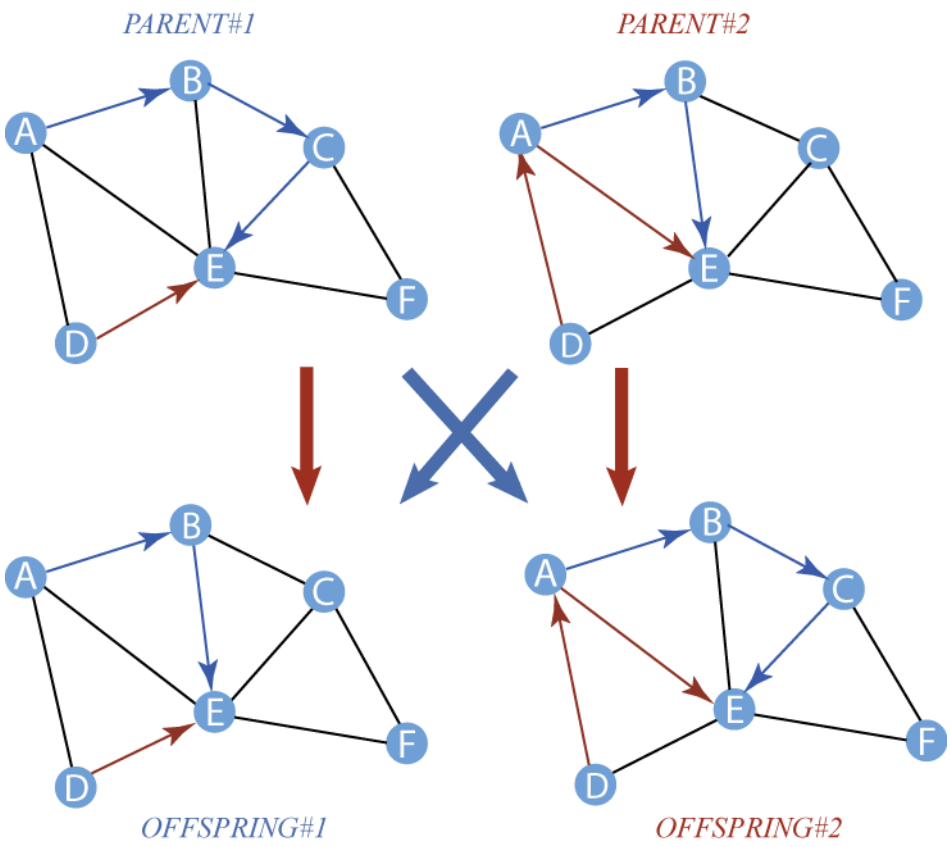

Figure 3.6 Demonstration of the crossover operation

3.3.5.3 Mutation

Mutation is the random adjustment to one part of the chromosome and often enables the recovery of good genetic material that may be lost through the generations. In this case, a binary mutation operates on a gene (bit) of an element (binary path code) of a chromosome and complements it. The application of mutation to one of the parents from *Figure 3.6*, $\{[A-B-C-E],[D-E]\}$, is shown in *Figure 3.7*. After the mutation operation on each lightpath, the new offspring (lightpath) is $\{[A-B-E],[D-A-E]\}$.

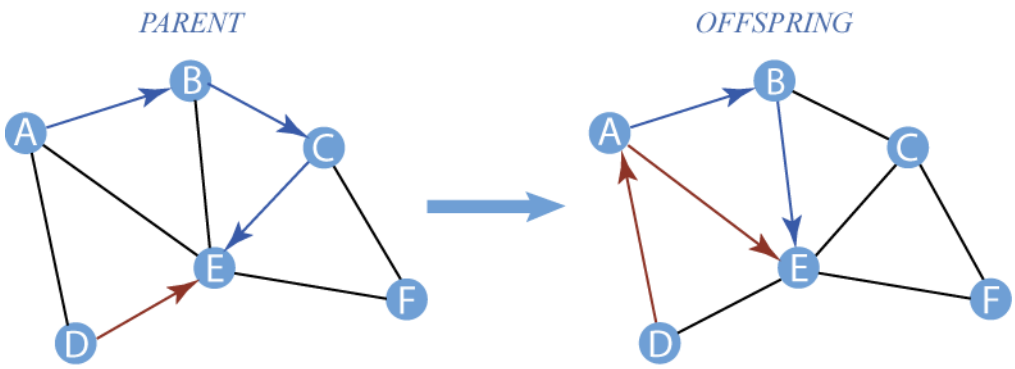

Figure 3.7 Demonstration of the mutation operation

3.3.6 Termination

The stochastic nature of GA search means that it can be difficult to specify convergence criteria to terminate the evolutionary cycle. Here, the GA is terminated after a predefined maximum number of generations and the quality of the best lightpaths evaluated according to the objective function. If no acceptable solutions are found, the GA is restarted or a fresh search initiated.

The complete GA diagram for designing the survivable DWDM optical mesh network is presented in *Figure 3.8*.

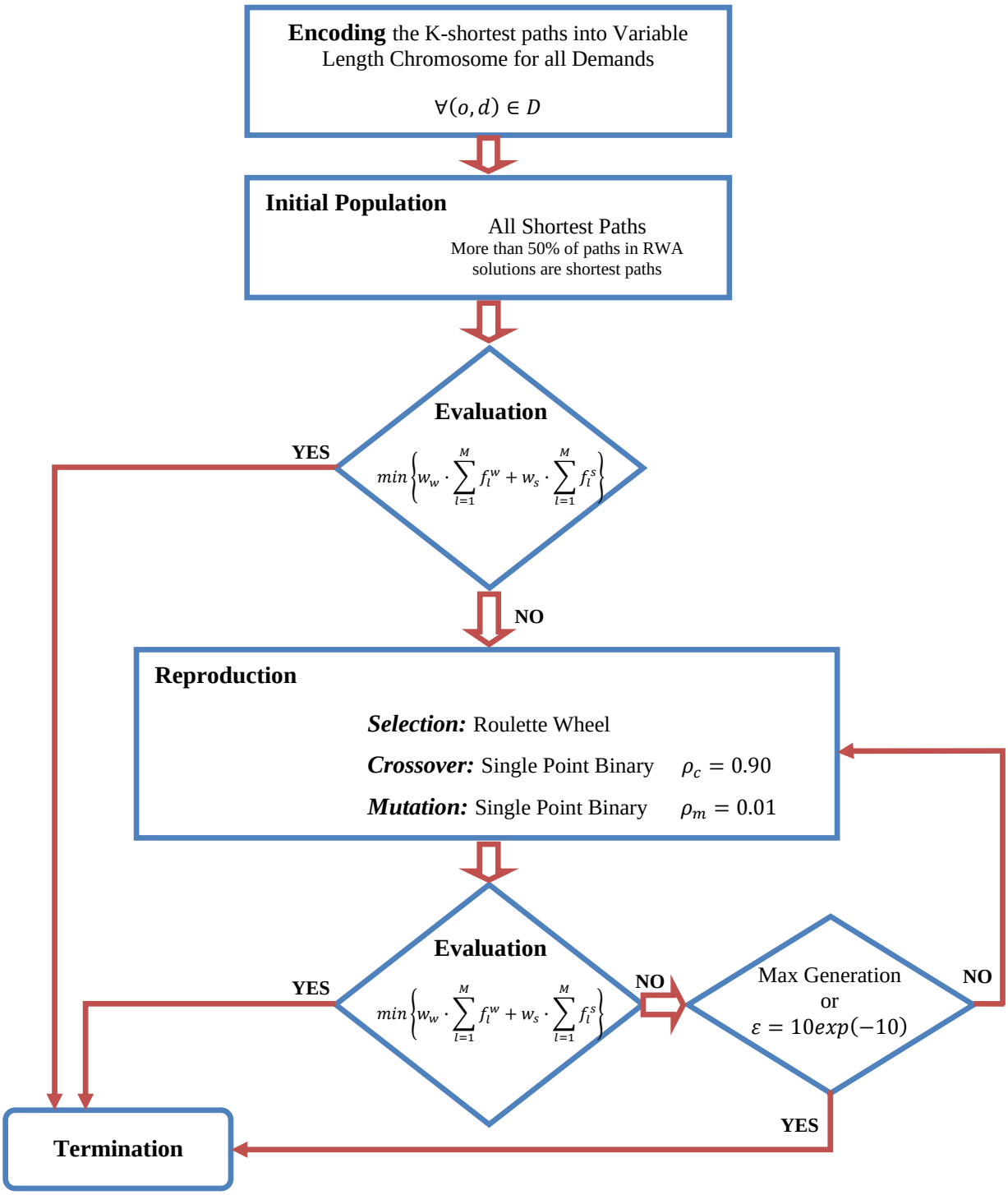

Figure 3.8 The diagram of a complete GA model

3.4 Simulation settings and benchmarking

The Pan European network [32], shown in *Figure 3.9*, (18 nodes, 35 links, and 4.7 average node degree) was used as a test bench backbone network. The solutions were achieved by considering 40 wavelengths per link, with five channels per link and eight wavelengths per channel. The attributes of the network have been summarised in *Table 3.1*. The network topology can also be represented by an adjacency matrix, like the one shown in *Formula 3.12*. It was assumed that all links in the network physical layer were bidirectional and that all nodes were capable of full wavelength conversion.

$$\Lambda = \begin{bmatrix}
0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0
\end{bmatrix}$$

Formula 3.12

Table 3.1 Link encoding and weight assignment

Node Pair	Link Assignment	Actual Link Length (km)	Capacity (wavelengths)
(1,3)	1	309	40
(1,8)	2	215	40
(1,9)	3	211	40
(1,14)	4	680	40
(1,15)	5	332	40
(2,9)	6	802	40
(2,10)	7	588	40
(2,13)	8	631	40
(3,6)	9	844	40
(3,7)	10	1471	40
(3,12)	11	1323	40
(3,14)	12	569	40
(3,15)	13	415	40
(4,7)	14	1504	40
(4,9)	15	699	40
(4,13)	16	1203	40
(4,15)	17	1155	40
(4,17)	18	359	40
(4,18)	19	912	40
(5,7)	20	1251	40
(5,16)	21	1506	40
(6,15)	22	429	40
(7,14)	23	791	40
(7,16)	24	889	40
(7,17)	25	1312	40
(8,17)	26	766	40
(9,15)	27	500	40
(9,17)	28	959	40
(10,13)	29	530	40
(10,15)	30	1826	40
(11,12)	31	650	40
(11,15)	32	2388	40
(14,17)	33	900	40
(18,16)	34	376	40
(18,17)	35	409	40

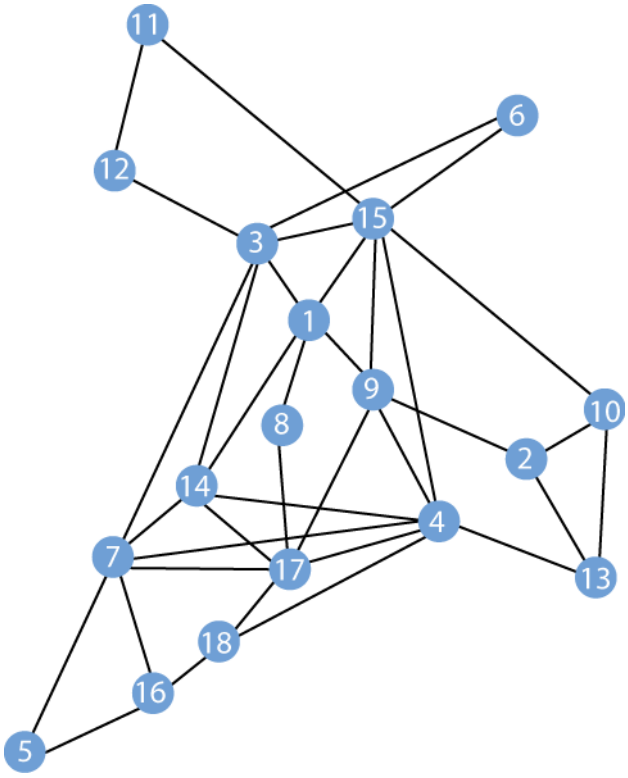

Figure 3.9 The Pan European network topology

3.4.1 Simulation setup

The simulation set-up was developed using *MathWorks MATLAB 7.1* and run on a parallel processing *SUN UNIX SOLAR* workstation system with 24 processors and 24 gigabytes of memory. The program script can be divided into four main block segments; *Figure 3.10* shows the interaction among them.

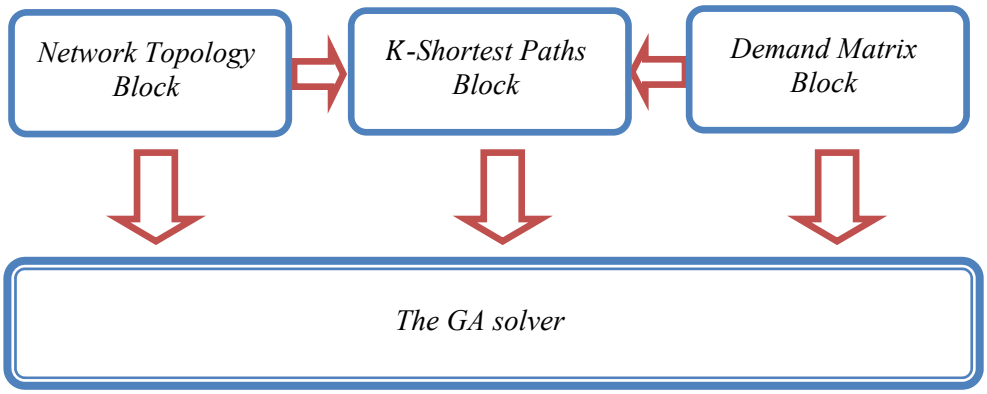

Figure 3.10 Block diagram of simulation setup

The '*Network topology*' block requires an adjacency matrix input like the one mentioned previously in *Formula 3.12*. The matrix maps the graphic topology of the network into the notation that the program can manipulate. The '*Demand matrix*' block is designed to import the set of traffic request pairs and the data volume required between them. The model of the demand matrix will be explained further in *Section 3.4.2*. After obtaining the network information, these data will be transferred to the 'K-*shortest paths*' block. In this block, the set contains the K-shortest paths is computed in respect to each request in the demand matrix. Finally the process of establishing the most utilized working and spare lightpaths according to the demand matrix is performed by the '*GA solver*' block.

3.4.2 Demand matrix modelling

DWDM optical mesh transport networks may actually contain a mixture of various services. However, if viewed in aggregate manner, all these high level services can be distilled down into a total demand for wavelengths between every (o, d) node pairs, where the set of these (o, d) pairs is referred to as the demand matrix. In the model employed, the following demand matrix was used

$$D = \begin{bmatrix} (1,11,10); & (2,7,6); & (3,4,7); & (6,4,5); & (5,17,8); \\ (6,11,9); & (17,10,6); & (11,4,11); & (13,8,13); & (18,12,7) \end{bmatrix}$$

each element of this matrix represents an $(o, d, demand)$.

3.4.3 GA parameters

The settings of the GA parameters can affect the accuracy of the solution, the complexity of search and the time or memory consumption. In this research, the GA parameter settings were tested within the range of suggested GA implementation

guideline presented in *Figure 2.4*. The testing showed that the settings summarized in *Table 3.2* could produce good results for the DWDM RWA problems.

Table 3.2 GA parameters

Population Size	50
Generations	100
Crossover Probability	$\rho_c = 0.90$
Mutation Probability	$\rho_m = 0.01$
Relative Accuracy Tolerance	$\varepsilon = 10e^{-10}$

3.4.4 The *K* factor

In this section, the right setting for the K-shortest algorithm is discussed. The simulation of the RWA problem was performed with the value of K ranging from 1 to 8, with results recorded in *Figure 3.11*. From the figure it may be seen that the quality of the solution found by GA becomes steady when $K \geq 4$. In order to minimize the overhead while gathering the best performance of the GA, $K = 4$ was chosen, i.e., only the first four shortest paths were considered to form the GA chromosomes.

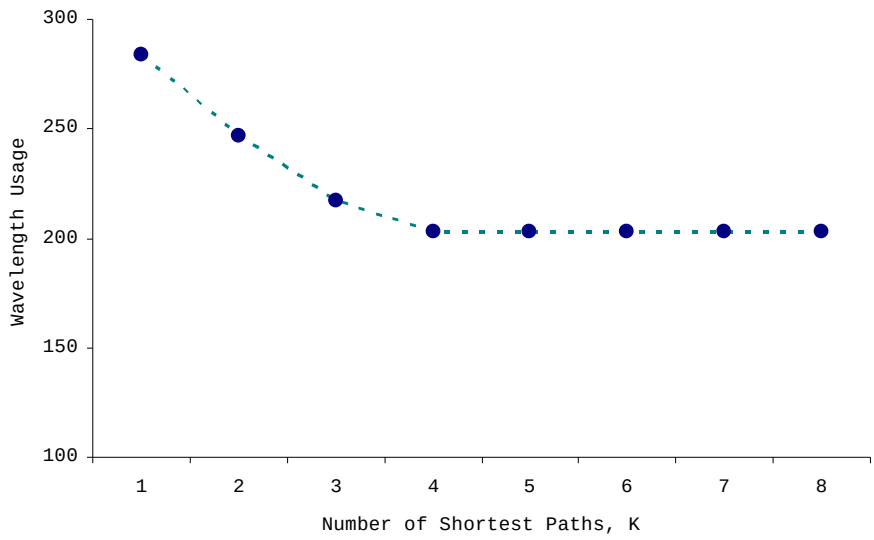

*Figure 3.11 The effect of the **K** factor*

3.5 Simulation Results

The GA model simulation results for the Pan European network can be viewed from three aspects, which are the solution of primary RWA, the solution of RWA with dedicated path protection and the solution of RWA with shared path protection. These are now considered turn.

3.5.1 Primary RWA

The RWA simulation results for the Pan European network with demand matrix D are shown in *Table 3.3*.

Table 3.3 RWA solution for the Pan European network

Node Pair	Demand	Working Path	Working Wavelength	Wavelength Utilisation
(1,11)	10	1-15-11	$(\lambda_1 \cdots \lambda_{10})$/all links	20
(2,7)	6	2-9-4-7	$(\lambda_1 \cdots \lambda_6)$/all links	18
(3,4)	7	3-15-4	$(\lambda_1 \cdots \lambda_7)$/all links	14
(6,4)	5	6-15-4	$(\lambda_1 \cdots \lambda_5)/[6-15]$; $(\lambda_8 \cdots \lambda_{12})/[15-4]$	10
(5,17)	8	5-7-17	$(\lambda_1 \cdots \lambda_8)$/all links	16
(6,11)	9	6-15-11	$(\lambda_6 \cdots \lambda_{14})/[6-15]$; $(\lambda_{11} \cdots \lambda_{19})/[15-11]$	18
(17,10)	6	17-9-2-10	$(\lambda_1 \cdots \lambda_6)$/all links	18
(11,4)	11	11-15-4	$(\lambda_1 \cdots \lambda_{11})/[11-15]$; $(\lambda_{13} \cdots \lambda_{23})/[15-4]$	22
(13,8)	13	13-4-17-8	$(\lambda_1 \cdots \lambda_{13})$/all links	39
(18,12)	7	18-4-15-3-12	$(\lambda_1 \cdots \lambda_7)$/all links	28
				203

Primary (working) wavelengths were assigned to paths using the simple but effective '*First-Fit strategy*' [9], which chooses the available wavelength with the smallest index. The table shows the connection requests between (o, d) node pairs with their volumes in terms of required wavelengths, the RWA solution and the working wavelength utilisation for each connection request.

Table 3.4 RWA solution of spare paths by DPP for Pan European network

Node Pair	Demand	Spare Path	Spare Wavelength	Wavelength Utilisation
(1,11)	10	1-3-12-11	$(\lambda_1 \cdots \lambda_{10})/\{[1-3],[12-11]\}$; $(\lambda_8 \cdots \lambda_{17})/[3-12]$	30
(2,7)	6	2-13-4-17-7	$(\lambda_1 \cdots \lambda_6)/\{[2-13],[17-7]\}$ $(\lambda_{13} \cdots \lambda_{19})/\{[13-4],[4-17]\}$;	24
(3,4)	7	3-14-17-4	$(\lambda_1 \cdots \lambda_7)$/all links	21
(6,4)	5	6-3-1-9-4	$(\lambda_1 \cdots \lambda_5)/\{[6-3],[3-1],[1-9]\}$; $(\lambda_7 \cdots \lambda_{12})/[9-4]$	20
(5,17)	8	5-16-18-17	$(\lambda_1 \cdots \lambda_8)$/all links	24
(6,11)	9	6-3-12-11	$(\lambda_6 \cdots \lambda_{14})/[6-3]$; $(\lambda_{18} \cdots \lambda_{26})/[3-12]$; $(\lambda_{11} \cdots \lambda_{19})/[12-11]$	27
(17,10)	6	17-4-13-10	$(\lambda_8 \cdots \lambda_{13})/[17-4]$; $(\lambda_1 \cdots \lambda_6)/\{[4-13],[13-10]\}$	18
(11,4)	11	11-12-3-7-4	$(\lambda_1 \cdots \lambda_{11})$/all links	44
(13,8)	13	13-10-15-1-8	$(\lambda_7 \cdots \lambda_{19})/[13-10]$; $(\lambda_1 \cdots \lambda_{13})/\{[10-15],[15-1],[1-8]\}$	52
(18,12)	7	18-17-9-15-11-12	$(\lambda_1 \cdots \lambda_7)/\{[18-17],[9-15]\}$; $(\lambda_7 \cdots \lambda_{13})/[17-9]$; $(\lambda_{20} \cdots \lambda_{26})/[15-11]$; $(\lambda_{12} \cdots \lambda_{18})/[11-12]$	35
				295

3.5.2 Dedicated Path Protection (DPP)

The DPP RWA solutions for the spare lightpaths are shown in *Table 3.4*. The primary and spare lightpaths of all requests are link disjoint, the scheme thus protects against any single link failure because at most one of the two working and spare lightpaths will fail.

3.5.3 Shared Path Protection (SPP)

In the SPP architecture, a link can be shared between spare paths if the working paths that protect the proposed spare paths are link disjoint. Let$S_l^s = \{(P_1^s, P_1^w, d^1), (P_2^s, P_2^w, d^2), \cdots, (P_j^s, P_j^w, d^j)\}$ be a set of spare paths, working paths and demand requests that spare paths are common on link l. This link can be shared between spare paths if $P_1^w \cap P_2^w \cap \cdots \cap P_j^w = \emptyset$ and the amount of spare resource, r^l, is reduced by the minimum of the common demand requests. The total amount of shared resource is: $R = \sum_l \sum_{S_l^s} \left((j-1) \cdot r^l\right), \forall l \in L$

The RWA problem was solved using SPP with the same resultant number of working wavelengths as DPP and this is shown in *Table 3.5*.

Table 3.5 Sharing strategy of spare paths by SPP for the Pan European network

Link	Path Pairs	Working Paths Situation	Reduced Capacity
$[3-12]$	(1,11), (6,11)	Joint	0
$[12-11]$	(1,11), (6,11)	Joint	0
$[17-4]$	(3,4), (17,10)	Disjoint	6
$[6-3]$	(6,4), (6,11)	Disjoint	5
$[13-10]$	(17,10), (13,8)	Disjoint	6
$[11-12]$	(11,4), (18,12)	Disjoint	7
			R=24

The total number of shared wavelengths needed was 24, reducing the spare wavelengths by 8.1% from the DPP case, which needed 295, this being the number of spare wavelengths.

Figure 3.12 shows the frequency of the wavelength assignments and it may be observed that the first fit algorithm assigned the lowest numbered wavelengths as expected, with the primary wavelengths making particular use of the first six wavelengths.

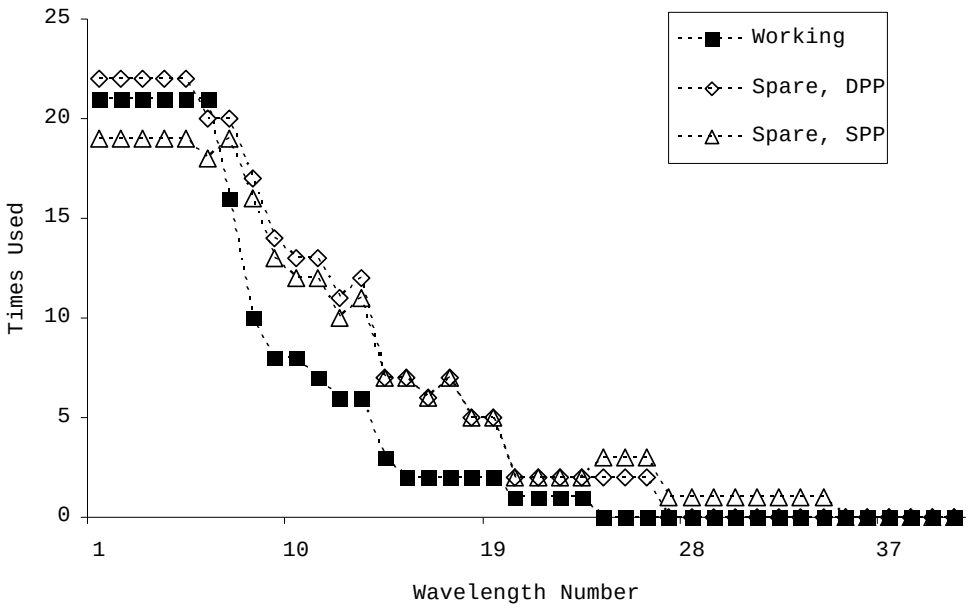

Figure 3.12 Wavelength distribution for the Pan European network

3.6 Summary

In *Part I* of this thesis a demonstration of the design of a the DWDM optical network by employing evolutionary algorithms has been provided. DWDM optical networks carry large traffic volumes and thus providing a high level of resiliency is an important and critical issue in their deployment. This is particularly true if they are to be employed as backbones for the provision of future IP services with enhanced QoS.

In this chapter, first an introduction to DWDM optical networks was presented, followed by a short review of recent research for designing survivable DWDM networks. This highlighted that ILP approaches have been extensively used for modelling different survivability schemes. However, the performance of such methods is dramatically reduced when the objective functions or constraints are non-linear.

The work has addressed the provision of spare capacity via a novel GA model based on variable length chromosomes. The RWA problem has been solved using the GA method and the results of applying it to the Pan European network have been investigated in the context of a single link failure. The optimum number of shortest paths was found to be four, with a greater number producing greatly diminished returns. The assignments using both DPP and SPP were investigated, with the latter producing a saving of 8.1%. The results indicate that GAs are a promising approach to tackle RWA problems in DWDM networks. Some extension of this work can be found in [33-36].

References

[1] O. Krauss, *DWDM and optical networks - An introduction to terabit technology*. Erlangen: Publicis Corporate Publishing, 2002.

[2] O. Gerstel and R. Ramaswami, "Optical layer survivability: a service perspective," *IEEE Communications Magazine,* vol. 3, pp. 104-113, 2000.

[3] C. S. R. Murthy and M. Gurusamy, *WDM optical networks: concepts, design, and algorithms*. Upper Saddle River, NJ: Prentice Hall PTR, 2002.

[4] S. Ramamurthy, L. Sahasrabuddhe, and B. Mukherjee, "Survivable WDM mesh networks," *J. Lightwave Technol.,* vol. 21, pp. 870-883, 2003.

[5] J. L. Kennington, E. V. Olinick, and G. Spiride, "Basic mathematical programming models for capacity allocation in mesh-based survivable networks," *Omega,* vol. 35, pp. 629-644, 2007.

[6] B. Mukherjee, *Optical Communication Networks* . New York: McGraw-Hill, July 1997.

[7] J. Zhang and B. Mukheriee, "A review of fault management in WDM mesh networks: basic concepts and research challenges," *Network, IEEE,* vol. 18, pp. 41-48, 2004.

[8] H. Pin-Han and H. T. Mouftah, "Shared protection in mesh WDM networks," *Communications Magazine, IEEE,* vol. 42, pp. 70-76, 2004.

[9] H. Zang, J. P. Jue, and B. Mukherjee, "A review of routing and wavelength assignment approaches for wavelength-routed optical WDM networks," *SPIE Optical Networks Magazine,* vol. 1, pp. 47-60, 2000.

[10] J. T. Park, "Resilience in GMPLS path management: model and mechanism," *Communications Magazine, IEEE,* vol. 42, pp. 128-135, 2004.

[11] H. Zang, C. Ou, and B. Mukherjee, "Path-protection routing and wavelength assignment (RWA) in WDM mesh networks under duct-layer constraints," *IEEE/ACM Transactions on Networking,* vol. 11, pp. 248-459, 2003.

[12] Y. Liu, D. Tipper, and P. Siripongwutikorn, "Approximating optimal spare capacity allocation by successive survivable routing," *IEEE/ACM Transactions on Networking,* vol. 13, pp. 198-211, 2005.

[13] X. Yang, L. Shen, and B. Ramamurthy, "Survivable Lightpath Provisioning in WDM Mesh Networks Under Shared Path Protection and Signal Quality Constraints," *J. Lightwave Technol.,* vol. 23, p. 1556, 2005.

[14] R. Shenai and K. Sivalingam, "Hybrid survivability approaches for optical WDM mesh networks," *Lightwave Technology, Journal of,* vol. 23, pp. 3046-3055, 2005.

[15] Z. R. Zhang, W. D. Zhong, and B. Mukherjee, "A heuristic algorithm for design of survivable WDM networks," *IEEE Communication Letters,* vol. 8, pp. 467-469, 2004.

[16] C. Ou, J. Zhang, H. Zang, L. H. Sahasrabuddhe, and B. Mukherjee, "New and improved approaches for shared-path protection in WDM mesh networks," *Journal of Lightwave Technology,* vol. 22, pp. 1223-1232, 2004.

[17] H. Pin-Han and H. T. Mouftah, "A novel survivable routing algorithm for shared segment protection in mesh WDM networks with partial wavelength conversion," *Selected Areas in Communications, IEEE Journal on,* vol. 22, pp. 1548-1560, 2004.

[18] C. W. Ahn and R. S. Ramakrishna, "A genetic algorithm for shortest path routing problem and the sizing of populations," *IEEE Transaction on Evolutionary Computation,* vol. 6, pp. 566-579, 2002.

[19] J. Arabas and S. Kozdrowski, "Applying an evolutionary algorithm to telecommunication network design," *IEEE Transactions on Evolutionary Computation,* vol. 5, pp. 309-323, 2001.

[20] J. Inagaki and M. Haseyama, "A genetic algorithm for routing with an upper bound constraint,," *IEICE Transaction of Information and System,* vol. E88-D, pp. 679-681, 2005.

[21] L. Chi-Chun and C. Wei-Hsin, "A multi-objective hybrid genetic algorithm for the capacitated multipoint network design problem," *IEEE Transactions on Systems, Man and Cybernetics, Part B,* vol. 30, pp. 461 -470, 2000.

[22] L. He, C. P.Botham, and C. D. O'Shea, "An evolutionary design algorithm for ring-based SDH optical core networks," *BT Technology Journal,* vol. 22, pp. 135-144, 2004.

[23] H. S. Yang, M. Maier, M. Reisslein, and W. M. Carlyle, "A genetic algorithm based methodology for optimizing multi-service convergence in a metro WDM network," *Journal of lightw ave Technology,* vol. 21, pp. 1114 -1133, 2003.

[24] S. Pierre and G. Legault, "A genetic algorithm for designing distributed computer network topologies," *IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics,* vol. 28, pp. 249-258, 1998.

[25] K. A. D. Jong and W. M. Spears, "Using genetic algorithm to solve NP-complete problems," *Proceeding of Third Interna tional Confer ence on Genetic Algorithms (ICGA),* 1989.

[26] Z. Drezner and S. Salhi, "Using hybrid metaheuristics for the one-way and two-way network design problem," *Naval Research Logistics,* vol. 49, pp. 449-463, 2002.

[27] S. V. Ukkusuri, T. V. Mathew, and S. T. Waller, "Robust transportation network design under demand uncertainty," *Computer-Aided Civil and Infrastructure Engineering,* vol. 22, pp. 6-18, 2007.

[28] S. Mittal and P. Mirchandani, "Implementation of K-shortest path dijkstra algorithm used in all-optical data communication networks," SIE 546 Project Report. 2004.

[29] S. J. Moore and M. C. Sinclair, "Deisgn of routing tables for a survivable military communications nework using genetic algorithms," in *IEEE Congress on Evolutionary Computation* Washington, USA, 1999, pp. 1788-1795.

[30] S. Kwong, D. W. F. Lam, K.S.Tang, and K.-F. Man, "Optimization of spare capacity in self-healing multicast ATM network using genetic algorithm," *IEEE Transactions on Industrial Electronics,* vol. 47, pp. 1334-1343, 2000.

[31] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*: Addison-Wesley publishing company, 1989.

[32] G. Birkan, J. Kennington, E. Olinick, A. Ortynski, and G. Spiride, "Practical Integrated Design Strategies for Opaque and All-Optical DWDM Networks:

Optimization Models and Solution Procedures," *Telecommunication Systems,* vol. 31, pp. 61-83, 2006.

[33] Y. S. Kavian, H. F. Rashvand, W. Ren, M. S. Leeson, E. L. Hines, and M. Naderi, "Network topology effect on QoS delivering in survivable DWDM optical networks," *Journal of Telecommunication and Information Technology,* vol. 1, pp. 68-71, 2009.

[34] Y. S. Kavian, H. F. Rashvand, W. Ren, M. Naderi, E. L. Hines, and M. S. Leeson, "Genetic algorithm quality of service design in resilient dense wavelength division multiplexing optical networks," *IET Communication,* vol. 2, pp. 505-513, 2008.

[35] Y. S. Kavian, H. F. Rashvand, W. Ren, M. S. Leeson, E. L. Hines, and M. Naderi, "QoS-based RWA in DWDM optical networks using genetic algorithms," in *Proceeding of IEEE Internationa l Conference on Transparent Optical Networks ICTON-MW08* Marrakech, Morocco, 2008, pp. 1-4.

[36] M. S. Leeson, Y. S. Kavian, W. Ren, E. L. Hines, and M. Naderi, "Sruvivable wavelength-routed optical network design using genetic algorithms," in *Proceeding of IEEE Internatio nal Conference on Transparent Optical Network, ICTON-MW07* Sousse, Tunisia, 2007, pp. 1-4.

Part II

WIRELESS NETWORK ROUTING OPTIMISATION

Chapter 4:

Swarm Intelligence

4.1 Introduction

In *Part II* of this thesis, the work on wireless mobile ad hoc network (MANET) routing optimisation will be presented. This second part of the thesis begins with an introduction to the particular intelligent systems application that has been applied to the areas of wireless MANET routing optimisation in this research work. The intelligent system application which forms the focus in the following chapters is one that is inspired by animal behaviour, which is often referred to as the swarm intelligence.

Swarm Intelligence (SI) is a type of artificial intelligence algorithm based on the collective behaviour of decentralized, self-organized systems. The expression was first introduced by Gerardo Beni and Jing Wang in 1989, in the context of cellular robotic systems [1]. There are two popular swarm inspired methods in the computational intelligence areas: particle swarm optimization (PSO) and ant colony optimisation (ACO). PSO originated as a simulation of simplified social systems. The original intent was to graphically simulate the movement of organisms such as a flock of birds or a school of fish. However, it was later found that the particle swarm model can be used as an optimizer [2]. ACO algorithms are the other main sub-area of the SI domain. ACO was inspired by the behaviour of ants and has many successful applications in dealing with discrete optimisation problems. Despite the fact that both ACO and PSO are called as SI algorithms, there is quite a difference between them. They are compared in more detail later in this chapter.

The aim of this chapter is to introduce the major SI algorithms, namely PSO and ACO. Overviews of PSO in *Section 4.2* and of ACO in *Section 4.3* are provided. In this thesis ACO is utilized in MANET routing optimisation. At the end of the chapter, a summary is provided to conclude the whole chapter.

4.2 Particle swarm optimisation

Particle swarm optimisation (PSO) is a population based stochastic optimisation technique, inspired by the social behaviour of birds flocking or fish ‘schooling’. PSO shares many similarities with evolutionary algorithms (EA) which have been described previously in *Chapter 2*. The system is initialised with a population of random solutions and searches for optima by updating generations. However, unlike EA, PSO has no genetic operators such as crossover and mutation. In PSO, the

potential solutions (called particles) progress through the problem space by following the current optimum particles. Compared to EAs, the advantages of PSO are that it is easy to implement and there are few parameters to adjust. PSO has been successfully applied in many areas: function optimisation, artificial neural network training, fuzzy system control and other areas where GAs can be applied [3].

4.2.1 The background

It is interesting to study the self-organizing behaviour of species living in groups where no leader can be identified, for example, a flock of birds, a school of fish, or a herd of sheep. Within these social groups, individuals have no knowledge of the global behaviour of the entire group, nor do they have any global information about the environment. Despite this, they have the ability to gather and move together, based on local interactions between individuals. From the simple, local interaction between individuals, more complex collective behaviour emerges, such as flocking behaviour, homing behaviour, exploration and herding [4-6]. A large number of studies of the collective behaviour of social animals have been conducted [6-10].

As an aid to better understanding of the coordinated group dynamics of social animal groups, a number of simulation studies have been carried out. Among them the term ‘particle swarm optimisation’ was first introduced by Kennedy and Eberhart [2], based on a social-psychological model of social influence and social learning [11]. Individuals in a particle swarm follow a very simple behaviour: emulate the success of neighbouring individuals. The collective behaviour that emerges is that of discovering optimal regions of a high dimensional search space.

In 1994, Millonas summarised the main principles for SI algorithms to make them act efficiently [12]. These principles are provided as follows, and will be discussed in the

following section to see how PSO algorithms observe these rules. A group of individuals in the PSO algorithms should obey the principles given below.

- **Proximity principle**: the ability to respond to quality factors in the environment.
- **Quality principle**: the ability to respond to quality factors in the environment.
- **Principle of diverse response**: the group activities should not be committed along excessively narrow channels.
- **Principle of stability**: modes of behaviour should not change every time the environment changes.
- **Principle of adaptability**: the ability to change behavioural mode when it is worth the computational cost.

4.2.2 The basic PSO algorithm

A PSO algorithm maintains a swarm of particles, where each particle represents a potential solution. In analogy with evolutionary computation paradigms, a swarm is similar to a population, while a particle is similar to an individual.

The global best PSO, or ***gbest*** PSO, is one of the basic PSO models. In this model, each particle keeps track of its coordinates in the problem space which are associated with the best solution (fitness) called $pbest$ that it has achieved so far. Another 'best' value that is the overall best value obtained so far by any particle in the population is tracked by the global version of the particle swarm optimizer. The associated location of this overall best value is called $gbest$. Each particle i in the ***gbest*** PSO is also equipped with two vectors, i.e., the velocity vector $V_i = [v_i^1, v_i^2, \cdots, v_i^D]$ and the position vector $X_i = [X_i^1, X_i^2, \cdots, X_i^D]$, where D stands for the dimensions of the solution space.

Algorithm 4.1 is a pseudo code which outlines the process for implementing the ***gbest*** PSO, *Figure 4.1* illustrates the flowchart of the basic PSO model. The detailed explanation and formulas are shown as follows [13]:

1. Initialise a swarm of particles with random positions and velocities on D dimensions in the problem space.
2. For each particle, evaluate the desired optimisation fitness function in D variables.
3. Compare the particle's current fitness p by evaluation with particle's $pbest$. If p is better than $pbest$, then set $pbest$ value equal to p, and the associated position coordinate equal to the current location in D dimensional space.
4. Compare fitness evaluation with the population's overall best value, $gbest$. If the current value is better than $gbest$, then reset $gbest$ to the current particle's array index and value.
5. Change the velocity and position of the particle according to *Formula 4.1* and *Formula 4.2*, respectively:

$$v_i^d = wv_i^d + c_1 rand_1^d\left(pbest_i^d - x_i^d\right) + c_2 rand_2^d\left(gbest^d - x_i^d\right) \qquad \textit{Formula 4.1}$$

$$x_i^d = x_i^d + v_i^d \qquad \textit{Formula 4.2}$$

   where w is the inertia weight, c_1 and c_2 are the acceleration coefficients, $pbest_i^d$ is the personal best position associated with particle i, x_i^d is the current position of particle i, and $rand_1^d$ and $rand_2^d$ are two uniformly distributed random numbers independently generated within [0, 1] [14].

6. Loop to Step 2 until a termination criterion is met; usually a sufficiently good fitness or a maximum number of iterations is reached.

The acceleration constants c_1 and c_2 in *Formula 4.1* represent the weighting of the stochastic acceleration terms that pull each particle toward $pbest$ and $gbest$ positions. Thus, adjustment of these constants changes the amount of "tension" in the system. Low values allow particles to roam far from target regions before being tugged back, while high values result in abrupt movement toward, or past, target regions [13].

Algorithm 4.1 ***gbest*** PSO [14]

Create and initialize a D-dimensional swarm, S;

```
BEGIN:
REPEAT
    FOR each particle i = 1, ···, S
        // set the personal best position
        IF f(pbest_i) < f(p_i)
            pbest_i = p_i
        END IF
        // set the global best position
        IF f(gbest) < f(pbest_i)
            gbest = pbest_i
        END IF
    END
    FOR each particle i = 1, ···, S
        Update the velocity using Formula 4.1
        Update the position using Formula 4.2
    END
UNTIL stopping condition is reached
```

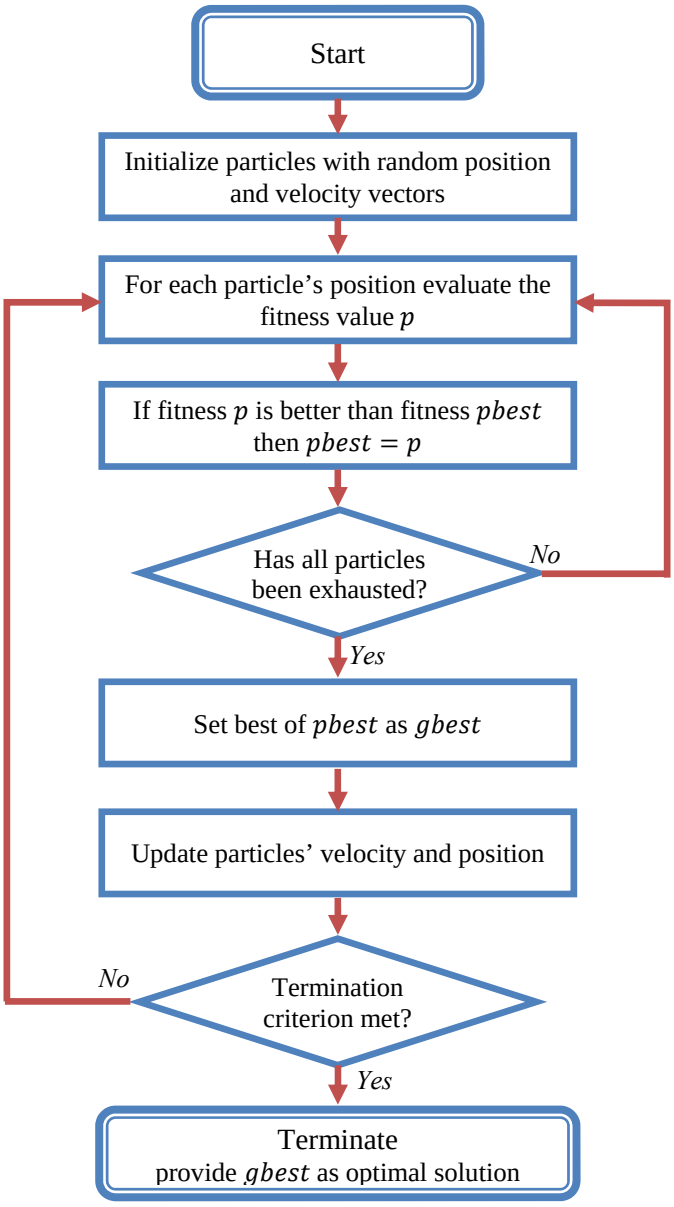

Figure 4.1 Flowchart for a basic PSO algorithm

In *Section 4.2.1*, the principles that ensure the efficiency of SI algorithms were introduced. PSO adheres to those principles as follows [15]. Multi-dimensional space calculations are carried out over a series of time steps, which addresses the **proximity principle**. The swarm of particles responds to quality factors in the form of the personal and neighbourhood best positions (**quality principle**). Allocation of responses between the personal best and neighbourhood best positions ensure a

diversity of response (**principle of diverse response**). The swarm changes its state only when the personal best and global best positions change, which addresses the **stability principle**. Lastly the swarm exhibits adaptive behaviour, since the state changes when the personal best and global best positions change (**principle of adaptability**).

Similar to EAs, the measure of the fitness requires a number of function evaluations, hence, can be computationally inefficient [16]. Further, the standard PSO algorithm can easily get trapped in the local optima when solving complex multimodal problems [17]. These weaknesses have restricted wider applications of PSO [18]. As a result, there have been seen many improved versions of the PSO model. Most modifications to the basic PSO are directed towards improving convergence of the PSO and increasing the diversity of the swarm [14]. The detailed literature of the PSO algorithms, the variations and their applications can be found in [3].

4.3 Ant colony optimisation algorithms

Ant Colony Optimization (ACO) is one of the most successful techniques of the wider field of swarm intelligence. The first ACO algorithms were proposed around 20 years ago. Since then, significant contributions to algorithmic variants, challenging application problems, and theoretical foundations have been realised. These have established ACO as a mature, high-performing metaheuristic for the solution of difficult optimisation problems. In this section, ACO is viewed from three perspectives, namely, where ACO comes from, what the characteristics of ACO algorithms are, and the framework of the ACO algorithms.

4.3.1 Foraging behaviour of ants

Many ant species have trail-laying trail-following behaviour when foraging (illustrated in *Figure 4.2* [19]): individual ants deposit a chemical substance called 'pheromone' as they move from a food source to their nest, and foragers follow such pheromone trails. Subsequently, more ants are attracted by these pheromone trails and in turn reinforce them even more. As a result of this autocatalytic effect, the optimal solution will emerge rapidly. The concentration of pheromone on a certain path is an indication of its usage. With time, the concentration of pheromone decreases due to diffusion effects. This property is important because it integrates dynamics into the path search process.

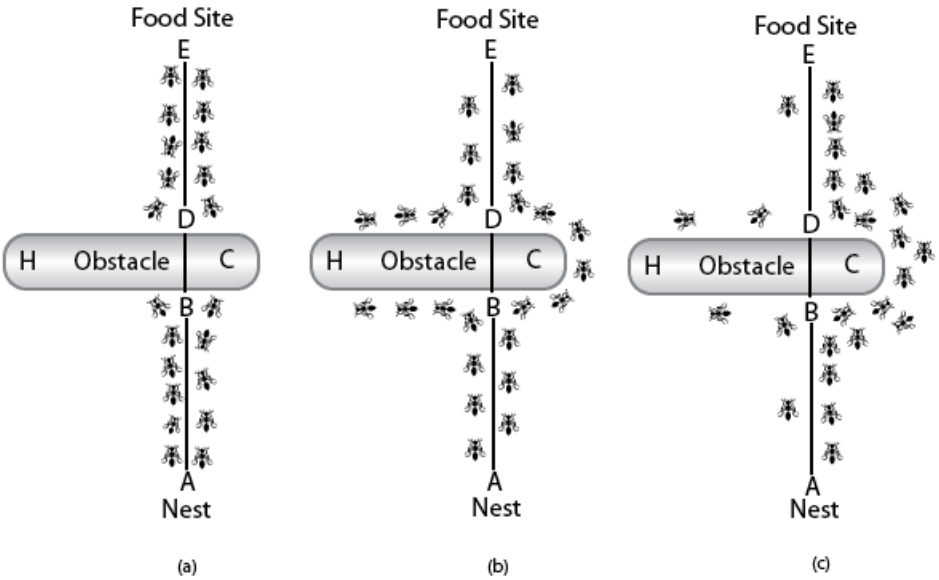

Figure 4.2 Ant colony foraging behaviour

In order to develop a formal model to describe the ant foraging behaviour, Deneubourg et al. [20] designed a simple experiment called the binary bridge experiment. In this experiment, the food source is separated from the next by a bridge with two equally long branches as illustrated in *Figure 4.3*. Initially, there is no pheromone on the two branches, which have, therefore, the same probability of being

selected by the ants. After a finite time period, one of the branches was selected, with most of the ants following the path, even with both branches being of the same length.

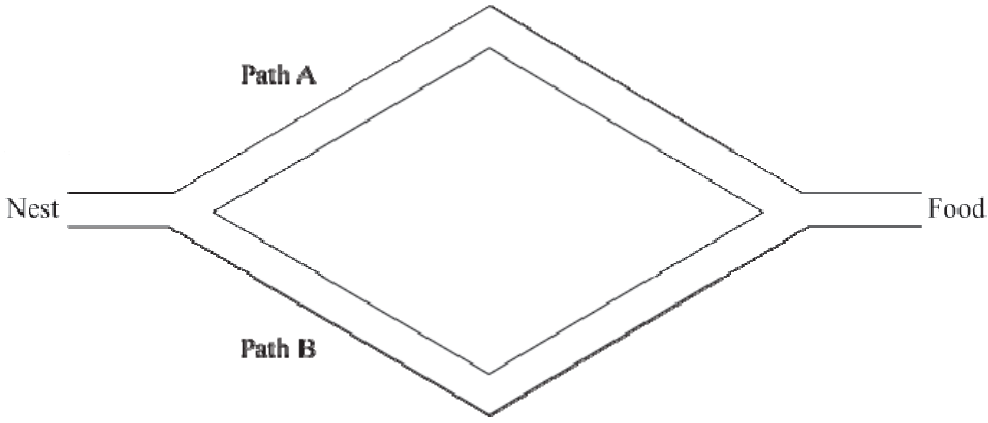

Figure 4.3 Binary bridge experiment

The selection of one of the branches is due to random fluctuations in path selection, causing higher concentrations on the one path.

From this experiment, a simple formal model was developed to characterize the path selection process [21]. For this purpose, it is assumed that ants deposit the same amount of pheromone and that pheromone does not evaporate. Let $n_A(t)$ and $n_B(t)$ denote the numbers of ants on paths A and B respectively at time step t. The probability of the next ant choosing path A at time step $t+1$ is given as,

$$P_A(t+1) = \frac{\left(c + n_A(t)\right)^{\alpha}}{\left(c + n_A(t)\right)^{\alpha} + \left(c + n_B(t)\right)^{\alpha}} = 1 - P_B(t+1)$$

Formula 4.3

where c quantifies the degree of attraction of an unexplored branch, and α is the bias to using pheromone deposits in the decision making process. The larger the value of α, the higher the probability that the next ant will follow the path with a higher pheromone concentration. The larger the values of c, the more pheromone deposits are required to make the choice of path non-random. Using the probability defined in

Formula 4.3, the decision rule of an ant that arrives at the binary bridge is expressed as follows: if $U(0,1) \leq P_A(t+1)$ then follow path A otherwise follow path B [22]. This model is one of the foundations of ACO algorithms, most of the modern ACO algorithms are the variations and improvement of the *Binary bridge experimental* model.

4.3.2 ACO algorithm characteristics

ACO algorithms are population-based stochastic search algorithms, designed to solve specific types of combinatorial optimisation problems. The first ant algorithm developed was the ant system [23, 24] and since then several improvements to the this system have been devised [25-28]. It should be noted that all the ACO algorithms have similar characteristics and common components, irrespective of the discrete optimisation problem being solved. The characteristics of ACO algorithms are summarised as follows.

ACO algorithms mimic the collective behaviour of real ant colonies, all these algorithms have a number of components in common, which include [29]:

- A colony of ants,
- A mechanism to generate and to activate ants,
- A pheromone evaporation mechanism,
- Daemon actions, and
- Stopping conditions.

Daemon actions are all those actions that cannot be performed by single ants. Examples of such daemon actions are the execution of local search algorithms to refine the constructed solutions, or the collection of global information which is used

to add more pheromone amounts to links. Stopping conditions are usually a set of criteria to terminate the search.

An ACO algorithm utilizes a colony of ants to search for multiple solutions in parallel. Ants traverse the entire search space, with each ant incrementally building a solution. Ants depend on local information in the form of pheromone deposits and heuristic information to decide which node to move to next. Ants also modify the pheromone concentration on each link in their paths, which enable cooperation among the ants to indirectly exchange information about the desirability of components. To achieve the task of constructing the optimal paths, ants in ACO algorithms have the following properties and characteristics [29].

- Each ant is assigned an initial state, which corresponds to the starting node.
- Each ant determines a set of feasible neighbourhoods of its current state. The feasible neighbourhood nodes must have valid transitions to the ant's current state.
- Each ant uses a probabilistic transition rule to select the next node from the set of feasible neighbourhood nodes.
- Each ant has the ability to modify the pheromone concentrations on each link visited as a means of communicating with other ants.
- An ant has a memory to keep information about the path constructed. The stored information enables ants to backtrack on the constructed path to reinforce pheromone concentrations on the links of that path.
- One or more termination conditions are associated with each ant.

The purpose of optimisation algorithms is to find the minimum or maximum cost of specific optimisation problems. However, not every optimisation problem can be

solved by ACO algorithms. When handling a specific optimisation problem, ACO algorithms require the problem to be represented by a graph, consisting of a finite number of nodes and links between nodes. Each node represents one of the components, and a link represents a transition from one node to the next. Each link is associated with a cost, and the objective of an ACO algorithm is to traverse this graph, in order to construct a minimum cost path, which represents a solution to the optimisation problem. Moreover, problems that ACO algorithms can be applied to must satisfy the following conditions [29, 30]:

- The search space is discrete;
- A set of finite constraints;
- A solution, usually represented as an ordered sequence of components;
- A cost function, which associates a cost to each solution generated by the search algorithm; each component added to a solution contributes to the total cost of the solution;
- A finite set of components from which solutions are constructed;
- A finite set of possible transitions among the components;
- A finite set of sequences of components, representing all possible valid combinations of components, to define the complete search space; validity, or rather feasibility, of sequences is determined by the set of constraints.

Due to their distributed and adaptive characteristics, one desirable feature of ACO algorithms is that they may allow for enhanced efficiency when the representation of the problem under investigation is spatially distributed and changing over time [30]. Network routing problems fall into a subset of problems that ACO algorithms can perfect entirely deal with. In fact, the dynamic nature of the routing problems has

made ACO algorithms for routing in communications networks particularly successful [30]. The ACO based routing protocols like AntNet [31], designed for packet-switched networks, have largely outperformed some of the state-of-the-art routing algorithms. These promising results provided the confidence for applying ACO based algorithms to the more dynamic and challenging network routing problem, MANET routing optimisation being such a one. In *Section 4.3.3*, a general framework of the basic ACO algorithm will be provided.

4.3.3 The framework of the ACO algorithms

The similar characteristics and common components of different variations of ACO algorithms that have been provided in the previous section culminated in a number of efforts to provide a general framework for ACO algorithms. One of the first ACO algorithm frameworks was introduced by Dorigo and Di Caro [32, 33], referred to as the ant colony optimisation meta-heuristic (ACO-MH). The ACO-MH behaviour is described in *Algorithm 4.2* , which illustrates how the components mentioned in *Section 4.3.2* are combined.

Algorithm 4.2 ACO Meta-heuristic

```
procedure ACO_meta-heuristic()
    1:  BEGIN:
    2:  WHILE termination_condition_not_satisfied
    3:      schedule_activates
                ants_generation_and_activity();
        pheromone_evaporation();
        daemon_acti          ons(); //optional
    4:  END WHILE
END procedure
procedure  ants_generation_and_activity()
    5:  WHILE available_resources()
    6:      schedule_the_creation_of_a_new_ant();
            new_activ_ ant();
    7:  END WHILE
```

```
END procedure
procedure new_active_ant()
    initialize_ant();
    M = update_ant_memory();
    WHILE current_state ≠ target_state
        A = read_local_ant-routing_table();
        P = compute_transition_probabilities(A, M, Ω);
        next_state = apple_ant_decision_policy(P, Ω);
        move_to_next_state(next_state);
        IF online_step-by-step_pheromone_update
            deposit_pheromone_on_the_visted_arc();
            update_ant_routing_table();
        END IF
        M = update_internal_state();
    END WHILE
    IF online_delayed_pheromone_update
        FOR each visited arc DO
            deposit_pheromone_on_the_visited_arc();
    update_ ant_routing_table();
        END FOR
    END IF
    terminate_ant();
END procedure
```

The high-level description in *Algorithm 4.2* outlines the three main components of ACO algorithms, namely *ants_generation_and_activity*, *pheromone_evaporation*, *and daemon_actions*, without specifying any details concerning how these are scheduled and synchronized. This allows both sequential and parallel implementations of ACO algorithms to be specified. Action *ants_generation_and_activity* creates a new ant and activates that ant. The procedure new_active_ant is called for each ant to construct a path. The procedure refers to an ant routing table, which maintains the routing information for each ant. Each entry of an ant's routing table is a value obtained by a functional composition of the pheromone and heuristic values of the link of the corresponding nodes. For most of the ACO algorithms, the function to calculate the

fitness metric for the link established between node i and j is similar to *Formula 4.4*, where τ, η represents the pheromone and heuristic values respectively.

$$\frac{\tau_{ij}^{\alpha}(t)\eta_{ij}^{\beta}(t)}{\sum_{u\in\mathcal{N}_i(t)}\tau_{iu}^{\alpha}(t)\eta_{iu}^{\beta}(t)} \qquad \textit{Formula 4.4}$$

Action *pheromone_evaporation* is a process of decreasing the intensities of pheromone trails over time, which enables ants to explore more search space, and prevents them from being trapped in a local optimal solution. The procedure *daemon_actions* is often used to gather useful global information by depositing additional pheromone. Since not all algorithms make use of *daemon_actions*, they are indicated as optional.

4.4 Summary

This chapter provides an overview of the main areas of swarm intelligence techniques, namely, PSO and ACO algorithms. Each algorithm is described from the perspective of how the algorithm originated from and the characteristic and general framework of the algorithm.

The PSO algorithms are similar to the evolutionary algorithms reviewed in *Chapter 2*. The main limitations of EAs and PSO algorithms are their computational inefficiency due to their iterative structure [16]. For highly dynamic networks, such as mobile ad hoc networks (MANETs), the topology of wireless links is rapid and constantly changing. In addition, the resources of such networks are usually extremely limited. These characteristics and properties become the bottleneck constraints for real time routing optimisation of such networks, and moreover, make it almost impractical to

apply the iterative based algorithms such as PSO and GA for solving the real time routing optimisation problems of such networks.

However, in the case of ACO algorithms, although the application types to which they may be applied are limited and have to satisfy a set of constraints, they have been extraordinarily successful in solving routing optimisation problems. Many ant based protocols including the ant based protocols for the MANETs have outperformed traditional classical routing protocols. These facts provide confidence in choosing ACO algorithms as a possible means to solve the MANET routing problem in this research work. In the following chapters (*Chapter 5* and *Chapter 6*), original work on MANETs routing optimisation will be presented.

References

[1] G. Beni and J. Wang, "Swarm Intelligence in Cellular Robotic Systems," in *Proceed. NATO Advanced Workshop on Robots and Biological Systems* , Tuscany, Italy, 1989.

[2] J. Kennedy and R. C. Eberhart, "Particle swarm optimization," in *Proc. IEEE int'l conf. on neural networks* , IEEE service center, Piscataway, NJ, 1995, pp. 1942-1948.

[3] A. P. Engelbrecht, *Fundamentals of computational sw arm intelligence* . Chichester: John Wiley, 2005.

[4] O. B. Bayazit, J. M. Lien, and N. M. Amato, "Roadmap-based flocking for complex environments," *10th Pacific Conference on Computer Graphics and Applications, Proceedings,* pp. 104-113, 2002.

[5] M. J. Mataric, "Interaction and intelligent behavior," in *Department of Electrical, Electronic and Computer Engineering*. vol. PhD thesis: MIT, 1994.

[6] C. W. Reynolds, "Flocks, herds, and schools: a distributed behavioral model," *Computer Graphics,* vol. 21, pp. 25-34, 1987.

[7] B. L. Partridge, "The structure and function of fish schools," *Scientific American,* vol. 246, pp. 114-&, 1982.

[8] T. J. Pitcher, B. L. Partridge, and C. S. Wardle, "Blind fish can school," *Science,* vol. 194, pp. 963-965, 1976.

[9] F. A. Sharpe, "Social foraging of southeast alaskan humpback whales." vol. PhD thesis, Burnaby, British Columbia: Simon Fraser University, 2000.

[10] C. H. Janson, "Experimental evidence for spatial memory in foraging wild capuchin monkeys, Cebus apella," *Animal Behaviour,* vol. 55, pp. 1229-1243, 1998.

[11] J. Kennedy and W. M. Spears, "Matching Algorithms to Problems: An Experimental Test of the Particle Swarm and Some Genetic Algorithms on the Multimodal Problem Generator," *Proceed ings of the IEEE Congress on Evolutionary Computation (CEC),* pp. 78-83, 1998.

[12] M. M. Millonas, "Swarms, phase transitions, and collective intelligence," in *Artificial Life II I*, C. G. Langton, Ed. Reading, MA: Addison-Wesley, 1994, pp. pp.417-445.

[13] R. C. Eberhart and Y. Shi, "Particle swarm optimization: developments, applications and resources," *Evolutionary Computation,* vol. 1, pp. 81-86, 2001.

[14] A. P. Engelbrecht, "Basic particle swarm optimization," in *Fundamentals of computational swarm intelligence* Chichester: John Wiley & Sons, Ltd, 2005, pp. 93-129.

[15] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory " pp. 39-43, 1995.

[16] G. Ciuprina, D. Loan, and I. Munteanu, "Use of intelligent-particle swarm optimization in electromagnetics," *IEEE Transactions on Magnetics,* vol. 38, pp. 1037-1040, 2002.

[17] J. J. Liang, A. K. Qin, P. N. Suganthan, and S. Baskar, "Comprehensive learning particle swarm optimizer for global optimization of multimodal functions," *Evolutionary Computation, IEEE Transactions on,* vol. 10, pp. 281-295, 2006.

[18] X. D. Li and A. P. Engelbrecht, "Particle swarm optimization: An introduction and its recent developments," in *Proc. Genetic Evol. Comput. Conf.*, 2007, pp. 3391-3414.

[19] R. Becker, J. L. Deneubourg, and S. Goss, "Trails and U-turns in the selection of the shortest path by the ant lasius niger," *Journal of Theoretical Biology,* vol. 159, pp. 397-415, 1992.

[20] J. L. Deneubourg, S. Aron, S. Goss, and J. M. Pasteels, "The self-organizing exploratory pattern of the argentine ant " *Journal of Insect Behavior,* vol. 3, pp. 159-168, March 1990.

[21] J. M. Pasteels, J. L. Deneubourg, and S. Goss, "Self-organization mechanisms in ant societies. I: Trail recruitment to newly discovered food sources," *Experientia. Supplementum,* vol. 54, pp. 155-175, 1987.

[22] A. P. Engelbrecht, "Ant colony optimization meta-heuristic," in *Fundamentals of computational swarm intelligence* Chichester: John Wiley & Sons, Ltd, 2005, pp. 93-129.

[23] M. Dorigo, "Optimization, learning and natural algorithms." vol. PhD thesis: Politecnico di Milano, 1992.

[24] M. Dorigo, "Learning by probabilistic boolean networks," in *Proceedings of the IEEE International Conference on Neural Networks*, 1994, pp. 887-891.

[25] L. M. Gambardella and M. Dorigo, "Ant-Q: A reinforcement learning approach to the TSP," in *Proceedings of ML-95, 12th International Conference on Machine Learning* 1995, pp. 252-260.

[26] L. M. Gambardella and M. Dorigo, "Solving symmetric and asymmetric TSPs by ant colonies," in *Proceedings of IEEE Interna tional Conferen ce on Evolutionary Computation*, 1996, pp. 622-627.

[27] V. Maniezzo and A. Carbonaro, "Ant colony optimization: An overview " in *Essays and Surveys in Metaheuristics* , C. Ribeiro, Ed.: Kluwer, 1999, pp. 21-44.

[28] T. Stutzle and H. H. Hoos, "Max-Min ant system and local search for the traveling saleman problem," in *Proceedings of the IEEE International Conference on Evolutionary Computation*, 1997, pp. 309-314.

[29] A. P. Engelbrecht, "General frameworks for ant colony optimization algorithms," in *Fundamentals of computa tional swarm intelligence* Chichester: John Wiley & Sons, Ltd, 2005, pp. 397-407.

[30] E. Bonabeau, M. Dorigo, and G. Theraulaz, "Swarm Intelligence: From natural to artificial systems " New York: Oxford University Press, 1999.

[31] G. Di Caro and M. Dorigo, "AntNet: Distributed stigmergetic control for communications networks," *Journal of Artificial Intelligence Research,* vol. 9, pp. 317-365, 1998.

[32] M. Dorigo and G. D. Caro., "Ant colony optimization: a new meta-heuristic," *Proceedings of the IEE E Congress on Evolutionary Computation,* vol. 2, p. 1477, 1999.

[33] M. Dorigo and G. D. Caro, "The ant colony optimization meta-heuristic," in *New Ideas in Optimiza tion*, D. Corne, M. Dorigo, and F. Glover, Eds.: McGraw-Hill, 1999, pp. 11-32.

Chapter 5:

Unicast Routing Optimisation for Mobile Ad Hoc Network

5.1 Introduction

A mobile ad hoc network (MANET) consists of a collection of wireless nodes, which dynamically form a temporary network, without using any existing infrastructure of wired networks or centralized administration [1]. The highly dynamic nature of MANETs results in frequent changes and unpredictability in network topologies,

adding difficulty and complexity to routing among the mobile nodes within the network. These added challenges, coupled with the critical importance of routing protocols in establishing communications among mobile nodes, make routing optimisation perhaps the most active research area within the MANET domain.

Current MANET routing protocols may generally be categorized as either proactive routing protocols or reactive on-demand routing protocols [2]. Proactive routing schemes like Destination Sequenced Distance Vector (DSDV) [3] constantly maintain and update routing tables of each node in the network, even for those nodes that currently have nothing to transmit. This reduces the available capacity of the network for actual data communication. The reactive on-demand routing protocols, such as Ad hoc On-demand Distance Vector (AODV) [4] and Dynamic Source Routing (DSR) [5], have the advantage of not having to maintain extensive routing tables for all destinations in the network. In AODV, for example, routes are only discovered on an as-needed basis and only maintained as long as they are necessary. However, due to the large end to end time delay, this may not be suitable for real-time data and multimedia communication applications.

In recent years, Swarm Intelligence (SI) based routing algorithms have been proposed for efficient routing in MANET. In general, SI based schemes are hybrid routing protocols that try to take advantage of both proactive and reactive routing protocols to balance the delay and control overhead. Most of these algorithms use combined traditional QoS metrics which often appear in the wired network optimisation literature such as measure of distance, end to end delay and available bandwidth to formulate the ant pheromone value, and in turn to determine the optimal communication paths. For highly dynamic MANET routing optimisation, there is a need for the introduction of new QoS metrics. If a MANET link which is currently

being used to send or receive data communication breaks, this will cause 'flooding' of the whole network, which is very costly for the limited MANET resources. In addition, it is not the case that all the nodes which participate in the real life MANET communication are dynamic. Rather, there are quite a few nodes that take part in the wireless communication activities which may be equipped with virtually unlimited power supplies because they are stationary. If the data is sent via these nodes, then clearly performance will be improved. On the other hand, if this fact is dismissed and all nodes are considered equal, letting them take an equal part in the routing and forwarding of packets, this may not be desirable. In an attempt to address this research gap, a new routing metric (the link stability) is introduced for MANET routing optimisation, and a novel 'Swarm Intelligence based Link Stability routing protocol (SILS)' is developed for MANETs [6].

The SILS routing protocol [6] is a unicast MANET routing protocol, which has a statistically based link stability function embedded in order to cope with real life wireless mobility scenarios. SILS is also the foundation of more complicated Swarm Intelligence based Multicasting with Stability metrics (SIMS) routing protocol, which will be illustrated in *Chapter 6*.

The structure of this chapter is organised as follows. In *Section 5.2*, the background to the MANET routing protocols is reviewed. In *Section 5.3*, the detailed design of the SILS unicast routing protocol is illustrated. The simulation and experimental results will be provided in *Section 5.4*. And finally, a summary of the chapter will be given in *Section 5.5*.

5.2 MANET routing protocols

The number of MANET routing protocols has been growing rapidly over recent years and in this section an overview of these will be given. The protocols may be categorised as for example traditional MANET routing protocol, swarm intelligence based hybrid routing protocol and MANET protocol with link stability issues. These are now considered in turn.

5.2.1 Traditional MANET routing protocols

Since the advent of DARPA packet radio networks in the early 1970s [7], numerous protocols have been developed for MANETs. Such protocols must deal with the typical MANET limitations, which include high power consumption, low bandwidth, and high error rate. As shown in *Figure 5.1*, these routing protocols may generally be categorized as: proactive routing protocols or reactive on-demand routing protocols.

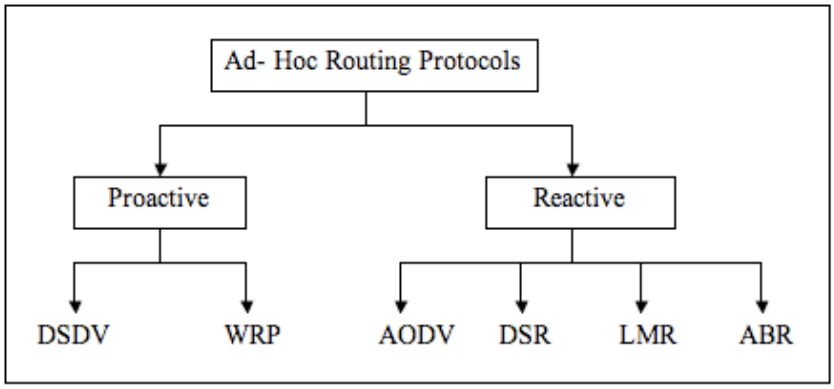

Figure 5.1 Categorization of ad-hoc routing protocols

5.2.1.1 Proactive routing protocols

In proactive (table-driven) protocols, nodes periodically search for routing information within a network. The control overhead for these protocols is foreseeable because it is independent of the traffic profiles and has a fixed upper bound. This is a general advantage of proactive routing protocols.

DSDV [3] is one of the typical proactive routing protocols for MANETs. It is based on the idea of the classical Bellman-Ford routing algorithm with certain improvements such as making it loop-free [2]. DSDV is the foundation of many other distance vector routing protocols such as AODV which is addressed in the next section.

5.2.1.2 Reactive on-demand routing protocols

The reactive on-demand routing protocols represent the true nature of MANETs, which is much more dynamic than infrastructured networks. Instead of periodically updating the routing information, the reactive on-demand routing protocols update routing information when a routing requirement is presented, consequently reducing the control overhead, especially in high mobility networks where the periodical update will lead to significant wasted overhead.

The AODV routing protocol [4] is a typical example of a reactive on-demand routing protocol. AODV is an improvement on DSDV because it typically minimizes the number of required broadcasts by creating routes on an on-demand basis, as opposed to maintaining a complete list of routes as in the DSDV algorithm. The AODV routing protocol will be described in more detail as it is chosen as a benchmark protocol in this research work.

Under the AODV protocol, if a node desires to send information to a destination node, it first looks up its own routing table to see if a valid route exists. If not, the source node broadcasts a *Route Request (RREQ)* message to its neighbours, which then forwards the request to their neighbours, and so on, until either the destination or an intermediate node with a route to the destination is located. Each node maintains its own sequence number, as well as a broadcast ID. The broadcast ID is incremented for

every *RREQ* the node initiates. Along with its own sequence number and the broadcast ID, the source node also includes the most recent sequence number it has for the destination. Intermediate nodes can reply to the *RREQ* only if they have a route to the destination with a destination sequence number equal to or greater than the one listed in the *RREQ*. If additional copies of the same *RREQ* are later received, these packets are simply discarded.

When the *RREQ* reaches the destination or an intermediate node (having a fresher route to the destination), it responds by sending a *Route Reply (RREP)* packet to the source. Periodic 'HELLO' broadcasts are used in AODV by the nodes in the network to inform each mobile node of other nodes in its neighbourhood. These broadcasts are used to maintain local connectivity. If a node along the route becomes unavailable, its upstream neighbour will propagate a link failure notification *Route Error (RERR)* message to each of its active upstream neighbours to inform the removal of that part of the route. A pseudo code of how a node processes *RREQ* message is shown in *Algorithm 5.1*.

Algorithm 5.1 Node *i* processing a *Route Request (RREQ)* Message

```
INPUT:
rreqMsg ← incoming Route Request Message
lastHop ← the node from which rreqMsg is received
BEGIN:
RoutingEntry = RoutingTable.get(rreqMsg.sender)
IF RoutingEntry is EMPTY
    Create a new RoutingEntry for rreqMsg.sender
END IF
IF RoutingEntry.seqID < rreqMsg.seqID
    Update RoutingEntry information:
```

```
            RoutingEntry.seqID = rreqMsg.seqID
            RoutingEntry.nextHop = lastHop
            RoutingEntry.hopsToSender = rreqMsg.hopsToSender
    END IF
    IF THIS_NODE == rreqMsg.destination
        Create Route Reply (RREP) Message
        Send RREP back to rreqMsg.sender
    ELSE IF rreq.TTL > 0
        rreqMsg.hopsToSender = rreqMsg.hopsToSender + 1
        rreqMsg.TTL = rreqMsg.TTL -1
        Rebroadcast rreqMsg
    END IF
```

5.2.2 Swarm intelligence based routing algorithms

AntNet [8], developed by Caro and Dorigo, is one of the well-known ACO based routing algorithms. It was originally designed for routing in packet-switch networks with the objective of optimising the performance of entire network. In AntNet, routing is achieved by launching forward ants at regular intervals from a source node N_s to a destination node N_d to discover feasible low-cost paths and by backward ants that travel from N_d to N_s to update pheromone tables at each intermediate node. The goodness metric of a path is reinforced according to the trip times of forward ants, and this goodness value is used to formulate the communication route from the source node N_s to the destination node N_d. Due to the success of AntNet, many ant based routing protocols [9-12] have been developed for MANETs.

5.2.3 Swarm intelligence versus traditional routing

In [13], Sim and Sun discussed the differences between swarm intelligence based routing and the traditional routing algorithms from the following three perspectives, and a summary of the comparison is given in *Table 5.1*.

Routing information: In traditional routing algorithms, a node N_i depends on the routing information furnished by all its neighbouring nodes to construct a complete routing table. And in SI based routing; the paths from a source to a destination are explored independently and in parallel.

Routing overhead: Generally speaking, routing in traditional routing algorithms is by means of the transmission of routing tables for each node N_i to every one of its neighbours. For a large network N, the routing table of N_i, which consists of a list of cost vectors to all other nodes in N, is large. Since each N_i needs to transmit its routing table to every one of its neighbouring nodes, the routing overhead can be very large. Routing in SI based routing algorithms is achieved by 'transmitting ants' rather than routing tables. Even though it is noted that the size of an ant may vary in different systems or implementations, in general, it is relatively small because ants are generally very simple agents.

Network adaptively: In dynamic networks transmitting large routing tables in short or regular intervals may incur large routing overheads. However, the possible flooding of the routing table at longer intervals may result in slower responses to changes in network topology. Since ants are relatively small they can be 'piggy backed' in data packets. Hence more frequent transmission of ants to provide updates of routing information may be possible. Therefore, using SI based algorithms for routing in dynamic network seems to be an appropriate area for research.

Table 5.1 summarizes the differences between traditional routing and SI based algorithms.

Table 5.1 Traditional routing algorithms vs SI algorithms

	Traditional Algorithms	**SI Algorithms**
Routing Preferences	*Based on transmission time/delay*	*Based on Pheromone concentration*
Exchange of routing information	*Routing information and data packet transmitted separately*	*Can be piggybacked in data packets*
Adaption to topology change	*Transmit entire routing table*	*Frequent Transmission of ants*
Routing overhead	*High Lo*	*w*
Routing update	*Update entire routing table*	*Update an entry in a pheromone table independently*

5.2.4 Link stability issues

In MANETs, the topology of the network changes frequently. Each time a link that is currently communicating breaks it will cause the current package to be dropped and also may cause the source node to perform a “route discovery” process that will result in “flooding” in the whole network, which is very costly. Therefore, the re-routing of existing connections should be carried out as seldom as possible. In other words, routes should be established along “stable”, i.e. durable paths. However, in the design of most MANET routing protocols [3-5, 11], this issue has not been addressed. Most of the protocols focus on parameters such as finding the shortest path to save bandwidth or finding the shortest time delay, which is likely to be a path with generally small hop counts.

There are a few protocols which aim to setup communication networks by establishing communication links via stable routes. An overview of such protocols is provided here. Associativity based routing (ABR) [14] is one of the protocols that prefers “stable” links over “transient” links. A link is considered to be stable if it

exists for a time duration of at least $A_{thresh} = 2r_{tx}/v$, where r_{tx} is the transmission range and v denotes the relative speed of two devices. It is left open how to determine the relative speed v among the mobiles which in turn determines A_{thresh}. Signal stability adaptive routing (SSA) [15] follows a similar approach. It distinguishes strongly connected from weakly connected links. A link is considered to be strongly connected, if it has been active for a certain predefined amount of time. Paths are established exclusively along stable links. Different prediction heuristics for a link's residual life time is introduced in [12] and [16]. Both approaches rely on the availability of GPS receivers to acquire distance and velocity information of neighbouring nodes. Apart from the disadvantages associated with these methods, e.g. unavailability in indoor environments or high demand for battery power. The problem with this approach is that the distance of a receiver is only a very vague estimate of link availability in realistic scenarios.

The proposed SILS protocol employs a simple statistically based method as the link stability function, and combines it with a SI algorithm aiming to utilize the routing bandwidth for MANET under certain mobility criteria. A detailed description of SILS can be found in the next section.

5.3 SILS unicast routing protocol

This section presents the SILS routing protocol, a specially designed MANET routing protocol aiming for the real life wireless mobility scenarios. In real life, the wireless network is often formed by a mixture of dynamic, static and relatively static nodes. For instance, if a wireless network has been setup within an office environment, desktop PCs or charging laptops can be considered as stable nodes, and in a motorway MANET scenario, the vehicles travelling in the same direction with the same speed

will be considered to be relatively stable nodes. The communication links established among these stable or relatively stable nodes are robust, and should be treated differently from the links formed by dynamic nodes.

SILS, a hybrid routing protocol, always tries to seek and prefer communicating via the stable links. This characteristic differentiates it from other SI based protocols. The SILS routing protocol mixes what is called 'link stability factor' with the traditional artificial ants pheromone function, to utilize the MANET bandwidth by trying to minimise 'flooding' in the network. Under the specially designed mobility scenario, the first priority for the SILS routing protocol is to identify the 'stable' links among all the available routes and to transmit data via as many 'stable' routes as possible. The SILS protocol also accounts for the hop count parameter in the routing optimisation. Due to the above mentioned characteristics, ideally, communication links established by the SILS protocol are robust and cost efficient.

5.3.1 SILS data structures

Under the SILS protocol, each node processes a *unique ID*. Each node in MANET is required to maintain a routing table together with an ant decision table. These key data structures will be explained in detail in this section.

5.3.1.1 Ant structure:

An artificial ant (AA) is a simple computational agent that models the foraging behaviour of real ants. AAs can be divided into forward ants and backward ants, with both have the same data structure. The following information is carried by an ant, π:

- The sequence number, *AntSeq*, of the AA;
- The ID of the ant, *AntID*, which is the (*node ID*, *AntSeq*) pair;
- The *sender* field, indicates the originator of the AA;

- The *destination* field, shows the destination the AA is travelling to;
- The forward flag, *forward*, to indicate the type of the AA;
- The number of nodes, m, which π visits, including the node π originated from;
- The nodes visited stack, $visitedNodes_{\pi}$, containing information about all the nodes $V = \{v_1, v_2, \cdots, v_m\}$ along the route, which can be reached by backtracking the ant π's movement;
- The pheromone intensity at node $v \in V$, p_v.

5.3.1.2 Ant decision table

An *ant decision table* at node i, A_i, is a data structure that stores pheromone trail information for routing from node i to a destination d via k possible next hop nodes $J = \{j_1, j_2, \cdots, j_k\}$. This pheromone trail information is used to compute the routing table. Each AA decision table entry $A_i(j, d)$ for node i maintains a row for the next hop-destination pair (j, d) along with the $\tau_i(j, d, t)$, $\eta_i(j, d)$, $\psi_i(j, d, t)$, and $a_i(j, d)$ values described below:

- $\tau_i(j, d, t)$ is the pheromone trail concentration left on a trail ij used as a first hop to destination d at current time t due to all the ants that have traversed the trail, taking into consideration pheromone evaporation. Thus, τ is thereby one of the factors to use to measure the goodness of trail ij. The pheromone amount $\tau_i^{\pi}(j, d, t)$ deposited by a backward ant π traversing from destination node d to the source node i via nodes $j \in J$ is given by the equation:

$$\tau_i^{\pi}(j, d, t) = 1 + \frac{p_i - p_d}{m} \quad (i, j, d \in V) \qquad \textit{Formula 5.1}$$

  where p_i and p_d are the pheromone amounts of ant π at nodes i and d, respectively, the value m is the number of hops that the ant π traversed from

node i to node d, and $V = \{v_1, v_2, \cdots, v_m\}$ denotes the set of m nodes visited by π.

The pheromone evaporation is an important characteristic of the ant colony algorithm. The pheromone evaporation function is used as negative reinforcement of the possible solution found by the AAs. In other words, the longer the pheromone was left on a trail, the smaller is its concentration. The 'evaporate' equations in the SILS protocol are defined as:

$$evaporate(\tau_i(j,d,t),\delta) = \delta \cdot C \cdot \tau_i(j,d,t) \qquad \textit{Formula 5.2}$$

and

$$\tau_i(j,d,t+\delta) = evaporate(\tau_i(j,d,t),\delta) + \tau_i(j,d,\delta) \qquad \textit{Formula 5.3}$$

where δ is the unit time passed since time t, C is the 'evaporate rate', and $\tau_i(j,d,\delta)$ is the pheromone amount laid by the following AAs.

- $\eta_i(j,d)$ is the heuristic value of going from i to d via the next hop j. $\eta_i(j,d)$ laid by the backward ant π, represents a measure of the distance to the destination d from i. The equation to compute $\eta_i(j,d)$ is given as:

$$\eta_i(j,d) = 1 + \frac{1}{m} \quad (i,j,d \in V) \qquad \textit{Formula 5.4}$$

The value m is the number of hops that the ant π traversed from node i to node d, and $V = \{v_1, v_2, \cdots, v_m\}$ denotes the set of m nodes visited by π.

- $\psi_i(j,d,t)$ is the link stability index for link ij that formulates the route to the destination d at current time t. If the time interval between the two periodically broadcasted ants is δ, then the equation for the link stability index is updated by the following group of ants at time $t+\delta$ is using:

$$\psi_i(j,d,t+\delta) \begin{cases} \sqrt{\psi_i(j,d,t)+1} & \text{If link } ij \text{ is still connected at time } t+\delta \\ 0 & \text{Otherwise.} \end{cases} \qquad \textit{Formula 5.5}$$

- The $a_i(j,d,t)$ value for a destination d represents the goodness value of choosing j as the next hop to reach d, and is computed using the following formula:

$$a_i(j,d,t) = \frac{\tau_i(j,d,t)^{\alpha} \times \eta_i(j,d)^{\beta} \times \psi_i(j,d,t)^{r}}{\sum_{l \in J} \tau_i(j,d,t)^{\alpha} \times \eta_i(j,d)^{\beta} \times \psi_i(j,d,t)^{\gamma}} \qquad \textit{Formula 5.6}$$

  where parameters α, β and γ define the relative importance of the pheromone concentration $\tau_i(j,d,t)$, the heuristic value of hop distance $\eta_i(j,d)$, and the link stability index $\psi_i(j,d,t)$, respectively.

5.3.1.3 Routing table

The routing table at node i, R_i, is a table containing the next hop entry to each destination. The routing table entry is computed by the composition of the pheromone values, the local heuristic values and the link stability index. A routing table entry to destination d contains the following information.

- The next hop j'. The communication path setup via the next hop j' to the destination d is the best available route from node i. This also means that the goodness value $a_i(j',d)$ is currently the highest value among those maintained at node i's ant decision table, A_i.
- The current route goodness measurement value $a_i(j',d)$.

5.3.2 Protocol description

The SILS routing protocol for MANETs is a flexible protocol in which the nodes in the network have proactive and reactive capabilities. A detailed outline of the SILS routing protocol is given as follows:

5.3.2.1 The route discovery process

SILS is not purely a proactive protocol, so no control packets will be sent out unless needed. If a route to a destination D is required, but not known at the source node S, S broadcasts forward reactive AAs to discover possible routes to D. Each forward ant has been assigned with an initial pheromone value ϕ_f, the value is decreased every time when the forward AA has been rebroadcasted. The forward AA dies when ϕ_f reaches 0. In this way, the depth of search to the destination node in the network can be determined.

If node X has been reached by a forward ant π, X will first check to see if it is already in π's $visitedNodes_\pi$ stack. If it is the case, that means a 'loop' has occurred, and π will be killed. X also checks the *destination* field of π to find out if it is the desired destination that π is travelling to. If it does not match, X puts its *unique ID* into π's $visitedNodes_\pi$ stack, and rebroadcast AA to its neighbours. Otherwise, if X is the desired destination D, the forward ant π then turns into a backward ant π' and travels back to π's sender S.

Algorithm 5.2 illustrates the process when a forward ant reaches node X.

Algorithm 5.2 Node X processing forward ant packet

```
INPUT:
fant ← incoming forward ant
BEGIN:
IF X ∈ fant.visitedNodes
    RETURN; // Loop occurred
ELSE IF X ≠ fant.destination
    fant.visitedNodes ← fant.visitedNodes ∪ X
    fant.pheromone ← fant.pheromone −1
ELSE IF X = fant.destination
    fant.forward ← FALSE  //Turn fant to backward ant bant
    Update the relevant tables kept at node X
    Send bant to back fant.sender
END IF
IF fant.forward & fant.pheromone > 0 THEN
    Rebroadcast fant
END IF
```

5.3.2.2 The route reinforcement process

Once the communication route to the destination D is established, the originator S will periodically broadcast forward proactive ants to the destination D until the route is no longer needed. In the same way as the process described in *Section 5.3.2.1*, these forward proactive ants will turn to backward ants after the destination D is reached, and then travel back to the originator S. On the way back from D to S, the backward ants deposit pheromone along the route. The amount of pheromone left on a link is calculated using *Formula 5.1*. The backward ants are required to calculate the heuristic value, and update the link stability table according to *Formula 5.4* and

Formula 5.5 respectively. With the aid of these AAs, the route from S to D is reinforced.

5.3.2.3 The route selection process

The entry of the routing table is computed by the composition of the pheromone value, the local heuristic value and the link stability index. After these relevant tables or values have been updated by the backwards ants, the goodness value of each possible link between the originator S and the destination D will then be calculated according to *Formula 5.6*, and in turn the routing table to D will be determined.

Based on the routing table, the data traffic will be distributed to D via each best scoring neighbour in the routing table. This mechanism for selecting routing makes sure that the data are forwarded along the most robust path.

5.3.2.4 The route maintenance process

The SILS protocol has introduced a route maintenance mechanism to deal with the dynamic characteristic of a MANET network.

When a route fails at an intermediate node X, that node buffers the packets which could not be routed and initiates a route discovery to find D. Node X also sends a *RERR* message back to the source node S at the same time. Additionally the pheromone via node X for the path S to D will be decreased, and the link stability vector at X will be reset.

If an *RERR* message is received by the source node S, that node buffers the packets, updates the pheromone table, and resends the packet via the current best available route. The pseudo code for handling the *RERR* message is provided in *Algorithm 5.3*.

Algorithm 5.3 Node K processing a *RERR (Route Error)* message

```
INPUT:
rerrMsg ← incoming Route Error message
BEGIN:
IF K.PheromoneEntry(rerrMsg.destination)≠EMPTY THEN
    K.PheromoneEntry(rerrMsg.destination)=
        2^(-1/2) × K.PheromoneEntry(rerrMsg.destination)
END IF
IF K == rerrMsg.destination THEN
    Re-calculate the best nextHop to the destination
    Re-send data message to nextHop
ELSE
    Keep forwarding rerrMsg until rerrMsg.destination is reached
END IF
```

5.4 Simulation and experimental results

To study the characteristics and evaluate the performance of the SILS routing protocol, simulation experiments have been conducted using the *Simulator for Network Algorithms (Sinalgo) environment* [17] developed by the Distributed Computing Group. The performance of SILS is compared with the ANSI (Ad hoc Networking with Swarm Intelligence) routing protocol [11] and AODV [4] for the same network and load characteristics.

5.4.1 Network mobility models

Node mobility is an important metric when evaluating MANETs and can be modelled in several ways. The performance of the SILS protocol is compared and evaluated under two different network mobility models, the settings applied to these mobility models are explained in the following sections.

5.4.1.1 Random waypoint mobility model

Random waypoint mobility model is a very popular and commonly used mobility model. It is implemented in popular network simulation tools, such as ns-2 [18], OPNET [19] and Sinalgo [17], and have been used in several performance evaluations of MANET protocols [4, 5, 11]. This mobility model is a simple and straightforward stochastic model that describes the movement behaviour of a mobile network node in a given system area as follows (see *Figure 5.2*) [20].

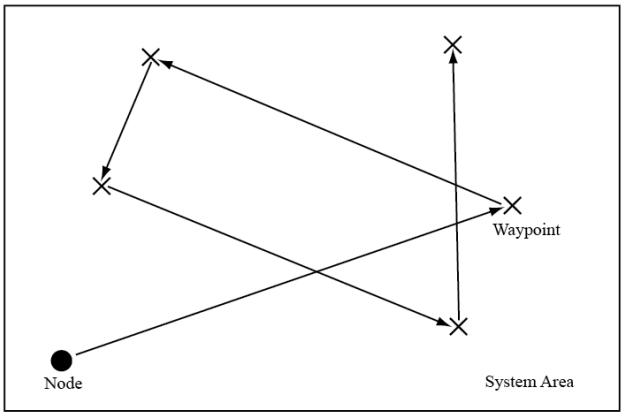

Figure 5.2 Illustration of the random waypoint movement.

A node randomly chooses a destination point (waypoint) in the area and moves with constant speed on a straight line to this point. After waiting a certain 'pause time', it chooses a new destination and speed, moves with constant speed to this destination, and so on. The destination points are uniformly randomly distributed within the system area. The node speed and waiting time are configured using a Gaussian distribution and Poisson distribution respectively.

5.4.1.2 Local area mobility model

This model is an extension of the *random waypoint mobility model* to simulate the real life local area mobility scenario. In this mobility model, a portion of the stationary nodes have been included to represent stationary nodes, such as desktop

PCs or charging mobile equipments, which commonly exist in the real life wireless network. The mobile nodes are implemented under the *random waypoint mobility model* as described in *Section 5.4.1.1*.

5.4.1.3 Network mobility model parameters

In the first set of experiments, all the nodes in the network use the *random waypoint mobility model*. Thanks to the *Sinalgo* simulation tool, the *random waypoint mobility model* has already been implemented; all that is needed is to assign the node mobility speed and the waiting time. In the current implementation of Sinalgo, the node mobility speed is set using a Gaussian distribution, the values were set to mean $\mu' = 1.5$ and variance $\sigma'^2 = 0.2$, which means the majority speed of the moving nodes range between 0.2 and 2.8 $(unit\ distance/unit\ time)$. The waiting time of the nodes is set via the Poisson distribution, and the mean value was set to lambda $\lambda' = 50$. This provides 50 unit times for the average node 'pause time'.

For the *local area mo bility model*, 10 immobile nodes have been deployed. These nodes may represent the desktop PCs or charging mobile equipments that participate in the MANET activities. The remaining 90 nodes are the mobile nodes under the *random waypoint* model. The speed and waiting time are set to the same values as in the previous simulation ($\mu' = 1.5$; variance $\sigma'^2 = 0.2$; $\lambda' = 50$).

5.4.2 Simulation environment setup

All simulations were performed using the same system area consisting of 100 nodes in a physical region of size 200×200 square units. All nodes have been equipped with a wireless network adapter with transmission radius of between 30 to 50 unit distances. To provide better observation of the simulation results, the reliable delivery option has been chosen for the simulation. This option ensures that during the wireless radio transmission, the data will never be lost nor become corrupt. The nodes in all the experiments are initially being distributed using the Grid distribution model as shown in *Figure 5.3*. It ensures that in all of our repeated experiments, the source and the destination nodes have the same initial distance and also have the same node density around them.

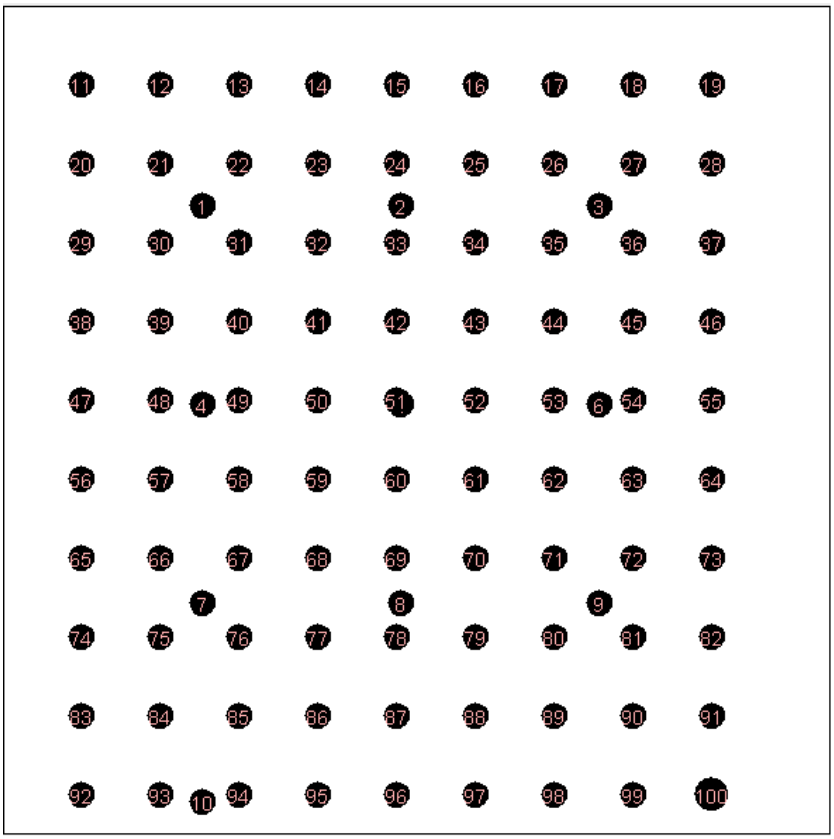

Figure 5.3 The initial distribution of the nodes

5.4.3 Protocols parameter settings

The settings of the parameters for the protocols have a great impact on their performance. When designing the experiments, these settings are carefully considered to ensure that the comparisons between these protocols are fair and meaningful.

The AODV protocol does not have a SI based structure, the implementation and all the settings of the protocol are set up according to [4, 21]. The SI parameter settings for the ANSI and SILS protocols are listed in *Table 5.2*.

Table 5.2 Parameter settings for SI based routing protocol

		ANSI Routing Protocol	SILS Routing Protocol
The pheromone concentration factor	$\boldsymbol{\alpha}$	$\alpha = 2$	$\alpha = 2$
The heuristic factor (hop count)	$\boldsymbol{\beta}$	$\beta = 2$	$\beta = 2$
The link stability factor	$\boldsymbol{\gamma}$	N/A	$\gamma = 2$
The initial ant pheromone amount	$\boldsymbol{\phi_f}$	$\phi_f = 4$	$\phi_f = 4$
Ants deployment time latency	$\boldsymbol{T_f}$	$T_f = 16$	$T_f = 16$

The current implementation of SILS assigns the relative importance of the pheromone concentration α, the relative importance of heuristic value of hop distance β, and the importance of link stability index γ with equal weight, which are $\alpha = \beta = \gamma = 2$. It should be noted that although these parameters are assigned with the same value, the SILS protocol is designed to select the link stability as the primary goal for routing optimisation. Simulation results show that the balance of the heuristic value and the pheromone concentration produces better results for SI based protocol with small or medium sized system area (node size less than 150). For the ANSI protocol, the same settings were provided, which were $\alpha = \beta = 2$. For both the SILS and ANSI protocols, the initial pheromone amount of the ants ϕ_f is set to 4. With ϕ_f too large,

the bandwidth and device battery will be wasted. On the other hand, if ϕ_f is too small, the ability of ants to search is limited, and therefore the optimal routes are likely to be missed. Similarly to ϕ_f, the setting of time the latency to deploy the following ants T_f, could also affect the network bandwidth and the limited resources. Thus, the setting of the parameter is dependent on the network characteristic, and should be set accordingly. In our designed experiments, with $\phi_f = 4$ and $T_f = 16$ time units, provides a good balance of the ant searching power and saving the limited bandwidth for SI based protocol.

5.4.4 Simulation results

The first simulations were performed under the *random waypoint mobility* model to study the performance of our newly developed protocol. For the purpose of comparison, AODV and ANSI are chosen to be the benchmark protocols. As the link stability algorithm of SILS is not designed to detect 'stable' links under the *random waypoint mobility* model, it is expected that the SILS protocol would produce similar results to ANSI, but better performance than AODV.

The first measurement was the link error rate, as it is the main reason why the limited MANET bandwidth gets wasted. After setting up the simulation program, traffic was created using constant bit rate (CBR) UDP flows. The source and destination nodes were chosen and the source node asked to continuously send $10, 20, 35, 50$ data packages to the destination node under each routing protocol. To minimize the statistical error during the simulation, every set of the experiments was repeated 10 times, and the average link failure rate of each protocol recorded and is shown in *Figure 5.4*.

Figure 5.4 Delivery failure count under the random waypoint mobility model scenario

From *Figure 5.4*, it can be seen that the SI based protocols, SILS and ANSI, have much higher package delivery ratio than the AODV protocol. The package delivery failure in SILS/ANSI is reduced by 40%-50% compared to AODV.

Then end to end delay from the source node and destination node was measured under each protocol, producing the graph shown in *Figure 5.5*. As the figure shows, unsurprisingly, the SI based protocols outperforms AODV; especially when transferring large amounts of data packages. Our measurement shows that when continuously transferring 50 packages, AODV requires 3 times longer than the SI based protocols under this designed simulation environment.

Figure 5.5 The end to end delay under the random waypoint mobility model scenario

These experimental results prove that the embedded ACO algorithm for the MANET protocol performs well in increasing MANET's package delivery ratio and reducing the end to end time delay. Another fact that can be observed from the experiments is that SILS and ANSI produce similar results under the *random waypoint network mobility* scenario. This is caused by the rapidly changing network topology under the current mobility model settings. When the topology of the network changes frequently, the SILS protocol's link stability function allocates similar stability score to each available route. Thus, the fitness value of each route is mainly dependent on the pheromone trail concentration and the heuristic value.

In the next set of experiments, the performance of these protocols was compared under the *local area mobility* model. The system area and the position of each node were initialized as in previous simulations. Again, the source node to continuously transferred data to the destination node with 10, 20, 35, 50 units of data packages

respectively under each protocol. Each process was repeated 10 times, and the average number of delivery failure recorded.

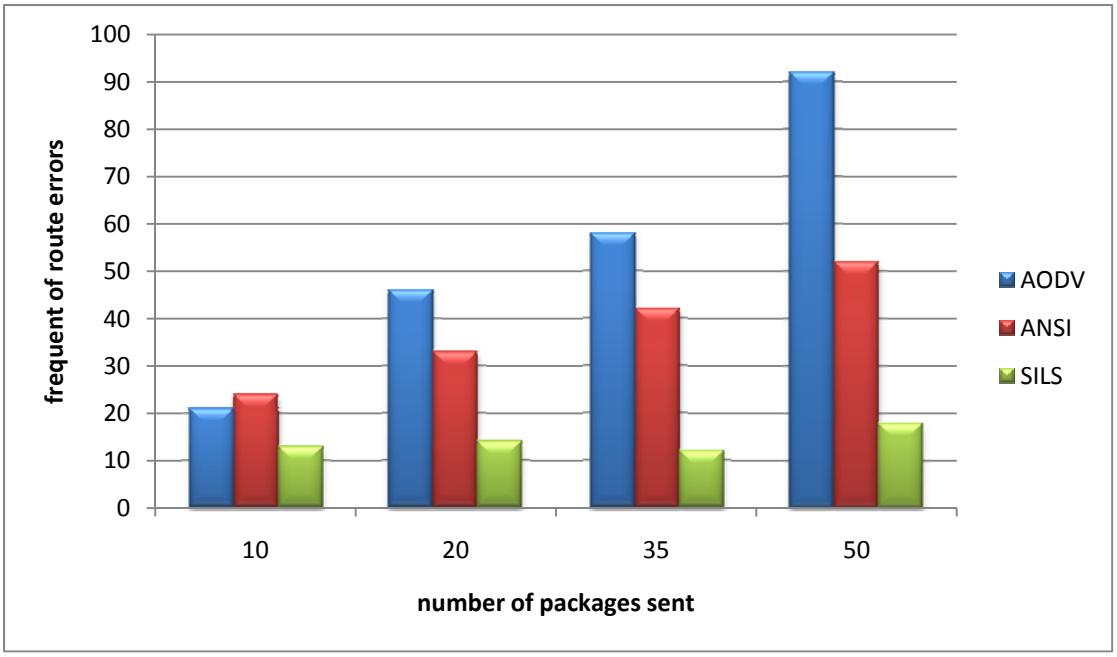

Figure 5.6 Delivery failure count under the local area mobility model scenario

It can be seen from *Figure 5.6* , that the performance of the AODV and ANSI protocols have not been affected much by the change of the mobility model. However, the route error rate via the newly proposed SILS protocol has been dramatically reduced. From the experimental results, it can be calculated that when transferring 50 units of data packages, the amount of delivery failure that occurs in SILS is reduced by around 80% and 67% compared to AODV and ANSI respectively. The reduction in the link failure rate results in faster end to end delivery as shown in *Figure 5.7*. More importantly, higher delivery success rate means there is a decrease in the amount of 'flooding' process that have to be performed for route recovery. Thus, the limited network bandwidth of MANET is utilized.

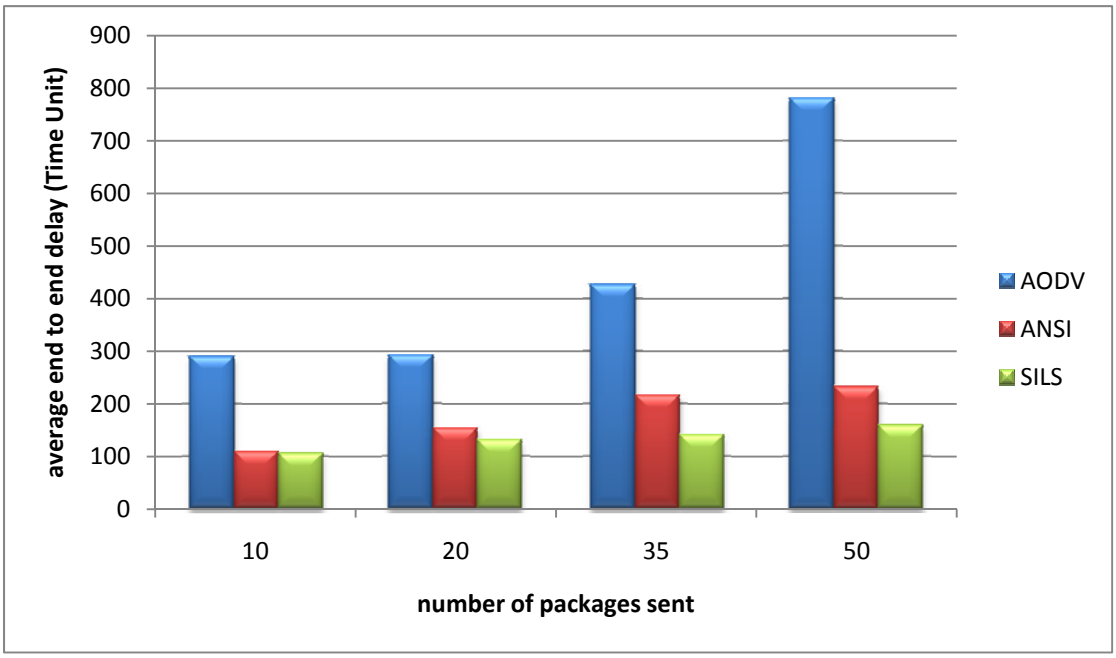

Figure 5.7 The end to end delay under the local area mobility model scenario

These simulations demonstrate that the simple statistical based link stability function can recognize the stable links under the *local area mobility* model.

5.5 Summary

This chapter provides an overview of the current MANET routing protocol and introduces a novel SI based routing protocol for MANET network.

The contents of this chapter are divided into three parts. The first part of the chapter covers the definition, characteristic and background to the MANET network. A comparison between the traditional routing protocols and SI based protocols for MANETs has also been given. One principle that is continually emphasised for MANET routing is to try to setup the communication routes via the 'stable' links whenever there is one available. Communicating via 'stable' links can reduce the number of route recovery process carried out by the routing protocol. In most of the cases, the route recovery process results in 'flooding' the network, which is very

costly for MANET communication. It is noticeable that the 'stable' links broadly exist in the real life wireless communication. However, most of the current MANET routing protocols do not take the advantage of 'stable' links and treat them as being equal to other links, obviously this is not desirable. For these reasons, the SILS protocol was developed. In the second part of this chapter, the SILS protocol is illustrated in detail; including the key data structures and the flow of the SILS routing protocol. In the last part of the chapter, the results of carefully designed simulations from which two main things that may be concluded:

1. The swarm intelligence based protocols can effectively reduce the occurrence of link failure within MANET, and shorten the end to end delay of MANET communication.
2. The statistically based link stability function works fine under the specially designed *local area mo bility* model. The SILS routing protocol reduces the amount of 'flooding' that occurs, thus improves the routing efficiency.

The SILS routing protocol is a unicast routing protocol. A more complex multicast MANET routing protocol will be presented in the next chapter.

References

[1] S. Corson and J. Macker, "Mobile Ad hoc Networking (MANET): Routing protocol Performance Issues and Evaluation Considerations," *Internet Draft: draft-ietf-manet-issues-01.txt,* 1998.

[2] E. M. Royer and T. Chai-Keong, "A review of current routing protocols for ad hoc mobile wireless networks," *Personal Communications, IEEE,* vol. 6, pp. 46-55, 1999.

[3] E. P. Charles and B. Pravin, "Highly dynamic Destination-Sequenced Distance-Vector routing (DSDV) for mobile computers," *SIGCOMM Computer Communication Review,* vol. 24, pp. 234-244, 1994.

[4] E. M. Royer, C. E. Perkins, and S. R. Das, "Ad Hoc On-Demand Distance Vector (AODV) Routing," *Proc. IEEE Workshop on Mobile Computing Systems and Applications,* pp. 90-100, 1999.

[5] D. B. Johnson and D. A. Maltz, "Dynamic Source Routing in Ad Hoc Wireless Networks," in *Mobile Computing*, T. Imielinski and H. Korth, Eds.: Kluwer Academic Publishers, 1996, pp. 153-181.

[6] W. Ren, E. L. Hines, M. S. Leeson, Y. S. Kavian, and D. iliescu, "Swarm intelligence based link stability routing optimisation algorithm for mobile ad hoc network," in *Intelligence Systems: Techniques and Applications*, E. Hines, M. Leeson, and etc., Eds. Netherlands: Shaker Publishing, 2008, pp. 379-398.

[7] J. Jubin and J. D. Tornow, "The DARPA packet radio network protocols," *Proceedings of the IEEE,* vol. 75, pp. 21-32, Jan 1987.

[8] G. Di Caro and M. Dorigo, "AntNet: Distributed stigmergetic control for communications networks," *Journal of Artificial Intelligence Research,* vol. 9, pp. 317-365, 1998.

[9] S. Marwaha, C. K. Tham, and D. Srinivasan, "Mobile agents based routing protocol for mobile ad hoc networks," in *IEEE Global Telecommunications Conference (GLOBECOM 02)*, Taipei, Taiwan, 2002, pp. 163-167.

[10] S. Ziane and A. Mellouk, "A swarm intelligent scheme for routing in mobile ad hoc networks," in *International Conference on Systems Communications* , Montreal, CANADA, 2005, pp. 2-6.

[11] S. Rajagopalan and C. C. Shen, "ANSI: A swarm intelligence-based unicast routing protocol for hybrid ad hoc networks," in *International Conference on Artificial Intelligence (ICAI 05)*, Las Vegas, NV, 2005, pp. 485-504.

[12] Z. Y. Wu, H. T. Song, X. M. Xu, and S. F. Jiang, "Ant-based stable disjoint path routing algorithm in MANETs," in *2nd International Conference on Pervasice Computing and Applications* , Birmingham, ENGLAND, 2007, pp. 297-302.

[13] K. M. Sim and W. H. Sun, "Ant colony optimization for routing and load-balancing: Survey and new directions," *IEEE Transactions on Systems Man and Cybernetics Part a-Systems and Humans,* vol. 33, pp. 560-572, Sep 2003.

[14] T. Chai-Keong, "Associativity-Based Routing for Ad Hoc Mobile Networks," *Wireless Personal Communications,* vol. 4, pp. 103-139, 1997.

[15] R. Dube, C. D. Rais, K. Y. Wang, and S. K. Tripathi, "Signal Stability-Based Adaptive Routing (SSA) for ad hoc mobile networks," *IEEE Personal Communications,* vol. 4, pp. 36-45, Feb 1997.

[16] S. M. Jiang, D. J. He, and J. Q. Rao, "A prediction-based link availability estimation for mobile ad hoc networks," in *20th Annual Joint Conference of the IEEE-C omputer-Society/IEEE-Communication-Society*, Anchorage, Ak, 2001, pp. 1745-1752.

[17] "Sinalgo - Simulator for Network Algorithms," http://dcg.ethz.ch/projects/sinalgo/index.html, (June 2009).

[18] "The Network Simulator - ns-2," http://www.isi.edu/nsnam/ns/, (June 2009).

[19] "OPNET," http://www.opnet.com/, (June 2009).

[20] B. Christian, H. Hannes, and P.-C. Xavier, "Stochastic properties of the random waypoint mobility model," *Wireless Networks,* vol. 10, pp. 555-567, 2004.

[21] C. Perkins, E. Belding-Royer, and S. Das, *Ad hoc On-Demand Distance Vector (AODV) Routing*: RFC Editor, 2003.

Chapter 6:

Multicast Routing Optimisation for Mobile Ad Hoc Network

6.1 Introduction

In the previous chapter, the SILS routing protocol [1] has been introduced for MANETs. This protocol employs a simple and straight forward statistical algorithm which is able to identify 'stable' links in certain wireless networking environments. The results of the statistical algorithm are then combined with artificial ants for MANET routing optimisation. Experimental results prove that under the *local area*

mobility model (see *Section 5.4.1.2*), the SILS protocol is able to effectively reduce network bandwidth wastage and reduce the end to end time delay for MANET unicast routings.

In this chapter, the 'link stability' concept from *Chapter 5* is continued but with a focus on designing a MANET routing protocol which enables multicast routings. Multicast routing is an important type of cast property for MANETs. It comes into play when a node needs to send the same message or stream of data to multiple destinations. In MANETs, the role of multicast services is potentially even more important because bandwidth and battery consumption can be reduced through multicast packet delivery. Supporting multicast routing under highly dynamic network situations is a major challenge for MANET multicasting routing protocols [2]. In this work, the SIMS (Swarm Intelligence based Multicasting with Stability metrics) routing protocol for MANETs has been developed and will now be presented. A new link stability adaptive algorithm based on the Kaplan-Meier statistical estimator is implemented for SIMS. Compared to the link stability algorithm that was employed by the SILS protocol, the new statistical algorithm is able to predict link stability under a wider range of mobility models.

The remainder of this chapter is organized as follows. In the next section, a number of MANET multicast approaches and protocols are reviewed. In *Section 6.3*, a number of possible methods to formulate the link stability metrics are discussed and the Kaplan-Meier survival estimator is introduced. Then in the next section, the detailed design and implementation of the SIMS protocol are presented. Finally, in *Section 6.5*, the performance of SIMS is evaluated and compared with other benchmarking MANET multicast routing protocols.

6.2 An overview of MANET multicast routing protocols

In a typical MANET environment, group-oriented communication is much more popular than one-to-one communication. Hence, multicast plays an important role in MANETs. Multicast protocols used in static networks do not perform well in MANETs due to their random node mobility and limited channel bandwidth.

There are various multicast protocols [3-8] that have been proposed to perform multicasting for MANETs. According to the type of routes the protocol creates, they can be classified as tree based, mesh based and hybrid multicasting protocols.

6.2.1 Tree based multicasting protocols

The tree based concept is inherited from the multicasting protocols utilised in wired networks, since efficiency can be achieved and robustness is not a critical issue in static networks. This type of protocol [4-6] forms a tree infrastructure with the source node as the root, thus there is only one single path between every pair of sender and receiver. Obviously, it is very efficient since only a small amount of routing information needs to be maintained.

MAODV (Multicast On-demand Distance Vector routing protocol) [4], an on-demand tree based protocol, is the multicast extension of AODV [9]. The multicast group is organized by using a tree structure, composed of the group members and routers [4, 10]. Broadcast is used to find the route and to construct the shared routing. A node which wants to join a multicast group or has data to send will broadcast a *Route Request (RREQ)* message. This message will be rebroadcasted by all the intermediate nodes until it reaches a member of the multicast tree node (see *Figure 6.1(a)*). The tree node can then reply with a *Request Reply (RREP)* message by unicast along the reverse path to the sender as illustrated in *Figure 6.1(b)*. If more than one *RREP*

message is received by the sender, the best route based on sequence number and hop count will be selected. An activation message will be unicasted along this selected path. Every intermediate node on this path will act as a router node, and add itself to the tree (see *Figure 6.1(c)*). This ensures the multicast tree has only a single path to any tree node. MAODV uses hard state in its routing table as opposed to the soft state. That is to say, the state information is updated when failure occurs rather than being updated periodically. If a link failure occurs, a route repair process will be performed.

MANSI (The Multicast for Ad hoc Networks with Swarm Intelligence) [5] is an on-demand, swarm intelligence based tree type multicast protocol. Similar to MAODV, MANSI determines a set of intermediate nodes which are called forwarding set to establish the multicast connections. A core which is the first active source within the

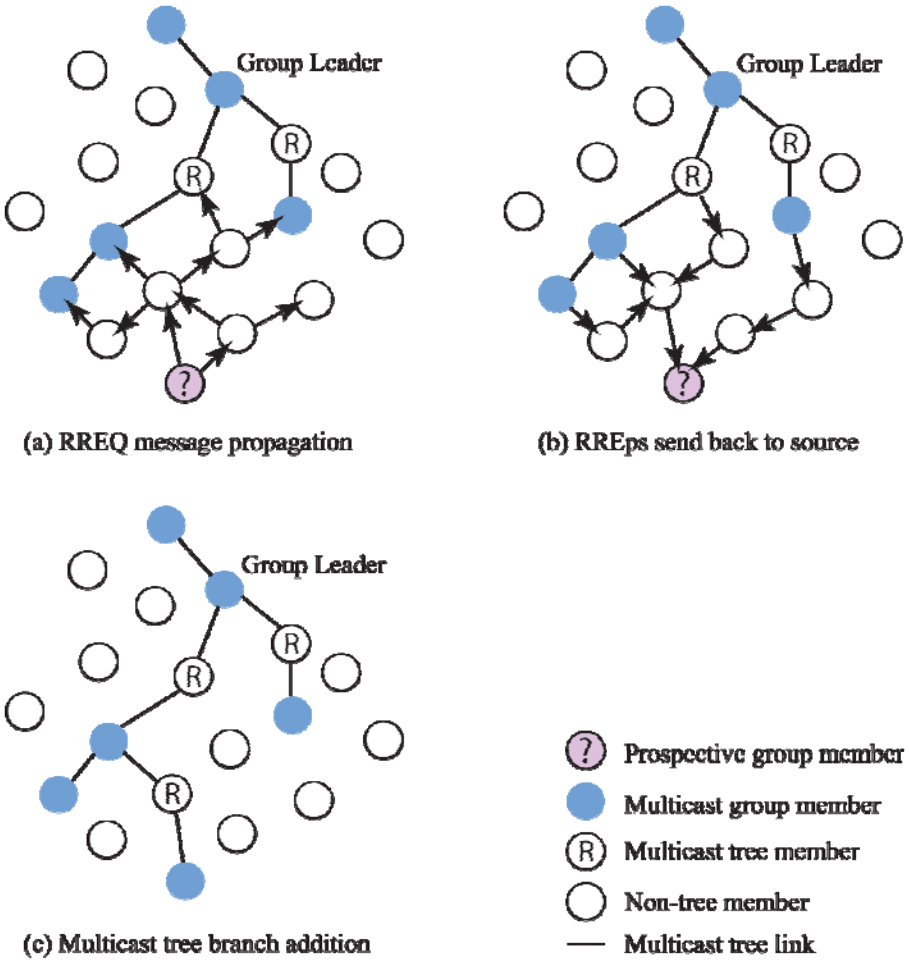

Figure 6.1 Join procedure in MAODV

group is responsible for the creation and maintenance of the multicast tree. MANSI does not rely on any unicast routing protocol. To maintain connectivity and allow new members to join, the core floods *CORE ANNOUNCE* periodically as long as there is more data to be sent. This process creates a forwarding set consisting of all the intermediate nodes on the paths on which *CORE ANNOUNCEs* are accepted and forwarded from the core to other group members, which are often shortest paths (see *Figure 6.2(a)*). However, group connectivity can be made more efficient. From *Figure 6.2(b)*, it can be see that if the group member *A* connecting the multicast tree via the forwarding set node *D*, the size of the forwarding set could be reduced, which lowers the total cost of forwarding data packets. Those possible shortcut routes are discovered with the help of artificial ants (AAs).

In order to optimise the multicast tree, each non-core node deploys *Forward Ants* that explores for better paths toward the core. If a *Forward Ant* arrives at a forwarding set node, it turns itself into a *Backward Ant* and travels back to its originator via the reverse path. On the way back, the *Backward Ant* estimates the cost of the discovered route and deposits pheromone along the path. These pheromone amounts are then used by subsequent *Forward Ants* that arrive at this node to make a decision which node they will travel to next, similar to how pheromone is used by biological ants. The pheromone amount, as well as the computed cost of the route, is updated on the node's local data structure.

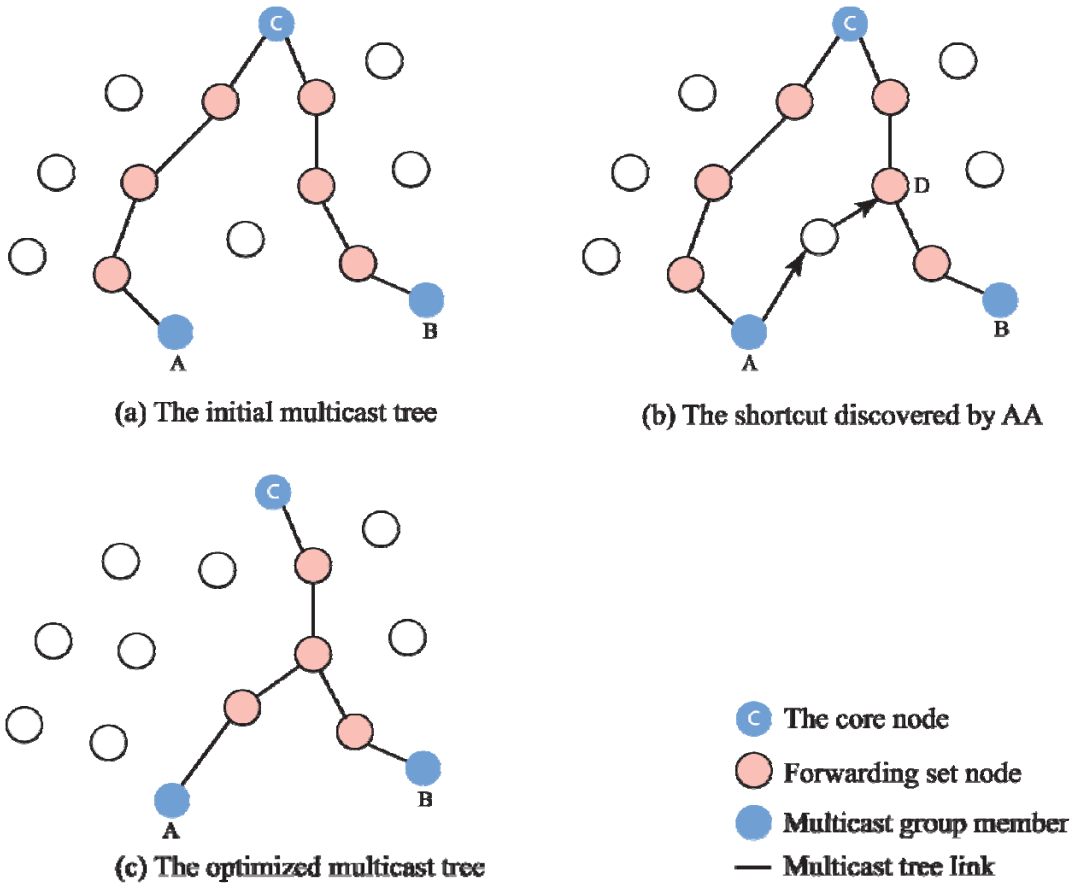

Figure 6.2 Multicast tree creation in MANSI

Sometimes multicast group members attempt to establish group connectivity via one another's forwarding path and none of them remains connected to the core, this phenomenon is defined as the race condition in MANSI. To prevent this happening, *height* is introduced. Each node of a forwarding set is associated with a *height* which is identical to the highest ID of the nodes that use it to connect to the core. In addition, the core has its height set to infinity. *Figure 6.3* shows an example illustrating how heights are assigned to forwarding nodes. A *Forward Ant* must stop and turn into a *Backward Ant* only when it meets a forwarding set node whose height is greater than the node ID who originated the ant. It ensures that the core, whose height is always the highest, will eventually be connected to all other members.

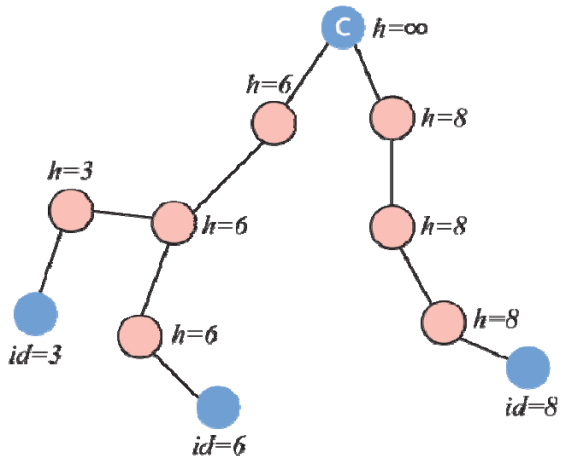

Figure 6.3 Example of node height allocation

By following the above rules, a majority of *Forward Ants* from each multicast group member will choose a path that connects to an existing forwarding set node with a smaller total path cost.

6.2.2 Mesh based multicasting protocols

In contrast to the tree based protocols, a mesh infrastructure builds multiple paths from senders to receivers. The resultant high redundancy structure delivers higher reliability because packets can be delivered even in the presence of links breakage. Thus in comparison with the tree based infrastructures, mesh based protocols [3, 7] are more robust but less efficient.

The On-Demand Multicast Routing Protocol (ODMRP) [7] is an on-demand, mesh-based multicast protocol that attempts to establish a forwarding group to forward the multicast messages. The source node of the group reactively establishes the routes via broadcasting the *Join Data* packet (see *Figure 6.4(a)*). If the non-duplicate *Join Data* is received by an intermediate node, it updates its routing table by adding the upstream node ID onto it. *Join Data* packet will be forwarded until a group member

node is reached. After receiving the *Join Data* packet, the group member then creates and broadcasts a *Join Table* packet to all its neighbours. When a node receives a *Join Table* packet, it checks if the *Next Node* field matches its own ID. If it does, the node realises that it is on the path to the source and thus is part of the forwarding group, and sets the *Forwarding Group Flag* on. It then broadcasts its own *JOIN TABLE* built upon matched entries. The *JOIN TABLE* is thus propagated by each forwarding group member until it reaches the multicast source via the shortest path (see *Figure 6.4(b)* and *Figure 6.5*). This process constructs (or updates) the routes from sources to receivers and builds a mesh of nodes as shown in *Figure 6.4(c)*, known as the *Forwarding Group* . The *Forwarding Group* is responsible for forwarding the multicast data.

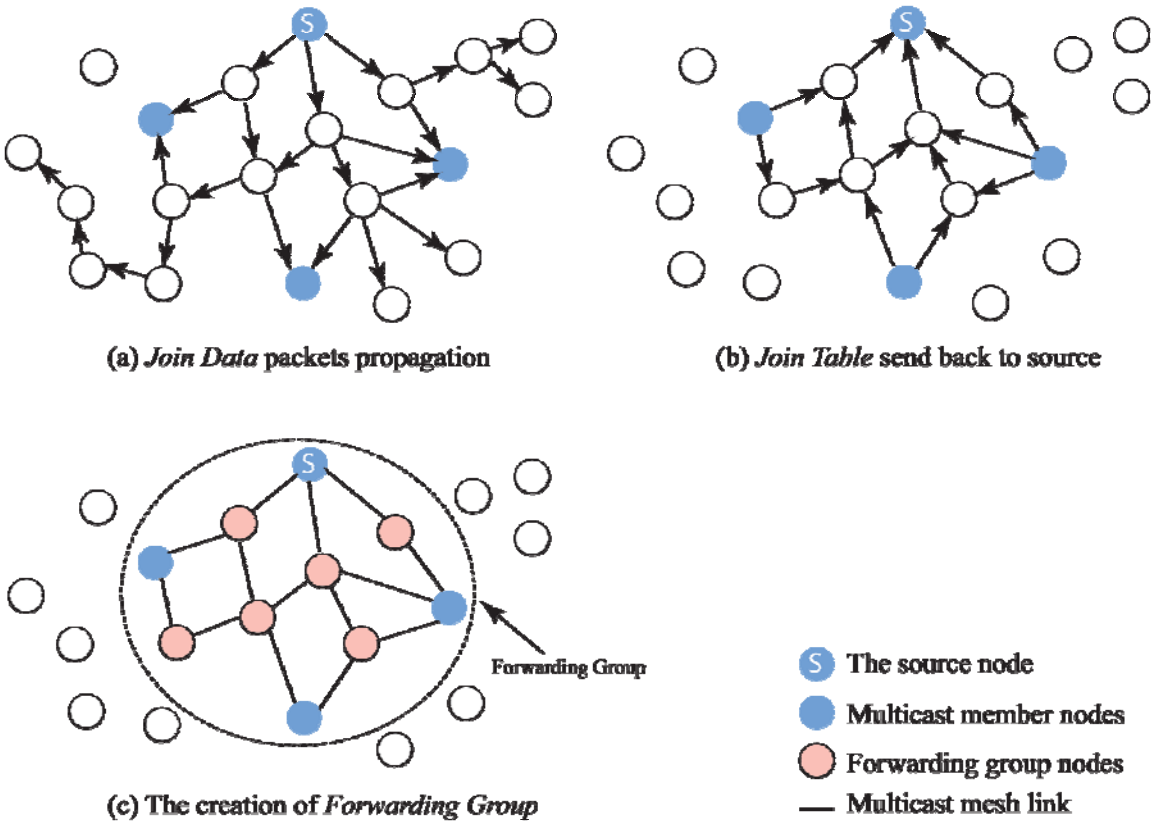

Figure 6.4 Forwarding Group creation in ODMRP

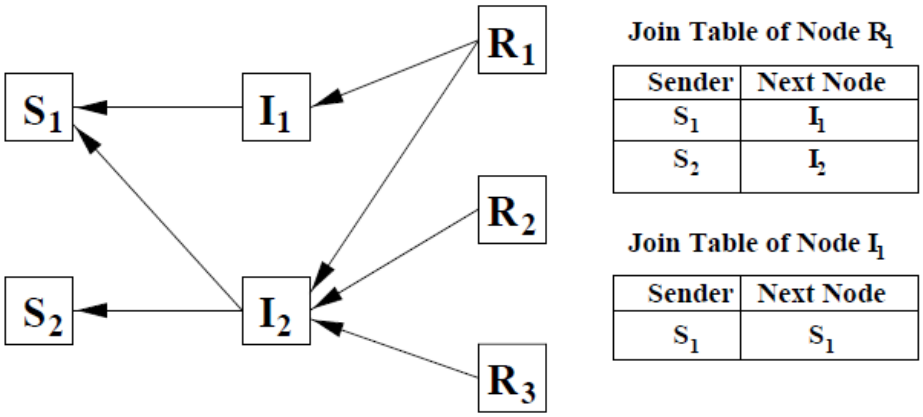

Figure 6.5 Example of ODMRP Join Table forwarding

ODMRP is a soft state protocol and there is no need for the group members to send explicit messages if it wants to leave the group. The *JOIN DATA* is broadcasted periodically by the multicast sender to refresh the membership information and update the routes information.

6.2.3 Hybrid multicasting protocol

AMRoute (Ad hoc Multicast Routing) [8] is one of the hybrid approaches for MANET multicast routings. This protocol is a combination of tree-based and mesh-based methods to seek both efficiency and robustness. It has two main procedures, which are mesh creation and tree creation. Initially, each group member declares itself as a logical core for its own group of size one. The Logical core is responsible for member and tree maintenance. To discover other disjoint mesh segments of the group, each core periodically broadcasts *Join Requ est* messages. When a *Join Request* message reaches a group member in a different mesh segment, A *Join Acknowledgment* message will be bounced back, and both nodes mark the other as the mesh neighbour. In this way, the mesh is created. After mesh creation, each core periodically transmits *TREE-CREATE* packets to mesh neighbours in order to build a

shared tree. When a member node receives a non-duplicate *TREE-CREATE* from one of its mesh links, it forwards the packet to all other mesh links. If duplicate *TREE-CREATE* messages are received, a *TREE-CREATE-NAK* is sent back along the incoming route. The node receiving a *TREE-CREATE-NAK* marks the link as mesh link instead of tree link.

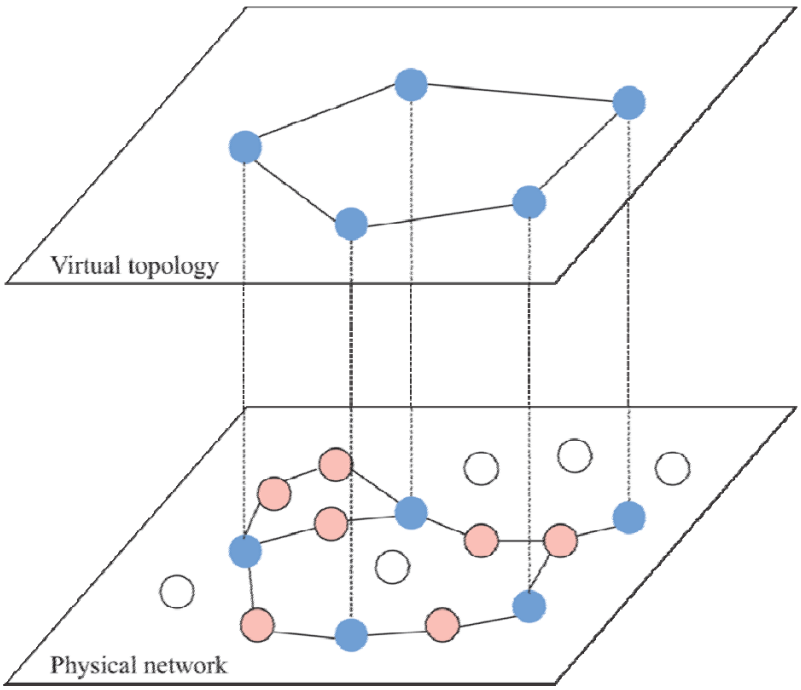

Figure 6.6 An illustration of the AMRoute virtual mesh links

The key characteristic of AMRoute is its usage of virtual mesh links to establish the multicast tree (see *Figure 6.6*). As long as routes between tree members exist via mesh links, the tree need not be readjusted when network topology changes. Non members do not forward data packets and need not support any multicast protocol. Thus, only the member nodes that form the tree incur processing and storage overhead. AMRoute relies on an underlying unicast protocol to maintain connectivity among member nodes. The major disadvantage of the protocol is that it suffers from temporary loops and creates non-optimal trees when mobility is present.

More detailed information about other multicasting protocols can be found in [11]. In [12, 13], performance evaluation and comparison of some described protocols is presented.

6.3 Link stability metrics

The importance of link stability in MANET has been addressed in *Chapter 5*. There are a few MANET protocols that take link stability into consideration when optimising routing. In the context of these protocols, many approaches are based on the assumption that old links are more stable than newer links. However, this is not valid in general for dynamic mobile networks. Other approaches [14-17], such as those reviewed in *Section 5.2.4* , have certain limitations for dynamic MANET communication. Nevertheless, a non-trivial dependency of a link's residual lifetime with respect to its current age maybe observed in many scenarios. This dependency can be utilised to select a stable link among several alternatives.

6.3.1 Possible link stability metric formulations

In this section, a brief overview of the possible formulation of the stability metrics is provided [18, 19].

Link with oldest age. Under several scenarios old links present a large expected residual life time for MANET, for example in the case of the *random waypoint mobility* scenario with long pause times; or the *local are a mobility* scenario with stationary stations which was described in *Section 5.4.1.2*. Thus, in these scenarios it is straightforward and easy to implement a strategy to select the oldest link. However, this strategy is only applicable in scenarios with a high probability of achieving a

sufficient number of long lived links. Otherwise, the oldest link might easily be the one whose expected residual life time is in the local minimum.

Link with youngest age. Almost all scenarios show a reduction in residual lifetimes in the early stages of a link being used, so the obvious choice is to select the fresh links for routing. However, this strategy does not perform well in rather static scenarios that show a large proportion of old links with a large expected residual lifetime, for example in a *local area mobility* scenario.

Link with maximum expected residual life time. The expected residual life time of a link may be calculated from the collected statistical data. Link stability metrics, which are measured from the statistical estimators, promises to be more adaptive than the previously presented link stability metrics. There are a number of statistical estimators that can be employed to estimate the expected link residual life time, the Kaplan-Meier estimator is one feasible option among them. The Kaplan-Meier estimator first appeared in 1958 [20], and was initially used to measure the survival probability of patients in the area of medical science. One attractive feature of the Kaplan-Meier survival analysis is that it has a way to take the incomplete sample data into the survival probability estimation. This feature makes the results of survivability estimation more meaningful and accurate and the estimator is discussed in the next section.

6.3.2 Kaplan-Meier survival probability estimator

In a MANET environment, the patterns of movement followed by the nodes play an important role in the performance of routing protocols. To develop better routing strategies, some researchers have focused on mapping wireless networks to social networks to gather information about different mobility models [21, 22]. The study of

the mobility models involves the measure of the inter-connection time among MANET links, which is a type of survival data. However it is difficult to gather the complete survival data for links in MANET since an inter-connection period is likely to start while the measurement is in progress, but end after the measurement period expires. The data that is impossible to avoid during the measurement are known as the censored data. It is common in network measurements, including the above mentioned studies, that they concentrate on fitting the distribution curve and ignore the censorship issue, which leads to inaccurate evaluation of the mobility model. To address this issue, SIMS employs the Kaplan-Meier estimator (also known as the product limit estimator) [20] to be the link stability function that predicts the MANET link inter-connection time. The Kaplan-Meier estimator was initially applied to measure the survival probability of patients in the area of medical science. The key special feature of the Kaplan-Meier estimator is that it has a method of dealing with censored data. Instead of simply ignoring the censored data, the Kaplan-Meier estimator uses them as incomplete samples and includes them in plotting the survival curves, this makes it a popular estimator in dealing with survival data. In this section the problem of survival data censorship, along with a study of the Kaplan-Meier estimator will be illustrated.

Under the SIMS routing protocol, each node is required to maintain a *neighbour table*, which contains its previous and present neighbouring inter-connection time. The terms '*link age*' and '*link life time*' are defined to represent the inter-connection time of an active link and an inactive link respectively. The *neighbour table* of a randomly selected node with ID *14* under SIMS can be decomposed as shown in *Table 6.1* and *Table 6.2*. From the tables it can be determined that node *14* has established 12 links with its neighbours up to the moment, 6 of them are present and the others are already

disconnected. The inter-connection time of 12 links range from 1 to 5 units of time. Based on this sample, the percentage of survived links after a time t can be calculated by *Formula 6.1*.

$$P_{survival}(t) = \frac{N(t)}{Sample\ Size} \qquad \textit{Formula 6.1}$$

Where $N(t)$ represents the number of current connected link after t.

Table 6.1 Node *14* active neighbour table

Node ID	Link Age
3	3
5	2
8	1
9	3
10	3
15	2

Table 6.2 Node *14* inactive neighbour table

Node ID	Link Lifetime
1	1
2	2
4	3
7	3
11	4
12	5

To determine $N(t)$, *Table 6.1* and *Table 6.2* have to be considered together as rearranged in *Table 6.3*.

Table 6.3 Links duration distribution table *I*

Link Lifetime	Sample Size	Active Links (censored)	Inact. Links	Surviving Links
1	12	1	1	?
2	?	2	1	?
3	?	3	2	?
4	?	0	1	?
5	?	0	1	?

When producing *Table 6.3*, problems occurred. The questions are: What shall be put in the question mark fields in the table. What shall be done with the actively connected links with *link age* a. In fact, there is no way to determine how long these active links will remain connected, thus the *link life time* for those links cannot be measured yet. However, if they were simply omitted from the study, valuable

information about the network could be lost. These data are referred to as censored data within the context of the Kaplan-Meier procedure. Kaplan and Meier, recognized that any attempt to salvage this censored information would involve a certain amount of approximation. They proposed a method whereby subjects who become unavailable during a given time period will be counted among those who survive through to the end of that period, but are then deleted from the number who are at risk for the next time period. Based on this rule, *Table 6.3* can then be completed as shown in *Table 6.4*, and the Kaplan-Meier procedure then calculates the survival probability estimate for each of the t time periods, except the first, as a compound conditional probability (see *Table 6.5*).

Table 6.4 Links duration distribution table *II*

Link Lifetime	Sample Size	Active Links (Censored)	Inact. Links	Surviving Links
1	12	1	1	11
2	10	2	1	9
3	7	3	2	5
4	2	0	1	1
5	1	0	1	0

Table 6.5 Links duration distribution table *III*

Link Lifetime	Sample Size	Active Links (Censored)	Inact. Links	Surviving Links	Kaplan-Meier Survival Probability Estimate
1	12	1	1	11	(11/12)=0.9167
2	10	2	1	9	(11/12)×(9/10)=0.8250
3	7	3	2	5	(11/12)×(9/10)×(5/7)=0.5893
4	2	0	1	1	(11/12)×(9/10)×(5/7)×(1/2)=0.2946
5	1	0	1	0	(11/12)×(9/10)×(5/7)×(1/2)×(0/1)=0.0000

For practical computational purposes, the same results shown in *Table 6.5* can be obtained more efficiently by using the Kaplan-Meier product-limit estimator

$$S(t) = \prod_{t_i < t} \frac{n_i - d_i}{n_i} \qquad \textit{Formula 6.2}$$

where $S(t)$ is the estimated survival probability for any particular link with an age of t unit time periods; n_i is the sample size which is being studied at the beginning of time period t_i; and d_i is the number of individuals whose link lifetime have been measured during time period t_i.

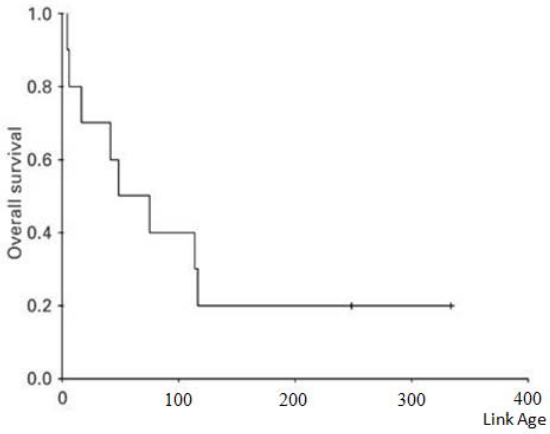

Figure 6.7 An overall survival plot of the Kaplan-Meier estimate

A plot of the Kaplan-Meier estimate of the survival function (see *Figure 6.7*) is a series of horizontal steps of declining magnitude which, when a large enough sample is taken, approaches the true survival function for that population. The value of the survival function between successive distinct sampled observations ("clicks") is assumed to be constant. On the plot, small vertical tick-marks indicate losses, where the MANET link lifetime data has been censored. When no truncation or censoring occurs, the Kaplan-Meier curve is equivalent to the empirical distribution.

6.4 SIMS multicast routing protocol

SIMS is a novel swarm intelligence based multicast MANET protocol which includes some of the best features of various multicast protocols that are described in the previous sections. For example, SIMS uses the tree structure like MAODV [4] and

MANSI [5] to improve the routing efficiency, employs the AAs and the *height* concept from MANSI to reduce the size of the forwarding set, and the mesh structure like ODMRP [7] and AMRoute [8] is also adopted to increase the routing robustness. Moreover, SIMS embeds a wildly adaptive stability function to cope with the dynamic characteristic of MANET. These features make SIMS a protocol with a good balance between the routing efficiency and the routing robustness. The detailed data structure and design process for SIMS will be provided in the following sections.

6.4.1 Overview of SIMS

The SIMS protocol allocates each node in the network a unique *ID*; each multicast group member is assigned a unique *multicast group address* . Although SIMS is described as a hybrid protocol, the multicast group is still organised by using a tree structure at the initial stage. In multicast tree, nodes which are not multicast group members but must exist to form the multicast are called *Routers*. Multicast group members and routers are all tree members and belong to the group tree.

Associated with each multicast tree, the multicast group member who wants to communicate with other multicast group members becomes the *group leader* of the tree. The group leader is responsible for creating and maintaining the multicast tree by the periodically broadcasting *Group-Hello (GRPH) messages*.

SIMS employs the *height* idea as in MANSI [5] to prevent a race condition where two or more members may attempt to join each other's path and isolate them from the rest of the group. Every router node in the multicast tree is associated with a *height* which is identical to the highest *ID* of the multicast member nodes that uses it to connect to its upstream. The *height* for the group leader is set to infinity, and the height of a multicast member node is set to its unique *ID*. In addition, to ensure the group leader

will not be isolated from the multicast tree, the downstream nodes are only allowed to connect to an upstream node with greater *height*. Figure 6.3 illustrates the *height* allocation process.

The embedded link stability metric is another factor that different SIMS from other existing multicast MANET protocols. SIMS embeds the Kaplan-Meier estimator into the swarm intelligence based algorithm in order to find more robust routing for MANET communications.

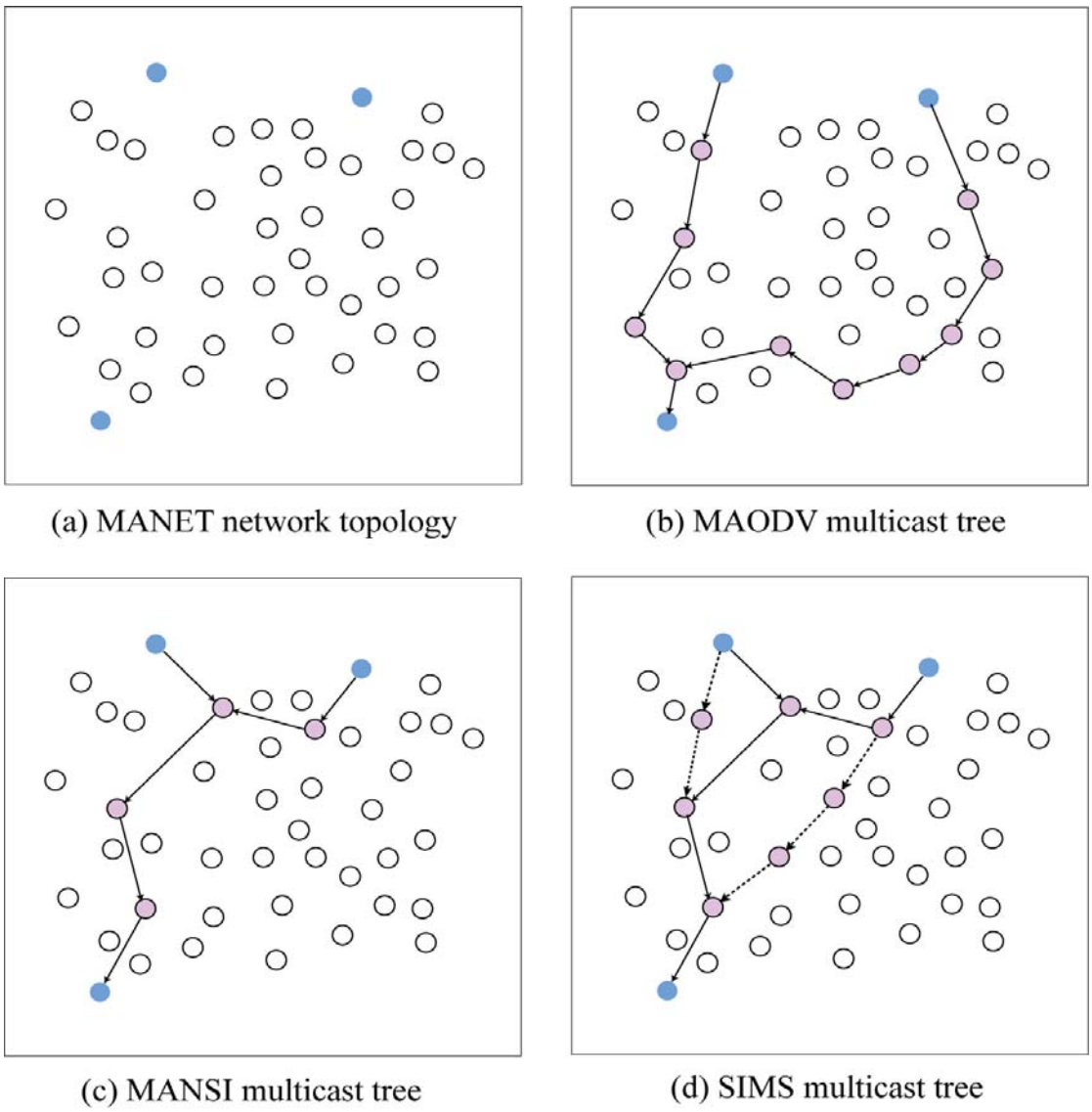

Figure 6.8 Sample network snapshots illustrating different multicast tree creation process

To enhance the robustness of multicast routing, each multicast group member under SIMS optionally keeps an extra set of routing information towards its upstream node, which forms a mesh structure. Although the mesh routing structure created by SIMS is more fragile compared with a pure mesh structure, especially if the backbone multicasts tree node are far apart from each other, SIMS has the advantage of bringing a good balance of robust and efficient for MANETs routings.

Figure 6.8 is a sample network snapshot, which shows the multicast tree creation process performed by different MANET multicast routing protocols.

6.4.2 The key data structure of SIMS

To store the substantial nodes and routing information, a considerable number of tables are managed by SIMS. A node with a unique *ID* i maintains the following tables for the multicast group g.

- *Neighbour Table*, $nbtab_i$: the node neighbouring information are obtained via a neighbour discovery protocol from the MAC or physical layer. The *Neighbour Table* is used by each node to keep its one-hop neighbouring information. $n_i(j)$ denotes a *Neighbour Table entry* corresponding to the neighbouring node j maintained at node i. Each table entry consists of three attributes, *isActive*, indicating if this neighbour is still active; *lastDuration* representing the previous inter-connection time for the link from i to the previous neighbour j, and *duration*, representing the current inter-connection time for the link from i to its currently active neighbour j.
- *Stability Table*, $stab_i$: keeps a metric of the *stability factor* $\psi_i(j,t)$ of links from node i to each neighbour j at time t. The *stability factor* $\psi_i(j,t)$ is a

double value in the range $[0,1]$, which is calculated using the Kaplan-Meier estimator to predict the stability of link ij.

- *Pheromone Table*, $phmtab_i$: keeps a set of *pheromone table entries* $\tau_i(j,h,t)$ for each destination d. $\tau_i(j,h,t)$ represents the pheromone intensity that node i maintains on the link to its neighbour j at time t, with respect to the *height* h. In other words, it indicates the ant π at node i's desirability of choosing j as the next hop to explore, as long as node i is of a lower *height* than h. The allowed range of pheromone intensity is $[0,1]$.
- *Unicast Route Table* , $urtab_i(d)$: keeps a record of the next hop information from i to destination d for unicast traffic.
- *Multicast Route Table*, $mrtab_i(g)$: lists the next hops for the tree structure of the multicast group, and must be maintained by every multicast group member. Each multicast routing entry is associated with a direction either downstream or upstream. The group leader has no upstream, while other nodes in the tree should have one and only one upstream.
- *Group Leader Table,* $gltab_i(g)$: records the current group leader's node *ID* and the next hop towards it. The table will be updated through information from the periodically broadcasted *GRPH* messages.

6.4.4 Ant Data Structure

An artificial ant (AA) is a multi-agent method inspired by the pheromone-based communication of biological ants. In our SIMS protocol an ant π carries the following information, and is updated each time when ant π arrives at the node.

- The forward flag, *forward*, is there to indicate if π is forward or backward. The forward AA is responsible for exploring possible paths towards the

destination. Once a valid path has been discovered, the forward AA turns to a backward AA and travels back to the originator where it was created. It is the backward AA's duty to update the routing and pheromone information on its way back to the originator.

- The *height* indicates the height of the forwarding node found by π. This field is used only after π has been turned into a backward AA.
- The number of hops, m, which π visited.
- The nodes-visited-stack, $visitedNodes_{\pi}$, containing information about nodes $V = \{v_1, v_2, \cdots, v_m\}$, that can be reached by backtracking the ant π's movement.
- The deterministic flag d, is used to indicate if π should always follow the current best route in order to obtain the actual current route cost, and then to decide if the multicast tree needs to be updated.
- The cost limit of the path, $costLimit$, is the hops that π is allowed to traverse after leaving its originator. This cost limit is ignored if the AA is deterministic since its goal is not to find a better cost, but to find the actual current best cost.
- The path pheromone intensity τ of the path that π visited.
- The path stability metric ψ of the path that π visited.

To ensure that the multicast tree constructed is efficient enough, each node maintains a pheromone table, $phmtab_i(d)$, and stability table, $stab_i(j)$ to keep track of the current healthier paths. After a forward ant π reaches the destination d, it turns into a backward ant and travels back to its originator. On the way back to the originator, π updates the heuristic value $\eta_i(j)$, the pheromone table entry, $\tau_i(j, h, t)$ and the stability metric, $\psi_i(j, t)$ according to the following formulas.

- $\eta_i(j)$ is the heuristic value of going from i to j. In our mapping, η is a measure of the distance to the destination going from i to d, when using next hop j.

$$\eta_i(j) = \frac{\beta}{m + \beta} \qquad \textit{Formula 6.3}$$

Where m is the hops count from i to d, and β is a positive integer value.

- $\tau_i(j, h, t)$ represents the pheromone intensity maintains at node i for the link to its neighbour j with respect to the height h after time t. The pheromone amount $\tau_i^{\pi}(j, h, t)$ deposited by a backwards AA π is defined as:

$$\tau_i^{\pi}(j, h, t) = \frac{1}{m + 1} \qquad \textit{Formula 6.4}$$

where m is the hop count from i to d.

An evaporation function is applied to the pheromone trail concentration to avoid convergence to a locally optimal solution. The pheromone intensity at time $t + \delta$ is calculated as follows:

$$evaporate(\tau_i(j, h, t), \delta) = \delta \cdot C \cdot \tau_i(j, h, t) \qquad \textit{Formula 6.5}$$

and

$$\tau_i(j, h, t + \delta) = evaporate(\tau_i(j, h, t), \delta) + \tau_i(j, h, \delta) \qquad \textit{Formula 6.6}$$

where $\tau_i(j, h, \delta)$ is the pheromone deposited on the trail by the following ants during the time span δ and C is the evaporative rate.

- $\psi_i(j, t)$ is the stability metric of the link ij with link age t. $\psi_i(j, t)$ is calculated by the Kaplan-Meier estimator (as shown in *Formula 6.2*), and

ranges [0,1]. The more $\psi_i(j,t)$ is close to 0, the more likely link ij with age t is going to break.

6.4.4 Protocol Description

The operation of the SIMS protocol can be divided into three phases, multicast tree creation, multicast tree evolution, and multicast tree maintenance. *Figure 6.9* lists the brief process of the SIMS protocol.

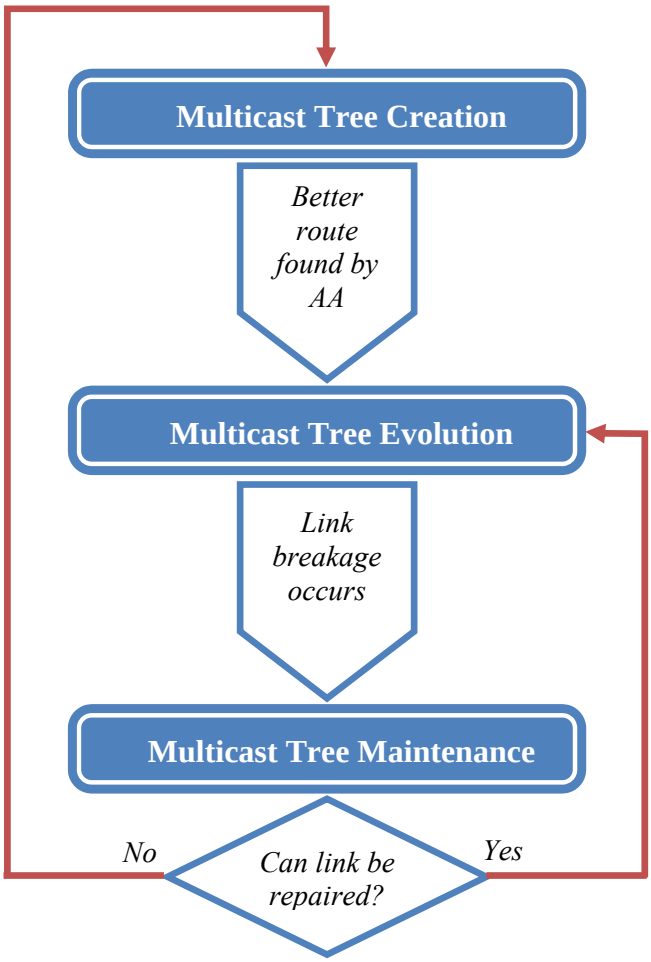

Figure 6.9 The structure of the SIMS protocol

6.4.4.1 Multicast tree creation

The SIMS protocol organises multicast group by using the multicast tree structure at the initial stage. Compared with traditional mesh based protocols, tree structure often

produces less control overhead. Since SIMS is an on-demand based protocol, it does not send out any control packets other than neighbour discovery messages when there is no active source of multicast traffic. If a member of the group has data to send and it sees that a group leader does not exist for the group yet, it declares itself as the group leader and floods the network with a *GRPH* to announce that it has becomes the group leader. The fields of a *GRPH* message contains the multicast group sequence number, the group leader ID, the next hop ID to the group leader, and the cost to the group leader. Under the SIMS protocol, only one group leader is allowed for the multicast group. To ensure that duplicate *GRPH* messages will not be processed, the message will be discarded if an announcement has been seen from the group leader with the same or smaller sequence number. The pseudo code of how node i processes a *GRPH* message is shown in *Algorithm 6.1*.

Algorithm 6.1 Node i processing a *GRPH* Message

INPUT:

grphMsg ← incoming *GRPH* Message

lastHop ← the node from which *GRPH* Message is received

BEGIN:

IF *group_leader*=INVALID_ADDRESS
OR $seqID_i(g) < grphMsg.seqID$

Update local information:

$group\_leader = grphMsg.groupLeader$

$seqID_i(g) = grphMsg.seqID$

Update cost in the Group Hello Message

$grphMsg.hopsToSender = grphMsg.hopsToSender + 1$

Rebroadcast *grphMsg*

END IF

When a fresh *GRPH* message is received by a multicast group member, it updates its group leader table entry with information containing the received *GRPH* message. If a multicast group node finds itself partitioned with no upstream paths, it will initiate a route discovery process by flooding the network with route request (*RREQ*) messages. Only the multicast group members with greater height values can reply to the *RREQ* message with corresponding route reply (*RREP*) message. All the routing information is cached during the propagation of the *RREP* message. Usually the multicast group node may receive more than one *RREP* message, in this case the node chooses the best available upstream node, caches the corresponding information in its multicast route table and produces a new message, multicast route activation (*MACT*), to graft a branch to the tree. After every multicast group member finds and establishes a path to its upstream, the initial multicast tree is then created.

6.4.4.2 Multicast tree evolution

Once the initial multicast tree is created, each group member which is not the group leader deploys a forward AA every ANT_INTERVAL time period in order to attempt to improve the efficiency of the multicast tree. Before an AA can be released, the node has to first decide which hop the ant will travel to, based on the pseudo code given in *Algorithm 6.2*. When the destination node is determined, the AA appends the destination node ID at the end of its visited node list.

Algorithm 6.2 Procedure *ReleaseForwardAnt(fant)* executed by node *i*

1: **PARAMETER:**

2: *fant* ← a forward ant to be released

3: **BEGIN**:

4: Compute a desirability, d_n, for node n, $n \in nbtab(i)$ from summations of only entries whose heights are higher than *fant.height* in the main pheromone table.

$$d_n = \begin{cases} 0 & \text{if } n \in fant.visitedNodes, \\ 1 + \sum\limits_{h > fant.height} \tau_i(n,h,t) & \\ & \text{if } \tau_i(n,h,t) \text{ exists,} \\ 0.5 & \text{otherwise.} \end{cases}$$

5: **IF** $\forall n, d_n = 0$ **THEN**

6: Return /* ant has no place to go*/

7: **END IF**

8: If *fant* is not deterministic (*fant.d=false*) and it is allowed to explore (*fant.costLimit>0*), *fant* decides to randomly choose a next hop *n*, where the probability of choosing *n* depends on its desirability as follows:

$$Prob(n) = {d_n}\Big/{\sum_{k \in nbtab(i)} d_k}$$

fant.costLimit ← *fant.costLimit* -1

9: Append *n* to *fant.visitedNodes* and send *fant* to *n*

Once a forward AA reaches a node j, j checks if its ID matches the node ID at the end of the ant's visited nodes field. If not, the AA is discarded. Otherwise, node j knows that this AA is intended for itself and accepts it. *Algorithm 6.3* shows how a forward AA is processed. First, node j checks if it is currently a forwarding node of the group and its *height* is greater than the ID of the ant's originator. If so, node j realises that the member who deployed the ant is eligible to join the group via node j itself. This ant is then turned into a backward ant by resetting its flag f. The cost metrics $\eta_i(j)$, $\tau_i(j,h,t)$ and $\psi_i(j,t)$ are then reset to zero in order to start computing the total cost on the way back, and the ant's *height* field is set to node j's height. The last entry of

its visited node stack $visitedNodes_{\pi}$ is removed in order to send this ant back to the previous hop. If the condition is not satisfied to convert the ant to a backward ant, node j increases the cost value. It then invokes the pseudo code shown in *Algorithm 6.2* to forward the ant to a next hop, if the updated cost does not exceed the limit.

Algorithm 6.3 Node j processing a forward ant packet.

```
INPUT:
fant ← incoming forward ant
BEGIN:
IF j = last entry in fant.visitedNodes THEN
    IF j.height > fant.height THEN
        Convert fant to a Backward
            fant.destination ← fant.sender
        fant.sender ← j
        fant.height ← j.height
        fant.forward ← FALSE
        Remove last entry from fant.visitedNodes and
            send fant back to fant.destination
    ELSE
        fant.cost ← fant.cost +1
        IF fant.cost < fant.costLimit THEN
            Invoke ReleaseForwardAnt(fant)
        END IF
    END IF
END IF
```

A forward AA turns into a backward AA after a node with a greater *height* value is reached. A backward AA is responsible for measuring the quality of the explored route, which involves the counting of the heuristic values, extracting the pheromone

intensity values and calculating the link stability factors. The SIMS protocol aims to optimise several factors simultaneously, sometimes this results in the multicast tree structure changing too frequently. Moreover, as the stability factor is in fact a statistical estimate, inaccurate prediction does occur, which will result in the multicast tree becoming unstable and increases its maintenance cost. To overcome these issues, SIMS introduces the following two methods.

Firstly, each optimisation factor τ, η and ψ is assigned importance weight α, β and γ respectively, which control the impact that each optimisation factor has on the routing quality measurement. When nodes are new to a network environment, as the network information gathered by the nodes is limited, the stability factor value is not expected to deliver precise predictions. In that case, the importance factor associated with the stability weight, γ, is tuned to its minimum, so that the inaccurate stability factor will have little impact on the routing optimisation, and the measurement of the route quality depends mainly on the combination of the heuristic value η and the pheromone intensity τ. As time goes by, the statistical estimate of the link stability becomes more reliable, so the γ value will be increased, making the stability factor dominant in network routing optimisation.

To guarantee that the multicast tree is robust and efficient, SIMS employs a mesh structure on some branches of the tree. Downstream nodes are allowed to have only one upstream node. However, if better quality routes are found by the backwards AA, the highest quality route is marked as the backup route and maintained by the downstream node along with the route to its upstream and this implementation forms the mesh structure. The reasons for maintaining both routes are due to the characteristics of the SI techniques, which are population oriented algorithms from

which an optimal solution does not immediately emerge. This means that when a better route is found by the AA, it is more likely to be a local optimal result. The backup route may become the primary upstream route if the original route to the upstream node is broken or the quality of the backup route stays consistent after a period of time. In addition, the mesh structure is optional, not all downstream nodes are necessary to have and maintain a backup route.

Algorithm 6.4 is a scenario that a node X hears a backward AA from Y, the pseudo code shows how the path quality and the pheromone table is measured and updated. The quality metric of a route, ik, is a combination of the pheromone intensity, $\tau(t)$, the heuristic function, η, and the stability metric, $\psi(t)$, and defined as:

$$goodness_i(k,T) = \tau_i(k,T)^{\alpha} \times \eta_i(k)^{\beta} \times \psi_i(k,T)^{\gamma} \qquad \textit{Formula 6.7}$$

where $\tau_i(k,T)$ is the pheromone intensity from i to k; $\eta_i(k)$ is the heuristic value of route ik; $\psi_i(k,T)$ is the stability metric for route ik, which is calculated using *Formula 6.8*. α, β and γ are the importance factors for τ, η and ψ respectively.

$$\psi_i(k,T) = \psi_i(j_1,t_1) \times \psi_{j_1}(j_2,t_2) \times \cdots \times \psi_{j_n}(k,t_k) \qquad \textit{Formula 6.8}$$

$j_1 \cdots j_k$ represent the intermediate hops from i to k, and $t_1 \cdots t_k$ representing the age of the corresponding link.

After updating the pheromone and the stability table, X checks its *visitedNodes* stack. If it is not empty, then X adds the calculated cost, removes the last entry of its *visitedNodes* stack, and sends the ant to the next hop node. If the destination node, d, is reached, then the goodness value carried by the ant, $goodness_d(s,T)$, is compared with the $goodness_d(u,T)$, which represents the goodness value from d to its currently upstream u. If the $goodness_d(s,T)$ scores better, and also $height_s >$

$height_d$, d will mark s as its backup route destination in its multicast table, and generates a deterministic forward AA π_d towards s. This deterministic forward AA acts like a *RREQ* message, except it also updates the routing information along the route. Once π_d arrives at s, the multicast routing information in s is updated, and a new multicast tree with a hybrid mesh structure is created and maintained.

Algorithm 6.4 Node *X* processing a backward ant packet

```
INPUT:
bant ← incoming backward ant
Y ← the node from which bant is received
BEGIN:
Update the pheromone amount, τ_X(Y, h, t), for link XY
Update the stability metric, ψ_X(s, T), for the path Xs
Calculate the goodness metric, goodness_X(s, T)
IF X ≠ last entry in bant.visitedNodes THEN
    Remove the last entry from bant.visitedNodes
    Send bant to bant.visitedNodes[lastElement]
ELSE  /* bant.destination is reached, and X = bant.destination */
    IF bant.goodness > X.goodness AND bant.height > X.height
        X.upstream ← bant.sender
        X.goodness ← bant.goodness
        Send determinate ant and maintain the route to bant.sender
    END IF
END IF
```

6.4.4.3 Multicast tree maintenance

SIMS uses a periodic one-hop neighbour hello messages to detect link breakage in the multicast tree. Once a downstream node establishes a route to an upstream node, it

takes on the responsibilities of maintaining the route. During every NEIGHBOUR_HELLO_INTERVAL period, the downstream node sends out *RREQ* messages towards its upstream. If no *RREP* message is received from upstream within a pre-defined waiting time, the downstream node realizes that the link is broken. The downstream node then searches its multicast route table to see if there is a backup route. If there is no route, the downstream node initiates a route recovery process, trying to repair the route or find a new upstream route. If the downstream node is unable to find an upstream node after MAX_RETRIES attempts, it then indentifies itself as partitioned, and will retry the route recovery process again until a *GRPH* message from the group leader is received.

The upstream node maintains the downstream node routing information in its multicast route table. The corresponding routing information gets updated when a fresh *RREQ* message is received from a downstream node. If the message has not been seen within a period of time, the upstream node considers the route to the specific downstream to be broken, and will delete the corresponding routing information in its multicast route table. The upstream node will not perform the route recovery process to repair the route to its downstream nodes.

6.5 MANET simulation design, performance and evaluation

The performance of the SIMS routing protocol is evaluated and compared with two other benchmark multicast routing protocols for MANET, namely, the MAODV [10] and the MANSI [5] routing protocols. All these routing protocols are simulated as previously under the *Sinalgo environment* [23] and compared with the same network mobility model and nodes density. In this section, the MANET fundamental

assumptions, the mobility model, the detailed MANET multicast simulation design and performance will be presented and discussed.

6.5.1 MANET fundamental assumptions

Before any routing protocol is applied to the specific MANET environment for simulation, the MANET assumptions are as follows.

- Each node under the MANSI and SIMS protocols has a means of gathering up-to-date neighbouring nodes information with a relatively low routing overhead. This can be achieved via a local neighbour discovery protocol implemented in the lower layer of the communication system model.
- The underlying links are bidirectional with the same routing overhead. As most MAC layers deployed in MANETs conform to the IEEE 802.11 standard, thus this assumption can be easily satisfied.
- Only one multicast group is allowed during the entire simulation process. Theoretically, the MANET multicast routing protocol should allow multiple multicast groups, and if more than one multicast group exists and the groups intersect with each other, the tree merging operation should be performed to build and maintain a new tree. However, the routing robustness of the MANET is the major issues in this research work, thus all multicast routing protocols are in their simplest versions with irrelevant multicast routing operations being omitted.

6.5.2 Mobility model parameters and survival analysis

This section presents the parameter settings for the specific mobility scenario that were chosen for the MANET multicast simulations, also their influence on MANET network stability are analysed. It starts with a brief introduction to the most popular

random direction mobility scenario . Then the influence of the general mobility parameters such as node density, transmission range, and node pause time are discussed.

Random Direction Scenarios. The *random direction model* is one of the most popular mobility models for simulating the users in a mobile wireless network [24]. The *Random direction model* is similar to the *random waypoint mobility model* that has been introduced in *Section 5.4.1.1*. Both models usually operate in a finite two dimensional plane, typically a square. Under both models, users traverse piecewise linear segments where speeds vary from segment to segment but they are in general maintained constant on a segment. The two models differ in one critical manner, namely how users choose the next segment to traverse. Under the *random waypoint model* a user chooses a point within the space with equal probability and a speed from some given distribution. On the other hand, under the *random direction model* a user chooses a direction to travel in, a speed at which to travel, and a time duration for this travel.

Because of its simplicity and the ability to demonstrate the MANET characteristics, the *random direction mobility scenario* has been chosen for our experimental simulations. 100 mobile nodes with walking speed have been setup in a region of 200×200 square unit area. The initial network topology was randomly distributed as shown in *Figure 6.10*. In order to minimise experiment errors, it is saved to be the initial network topology for the rest MANET simulation experiments. In *Table 6.6*, a summary of the MANET network environment and *random direction model* parameters are presented. Some of the parameters, e.g., node density, node speed, are related to the MANET network routing stability, and will be discussed later in this section.

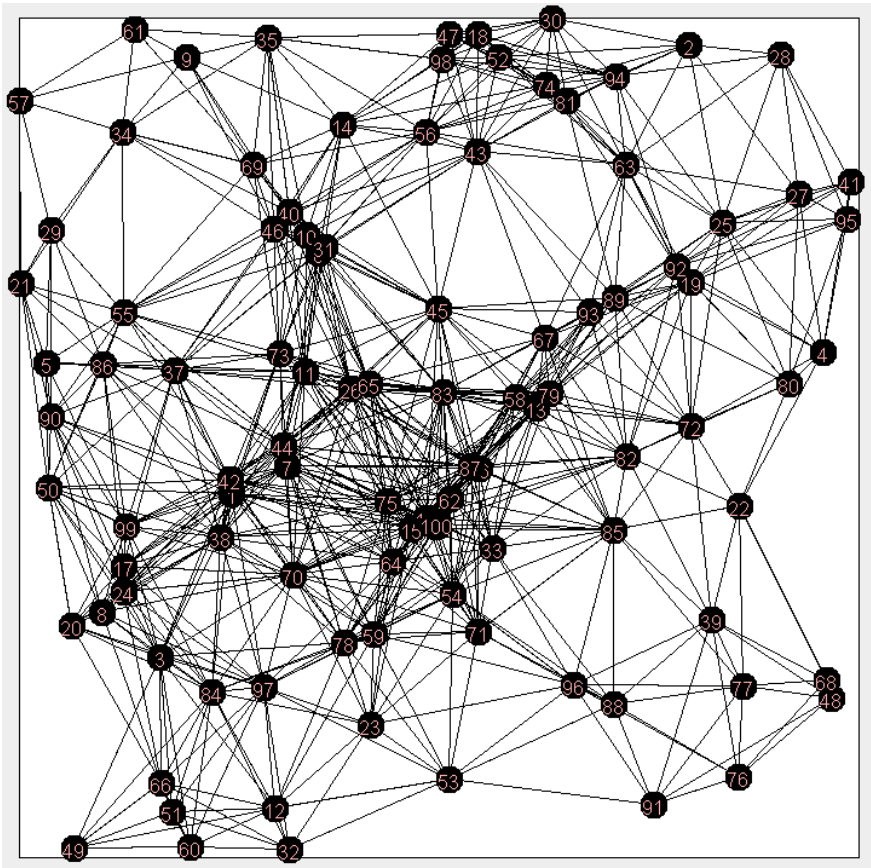

Figure 6.10 The initial network topology for MANET

Table 6.6 Summary of Network environment parameter settings

Simulation Size	200×200
Number of Nodes	100
The Speed of Nodes (Gaussian distribution)	Mean $\mu' = 0.3$ Variance $\sigma'^2 = 0.5$
Nodes Moving Duration (Uniform distribution)	$Min = 2$ $Max = 5$
Transmission Radius	$rMin = 30$ $rMax = 50$
Pause Time (Poisson distribution)	$\lambda' = 80$

Node Density. SIMS uses a statistically based metric to evaluate the stability of the communication routes. The statistical approaches benefit from an increased node density for two reasons. First, the accuracy of the link lifetime statistics will be improved by an increased node density, because more samples may be recorded. Furthermore, a higher node density also leads to a higher network connectivity, which in turn has the effect of offering more alternate paths.

Transmission Range. Node density and transmission range are closely related parameters. In fact, increasing the node density has the same effect as increasing the transmission range, or decreasing node velocity. Consequently, higher transmission ranges lead to longer link lifetimes and thus to fewer rediscoveries in general. In this simulation, the transmission radius of the nodes is set between 30 to 50 unit distances with free space propagation.

Pause Times. Pause times introduce stationary nodes to a scenario leading to an increased fraction of old links and thus to a higher average lifetime of links. *Sinalgo* has provided a toolbox which enables us to set different pause times for each node via the Poisson distribution with the Poisson distribution factor λ set at 80.

Survival Analysis. Each mobile node under the SIMS protocol is equipped with a memory to record the link connection durations. In general, the resources of MANET nodes are limited, thus in the current version of SIMS, nodes are required to only keep the connection information of their last known active and inactive links, the rest of the link connection data are abandoned due to the limited memory constraint. Although link lifetime estimates based on such data are not 100% accurate, they do provide reference metrics to guide SIMS in its choice of relatively stable communication paths. Moreover, this implementation has the advantage of being able to get rid of the

outdated link information and this makes SIMS more adaptable to a broader range of mobility scenarios.

Based on the initial network topology shown in *Figure 6.10* and mobility settings provided in *Table 6.6*, the simulation kept running for a period of 10000 time unit. *Figure 6.11* is plotted according to the link duration data from nodes under the SIMS protocol. In the figure, the blue curve represents the global link survival probability, where survival data are gathered from all nodes in the network, and the red curve is plotted according to the survival data which is randomly selected from the population. From the figure, it can be seen that under the current mobility settings, around 40% of the population could survive after a 10000 time unit span. However, to achieve 80% above survivability, the figure suggests that the link age should be younger than 91 time units.

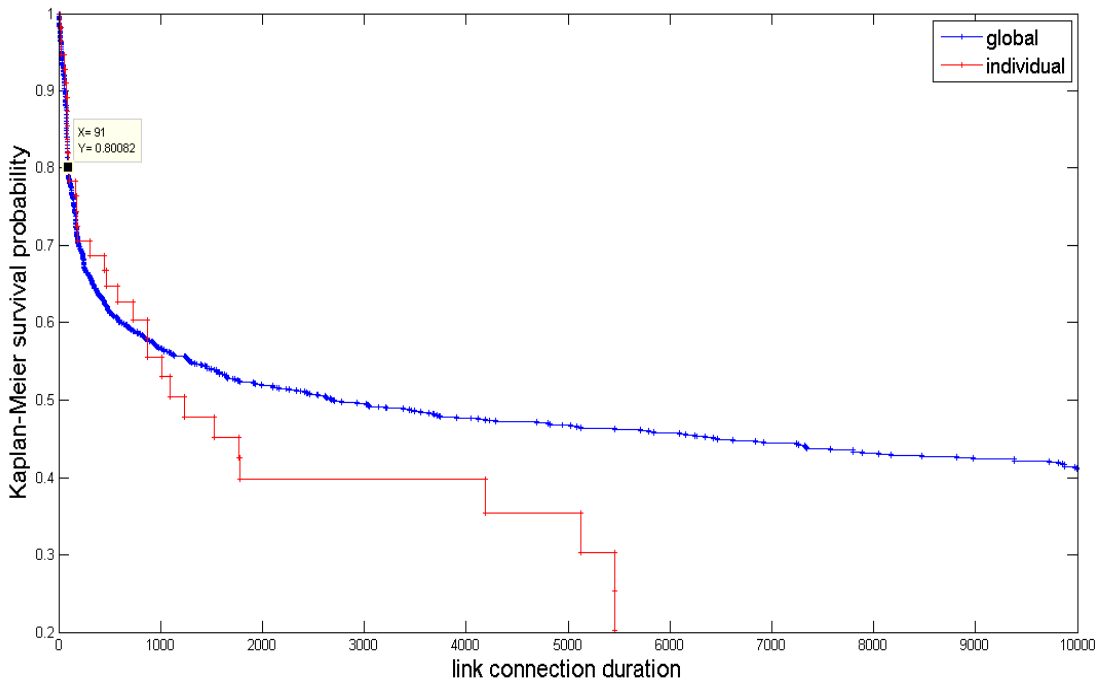

Figure 6.11 Random direction mobility model link survival probability

6.5.3 Protocol parameters

Parameter settings for the protocols are important factors that could influent the simulation performance. To make sure the routing protocols are able to achieve their best performance, all the protocols parameters were empirically tuned manually to the values shown in *Table 6.7*.

Table 6.7 Parameter settings for MANET multicast routing protocols

		MAODV	MANSI	SIMS
General parameters	Time latency for one-hop message	15	15	15
	Max time to wait for RREPs	30	30	30
	Frequency of group hello message (*GRPH*) broadcasts	100	100	100
Ants parameters	The pheromone concentration factor $\boldsymbol{\alpha}$	N/A	1	1.5
	The heuristic factor $\boldsymbol{\beta}$		1	10
	The link stability factor $\boldsymbol{\gamma}$		N/A	1.0
	Maximum pheromone concentration value		1.0	1.0
	Pheromone evaporate rate		0.99	0.99
	Initial pheromone amount		15	15
	Time latency of ant deployment		15	15

The protocols parameters can be categorised into general parameters and ants parameters. In the general category, the one-hop message time latency indicates how frequent each downstream node checks the connectivity to its upstream. With an extremely large time latency value, the downstream nodes have a risk of getting outdated routing information; on the other hand, high frequency connectivity checks increases the routing overhead. This applies to the frequency of broadcasting the

GRPH as well. When a downstream node requests a route to the group leader, it sends out *RREQ* requests, and waits for a predefined time latency to see if a better upstream candidate is available. This predefined time is set in the 'Max time to wait for *RREP*s' field in *Table 6.7*. Every multicast group member is required to keep up-to-date multicast group information, and this is done via the broadcasting of the *GRPH* messages. To balance this in the simulations, the *GRPH* messages are broadcasted every 100 time unit.

The MANSI and SIMS protocols are SI based protocols that employ AAs for the routing optimisation. The ant parameters are clearly illustrated in *Table 6.7*. During the routing optimisation three weight coefficients α, β and γ, are the importance factors associated with the ant pheromone concentration, the heuristic hop distance and the stability metrics respectively. To prevent the ant pheromone value becoming too large, which would trap route search into a local optimal, the pheromone value has been set with an upper bond, and a pheromone evaporate function is implemented as well for this reason. All the AAs are assigned with an initial pheromone value, the value decreases every time when ant jumps to the next hop. This enables the initial pheromone value to control the scope that an ant is allowed to search. Finally, the 'Time latency of ant deployment' parameter determines the frequency with which ants are created and deployed. In the same way as for the 'Time latency for one-hop message' parameter, extremely large or small values will decrease the performance of the protocols in different ways.

6.5.4 Simulation and results

Routing robustness is the factor in which SIMS differs from other MANET multicast benchmarking protocols. To verify the performance of the proposed protocol, SIMS together with MAODV and MANSI are simulated through sets of specially designed

experiments based on the parameter settings stated in *Section 6.5.3*. The experiments are designed to investigate the MANET multicast routing protocols from three perspectives, namely, routing robustness, percentage of multicast delivery ratio and routing overhead. In order to minimise errors, all the single simulation experiments were repeated for 10 times, and the average values were determined.

Multicast tree robustness. First, an investigation determines if there are any differences in multicast tree robustness between SIMS and other benchmark protocols. Under the multicast tree based MANET protocols, each time when a downstream node is no longer part of the multicast tree, it tries to establish a new connection by 'flooding' the network, such an operation is very costly to MANET routings. Too see SIMS performance in terms of reducing the 'route rediscovery processes', the following set of experiments were designed.

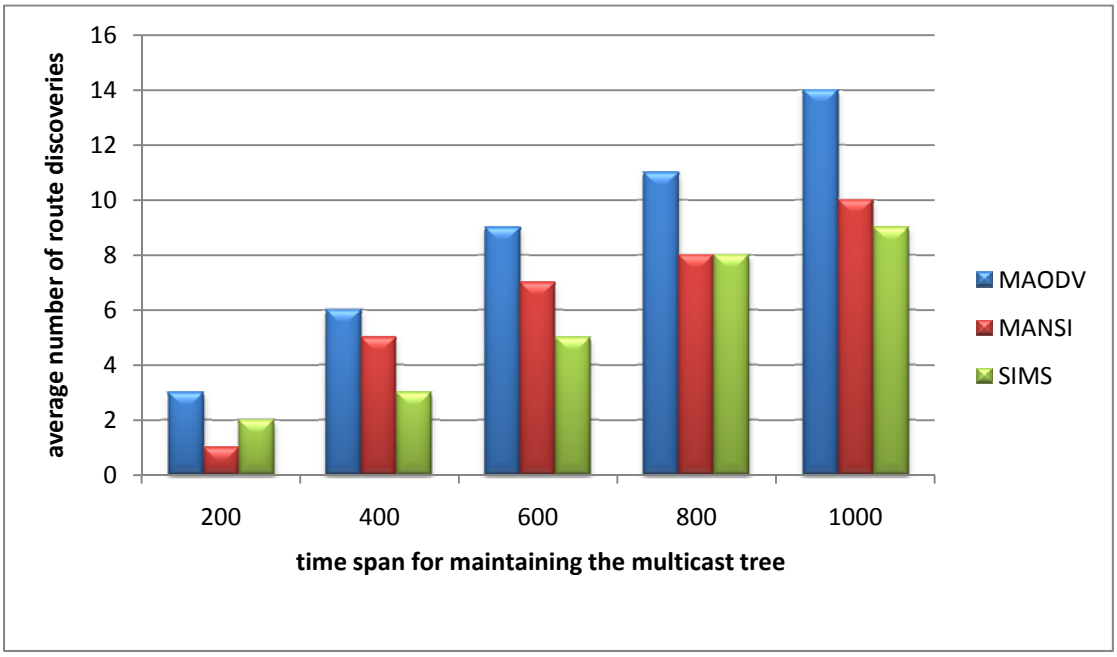

Figure 6.12 Multicast tree robustness of 5 nodes formed multicast group

In the first set of trials, a multicast tree with 5 multicast members was created. Then the frequency that route discoveries are performed with respect to the time span which ranges from 200 to 1000 unit time under the different protocols were recorded and illustrated in *Figure 6.12*. It is worth noting that the multicast tree formed from only 5 nodes is more likely to be a 'star' shaped, which means that all leaf nodes are connected to the group leader directly. Thus the robustness of the tree is totally dependent on the selected route stability at the time, and ideally all protocols should produce a similar performance. However, the SI based protocols keep evolving whenever a better route is available. With shorter hops and fresher routes, multicast trees created by SI based protocols has less chance of breaking, this is reflected in *Figure 6.12* as the SI based protocols perform slightly fewer 'route rediscoveries' than the MAODV protocol. This simulation suggests that the SI based protocols are able to setup multicast communication via a minimum sized forwarding set of routers.

In the second set of tests, the size of the multicast group was increased to 9, and the experimental results are shown in *Figure 6.13*. It can be seen that with 9 nodes, a proper tree structure can be built. A tree structure is often formed by a trunk or a backbone and a set of branches and leaves. If a branch or a leaf is cut off from the tree, only a single route rediscovery is needed. But if the backbone of the tree falls apart, the tree has no choice but to be rebuilt.

In MAODV, the multicast tree is built via the shortest paths algorithm, which are more likely to utilise the branches of the tree, making this approach relatively more robust. MANSI employs AAs to discover the smallest possible sized packets forwarding set [5]. In this structure, a single broken link may make it necessary to rebuild the multicast tree. This makes MANSI perform relatively poorly when long time span communication is required. With the Kaplan-Meier estimator and its hybrid

routing structure, SIMS outperforms all other benchmark protocols. In our experiments, the average frequency with which the route rediscovery process performed is only 13, whereas it is 17 and 38 (decreases of 24% and 66%) for MAODV and MANSI respectively.

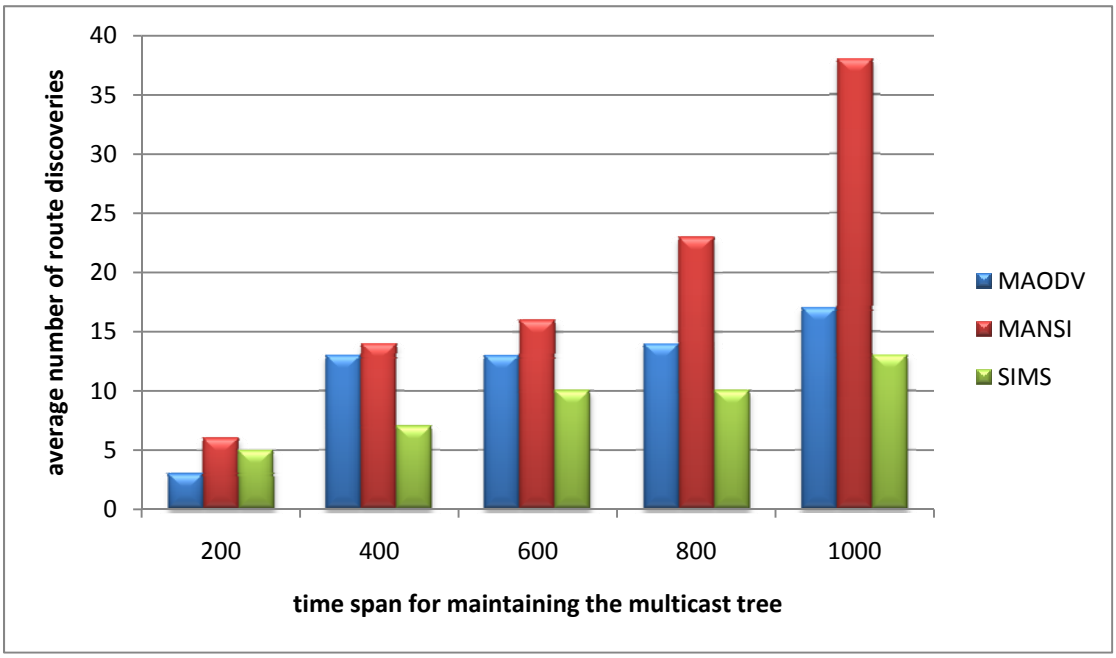

Figure 6.13 Multicast tree robustness of 9 nodes formed multicast group

Packets delivery ratio. The packets delivery ratio is another important metric to measure the performance of the routing protocols and is calculated as the number of data packets received by the destination nodes divided by the data packets transmitted by the source nodes. At this time, it is necessary to find out how the delivery ratio varies with respect to increase in any of the multicast group size.

In order to get more accurate Kaplan-Meier survival estimation, the SIMS protocol should run for an observation time span before the experiments. The observation time span must be set to satisfy two conflicting goals: it must be long enough to allow for the collection of a representative sample of link life time. On the other hand, a too long observation span bears the risk of using outdated information. In our simulation,

the observation time span is chosen according to the study of the mobility model as shown in *Figure 6.11*. From the figure it can be seen that the link survivability sharply decreases at the beginning and becomes steady after 1000 time units, which suggests 1000 time unit should be a reasonable value for the observation time span.

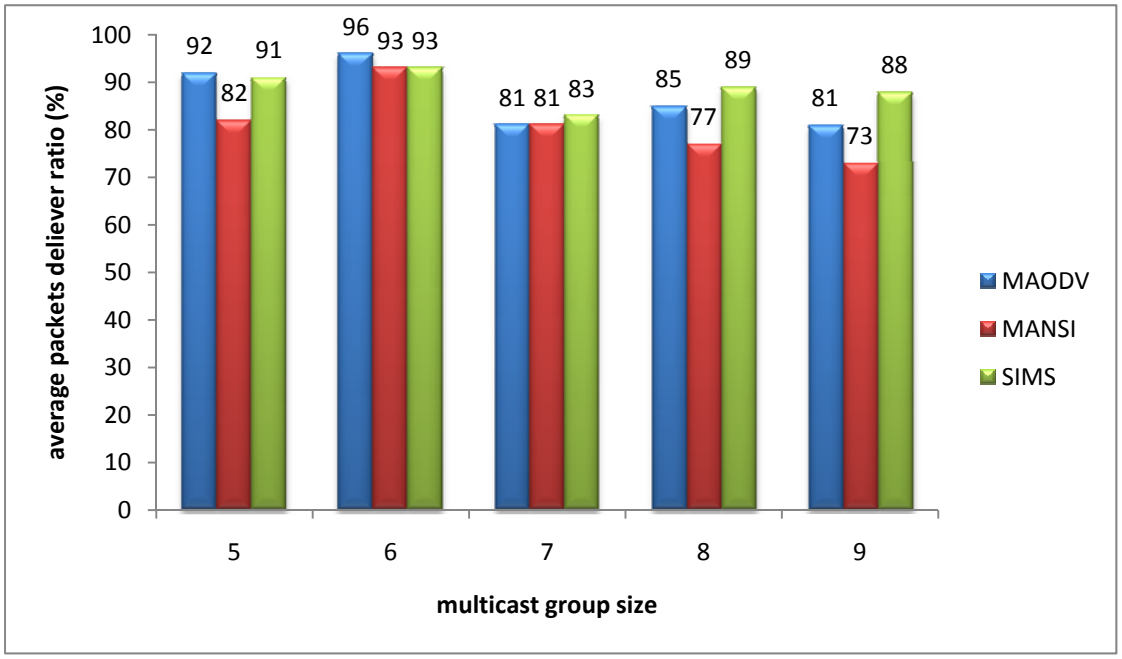

Figure 6.14 Multicast packet delivery ratio for varying multicast group size

In this set of experiments, the simulation first waits for the multicast communication setup under each routing protocol among 5 – 9 group members, and then ask the group leader to send total 100 data packages to each multicast member via the constant bit rat (CBR) at every 2 time unit interval. The results are plotted in *Figure 6.14*. In this experiment, all protocols produced fairly good result with all the delivery ratios being above 70%. In all protocols the packet delivery ratio shows a trend of decreasing when more multicast members are involved. SIMS shows its advantages especially if the size of the multicast group is large, with 7% and 15% improvement in success delivery ratio compared to MAODV and MANSI respectively.

Maintaining overhead. So far the performance of SIMS is very satisfactory. In this part of the simulation experiments, the interest is in the possible negative side of the SIMS protocol, which is the multicast tree maintaining overhead.

It is difficult to compare the routing overhead between SI based protocols and traditional MANET protocols like MAODV using the metric in terms of the quantity of control messages which were sent. SI based routing protocols optimise the routing by periodically deploying the AAs. Although in general more control messages are sent with ant based protocols, control messages in ant based protocols are often relatively small in size [25].

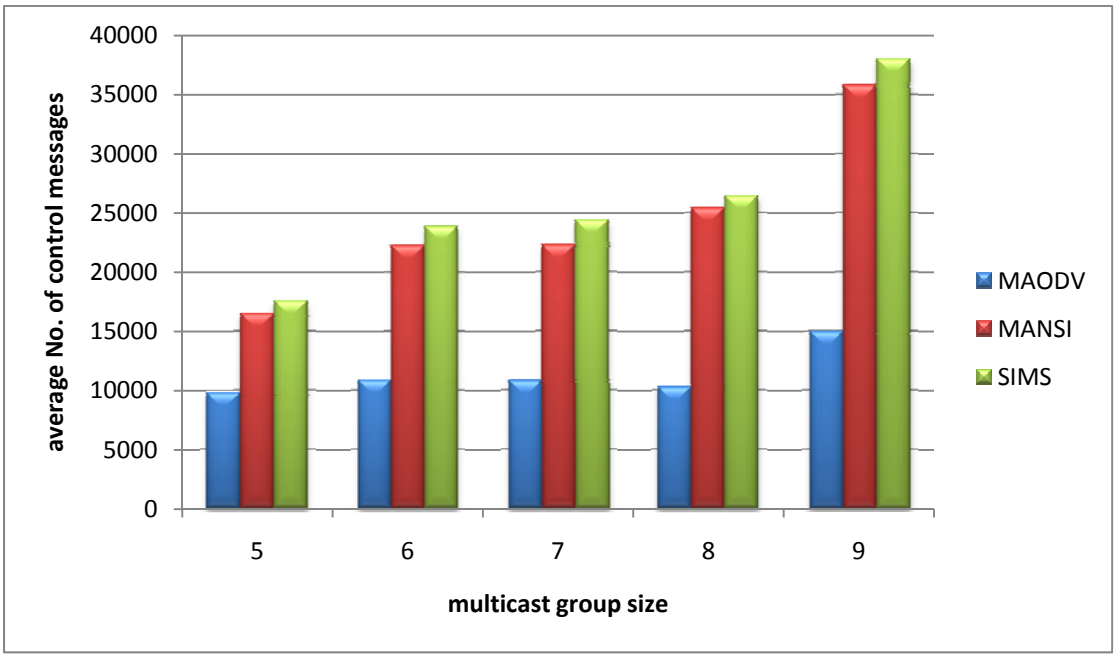

Figure 6.15 Control overhead of maintaining multicast tree for 100 unit times

Due to the constraint of our simulation software, it is not possible to measure the routing overhead using the metric of data size. In the simulation, the cost is measured by the number of control messages instead. Although it is already known that MAODV will generate much fewer control messages, the MAODV simulations are still performed and the results plotted as a reference. In the simulation experiments,

the quantity of control messages needed to maintain the multicast tree for a time span of 100 unit time was recorded. *Figure 6.15* presents the simulation results. When comparing SIMS with MANSI, around 4% – 9% more control messages are generated by the SIMS routing protocol. This is because of SIMS's hybrid structure, more control messages are needed for the forwarding set evolving and maintenance. Considering the outstanding results from the two previous sets of experiments, this amount of extra routing overhead is totally tolerable.

6.6 Summary

This chapter focuses on multicast routing for MANET and provided a detailed design process and performance analysis of a novel SI based on-demand hybrid multicast protocol for MANET.

At the beginning in *Section 6.2*, a number of MANET multicast protocols, including tree based, mesh based and hybrid protocols, were reviewed. In *Section 6.3*, the importance of link stability metric for MANET routings is discussed. The advantages of the statistical based Kaplan-Meier algorithm for estimating the link survival time have been addressed. The detailed implementation of Kaplan-Meier algorithm has also been presented in this section.

In *Section 6.4* and *Section 6.5*, the SIMS protocol is introduced and it performance is evaluated via 3 sets of simulation experiments. Two other MANET multicast routing protocols are also implemented and used as benchmarking protocols for SIMS performance evaluation. Through the specially designed simulation experiments, it was found that the stability of multicast tree created by SIMS is improved when compared to MAODV and MANSI. A robust multicast tree in multicast MANET

routing results in a low frequency route rediscovery processes and higher packets delivery ratio. Although the SIMS routing overhead is slightly higher than MANSI, considering the improvements that SIMS is able to provide such cost is tolerable. Above all, SIMS is a protocol that can provide routing efficiency, and moreover, in addition routing robustness.

References

[1] W. Ren, E. L. Hines, M. S. Leeson, Y. S. Kavian, and D. iliescu, "Swarm intelligence based link stability routing optimisation algorithm for mobile ad hoc network," in *Intelligence Systems: Techniques and Applications*, E. Hines, M. Leeson, and etc., Eds. Netherlands: Shaker Publishing, 2008, pp. 379-398.

[2] I. Chlamtac and S. Kutten, "On Broadcasting in Radio Networks--Problem Analysis and Protocol Design," *Communications, IEEE Transactions on,* vol. 33, pp. 1240-1246, 1985.

[3] J. J. Garcia-Luna-Aceves and E. L. Madruga, "The Core-Assisted Mesh Protocol," *IEEE Journal on Selected Areas in Communications,* vol. 17, pp. 1380-1394, August 1999.

[4] M. R. Elizabeth and E. P. Charles, "Multicast operation of the ad-hoc on-demand distance vector routing protocol," in *Proceedings of the 5th a nnual ACM/IEEE international conference on Mobile computing and networking* Seattle, Washington, United States: ACM, 1999.

[5] C. C. Shen and C. Jaikaeo, "Ad hoc multicast routing algorithm with swarm intelligence," *Mobile Networks & Applications,* vol. 10, pp. 47-59, Feb-Apr 2005.

[6] K. Chen and K. Nahrstedt, "Effective Location-Guided Overlay Multicast in Mobile Ad Hoc Networks," *International journal of wireless and mobile computing, special issue on group communications,* vol. 3, p. 2005, 2005.

[7] S.-J. Lee, W. Su, and M. Gerla, "On-Demand Multicast Routing Protocol (ODMRP) for Ad Hoc Networks," in *Internet-Draft, draft-ietf-manet-odmrp-01.txt*, June 1999.

[8] J. Xie, R. R. Talpade, A. McAuley, and M. Liu, "AMRoute: Ad Hoc Multicast Routing Protocol " *Mobile Networks & Applica tions,* vol. 7, number 6, pp. 429-439, December 2002.

[9] E. M. Royer, C. E. Perkins, and S. R. Das, "Ad Hoc On-Demand Distance Vector (AODV) Routing," *Proc. IEEE Workshop on Mobile Computing Systems and Applications,* pp. 90-100, 1999.

[10] E. P. Charles and M. R. Elizabeth, "Multicast Ad hoc On-Demand Distance Vector (MAODV) Routing," *IEFT, Internet Draft: draft-ietf-manet-maodv-00.txt,* 2000.

[11] C. M. Cordeiro, H. Gossain, and D. Agrawal, "Multicast over Wireless Mobile Ad Hoc Networks: Present and Future Directions," *IEEE Network,* vol. 17, pp. 52-59, 2003.

[12] L. Sung-Ju, W. Su, J. Hsu, M. Gerla, and R. Bagrodia, "A performance comparison study of ad hoc wireless multicast protocols," in *INFOCOM 2000. Nineteenth Annual Joint Conferen ce of the IEE E Computer and Communications Societies. Proceedings. IEEE*, 2000, pp. 565-574 vol.2.

[13] A. Vasiliou and A. A. Economides, "Evaluation of multicasting algorithms in MANETs," in *Proceedings 3rd World Enformatika Confer ence (WEC'05) - International Conferen ce on Telecommunications and Electronic Commerce (TEC 2005)*, 2005.

[14] T. Chai-Keong, "Associativity-Based Routing for Ad Hoc Mobile Networks," *Wireless Personal Communications,* vol. 4, pp. 103-139, 1997.

[15] R. Dube, C. D. Rais, K. Y. Wang, and S. K. Tripathi, "Signal Stability-Based Adaptive Routing (SSA) for ad hoc mobile networks," *IEEE Personal Communications,* vol. 4, pp. 36-45, Feb 1997.

[16] Z. Y. Wu, H. T. Song, X. M. Xu, and S. F. Jiang, "Ant-based stable disjoint path routing algorithm in MANETs," in *2nd International Conference on Pervasice Computing and Applications* , Birmingham, ENGLAND, 2007, pp. 297-302.

[17] S. M. Jiang, D. J. He, and J. Q. Rao, "A prediction-based link availability estimation for mobile ad hoc networks," in *20th Annual Joint Conference of the IEEE-C omputer-Society/IEEE-Communication-Society*, Anchorage, Ak, 2001, pp. 1745-1752.

[18] M. Gerharz, C. de Waal, M. Frank, and P. Martini, "Link stability in mobile wireless ad hoc networks," in *Local Computer Networks, 2002. Proceedings. LCN 2002. 27th Annual IEEE Conference on*, 2002, pp. 30-39.

[19] M. Gerharz, C. de Waal, P. Martini, and P. James, "Strategies for finding stable paths in mobile wireless ad hoc networks," in *Local Computer Networks, 2003. LCN '03. Proceedings. 28th A nnual IEEE Interna tional Conference on*, 2003, pp. 130-139.

[20] E. L. Kaplan and P. Meier, "Nonparametric Estimation from Incomplete Observations," *Journal of the American Statistical Association,* vol. 53, pp. 457-481, 1958.

[21] A. Helmy, "Small worlds in wireless networks," *IEEE Communications Letters,* vol. 7, pp. 490-492, October 2003.

[22] V. Srinivasan, M. Motani, and W. T. Ooi, "Analysis and Implications of Student Contact Patterns Derived from Campus Schedules " in *Proceedings of the Twelfth Annual International C onference on Mobile Computing and Networking (MobiCom)* Los Angeles, CA, September 2006.

[23] "Sinalgo - Simulator for Network Algorithms," http://dcg.ethz.ch/projects/sinalgo/index.html, (June 2009).

[24] C. Bettstetter, "Mobility modelling in wireless networks: categorization, smooth movement, border effects," *ACM Mobile Computing and Communications Review,* vol. 5, pp. 55-67, July 2001.

[25] K. M. Sim and W. H. Sun, "Ant colony optimization for routing and load-balancing: Survey and new directions," *IEEE Transactions on Systems Man and Cybernetics Part a-Systems and Humans,* vol. 33, pp. 560-572, Sep 2003.

Chapter 7:

Conclusions and Suggestions for Further Work

7.1 Introduction

This chapter summarises the main findings of this research, presents the conclusions and provides an outline of some ideas for further work. The research presented in the thesis provides the basis for the delivery of more practical routing protocols for optical and mobile ad hoc networks. It thus contributes to what may one day become the standard routing protocols for the next generation communication networks. This

can provide the future data communication with more flexibility and much improved QoS services.

This chapter is organised as follows. *Section 7.2* summarises the main findings of this research work, whilst *Section 7.3* presents the main conclusions. *Section 7.4* discusses further work and future potential avenues of research and development.

7.2 Summary of main findings

The main findings and the major original contributions to knowledge are presented as follows:

- **Genetic algorithm application for designing robust DWDM optical cores.**

  Robustness is gaining importance in designing future communication networks. In *Chapter 3* , an application of GAs to design robust DWDM optical mesh transport networks was illustrated. A novel way of representing optical lightpaths using GA binary chromosomes was also presented in *Chapter 3* . The simulation was performed using the Pan-European network test bench in order to achieve efficient capacity allocation for different demand scenarios. The simulation results demonstrated that using a GA is a promising approach for the robust core design of high speed DWDM optical mesh networks.

- **Link stability analysis for MANET**

  In *Chapter 5* and *Chapter 6* , the importance of link stability issues for MANET were discussed. In *Chapter 5*, a wireless network mobility model, the *local area mobility* model, was introduced to represent the real life wireless mobility scenarios. One attribute of this model is that it introduced a number

of absolutely stationary nodes into the mobility scenario. Aimed at this particular mobility scenario, an *oldest link best link* stability metric was implemented. To cope with more dynamic mobility models, possible solutions have been discussed in *Chapter 6*. The Kaplan-Meier survival estimator is one of the most suitable tools for the MANET link stability metric. The application of the Kaplan-Meier survival estimator in MANET routing optimisation has been demonstrated.

- **Swarm intelligence based unicast routing protocols design for MANET.**

  In *Chapter 5* , SILS, a unicast routing protocol for MANET, has been presented. It was implemented based on a swarm intelligence algorithm to improve efficiency and robustness in MANETs. An *oldest links best* type statistical metric was employed by SILS to search for the stable links. This metric is particularly successful when measuring link stability under the *local area mobility* model. Compared with AODV and ANSI, SILS achieves an improvement of 80% and 67% respectively in the need to perform routing rediscoveries.

- **Swarm intelligence based multicast routing protocols design for MANET.**

  The SIMS routing protocol introduced in *Chapter 6* is an extension of SILS to enable multicast routing and to be more adaptive to more dynamic mobility scenarios. The Kaplan-Meier survival estimator was embedded into the SIMS ant algorithms. In order to gain more routing robustness, part of the network was organised using a mesh structure. With the help of the Kaplan-Meier based stability metric and the hybrid routing structure, simulation results showed that 'routing rediscoveries' can be successfully reduced. The

improvement in the result varies with respect to the multicast group size. For a multicast group formed containing nine nodes within a 100 node network, there are decreases in the need to perform routing rediscoveries of 24% and 66% in comparison with MAODV and MANSI respectively. The beneficial consequences are improvements of 7% and 15% in successive delivery ratio compared to MAODV and MANSI respectively.

7.3 Conclusions

The objective of the thesis is the application of intelligent system algorithms to improve the routing efficiency and in particular the routing robustness for next generation communication networks. The next generation of *Fibre-Wireless* networks are formed by the optical mesh networks which act as the communication backbones for long distance communication, and the wired or wireless network for the local area data communication. In this thesis, the detailed focus is on both smarter ways of designing survivable DWDM optical mesh networks and routing optimisation for medium sized self organised MANETs. This thesis has tried to address the objective by solving the following three main problems:

1. *How to apply GAs to the design of survivable DWDM optical mesh transport networks.*
2. *How to improve the routing efficiency and routing robustness of the unicast mobile ad hoc network using the swarm intelligence techniques.*
3. *How to extend problem 2 and to implement a multicast MANET routing protocol, and is able to adapt to more dynamic mobility scenarios.*

From the summary presented in *Section 7.2*, a number of results can be described as follows. First, it can be concluded that GAs are a promising approach to tackle routing wavelength assignment problems in DWDM optical networks. The GA approach has the pertinent advantage of dealing with non-linear objective functions or constraints which are unable to be solved via the traditional integer linear (ILP) algorithms. In this thesis, the DWDM network routing assignments were optimised for both dedicated path protection (DPP) and shared path protection (SPP) under the Pan European network model. It was shown that the SPP approach is able to produce a saving of 8.1% in the overall network bandwidth.

In the context of MANET unicast protocol design, it has been shown that the swarm intelligence method of network routing allows for improved routing efficiency. Simulation results showed that for a medium sized network, swarm intelligence based protocols in general result in better routing efficiency than the traditional MANET protocols. Furthermore, it can be concluded that various MANET link stability metrics can be combined with the swarm intelligence algorithms to improve the routing robustness.

The final conclusion relates to the multicast MANET routing protocol; to its design and performance. In the context of this work, multicast routing for MANETs has been optimised through the swarm intelligence algorithm, the Kaplan and Meier statistical estimator and also by its hybrid mesh-tree routing structures. It is concluded that the above mentioned approach can be successfully applied to multicast MANET routing protocol design, due to being cost efficient and impressive performance during the simulations. This approach can confer the additional advantage of not adding overly-complex in implementing the protocol, thereby becoming practical to transfer it from the laboratories to industry.

In conclusion, the contribution of this thesis is in the demonstration that intelligent systems, such as evolutionary techniques and swarm intelligence based algorithms, can be successfully applied to achieve routing robustness and efficiency in next generation communication network routing protocol design.

7.4 Further work

The work presented in this thesis can be extended on a range of fronts. For example in the optical network domain, delivery of the required QoS in IP-over-DWDM networks would face uncertain environment capacity problems due to demand variations over different periods of network operation. Therefore, future demand uncertainties should be considered in planning DWDM transport backbones for capacity sensitive telecommunication applications with enhanced quality of service.

In this work for MANET routing protocols design, the number of deployed ant messages can be further reduced. Swarm intelligence based routing protocols require the periodically deployment of ant message to achieve routing improvement. Although ant messages are generally small in size; reduced ant deployment could be adding up achieving more routing efficiency and reducing mobile node battery consumption.

MANET routing optimisation can be classified into the multi-objective optimisation algorithms domain. Usually these types of algorithms require combination of sets of parameters to be tuned accordingly. In this work, the values of the protocol parameters were adjusted manually purely based on empirical tests and experience of the network mobility attributes. The problem of choosing the perfect values for the protocol parameters is another NP-hard problem, which is difficult to solve via

traditional methods. In [1], David etc. proposed a methodical way of using GAs for automated selection of parameters in a MANET networking system. Similar GA systems can be implemented in future work to further improve MANET routing efficiency.

Reference

[1] M. David and R. Jason, "Optimizing parameters of a mobile ad hoc network protocol with a genetic algorithm," in *Proceedings of the 2 005 conference on Genetic and evolutionary computation* Washington DC, USA: ACM, 2005.