Application of Genetic Algorithm to Wireless Communications

By

Xiang Wu

A dissertation submitted to the Faculty of Engineering

University of Newcastle upon Tyne

In partial fulfilment of the requirements for the degree of

Doctor in Philosophy

December 2004

Dedicated to My Beloved Parents

Chuanli Hong and **Qinfen Wu**

Acknowledgement

This research would not have been possible without the enormous support and guidance I have received from many generous people.

First of all, I am deeply grateful to my major supervisor, Professor Bayan S. Sharif, for the excellent guidance, many invaluable suggestions and stimulating discussions. It has been a pleasure and a privilege to benefit from his skills and knowledge.

I would like to acknowledge the School of Electrical, Electronic and Computer Engineering, which is led by my second supervisor, Prof. Oliver R. Hinton, whose excellent facilities and significant research output have made great contribution to this dissertation.

I would also like to thank the British Council (ORS) and the University of Newcastle for their financial support throughout the study. Without this help, the present study cannot be completed.

Finally, I would like to conclude with an acknowledgment to my parents, which I have saved for the very last. Words cannot express my appreciation to them for their affection, guidance, sacrifice, and perseverance from my first day of life. Nothing I ever do can possibly repay the debt of gratitude I owe them.

Abstract

Wireless communication is one of the most active areas of technology development of our time. Like all engineering endeavours, the subject of the wireless communication also brings with it a whole host of complex design issues, concerning network design, signal detection, interference cancellation, and resource allocation, to name a few. Many of these problems have little knowledge of the solution space or have very large search space, which are known as non-deterministic polynomial (NP) -hard or -complete and therefore intractable to solution using analytical approaches. Consequently, varied heuristic methods attempts have been made to solve them ranging from simple deterministic algorithms to complicated random-search methods.

Genetic algorithm (GA) is an adaptive heuristic search algorithm premised on the evolutionary ideas of evolution and natural selection, which has been successfully applied to a variety of complicated problems arising from physics, engineering, biology, economy or sociology. Due to its outstanding search strength and high designable components, GA has attracted great interests even in the wireless domain.

This dissertation is devoted to the application of GA to solve various difficult problems spotlighted from the wireless systems. These problems have been mathematically formulated in the constrained optimisation context, and the main work has been focused on developing the problem-specific GA approaches, which incorporate many modifications to the traditional GA in order to obtain enhanced performance.

Comparative results lead to the conclusion that the proposed GA approaches are

generally able to obtain the optimal or near-optimal solutions to the considered optimisation problems provided that the appropriate representation, suitable fitness function, and problem-specific operators are utilised.

As a whole, the present work is largely original and should be of great interest to the design of practical GA approaches to solve realistic problems in the wireless communications systems.

Contents

List of Figures

List of Tables

List of Publications

- **Journals**

X. Wu, B.S. Sharif, and O.R. Hinton, "An Improved Resource Allocation Scheme for Plane Cover Multiple Access Using Genetic Algorithm," ***IEEE Transactions on Evolutionary Computation, vol. 9, no. 1, pp.74-81, 2005***.

X. Wu, T.C. Chuah, B.S. Sharif, and O.R. Hinton, "Adaptive Robust Detection for CDMA Using A Genetic Algorithm," ***IEE Proceedings Communications***, vol. 150, pp. 437-444, 2003.

X. Wu, B.S. Sharif, and O.R. Hinton, "Genetic Algorithm based Adaptive Channel Allocation Schemes for Wireless PCN," submitted to ***Elsevier Applied Soft Computing***, Jun, 2004.

X. Wu, B.S. Sharif, and O.R. Hinton, "Solving Minimum Power Broadcast Problem in Wireless Ad Hoc Networks using Genetic Algorithm," submitted to ***IEEE Transaction on Wireless Communications***, Dec, 2004.

- **Conferences**

X. Wu, B.S. Sharif, and O.R. Hinton, "Adaptive Channel Allocation Schemes for Wireless PCN: A Genetic Algorithm Approach," The joint conference of 10th Asia-Pacific Conference on Communications (***APCC2004***) and 5th International Symposium on Multi-Dimensional Mobile Communications (***MDMC2004***), Beijing, China, Aug.29 –Sep.1, 2004, pp. 76-79.

X. Wu, B.S. Sharif, and O.R. Hinton, "Genetic Algorithm Based Optimal Resource Allocation for Plane Cover Multiple Access Scheme," in Proc. 2nd International Conference on Communications, Circuits and Systems (***ICCCAS'04***), China, June 27-29, 2004, pp. 1113-1117.

X. Wu, T.C. Chuah, B.S. Sharif, and O.R. Hinton, S. Sali, "Robust Adaptive Multiuser Detection: A Genetic Algorithm Approach," in ***IMA*** 2nd Int. Conf. on Mathematics in Communications, United Kingdom, Dec. 2002.

1 Introduction

This chapter presents the background of the subject material that has motivated the research directions, states the main contributions, and also gives the outline of the dissertation.

1.1 Motivation

During the last two decades there has been tremendous interest and progress in the field of wireless communication networks [1], [2]. These wireless networks provide users with the ability to communicate anytime, anywhere, and in any form.

Like all engineering endeavours, the subject of wireless communication also brings with it a whole host of complex design issues, concerning network design, signal detection, interference cancellation, and resource allocation, to name a few. One of the interesting trends in these researches has been the growing incorporation of artificial intelligence (AI) techniques into such systems. The reasons for that are twofold: First, a large class of problems arising from real wireless systems can be formulated to the optimisation problems and involves searches for the best solution among many potential possibilities. Second, most of these optimisation problems are NP-complete in nature, characterised by search spaces increasing exponentially with the size of the input parameters [3]. Therefore, these problems are usually intractable to solution using analytic approaches or simple deterministic algorithms running in polynomial time. To cope with that, heuristic and stochastic optimisation approaches, such as Simulated Annealing (SA), Tabu Search (TS), Neural Network (NN), and GA, offer more promising alternatives.

The aim of this thesis is to contribute to this important research area by studying existing wireless communication systems, identifying several important optimisation problems, and proposing solutions. This thesis is most concerned with designing problem-specific GA approaches due to its outstanding search strength and high designable components.

GA is a heuristic search method introduced in the 1970s by John Holland [4]. After decades of research on GA, it is used to solve a wide range of problems and shown to be superior to many alternative methods [5], [6]. GA attracts most interest in the field of AI since it performs multi-directional search by manipulating and maintaining a population of potential solutions and tends to focus increasingly on areas with deeper minima. In contrast to other heuristic approaches, such as SA, which only examines one point at a time (one-dimensional) in the search, GA is not biased toward local minima. Although GA has been studied for over two decades, implementing it is often as much an art as designing efficient heuristics. Much of the GA literature is devoted to relatively simple problems. Simplistic application of GA to particular problems often produces reasonable results, but naive application of GA to more realistic problems often results in poor performance. This is due to both the nature of the genetic search and the relationships between the genetic representation and the genetic operators.

The research primarily focuses on the optimisation problems in wireless communication systems and the problem-specific GA approaches has been proposed. A practical improvement to the implementation of GA will be attempted in areas where knowledge of the problem domain is available. Therefore, various modified GA approaches are designed according to the different requirements that the specific problems exhibit. The main work concentrates on formulating the optimisation problems, choosing the appropriate genetic representation of the problems to be solved, employing specialised initialisation, and devising the appropriate genetic operators, etc. The primary goal of the research is to investigate the theory and the application of GA by solving realistic wireless optimisation problems.

1.2 Research Problems

Efficient multiple access methods

There are three major multiple access techniques, namely Time Division Multiple Access (TDMA), Frequency Division Multiple Access (FDMA) and Code Division Multiple Access (CDMA). In TDMA/FDMA, the entire bandwidth is divided into narrow channels in the form of time-slot or frequency bands. Each user is assigned a particular channel, which is not shared by other users. This technique is widely employed in the first-generation (1G) and second-generation (2G) wireless systems [7], [8]. The fundamental problem of such systems is how to share the limited resource by as many users as possible whilst maintaining good quality of service, which is referred to as channel allocation problems. Chapter 3 addresses a typical channel allocation problem and deals with it in an optimisation sense using GA.

The important feature of TDMA/FDMA systems is that the various users are operating in appropriate channel reuse pattern, but this also results in low system capacity. Based on the TDMA concept, Plane Cover Multiple Access (PCMA) has been proposed to improve the attainable system capacity by introducing multiple reuse patterns to the system users. Consequently, a new allocation scheme needed to be considered, which is studied in Chapter 4.

CDMA is another efficient access method designed to increase the system capacity, which enables different users operating in the same frequency spectrum by assigning a different code signature to each user. In contrast to TDMA/FDMA systems, CDMA systems are interference-limited instead of bandwidth-limited. Therefore, the most important issue in such systems is to suppress the interference imposed by other users. The multiuser detection (MUD) technique has therefore attracted most interest in CDMA research areas. Chapter 5 addresses MUD problem in CDMA systems using GA.

Flexible network architecture

In traditional wireless systems, a cellular scenario consists of mobile nodes linked via a radio network to an infrastructure of switching controlling centre (base station). However, due to the movement nature of mobile nodes, a so-called ad-hoc network scenario has been introduced and it is envisioned to be integral to future wireless networks. In an ad-hoc network, mobile nodes connected by wireless links in the absence of any fixed infrastructure. The mobile nodes are free to move randomly and organise themselves arbitrarily, and networks may operate in a standalone fashion or may be connected to the larger Internet. Such networks are of interest because they do not require any prior investment in fixed infrastructure. Instead, the network nodes agree to relay each other's packets toward their ultimate destinations, and the nodes automatically form their own cooperative infrastructure. Though ad hoc networks are attractive, they are more difficult to implement than fixed networks and a lot of design issues remain open. Chapter 6 is devoted to the allocation problem in TDMA-based mobile ad-hoc network (MANET). Although it is similar to the channel allocation in the traditional network, many new features concerned with ad-hoc nature need to be given special consideration. In Chapter 7, more general problem, broadcast operation, is considered in ad-hoc networks, which is widely used regardless of the underlying access methods (TDMA, FDMA or CDMA).

1.3 Main Contributions

Throughout this thesis the problem-specific GA approaches are developed according to the different requirements, and the following contributions to the research area of wireless communication and GA are made:

1. A discrete event simulator is implemented for simulating Personal Communication System (PCS) networks. The simulator is fully component-based and event-oriented, in which each simulated object can be viewed as an individual functional element.

2. Real-encoded GA based adaptive channel allocation schemes are proposed for PCS networks, which have been shown to be able to find the appropriate allocated and/or reserved channels in terms of Grade of Service (*GoS*) for the considered wireless system.

3. Real-encoded GA based resource allocation scheme is proposed for PCMA system, which has been shown to exhibit powerful search capability in maximising the attainable capacity of packet-based wireless networks.

4. CDMA real-encoded GA based MUD is developed, which has been shown to have good adaptive impulsive noise rejection capability without the need for a separate channel estimator.

5. A permutation-encoded GA based broadcast scheduling scheme is proposed for TDMA-based MANET, which has been shown to be able to obtain the optimum

schedule results and outperform many other alternative methods.

6. A permutation-encoded GA based approach is proposed for solving the minimum power broadcast (MPB) problem in wireless ad hoc networks, which has been shown to be able to attain competitive results compared with many recent reported methods.

1.4 Dissertation Outline

Chapter 2 provides an introduction to optimisation problems and related algorithms, which forms the theoretical foundation of this dissertation.

In Chapter 3, channel allocation in PCS networks is formulated as a combinatorial optimisation problem. GA-based channel allocation approaches have been developed in order to find the appropriate number of allocated and/or reserved channels in terms of *GoS* for a given period of time. The proposed GA-based schemes are shown to have much more adaptability to the traffic load than the traditional fixed allocation schemes, and they outperform other heuristic methods such as TS and SA. Both the analytical and simulated models of the GA-based allocation schemes are presented. The call-initiated discrete event simulator has been employed to simulate a PCS cellular network, which is the test-bed for the proposed channel allocation schemes.

In Chapter 4, an improved resource allocation scheme is proposed, which uses GA in conjunction with the recently developed PCMA scheme in order to maximise the attainable capacity of packet-based wireless communication networks. Computer simulation results suggest that the proposed GA approach outperforms the uniform and the Greedy algorithm based '*min*' methods in terms of serviced users. The portable-initiated discrete event simulator has been employed to simulate a PCMA wireless network, which is the test-bed for the proposed resource allocation scheme.

In Chapter 5, for a CDMA system a novel adaptive robust multi-user detector for CDMA using GA is proposed. The GA-based detector implicitly implements the Huber's *M*-estimator and is robust against heavy-tailed impulsive noise. The novel feature of this GA-based detector lies in joint symbol detection and adaptive estimation of the cut-off parameter of the *M*-estimator's object function through a GA optimisation strategy. In particular, the GA-based detector carries out a multi-point search by manipulating and maintaining a population of candidate solutions for different values of the cut-off parameter to encourage information formation and exchange. Since the GA treats the cut-off parameter as one of its optimisation parameters, the need for a separate channel estimator is thus eliminated. Simulation results are provided to examine the evolutionary behaviour and the detection performance of the proposed GA-based detector. It is shown that the GA approach provides good adaptive impulsive noise rejection capability.

In chapter 6, a novel permutation-encoded GA is proposed for solving the optimum broadcast scheduling (OBS) problem in TDMA-based MANET. The problem is widely known as NP-complete and diverse heuristic algorithms were reported to solve this problem recently. GA approach is developed to cooperate with a deterministic greedy-like algorithm to obtain the optimum schedules. The novel feature of the proposed GA approach lies in employing the permutation encoding strategy, and thus obviates the invalid solutions encountered by conventional binary encoding scheme. Consequently, the problem search space is reduced to a great extent and the GA becomes more efficient in searching for the optimum solutions. A variety of simulation scenarios were conducted to examine the evolutionary behaviour and the scheduling performance of the proposed GA approach. Simulation results suggested that the proposed GA approach exhibits superb search strength in obtaining lower frame length whilst keeping higher slot utilisation compared with other recently proposed methods.

In chapter 7, a novel permutation encoded GA is proposed for solving the MPB problem in wireless ad hoc networks, which has also been proven to be NP-complete. The MPB problem has been mathematically formulated as a constrained optimisation problem using a graph representation, and a GA-based approach is developed to cooperate with a deterministic greedy-like algorithm to obtain the MPB tree. The powerful search capability of GA is a key factor in improving the system performance in terms of the total consumption power. A variety of simulations were conducted to examine the performance of the proposed GA approach, and the results indicate that the proposed GA approach significantly outperforms the greedy-based Broadcast Incremental Power (BIP) algorithm and exhibits competitive search strength compared with other recently proposed optimisation methods.

Finally, Chapter 8 presents a summary of the thesis and looks ahead to future work arising from this dissertation.

2 Optimisation Problems and Algorithms

This chapter provides a brief introduction of optimisation problems and an overview of the related optimisation algorithms. This preliminary section is useful to show the context, in which optimisation problems in wireless systems need to be solved in this thesis.

2.1 Optimisation Problems

As in all empirical sciences, optimisation problems are abundant in wireless systems, which can be perceived and modelled by optimising the value of an objective function, under stated feasibility constraints. In many cases of practical relevance, and particularly in most highly nonlinear models, the optimisation problems do not warrant the global optimality of solutions found by local scope search approaches. Since the number and quality of the local solutions are typically unknown, there is a strong motivation for seeking the globally best solutions, which is the main objective of global optimisation algorithms [9].

The goal of an optimisation problem can be described as follows: find the combination of variables (independent parameters), which optimises a given quantity, possibly subject to some restrictions on the allowed parameter ranges. The quantity to be optimised (maximised or minimised) is termed the objective function, the parameters which may be changed in the quest for the optimum are called decision or control variables, and the restrictions on allowed parameter values are known as constraints.

2.1.1 Decision Variables

The decision variables can be denoted by a vector $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$, and each element is within a certain permissible range. The variables can take either continuous or discrete values. A continuous variable is one that takes any value in the range of the variation in its region. A discrete variable is one that takes only isolated values, typically from a list of permissible integer values.

2.1.2 Objective Function

Almost all optimisation problems have a single objective function $f(\boldsymbol{x})$, which is to be minimised or maximised. However, there exist two types of interesting exceptions: the first one has no objective function defined, in which the goal is to find a set of variables that satisfies the constraints only. This type of problems is usually called a *feasibility problem*. The second one contains multiple defined objective functions and is generally referred to as multi-criteria optimisation. These different objectives may not be compatible and are typically reformulated as single-objective problems by either forming a weighted combination of the different objectives or replacing some of the objectives by constraints.

2.1.3 Constraints

The constraints are not essential. In unconstrained optimisation, there are no limitations on the values of the variables. However, in a realistic world, most optimisation problems are constrained on the parameters and these constraints make certain solutions invalid. Note that there are two types of constrains including equality and inequality ones.

2.1.4 Standard Formulation

From the above subsections, the final formulation of the optimisation problem can be mathematically stated as follows

Minimise/Maximise $\quad f(\boldsymbol{x}), \quad \boldsymbol{x}=(x_1, x_2, \ldots\ldots, x_n)$

Subject to

$$g_i(\boldsymbol{x}) \leq 0, \qquad i = 1,\ldots, m$$

$$h_i(\boldsymbol{x}) = 0, \qquad i = 1,\ldots, p$$

where $f(x)$ is the objective or cost function, $\boldsymbol{x}$ is the vector of the independent variables, $g_i(\boldsymbol{x})$ is the set of inequality constraint functions, and $h_i(\boldsymbol{x})$ is the set of equality constraint functions. Taken together, $g_i(\boldsymbol{x})$ and $h_i(\boldsymbol{x})$ are known as the problem functions.

2.2 Optimisation Algorithms

The most common approach to solve an optimisation problem is to apply gradient-based search methods like the well-known Quasi-Newton methods [10]. These methods are highly efficient and well developed for general applications. However, the main drawbacks are that these methods make strong assumptions on the continuity and differentiability of the objective function $f(\boldsymbol{x})$. Additionally, the solution strongly depends on the initial design because only local solutions are possible.

Unlike gradient-based algorithms, global optimisation algorithms circumvent these restrictions, which are very useful when the search space is likely to have many minima, making it hard to locate the true global minimum. It should be noted that global optimisation approaches usually involve a stochastic element, and may therefore not guarantee to give the true global minimum, nevertheless in almost all cases they are able to find very good solutions, where other techniques fail completely.

A great many methods have been proposed for global optimisation, these include greedy methods, exhaustion, branch and bound, random search, and methods inspired by the natural world. Here, we give an overview of widely used optimisation methods, which by no means cover the field but rather are intended to be a sample of those available.

2.2.1 Simulated Annealing

As its name implies, the SA is a random-search technique, which exploits an analogy between the way in which a metal cools and freezes into a minimum energy crystalline structure (the annealing process) and the search for a minimum in a more general system.

The concept of SA is based upon [11], [12], which was originally proposed as a means of finding the equilibrium configuration of a collection of atoms at a given temperature. The principles involved in SA are very similar. Each point in the search space has an energy associated with it, which indicates how good it is. The goal is to find the point with the minimum energy. The algorithm starts off at an arbitrary point, at each step chooses some neighbour of the current point and moves to that point with a certain probability. Neighbours are points that are close to each other in a problem-dependent fashion. The probability of transition is a function of the energy difference between the two points and a global time-dependent parameter called the *temperature*.

SA operates on a single point, and at each step a point x' in the neighbour $N(x)$ is generated from the current point x. If the new point has a lower cost function than x it is accepted unconditionally, but even if it has a higher cost it is accepted probabilistically using a Boltzmann probability described below. This acceptance probability is proportional to the temperature T of the annealing process, which is lowered gradually as the algorithm proceeds.

$$p = \boldsymbol{min}(1, exp(-\frac{\delta f}{T}))$$

The main procedures in SA are illustrated in Figure 2.1.

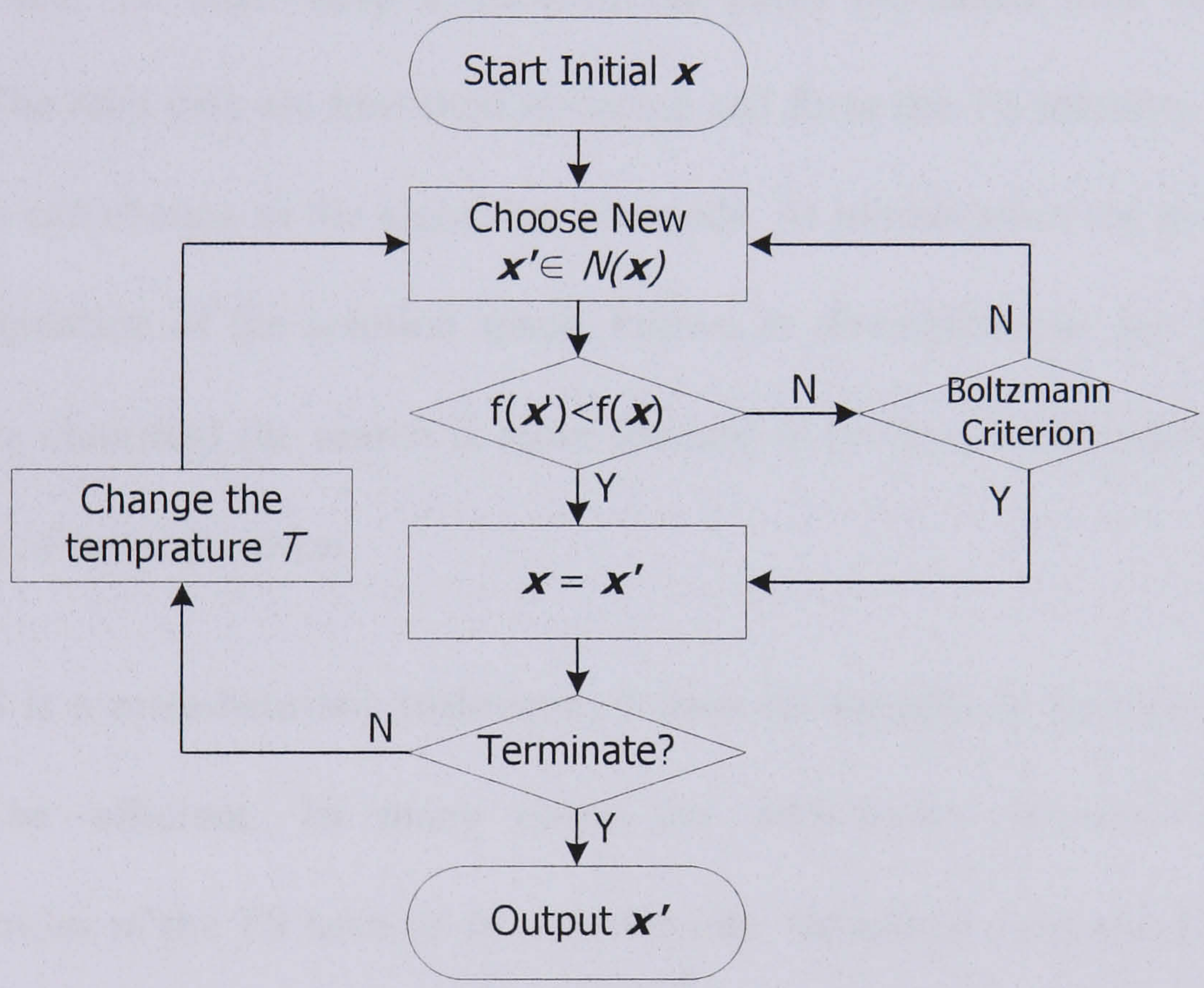


Figure 2.1: Main procedures in SA.

It has been proved that by carefully controlling the rate of cooling of the temperature, SA can find the global optimum. However, this requires infinite time. Fast annealing and very fast simulated re-annealing (VFSR) or adaptive simulated annealing (ASA) are each in turn exponentially faster and overcome this problem [13].

2.2.2 Tabu Search

Whereas SA performs an optimisation by randomly generating a new state and applying the Boltzmann rule to decide whether to accept or reject, TS uses sophisticated history mechanisms to avoid evaluating solutions that are already known. Therefore, TS is a non-random meta-heuristic algorithm as described in [14].

The optimal solution is discovered over time by performing a given action, determining its consequences and using those consequences in performing future actions. Hence, TS must keep a track of the paths (so-called *tabu list* ***L***) it has traversed. The tabu lists are historical in nature and form the TS memory. The role of the memory can change as the algorithm proceeds. At initialisation the goal is make a coarse examination of the solution space, known as *diversification*, but as candidate locations are identified the search is more focused to produce local optimal solutions in a process of *intensification*.

Because TS is a meta-heuristic technique, it must be adapted to the problem at hand for it to be efficient. In many cases the differences between the various implementations of the TS have to do with the size, variability, and adaptability of the memory to a particular problem domain. Like all search algorithms, TS performs better the more problem specific information is encoded into the procedure, which is especially important in determining how to dynamically generate the local neighbourhood.

A typical algorithm for TS is described in Figure 2.2.

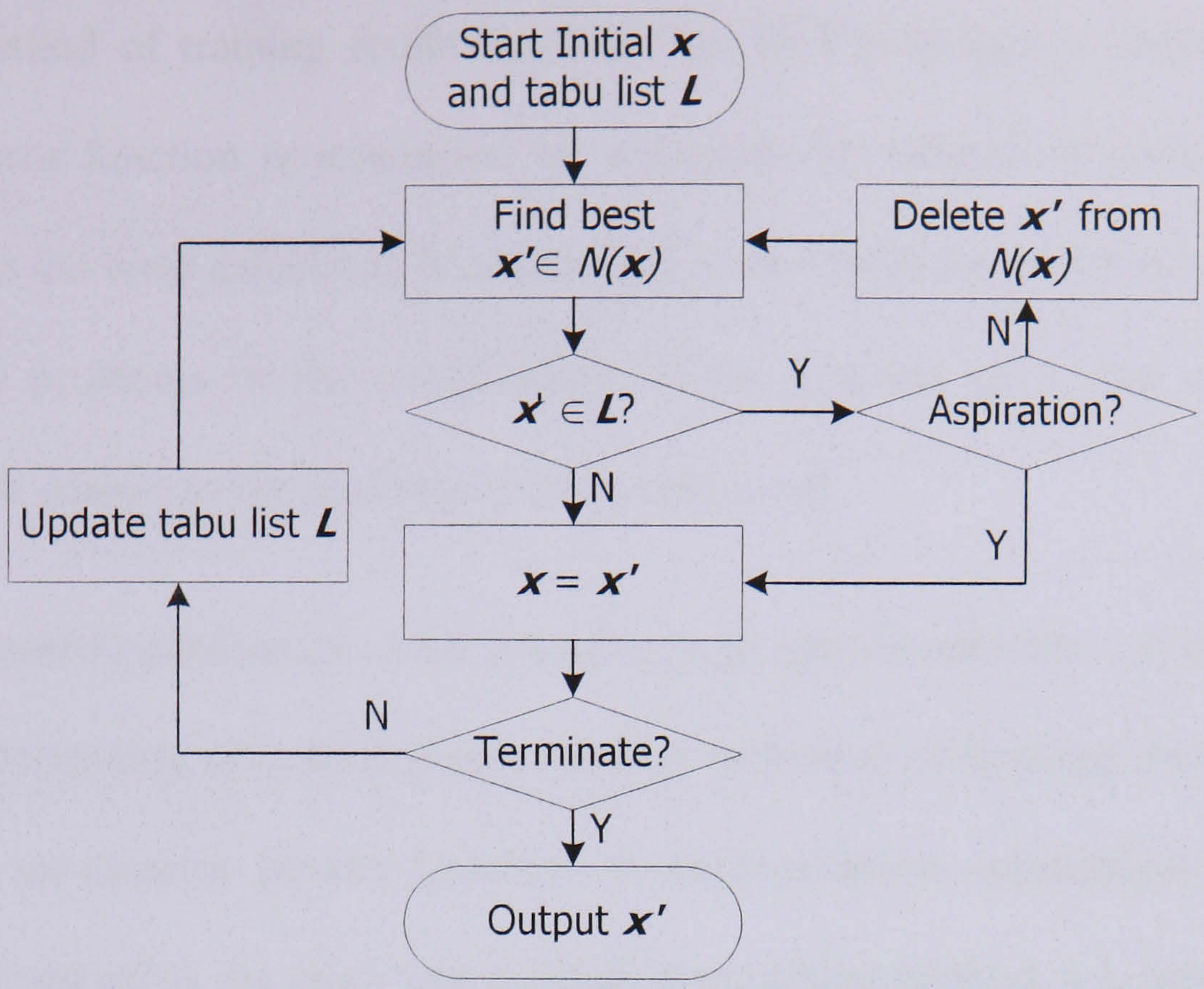


Figure 2.2: Main procedures in TS.

2.2.3 Neural Network

Neural networks derive their inspiration from biological neuron systems and the brain. A neural network consists of a number of sub-units called neurons. A typical neuron consists of inputs, a summing function, a limiting or threshold function and outputs. The interconnected parallel neuron a network and a diagram of an actual neuron are shown in Figure 2.3.

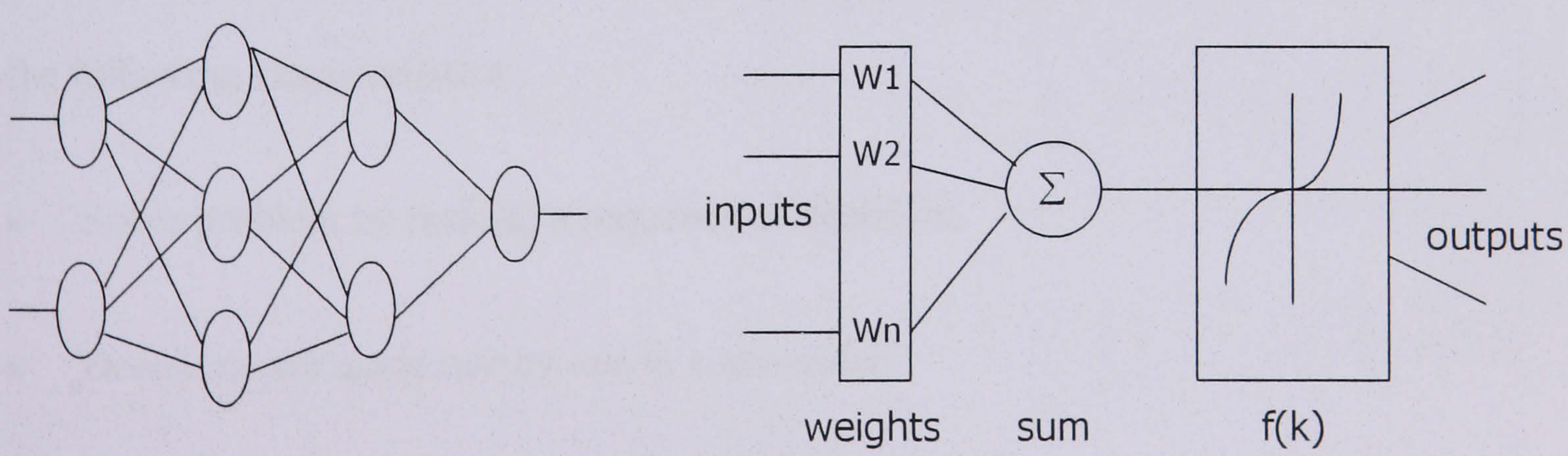


Figure 2.3: The structure of NN.

For neural networks to perform any useful function they must be trained. The most common method of training found has been the back-propagation method [15], in which an error function is minimised by changing the weights of each synapse in proportion to the error calculated. Fundamental to the utilisation of neural networks to optimisation problems is the computation of the gradient of a cost or objective function with respect to the parameters being optimised.

For a constrained optimisation, hard bounds can be introduced either on the inputs or on outputs depending on convenience. Another technique of limiting these problems is by using an exterior penalty function. A multi-criterion optimsation problem is generally solved using the mini-max method, a weighting method, a L_p-norm method, etc.

2.2.4 Greedy Method (GM)

The greedy method solves a given optimisation problem by going through a sequence of (feasible) choices [16]. The sequence starts from well-understood starting configuration, and then iteratively makes the decision that seems best from all those that are currently possible. There is no back-tracking in its searching procedure. The greedy method assumes that a local optimum is part of the global optimum. GM has the following characteristics:

- Solve problem by making a sequence of decisions
- Decisions are made one by one in some order
- Each decision is made using a greedy criterion
- A decision, once made, is (usually) not changed later

2.2.5 Genetic Algorithm

GA is a heuristic search algorithm inspired by the genetic mechanisms of natural species evolution [5], [17]. The basic idea of GA is to represent solutions to a particular problem as individuals in a competing population. As generation passes the more fit individuals, representing better solutions, evolve to produce an optimal solution. Two of the most common GA implementations are 'simple' and 'steady state'. The simple GA is a generational algorithm in which the entire population is replaced each generation. In contrast, in the steady state GA only a few individuals are replaced each generation, which is often referred to as overlapping populations.

The main procedures of GA can be identified as Figure 2.4. Usually, individuals are encodes as chromosome-like string. GA simulates the evolutionary process by generating an initial population of individuals (***P***) and iteratively applying genetic operators i.e. selection, crossover, and mutation, in each reproductive cycle. First, selection is executed to select a couple of individuals for crossover (with a probability of p_c) and mutation (with a probability of p_m) operations. The new generated children are put into an intermediate population, called offspring ***P'***. The reproduction process is repeated until the offspring is full i.e. a predefined number of children have been produced. The offspring can either replace all, or a part of the individuals in the initial population. The so-called incremental replacement is commonly used, in which the less fit individuals are replaced. The execution terminates when the population is convergent or a given number of generations have been run through.

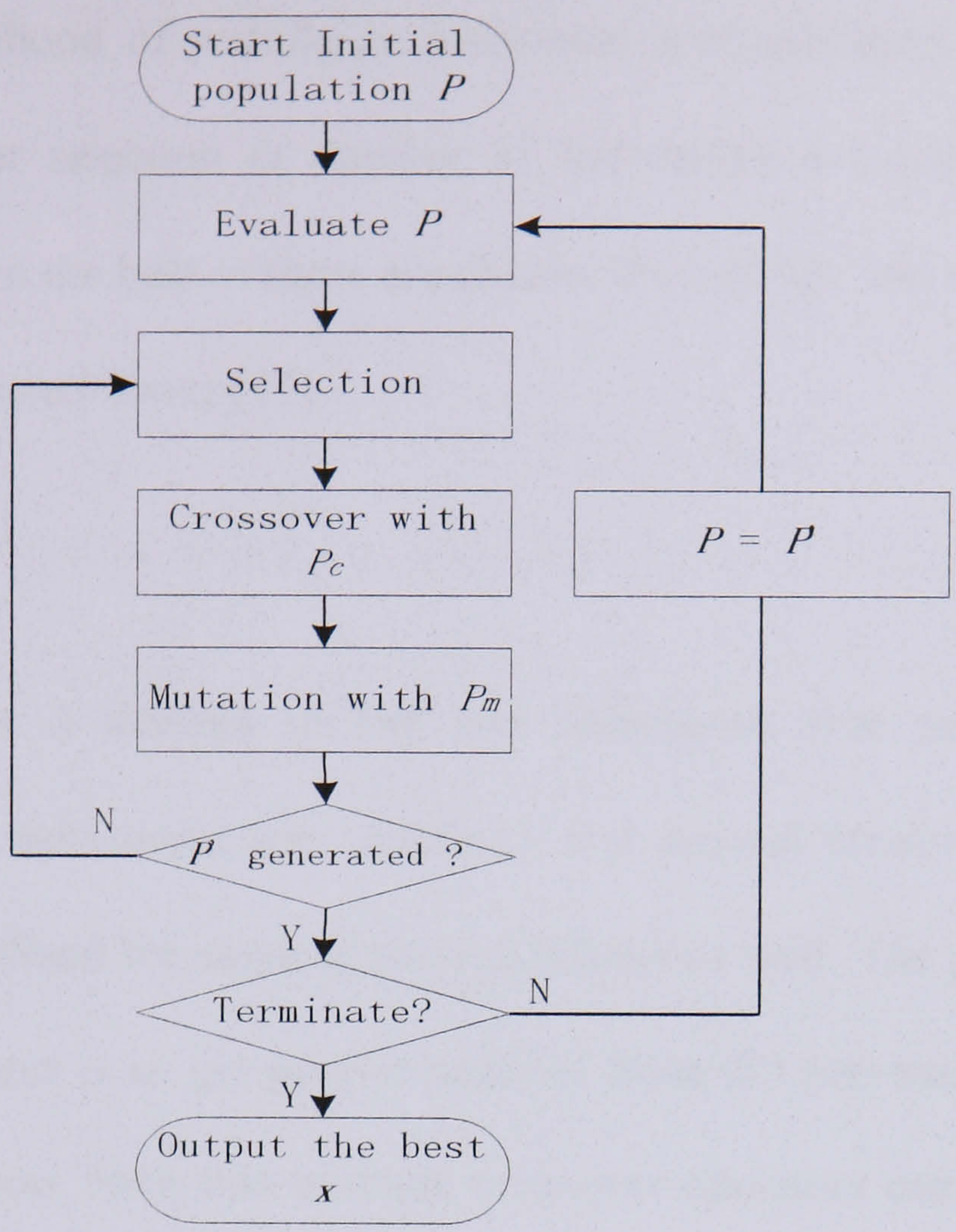


Figure 2.4: Main procedures in GA.

Representation

There are many representations for the individual genomes in the GA including binary-coded, real-coded and permutation-coded schemes. The encoding scheme determines the corresponding genetic operators (initialisation, mutation, crossover, and selection) to be used. Note that each individual must represent a complete solution to the problem need to be optimised.

Selection

The selection method determines how individuals are chosen for mating. If a selection method that picks only the best individual is employed, the population will quickly converge to that individual. However, worse individuals should also be given consideration during the evolution so that the GA can maintain a good exploration in the problem search space. Some of the more common methods include roulette wheel selection (the likelihood of picking an individual is proportional to the individual's fitness), tournament selection (a number of individuals are picked using roulette wheel selection, then the best of these are chosen for mating), and rank selection (pick the best individual every time) [17].

Crossover

Typically crossover is defined so that two individuals (the parents) combine to produce two more individuals (the children). But asexual crossover or single-child crossover can be defined for some specific problems as well. The primary purpose of the crossover operator is to get genetic material from the previous generation to the subsequent generation. Note that multiple crossover operators can be used during an evolution.

Mutation

The mutation operator introduces a certain amount of randomness to the search. It can help the search find solutions that crossover alone might not encounter. Note that multiple mutation operators can also be used during an evolution.

Replacement

Replacement schemes are used by GA with overlapping populations to determine how the new individuals will be assimilated into the population. Replace-worst and replace most-similar are the only really useful replacement schemes. Sometimes replace-parent can be effective, but usually when the parents are similar to the offspring and this is just replace-most-similar [17].

2.3 Comparison of Optimisation Algorithms

The major strengths of SA are that it can walk over and around peaks of different heights and widths with the same step size. In fact, it may climb down the other side of the adjacent hill and climb up another hill; hence it can stochastically guarantee to find an optimal solution. However, the fundamental weakness of SA is that it requires, for solutions to be unique, the energy surface to be generally convex. Because of this constraint, it is often not a good method to use for many problems, which have more than one energy wells. Besides, the final solution of SA is often highly dependent on the initialisation and cooling schedule. This would indicate a non-convex function, and there is no way of overcoming the problem except to run the program from many different initial conditions.

TS has the advantage of not using hill-climbing strategies and its performance could also be enhanced by branch and bound techniques. However, the mathematics behind this technique was not as strong as those behind NN or SA. Furthermore, a solution space would have to be generated. Hence, TS would require knowledge of the entire operation at a more detailed level [18].

NN has the advantage that the entire operations process can be treated as a black box, which would ease the burden of having to model the entire system. It has, however, the disadvantage of requiring gathering data and training the network. Furthermore, the performance of the optimisation would be heavily dependent on the quality of the data used [18].

The principal advantage of greedy methods is that they are usually straightforward, easy to understand and easy to code. Their principal disadvantage is that for many problems there is no greedy method. More precisely, in many cases there is no guarantee that making locally optimal improvements in a locally optimal solution yields the optimal global solution.

Of all the methods discussed so far, the GA perhaps is the most robust. The features of GA are different from other search techniques in several aspects. First, GA is a multi-dimensional method that searches many peaks in parallel, and hence reducing the possibility of local minimum trapping. Second, GA works with a coding of parameters instead of the parameters themselves. The coding of parameter will help the genetic operator to evolve the current state into the next state with minimum computations. Third, GA evaluates the fitness of each string to guide its search instead of the optimisation function. GA only needs to evaluate objective function to guide its search, which means that there is no requirement for derivatives or other auxiliary knowledge. Hence, there is no need for computation of derivatives or other auxiliary functions. Finally, GA explores the search space, where the probability of finding improved performance is high. However, like other population-based technique, GA requires very intensive computation. Nevertheless the availability of supercomputing resources can mitigate this drawback and makes GA a good candidate for most optimisation problems. It is worth noting that the GA is also susceptible to the problems of local minima or maxima, in certain encodings. Therefore, problem-specific modifications appropriate for a given environment to suit the design requirement, are encouraged to be introduced to chromosome representation, genetic operators, replacement scheme, and implementation parameters to achieve enhanced performance. Those techniques would be discussed in more details in the following chapters.

In this thesis, mainly the GA approaches are investigated due to their superior strength in the optimisation domain. A number of problems in wireless systems will be formulated as optimisation problems and processed. Although the problems studied in this thesis by no means cover the entire range of possible applications of the techniques to wireless systems, however, they do represent most attempts to date and show a good effort of the considered subject.

3 Application to Channel Allocation in PCS Networks

3.1 Introduction

With the increased demand for wireless PCS networks, the limited radio resource becomes much more valuable. The efficient utilisation of radio channels with a simultaneous increase of traffic capacity requires proper channel allocation schemes (CAS), which not only guarantee best overall network performance but also have flexible adaptability to non-uniform traffic distribution.

A wireless PCS network is typically divided into cells each of which is allocated a number of channels. The mobile users in a cell are served by a base station. There are two types of calls that may be generated by mobile users, which compete for these channels: new calls and handoff calls. Call blocking occurs when a call arrives at a cell and base station finds no free channels available. One distinguishes between two kinds of blocking, the first is called new call blocking probability (P_b) that refers to the probability that a new call finds all channels busy on its origination in a certain cell; the second is called handoff call blocking probability (P_h), which refers to the probability that a handoff call finds all channels busy on its arrival at its target cell [19]. The *GoS* in PCS network is mainly determined by these two quantities [20]. Since each arriving call, whether it is a new call or a handoff call, will occupy a channel for a period of time if it is granted to service, then the new call blocking probability and handoff blocking probability cannot be decreased simultaneously, and a trade-off must be made [21]. As premature termination of a handoff call is usually more objectionable than the rejection of a new call, it is widely believed that good CAS schemes must give higher priority to the handoff call requests as compared to new call requests [22]. A good overview of various handoff priority-based CAS schemes could be found in [23]. They can be broadly classified into two categories: Guard Channel Schemes (GCS) and Handoff Queuing Schemes (HQS) [20]. In GCS, some of the channels in each cell are reserved for handoff arrival calls only while the rest of channels are shared by both new and handoff calls. In HQS, either new calls or handoff calls are queued for service when no free channels are available.

This study concentrates on the GCS. The critical element in the GCS is channel allocation and reservation. Since the traffic patterns are usually non-uniform, the question arises as to how many channels should be allocated and/or reserved in the different cells so that the best overall system performance can be achieved. Clearly, increasing the number of guard channels will reduce handoff call blocking probability but at the same time, it may increase the new call blocking probability and vice versa. It is therefore very important to choose the appropriate number of channels so that handoff call blocking probability is guaranteed to be under the desired threshold and the sacrifice to new call blocking probability is kept at a minimal level. A multi-period representation of the traffic has been proposed in [24]. In that case, for a given time period, during which the new call traffic rate is supposed to be constant, it needs to determine the proper number of channels to be allocated and/or reserved in each cell. The proposed algorithms are implemented using a GA. By exploiting the powerful search capability of GA, the proposed schemes have been shown to be able to achieve optimal network performance in terms of GoS and have the ability to adapt to non-uniform traffic distribution.

The organisation of this chapter is as follows. Section 3.2 presents the assumptions and analytic model of wireless PCS networks. This is followed by the problem formation of the considered cases in Section 3.3. Section 3.4 describes the proposed GA-based adaptive channel allocation schemes in details. In Section 3.5, the simulation environment is described. Section 3.6 gives the results and discussions. Finally, Section 3.7 concludes this chapter.

3.2 Assumptions and Analytic Model

In this study, it is assumed that for a given time period, the traffic is characterized by the arrival of new calls and by the transition probabilities of handoff calls. New arrival calls follow a Poisson process with arrival rate $\lambda_{i,nc}$, which depends on the cell i. Handoff arrival calls in each cell also follow a Poisson process with arrival rate $\lambda_{i,hi}$, which will be the same as handoff departure rate $\lambda_{i,ho}$ when the network traffic is uniform. When the mobile is accepted by a certain cell, it may reside in that cell for a period of time t_m, which is called sojourn time or dwell time. It is assumed that t_m is a random variable, which is exponentially distributed with mean $1/\eta$. It is also assumed that call holding time is exponentially distributed with mean $1/\mu$. These assumptions conform to those proposed in [25], [26]. The call holding time is the amount of time that the call would remain in progress if it continues to completion without forced termination due to handoff failure [27].

Let $P_{i,nc}$ and $P_{i,hi}$ be the blocking probability of new call and handoff call in each cell, respectively. The overall GoS depends on the balanced sum of both new call blocking probability and handoff call blocking probability in each cell and is defined as follows

$$GoS = \sum_i \frac{\lambda_{i,nc}}{\lambda_{nc}} P_{i,nc} + \alpha \sum_i \frac{\lambda_{i,hi}}{\lambda_{hi}} P_{i,hi} \qquad (3.1)$$

where

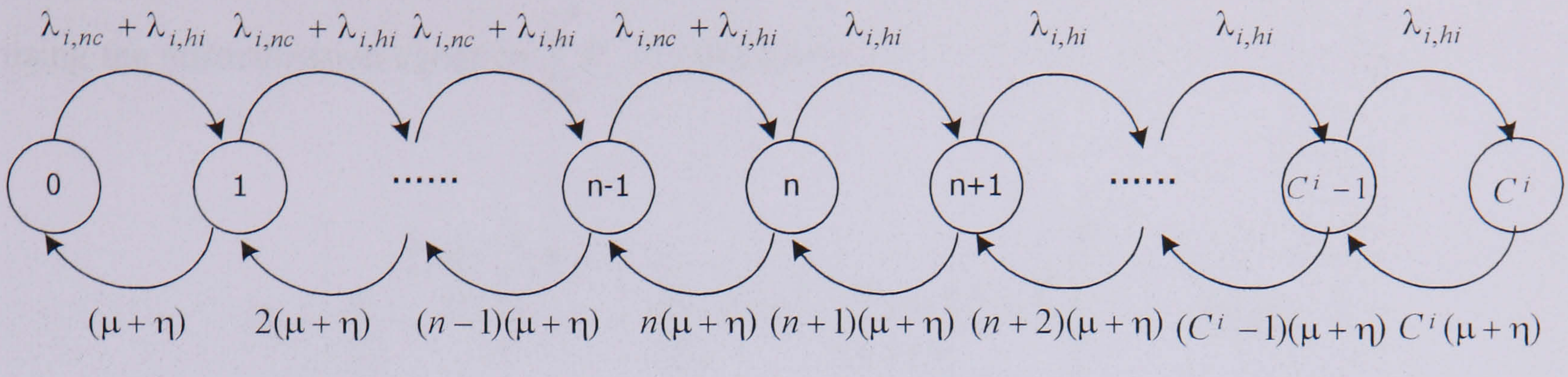
$$\lambda_{nc} = \sum_i \lambda_{i,nc} \tag{3.2}$$

$$\lambda_{hi} = \sum_i \lambda_{i,hi} \tag{3.3}$$

Here, α is the penalty parameter. The *GoS* performance measure penalises dropping of a handoff α times more than blocking a new call request. By adjusting the parameter, a balance between new call blocking probability and handoff call blocking probability could be achieved. By default, the algorithms are conducted by setting α=10, which complies with the previous studies.

The considered system model in this study is an arbitrary topology of N cells with hexagonal cell shape. The channel allocation and channel reservation vectors are denoted as $C = [C^1, \cdots, C^N]$ and $H = [H^1, \cdots, H^N]$ respectively, where C^i is the number of channels allocated to cell i, while H^i is reserved for handoff calls only. The above system can be modelled by a continuous-time Markov process, and since the non-prioritised schemes (NPS) have much in common with reserved channel schemes (RCS) as regard to the state space, therefore only the Markov model of RCS is described here as example. More details about the model could be found in [26].

The state diagram that describes the RCS system performance is shown in Figure 3.1.



Note: $n = C^i - H^i$

Figure 3.1: The Markov state transition diagram for reserved channel schemes

If there is a new call arrival in cell i, it will be blocked if the number of idle channels at that time is less than n, where n=C^i-H^i, otherwise, the new call will be accepted. Handoff calls will only be blocked if the number of idle channels is less than C^i, therefore resulting in a premature termination. Let P_j be the statistical-equilibrium probability of j busy servers, then for $j\in[0,n)$ a transition rate from state P_j to P_{j+1} is given by $\lambda_{i,nc+}\ \lambda_{i,hi}$ and a transition from state P_{j+1} to P_j is given by $(j+1)(\mu+\eta)$ because of the exponential channel occupancy time distribution. For $j\in[n,C^i)$, a transition rate from state P_j to P_{j+1} is given by $\lambda_{i,hi}$ and a transition from state P_{j+1} to P_j is given by $(j+1)(\mu+\eta)$ because there are H^i channels reserved for handoff calls only. Therefore, when n channels are busy at one time, a new call will be blocked whilst a handoff call will be attended to. Based on the state diagram of Figure 3.1, the steady-state probability P_j is defined as

$$P_j = \frac{\left(\frac{\lambda_{i,nc}+\lambda_{i,hi}}{\mu+\eta}\right)^j}{j!}P_0, \qquad j\in[0,n) \tag{3.4}$$

and

$$P_j = \frac{(\lambda_{i,hi})^{j-n}(\lambda_{i,nc}+\lambda_{i,hi})^n}{(\mu+\eta)^j\, j!}P_0, \qquad j\in[n,C^i) \tag{3.5}$$

using the normalization equation $\sum_{j=0}^{C_i} P_j = 1$, this gives

$$P_0 = \left[\sum_{j=0}^{n} \frac{(\frac{\lambda_{i,nc} + \lambda_{i,hi}}{\mu + \eta})^j}{j!} + \sum_{j=n+1}^{C_i} \frac{(\lambda_{i,hi})^{j-n} (\lambda_{i,nc} + \lambda_{i,hi})^n}{j!(\mu + \eta)^j} \right]^{-1} \tag{3.6}$$

In this case, the probabilities $P_{i,nc}$ and $P_{i,hi}$ are given by

$$P_{i,nc} = \sum_{j=n}^{C^i} P_j = \frac{\rho_i^{C^i - H^i} \sum_{k=C^i - H^i + 1}^{C^i} \rho_{i,hi}^{k - C^i + H^i}}{\sum_{k=0}^{C^i - H^i} \frac{1}{k!} \rho_i^k + \rho_i^{C^i - H^i} \sum_{k=C^i - H^i + 1}^{C^i} \frac{1}{k!} \rho_{i,hi}^{k - C^i + H^i}} \tag{3.7}$$

and

$$P_{i,hi} = P_{C^i} = \frac{\rho_i^{C^i - H^i} \sum_{k=C^i - H^i + 1}^{C^i} \rho_{i,hi}^{k - C^i + H^i}}{\sum_{k=0}^{C^i - H^i} \frac{1}{k!} \rho_i^k + \rho_i^{C_i - H_i} \sum_{k=C^i - H^i + 1}^{C^i} \frac{1}{k!} \rho_{i,hi}^{k - C^i + H^i}} \tag{3.8}$$

where

$$\rho_i = \frac{\lambda_{i,nc} + \lambda_{i,hi}}{\mu + \eta} \tag{3.9}$$

$$\rho_{i,hi} = \frac{\lambda_{i,hi}}{\mu + \eta} \tag{3.10}$$

It should be noted that Equations (3.7) and (3.8) could also be used for a non-prioritized channel allocation scheme when H^i=0.

If each cell is modelled as a Markov queuing system, it will have two kinds of traffic streams: new calls and handoff calls. According to the previous assumptions, the new call arrival rate $\lambda_{i,nc}$ in cell i at any time can be acquired from multi-period representation of the traffic. We denote the new call blocking probability and handoff call blocking probability in cell i as $P_{i,nc}$ and $P_{i,hi}$, respectively. Applying the results in [26], [28] to our cases, the handoff departure rate $\lambda_{i,ho}$ and handoff arrival rate $\lambda_{i,hi}$ in cell i can be calculated as follows

$$\lambda_{i,ho} = \frac{(1-P_{i,nc})\lambda_{i,nc}\eta}{\mu+\eta-\eta\sum_{k=1}^{6} q(i,k)(1-P_{k,hi})} \tag{3.11}$$

and

$$\lambda_{i,hi} = q(i,k)\sum_{k=1}^{6} \lambda_{k,ho} \tag{3.12}$$

where q(i,k) is the probability that a mobile user with an engaged call in cell i leaves the cell (handoff departure) by the kth side. In this study, it assumes that the user leaves the cell by each of the sides with equal probability, hence q(i,k) =1/6. Details about derivation of the above Equations can be found in [26].

Equations (3.7), (3.8), (3.11) and (3.12) form a set of recursive formulae, which could be solved by an iterative method as proposed by [26], and a short description is presented here to enhance clarity. An important difference here is that this study deals with GoS, which is the accumulation of all cells' blocking probability. After the iteration procedure converges, all new call and handoff call blocking probabilities are added up as the GoS, which will be viewed as the fitness value for the proposed GA approaches.

Input: C^n, H^i, μ, η, and $[\lambda_{1,nc}, \lambda_{2,nc}, \ldots\ldots, \lambda_{N,nc}]$

Output: [$P_{1,nc}$, $P_{2,nc}$, ……, $P_{N,nc}$], [$P_{1,hi}$, $P_{2,hi}$, ……, $P_{N,hi}$] and GoS

Step 3.1) Initialize each $\lambda_{i,ho}$=0.2 x $\lambda_{i,nc}$, δ_i=1

Step 3.2) If each $|\delta_i| \leq 0.0001$, then go to Step 3.5.

Step 3.3) Compute $P_{i,nc}$ and $P_{i,hi}$ according to Equations (3.7) and (3.8).

Step 3.4) Compute the new value of $\lambda_{i,ho}$ using Equation (3.11) and the new value of $\lambda_{i,hi}$ using Equation (3.12). Let δ_i be the difference between old $\lambda_{i,ho}$ and the new $\lambda_{i,ho}$, go to Step 3.2).

Step 3.5) Compute the GoS by using Equation (3.1).

3.3 Problems Formulation

As for the considered problems, the goal is to find the optimal number of allocated channels and reserved channels in each cell so that the GoS achieves the optimal minimum. In fact, there are three cases in this study:

Case 1: Each cell has an even number of channels allocation, which means that all C^i have a fixed value. GA approach is used to find the optimal number of channels H^i reserved for handoff call in each cell.

Case 2: Each cell has a fix number of channels reserved for handoff calls, which means that all H^i have a fixed value. GA approach is used to find the optimal number of channels C^i allocated in each cell.

Case 3: Each cell has no fixed number of channels allocated or reserved. GA approach is used to find the optimal number of channel allocation C^i and channel reservation H^i in each cell.

Case 1 is a simple uniform fixed channel allocation (FCA) scheme, in which the available channels are segmented among all cells within a cluster. The channel sets are reused in neighbouring clusters, with co-channel constraints having to be satisfied. In this scheme, the same number of nominal channels is allocated to each cell, which is determined by

$$F = \frac{M}{Z} \tag{3.13}$$

where M is the number of available channels to the whole PCS network and Z is the cluster size. For this case, the channel allocation vector $C = [C^1, \cdots, C^N]$ is assumed to be known, and the channel reservation vector $H = [H^1, \cdots, H^N]$ is required that minimises the GoS. An upper bound should be fixed for each point H^i of the vector so that the GA approach can search in a reasonable solution region. As the new call arrival rate and the handoff arrival rate are of the same order of magnitude, then Equations (3.7) and (3.8) imply that the maximal value of H^i should be limited within 4 or 5 in order to satisfy $P_{i,nc} \leq 100 \times P_{i,hi}$. Otherwise the new call blocking probabilities would be too high. In all considered cases, the maximal value is set to 4.

Case 2 is based on the concept of non-uniform compact pattern [29], [30]. To give a brief description of the compact pattern, consider N cells and M channels in the PCS network. The allocation of a channel, say channel k, to the set of co-channel cells forms an allocation pattern of channel k and is denoted as π_k [30]. π_k is described by a set of indicator functions $\{I_1(k), I_2(k), \ldots\ldots, I_N(k)\}$ where

$$I_i(k) = \begin{cases} 1, & \text{channel } k \text{ allocated to cell } i \\ 0, & \text{otherwise} \end{cases} \tag{3.14}$$

Let the compact allocation pattern of a channel be the pattern with minimum average distance between co-channel cells. Assuming the minimum co-channel reuse distance is three cell units, the total number of compact patterns is 14 including clockwise ones and anticlockwise ones. For this case, the channel reservation vector $H = [H^1,\cdots,H^N]$ is assumed to be known, and it needs to find the pattern allocation vector $P = [P^1,\cdots,P^M]$ that minimises the GoS. The maximal number of each point of the vector is set to be the total number of compact patterns, i.e. 14. Then the channel allocation vector $C = [C^1,\cdots,C^N]$ can be subsequently calculated from the pattern allocation vector $\boldsymbol{P}$.

In case 3, which is a combination of the above two cases, the channel allocation vector $C = [C^1,\cdots,C^N]$ and channel reservation vector $H = [H^1,\cdots,H^N]$ are both unfixed in advance and they should be determined by the GA approach by means of minimising the GoS.

All these cases are combinatorial optimisation problems in nature and presumably there is even no efficient algorithm to find an approximately optimal solution. Although some other heuristic methods such as TS [25], GM [31] and SA [32] have been proposed to tackle these problems, this study consider GA as an alternative optimisation approach, which has been shown to outperform other heuristic methods in a wide range of applications [5],[6].

3.4 GA-based Channel Allocation Schemes for PCS networks

GA is an adaptive and robust optimisation and search technique, which borrows the ideas of natural selection and 'survival of the fittest' from natural evolution. Therefore, GA can easily search for potential solutions to solve complex problems in a general, representation-independent manner. Such a search is not guided by stringent mathematical formulation but often requires balancing two conflicting objectives: exploiting the best solutions and exploring the search space.

For all studied cases in this chapter, we choose real-coded GA, which has many advantages over binary GA [33]. It should be noted that real value chromosomes should be converted into integer ones when they are evaluated in the fitness function (also called energy function). In the following subsections, we outline the development of the GA approach in channel allocation schemes, whose structure is depicted in Figure 3.2.

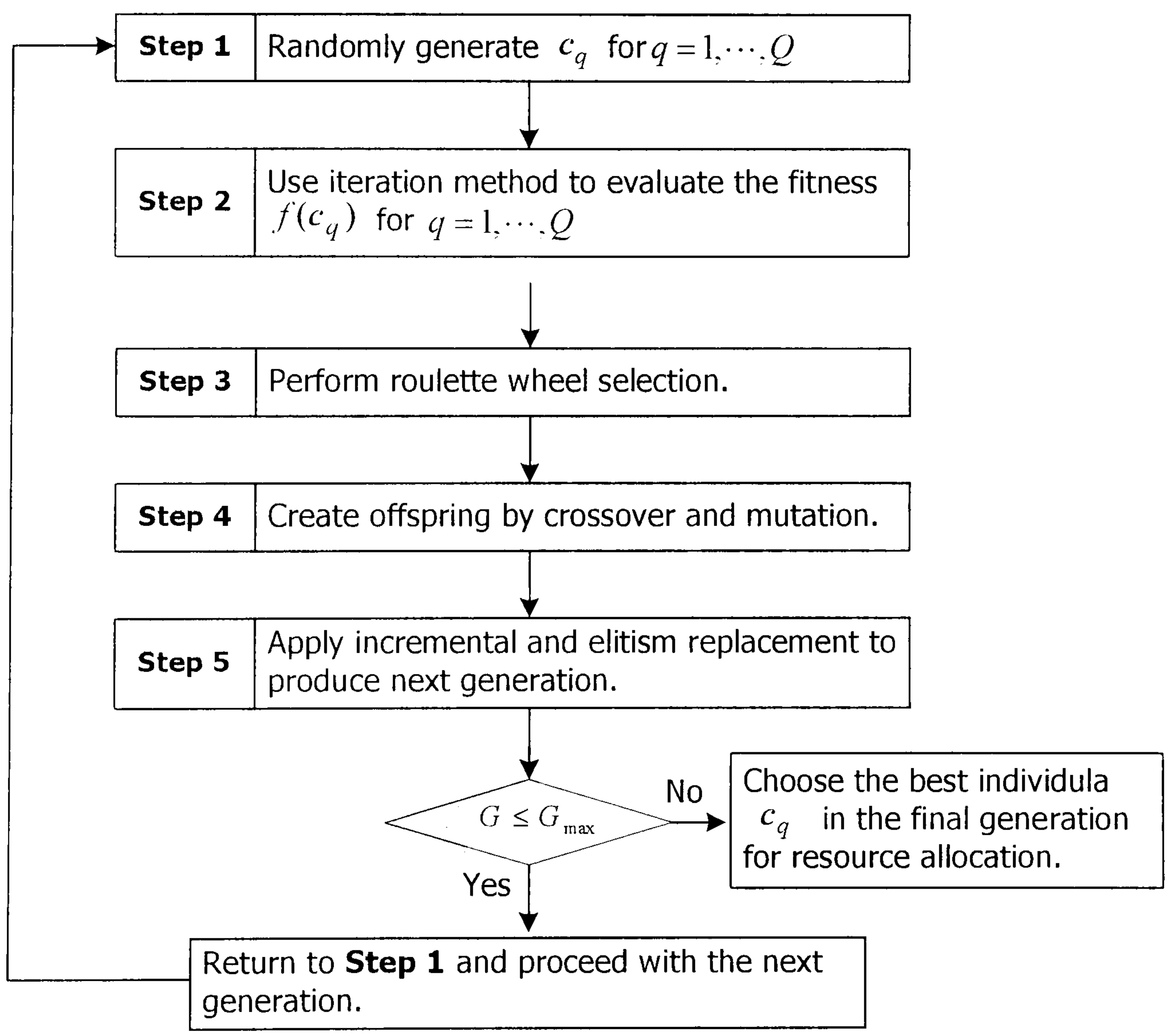


Figure 3.2: Flowchart depicting the structure of the proposed GA channel allocation approaches for PCS system.

3.4.1 Chromosome Representation

The encoding scheme of chromosomes has a major impact on the performance because it can severely limit the search space observed by the system. This study use real-valued representation for all cases under consideration. For case 1, it needs to find the optimal channel reservation vector $H = [H^1, \cdots, H^N]$, therefore an N-dimensional real-valued vector $c = [c^1, \cdots, c^N]$ is employed, where each component corresponds to the number of channels reserved in each cell. For case 2, it needs to find the optimal pattern allocation vector $P = [P^1, \cdots, P^M]$, therefore an M-dimensional real-valued vector $c = [c^1, \cdots, c^N]$ is used, where each component corresponds to the number of pattern that each channel belongs to. For case 3, it needs to find both the optimal channel reservation vector $H = [H^1, \cdots, H^N]$ and the optimal pattern allocation vector $P = [P^1, \cdots, P^M]$. Therefore, we use (N+M)-dimensional real-valued vector $c = [c^1, \cdots, c^{N+M}]$, where the first N components correspond to the number of channels reserved in each cell, and the last M components correspond to the number of patterns that each channel belongs to.

3.4.2 Initialisation

The population of real-coded chromosomes $\{c_q = [c_q^1, \cdots, c_q^K], q = 1, \cdots, Q\}$ is initialised randomly, where K is the number of chromosomes, Q is known as the population size. The lower and upper bound for each variable c^i in the chromosome is denoted by a_k and b_k, respectively. Therefore, for channel reservation cases, a_k=0 and b_k=4. For pattern allocation cases, a_k=1 and b_k=14. The purpose of using random generation is to distribute the initial trial solutions to a highly diversified search space.

3.4.3 Fitness Evaluation

The objective function in Equation (3.1) provides the mechanism for evaluating the fitness of each chromosome. By convention, the fitness function should be a positive value. Since GoS is non-negative, the fitness value (which is to be minimised) of each chromosome $f(c_q)$ is calculated directly from Equation (3.1).

3.4.4 Genetic Operators

Based on the fitness function defined above, three basic types of genetic operators are required to modify the population: selection, crossover, and mutation. In this section, some genetic operators used in the proposed GA approaches will be presented. More discussion about real-valued genetic operators can be found in [34].

3.4.4.1 Selection

Selection is a process used for choosing parent chromosomes to participate in reproduction for the next generation. The reproductive opportunity of an individual parent is normally granted in direct proportion to its fitness value so that highly fit chromosomes contribute more copies to the mating pool than do poor ones. Among the many selection schemes available, this study uses the roulette wheel sampling scheme [5].

3.4.4.2 Crossover Operators

Crossover is a crucial operator that combines two or more parent chromosomes to produce new offspring chromosomes. A suitably designed crossover can significantly accelerate the search process. Three crossover operators are considered: simple one-point crossover, arithmetic crossover, and heuristic crossover [34]. As will be seen later, the combined use of these crossover operators can lead to enhanced performance.

Let c_r and c_q be two selected parent chromosomes. One-point crossover creates two offspring chromosomes $\tilde{c}_r$ and $\tilde{c}_q$ by randomly selecting a crossover point and exchanging segments after that point. Arithmetic crossover produces the offspring as follows:

$$\tilde{\mathbf{c}}_r = z\mathbf{c}_r + (1-z)\mathbf{c}_q \tag{3.15}$$

$$\tilde{\mathbf{c}}_q = z\mathbf{c}_q + (1-z)\mathbf{c}_r \tag{3.16}$$

where $z \sim U(0, 1)$. The rationale of using arithmetic crossover for the proposed GA detector is to reduce detection delays, since some previous studies have shown that arithmetic crossover could increase the convergence rate [34].

On the other hand, heuristic crossover is distinct from both simple and arithmetic crossover because it uses values of the fitness function in determining the search direction. Assuming that the parent c_r is better than c_q in term of the fitness value, i.e., $f(c_r) \geq f(c_q)$ for maximisation and $f(c_r) \leq f(c_q)$ for minimisation problems, then heuristic crossover generates a new offspring $\tilde{c}_r = c_r + z(c_r - c_q)$ and sets $c_q = \tilde{c}_r$. If $\tilde{c}_r$ is not feasible, i.e., $\tilde{c}_r^k < a_k$ or $\tilde{c}_r^k > b_k$ for all $k = 1, \cdots, K$, then another random draw of z is performed to produce a new offspring $\tilde{c}_r$. If no new offspring satisfies the constraints after W attempts, then the operator halts and produces no offspring. The main reasons for incorporating heuristic crossover are to assist in fine local tuning and search in the most promising direction.

It should be noted that crossover is not always invoked. After selecting a pair of parents, the algorithm implements crossover only if a random variable z ($z \sim U(0,1)$) is greater than the crossover rate p_c. Otherwise, the parents remain unaltered. Typical values of p_c lie within the range of 0.6-0.9 [5].

3.4.4.3 Mutation Operators

The mutation operator randomly alters some genes' values in a chromosome with a probability determined by the mutation rate p_m. This can result in entirely new offspring chromosomes. Mutation is used very sparingly in most GAs. Typically, the mutation rate p_m is less than 0.1 [5].

We consider multi-non-uniform mutation [34], which is a dynamic (population dependent) mutation operator aimed at improving single-element tuning and reducing the drawback of random mutation. It is defined as follows: For a parent $\boldsymbol{c}_q$, randomly select an element c_q^k and perform mutation to yield $\tilde{c}_q^k$ as

$$\tilde{c}_q^k = \begin{cases} c_q^k + \Delta[G,(b_k - c_q^k)], & \text{if } z_1 \sim U(0,1) < 0.5 \\ c_q^k - \Delta[G,(c_q^k - a_k)], & \text{if } z_1 \sim U(0,1) \geq 0.5 \end{cases} \tag{3.17}$$

where G denotes the current generation number, and the function $\Delta[G,y]$ is defined as [34]:

$$\Delta[G,y] = (y(1 - \frac{G}{G_{\max}}))^b \tag{3.18}$$

where $z_1, z_2 \sim U(0,1)$ with uniform distribution on the interval [0, 1], G_{max} is the maximum number of generations, and b is a shape parameter, which determines the degree of dependency on generation number (it uses b=3 in the following). The function $\Delta[G, y]$ returns a value in the range [0, y], such that the probability of $\Delta[G, y]$ being close to 0 increases as G increases. This causes the operator to search the space uniformly at the beginning (when G is small), and very locally at later generations [34]. The multi-non-uniform mutation applies the non-uniform mutation operator to all the elements in $\boldsymbol{c}_q$.

3.4.5 Replacement

After a predefined number of offspring has been produced through the above genetic operators, a replacement strategy is required in order to modify the old population with the new generation [35]. We use the so-called incremental replacement, in which children will have the chance to compete with some of the parent individuals. Besides, a so-called elitist strategy could be also used to improve algorithm performance [5], which is also adopted in this study. The elitist strategy appends the best performing chromosome of a previous generation to the current population, and thus ensures that the chromosome with the best fitness value always survives to the next generation.

3.4.6 Termination

Termination is a criterion by which the GA decides whether to continue searching or to stop the search. Typical termination criterion of the GA involves either satisfying a problem-specific success indicator or completing a specified number of generations to be run, G_{max}. Since in the studied problems, the number of iterations required to reach a predefined energy function is not known in advance, this study adopts the latter strategy to avoid excessively high complexity and detection delay.

3.5 Simulation Environment

The analytic models are validated by using discrete event simulation experiments. To simulate a very large wireless PCS network, a wraparound hexagonal topology is employed, which is shown in Figure 3.3(a) [26], [36].

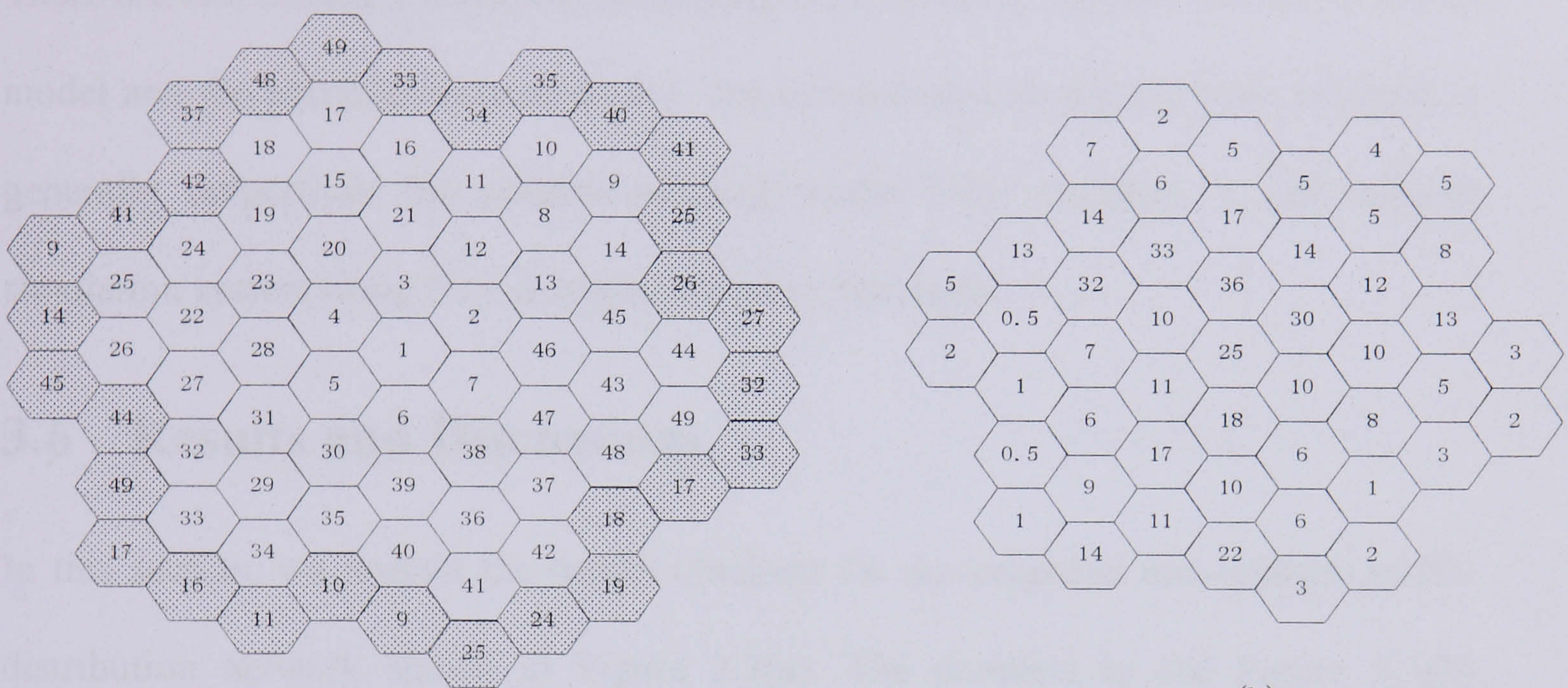


Figure 3.3: Simulation Environment (a) Wraparound Topology of Cellular Network (with cell numbering) (b) Non-uniform traffic pattern (load in Erlangs).

This approach eliminates the boundary effect that occurs in an unwrapped topology. There are 49 cells in the simulated PCS network and the reuse pattern is assumed to be seven to ensure the desired co-channel interference requirements. The mobility behaviour of mobiles in the simulation is described by a two-dimensional random walk [37]. In this model, a mobile stays in the coverage area of a cell for a period of time (sojourn time) that has an exponential distribution with mean $1/\eta$. The mobile then moves to one of the six neighbouring cells with the same routing probabilities of 1/6. As shown in Figure 3.3(a), the white 49 cells are the actual simulated cells and the bordering shaded cells are used to create wraparound topology [26]. Figure 3.3(b) shows a random non-uniform spatial traffic pattern taken from a real cellular system

[26]. The number in each cell represents the offered load and ranges from 0.5 to 36 Erlangs and the average traffic load is 10 Erlangs per cell. This data is used as the network traffic representation for a given time period. This base load is changed proportionately to investigate other offered traffic load conditions.

There are two common models in simulating PCS networks, namely the call-initiated model and the portable-initiated model. The call-initiated model has been reported to generally outperform the portable-initiated model [38]; therefore, a call-initiated simulation system using C++ is implemented in this study.

3.6 Results and Discussions

In this section, we present the results obtained for the proposed non-uniform traffic distribution network shown in Figure 3.3(a). The numbers in the Figure 3.3(b) represent 10 Erlangs per cell traffic condition. These numbers are changed proportionately from 2 Erlangs to 12 Erlangs per cell to determine the network performance under other offered load traffic demands. A summary of the various GA parameters for the proposed schemes is given in Table 3.1.

Table 3.1: Summary of GA parameters used for the channel allocation simulations for PCS system.

Parameter	Value/Type
Population size, Q	50
Generation, $G_{\max}$	200
Representation	Real-valued
Initialisation	Random
Generation selection	Roulette wheel
Crossover operators	Simple + Heuristic + Arithmetic
Crossover probability, p_c	0.88
Mutation operator	Multi-non-uniform
Mutation probability, p_m	0.08
Replacement	Incremental + Elitism

Figures 3.4-3.6 show the results comparison between different channel allocation schemes (both analytical and simulated), namely, non-prioritized fixed channel allocation scheme(FCA-NPS), reserved one fixed channel allocation scheme(FCA-RCS1), reserved four fixed channel allocation scheme(FCA-RCS4), and GA based adaptive reservation channel allocation scheme(FCA-ARCS). From Figure 3.4 and Figure 3.5, it can be seen that FCA-NPS has the lowest new call blocking probability and highest handoff call blocking probability in all four schemes since there is no channel reserved to handoff calls. While FCA-RCS4 has the lowest handoff call blocking probability and highest new call blocking probability in all four schemes because it reserves the most channels for handoff arrival calls. However, as for the overall network performance measurement - *GoS*, the GA based adaptive reservation channel allocation scheme proves to be more efficient in both analytical (solid lines) and simulated results (dashed lines) as shown in Figure 3.6. Under light offered load

conditions (below 4 Erlangs per cell), it is significantly better than FCA-RCS4 scheme and marginally better than FCA-RCS1, whereas under heavy offered load conditions (above 10 Erlangs per cell), it is significantly better than FCA-RCA1 and marginally better than FCA-RCS4. It is shown that FCA-ARCS could adapt to different traffic load and hence determine the best reserved number of channels in each cell, thus achieving the optimal network performance in terms of *GoS*.

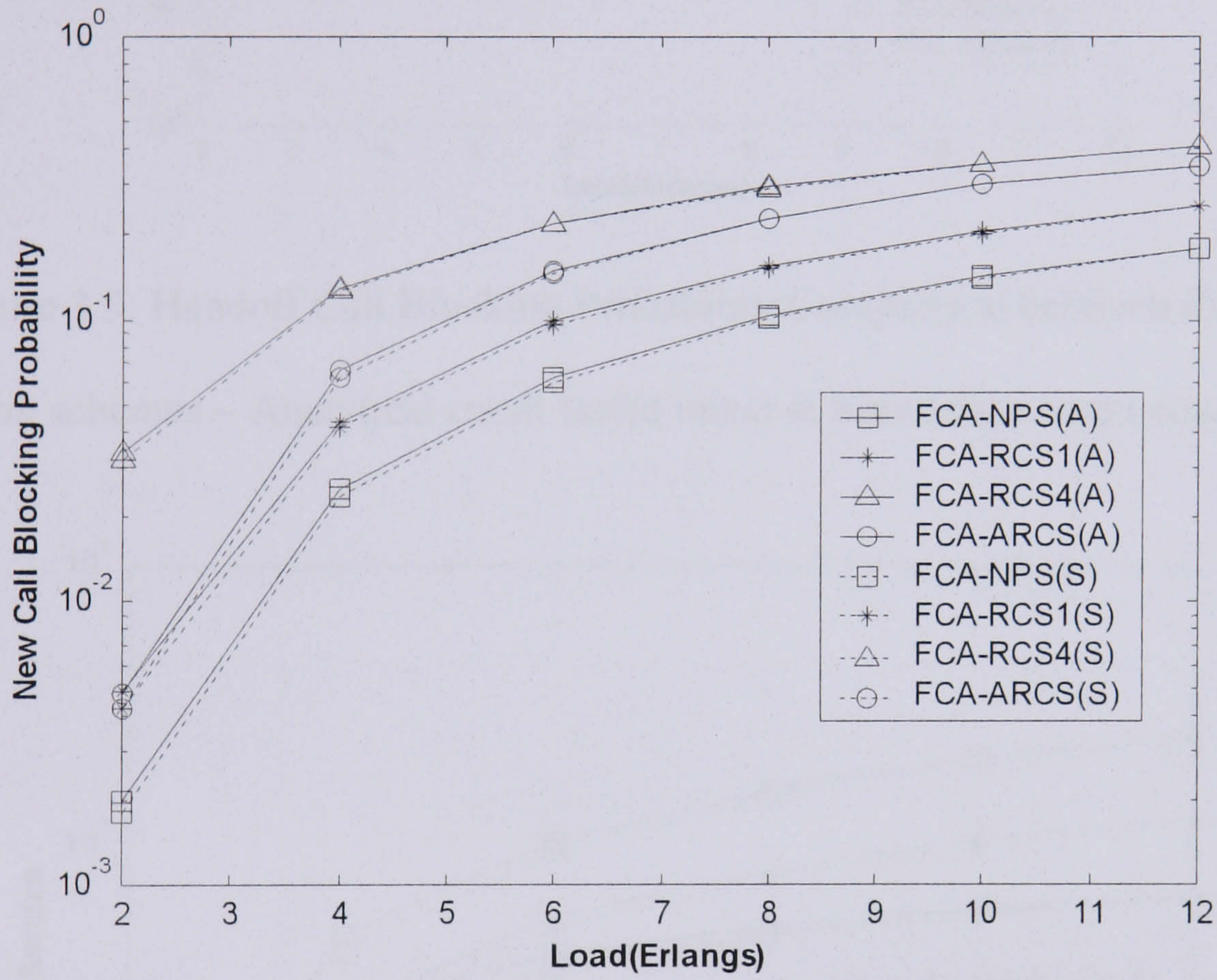


Figure 3.4: New Call Blocking Probability Comparison between different allocation schemes – Analytical result (solid lines) & Simulation result (dashed lines).

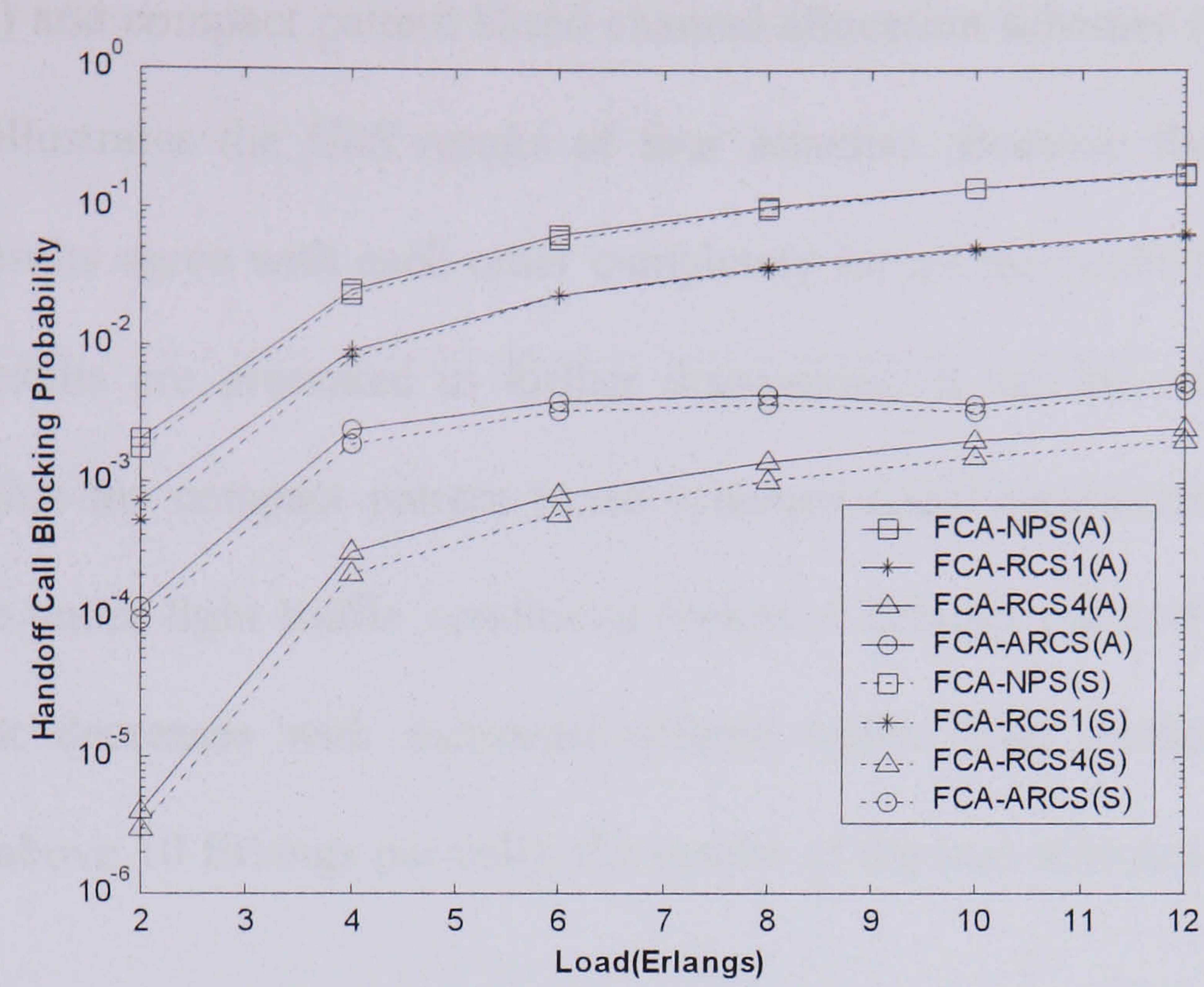


Figure 3.5: Handoff Call Blocking Probability Comparison between different allocation schemes – Analytical result (solid lines) & Simulation result (dashed lines).

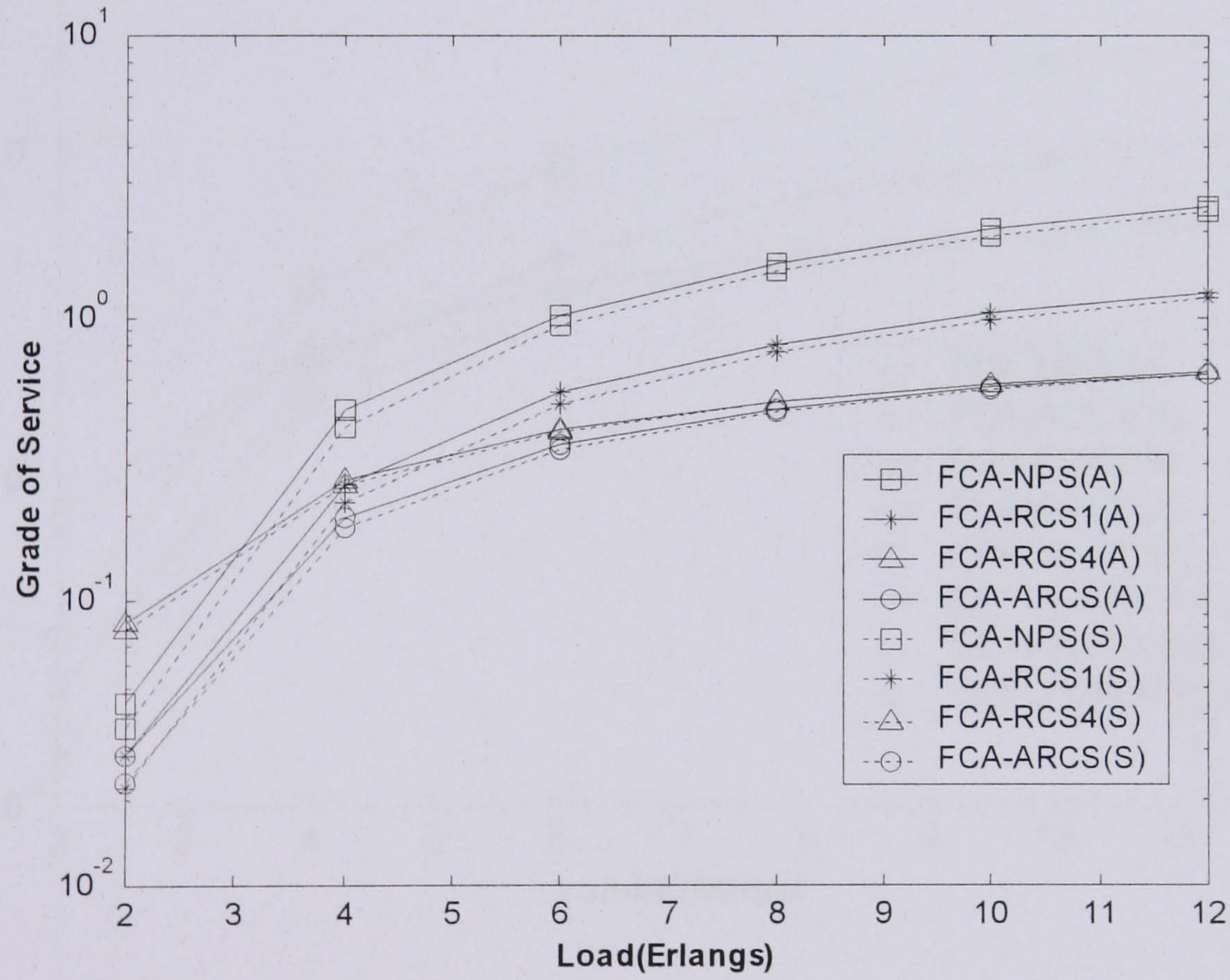


Figure 3.6: Grade of Service Comparison between different allocation schemes – Analytical result (solid lines) & Simulation result (dashed lines).

In order to examine the difference between normal fixed channel allocation schemes

(FCA-based) and compact pattern based channel allocation schemes (CPFCA-based), Figure 3.7 illustrates the *GoS* results of four schemes. Because the analytical and simulated results agree with each other completely for all the studied cases, only the simulated results are presented in further discussions. It can be seen clearly from Figure 3.7 that the compact pattern based schemes could significantly improve the performance under light traffic conditions (below 4 Erlangs per cell). However, the improvement decreases with increased offered traffic load. Under heavy traffic conditions (above 10 Erlangs per cell), the results of the two schemes become almost identical.

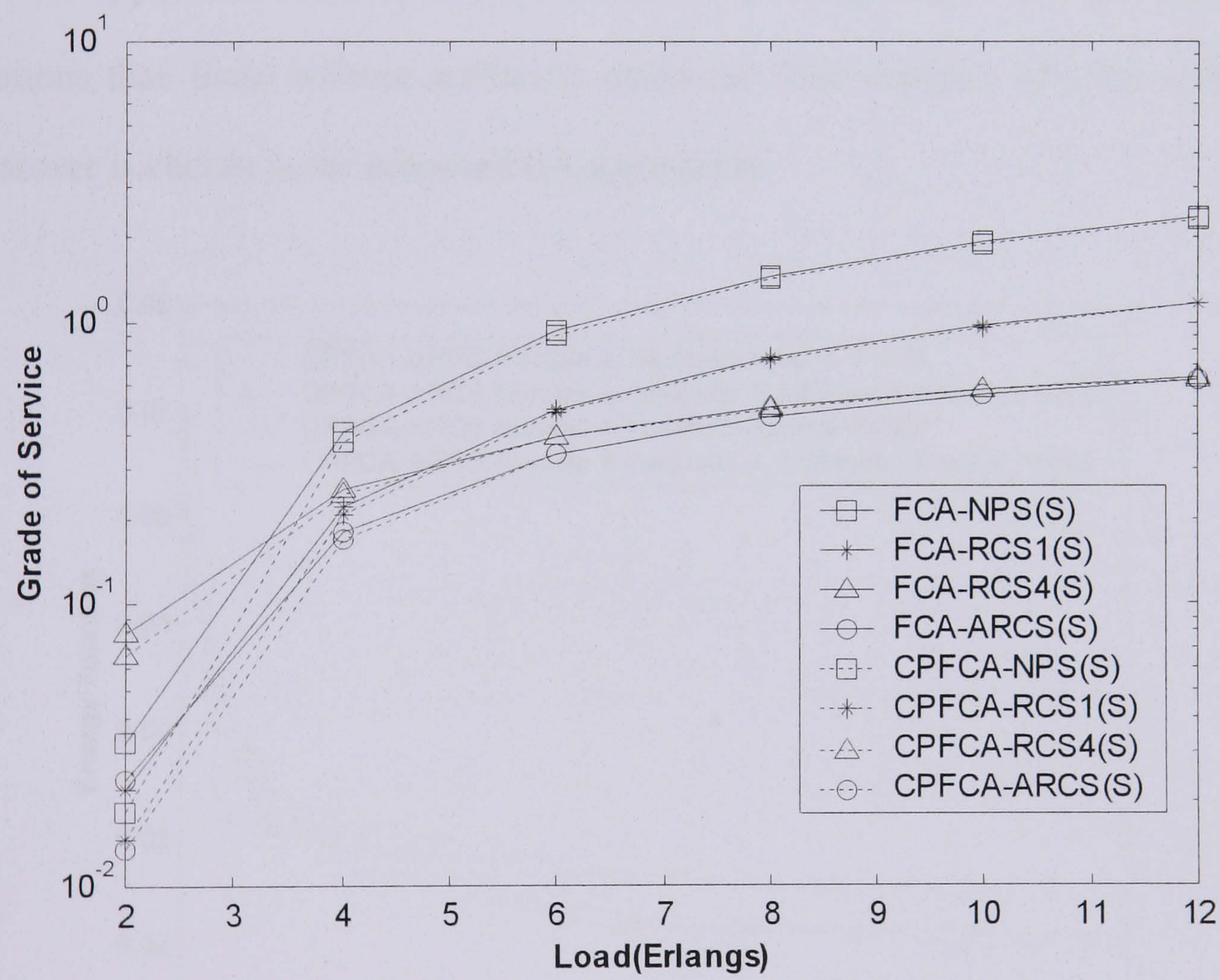


Figure 3.7: Grade of Service Comparison between FCA-based (solid lines) and CPFCA-based different schemes (dashed lines) – Simulated Result.

Figure 3.8 demonstrates the evolution of the fitness function with respect to the number of generations for CPFCA-ARCS using different populations (size of 50 or 80) and different crossover operators (with or without arithmetic crossover). The abscissa indicates the number of generations, while the ordinate shows the best function evaluation found in each generation so far. The traffic load is chosen as 2 Erlangs per cell, which is a light traffic load condition. It is observed that the performance of the proposed scheme improves as the population size increases, since the search space becomes larger. However, the improvement is very marginal and it becomes unjustified compared with the excessive amount of processing time required for each generation. Hence, for the system under consideration, population size of 50 provides a good compromise between performance and complexity. Moreover, it appears that the GA approaches with arithmetic crossover converge faster and provide better solutions than those without arithmetic crossover. This explains why the arithmetic crossover is chosen in the proposed GA approaches.

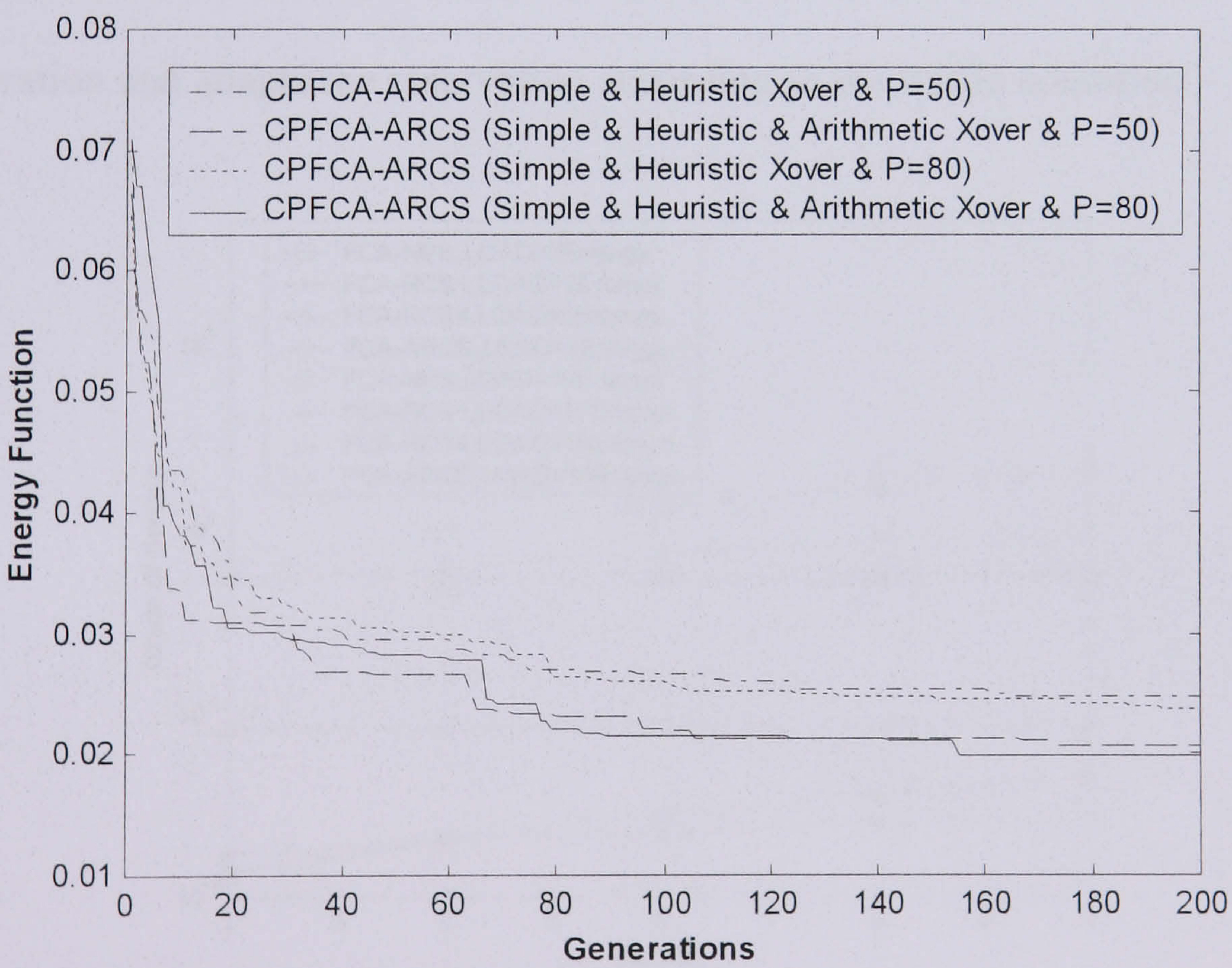


Figure 3.8: Evolution of the energy functions with respect to the number of generations for CPFCA-ARCS using different crossover operators and different population sizes.

Figures 3.9-3.11 demonstrate the effect of the penalty parameter α on the performance in light traffic load (2 Erlangs) and heavy traffic load conditions (10 Erlangs) by choosing α to vary from 2 to 10. Figure 3.9 shows a comparison of *GoS* performance among four different channel allocation schemes, while Figure 3.10 and Figure 3.11 show the new call blocking probability and handoff call blocking probability respectively. Figure 3.9 shows that GA based scheme FCA-ARCS exhibits the best adaptability in both light and heavy traffic conditions. When the penalty parameter α is increased, FCA-ARCS tends to increase reservation slightly in light traffic conditions and heavily in heavy traffic conditions, respectively. From Figure 3.10 and Figure 3.11, it can be clearly seen that both non-prioritized channel allocation schemes (NPS) and fix channel reservation schemes (FCA-RCS1 or FCA-RCS4) have fixed values for both new call and handoff call blocking probability, while FCA-ARCS has flexible ones because it takes the penalty parameter into its allocation consideration and adapts the reservation according to the traffic condition.

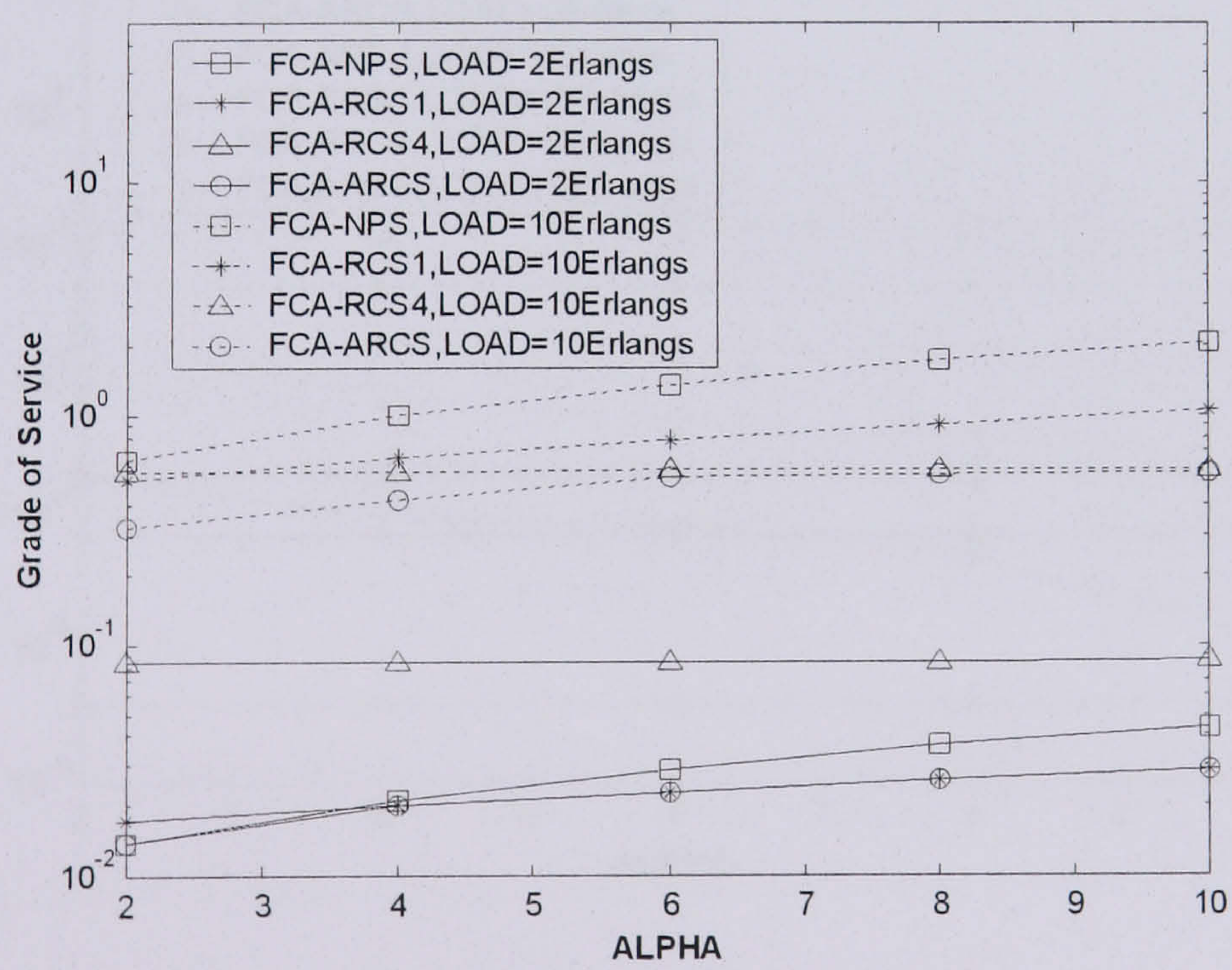


Figure 3.9: Grade of Service Comparison between different channel allocation schemes with penalty parameter α under light traffic load (2 Erlangs) and heavy traffic load (10 Erlangs)

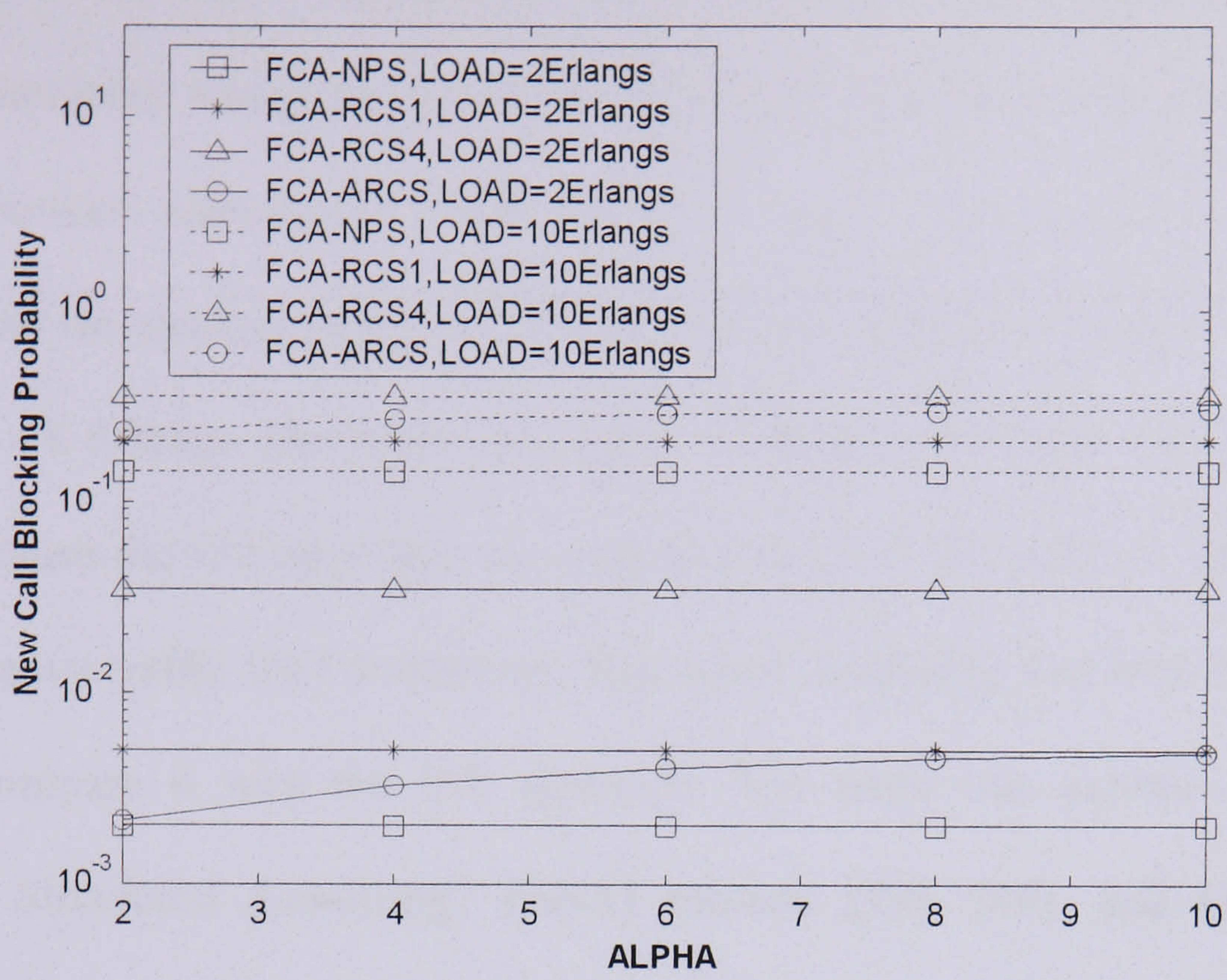


Figure 3.10: New Call Blocking Probability of different channel allocation schemes with penalty parameter α under light traffic load (2 Erlangs) and heavy traffic load (10 Erlangs)

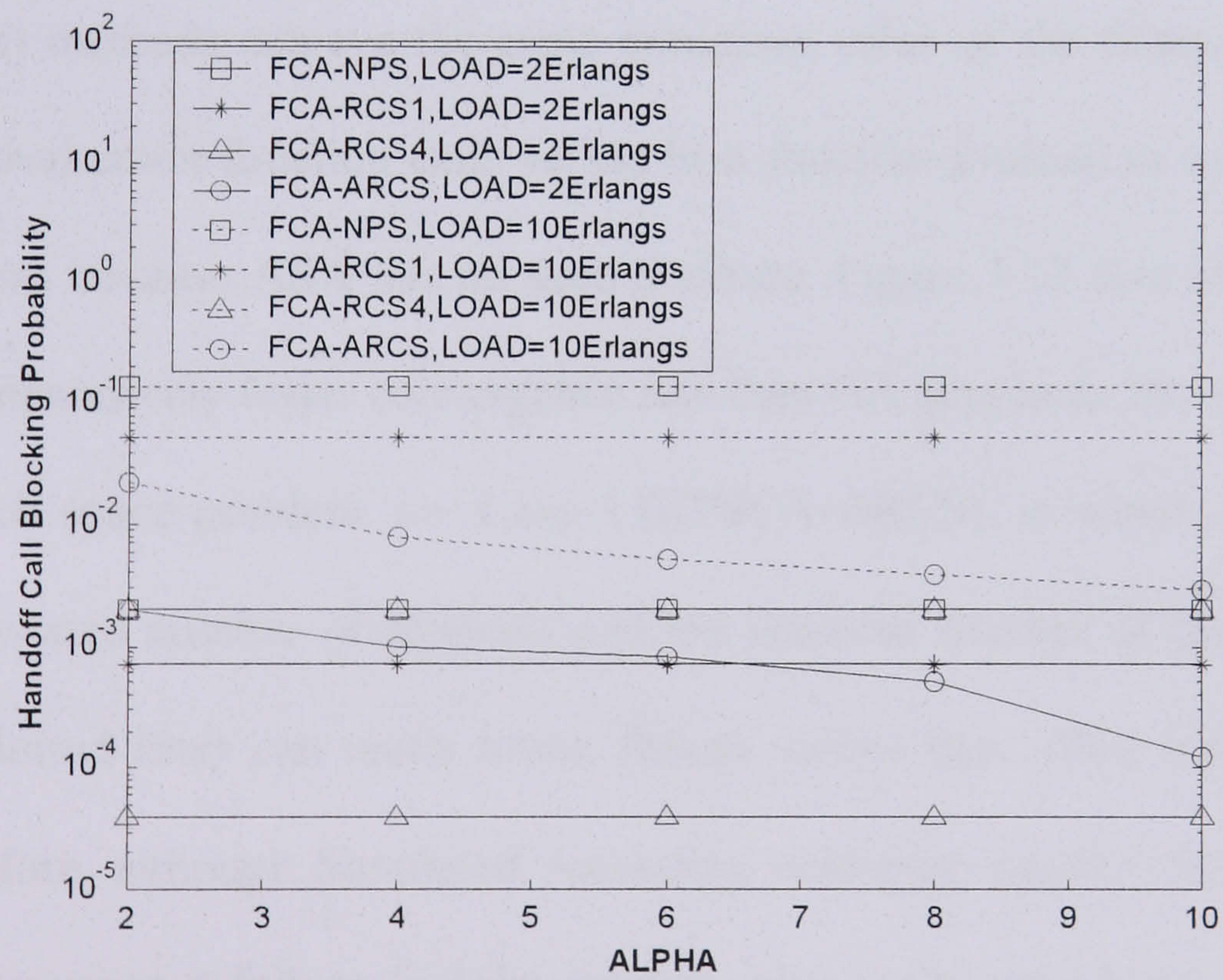


Figure 3.11: Handoff Call Blocking Probability of different channel allocation schemes with penalty parameter α under light traffic load (2 Erlangs) and heavy traffic load (10 Erlangs)

A comparison between GA approach and other heuristic methods such as Tabu Search [25] and Simulated Annealing is also considered. According to the results from [25], the improvements achieved by Tabu Search over the fixed channel schemes are very marginal and the outperforming merit only exist at a particular region of traffic load, around 6.5 - 8 Erlangs. However, this is not the case according to our GA simulation results, because the GA approach can outperform the fixed channel schemes both in light and heavy traffic load conditions. Simulated Annealing was also investigated in order to compare it with the GA approach. The work was carried out using the "Adaptive Simulated Annealing" (ASA) toolbox [39], [40], and GA and ASA simulations were carried out for all three studied cases. It shows when the search space is comparatively small, i.e. Case 1 (FCA-ARCS) and Case 2 (CPFCA-RCS1 or CPFCA-RCS4), GA approach shows the same optimsation strength as ASA methods. Figure 3.12 demonstrates Case 1 for illustration, where both GA (solid line) and ASA (dashed line) methods achieve the same minimum value of the fitness function. The number of evaluation function calls versus best function evaluation was used instead of generations because ASA has no such attribute. Figure 3.12 also shows that ASA exhibits comparatively faster convergence rate than GA approach. However, as for the largest search space problem, i.e. Case 3 (CPFCA-ARCS), in which it should adapt both the allocated number of channels and the reserved number of channels, the GA approach (dotted line) can reach lower fitness values than ASA method (dash-dot line). Therefore, although Simulated Annealing converges slightly faster to a steady cost value, however, it fails to find the optimal value in the considered experiments.

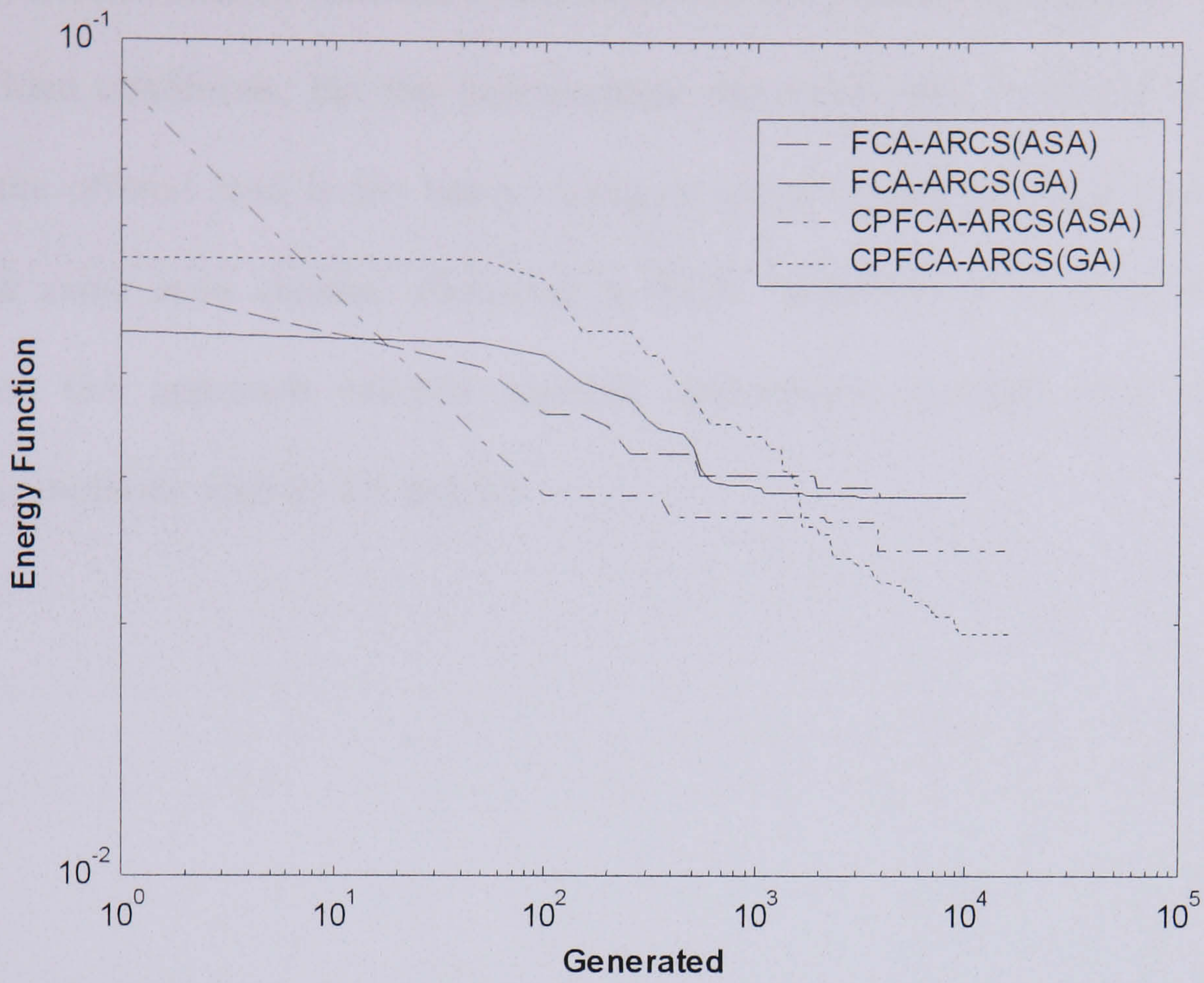


Figure 3.12: Convergence rates comparison between GA and ASA based methods.

3.7 Concluding Remarks

In this chapter, GA-based adaptive channel reservation and allocation schemes are proposed, analytical and simulation models are also presented. The analytical model was validated by discrete event simulations for all studied cases. For a given traffic representation, the GA approach carries out a multi-point search by manipulating and maintaining a population of candidate solutions to find the appropriate number of channels that need to be allocated and/or reserved in each cell in terms of the defined *GoS*.

Simulation results demonstrate that the adaptive reserved FCA scheme significantly outperforms the excess fixed reserved FCA schemes under light traffic load conditions, and significantly outperforms the less fixed reserved FCA schemes under heavy traffic load conditions. The proposed GA approach also proves that compact pattern based allocation schemes could improve FCA performance greatly under light traffic load conditions, but the improvement decreases with increased traffic load. When the offered load is too heavy, compact pattern based GA approaches always find the same even channel allocation as FCA. Moreover, it is observed that the proposed GA approach exhibits superior optimisation strength than some other heuristic methods such as TS and SA.

4 Application to Plane Cover Multiple Access Scheme

4.1 Introduction

Diverse mobile services and the commercial success of cellular networks have stimulated a great demand for spectrally efficient wireless systems. In the near future, next-generation wireless systems are envisioned to support broadband multimedia applications, which include not only voice, but also data traffic such as images, video, etc. [41], [42]. However, the traditional one-seventh resource reuse scheme only has a maximum system utilisation of 1/7 since each cell can only use 1/7 of the total resources even with zero co-channel interference. Such a disappointingly low efficiency cannot grant the foreseeable multimedia demands for wireless system.

In order to overcome the existing problems major changes in network scenario must be invoked. Since spectral capacity is a scarce resource in the wireless environment, efficient utilisation of the limited capacity by using multi-access strategy and new resource allocation is not only desirable but is also imperative for next-generation wireless systems. There has been a significant research effort devoted to achieve greater system capacity in recent years [23]. Capture Division Packet Access (CDPA) as proposed in [43] introduces the error recovery mechanisms to remove the limitation imposed on the cell reuse factor and hence increases the system capacity. Reuse Partitioning (RP) permits more intense reuse for those users for which transmission quality will not be compromised, thus achieving greater system utilisation [44]. PCMA has also been studied as a means of improving the attainable system capacity.

PCMA seeks to maximise the number of parallel transmissions among cells by defining virtual cells, in which users transmit using different reuse factors [45]. It has been shown that PCMA can provide even greater system capacity under a variety of propagation conditions and can also provide minimal delay relative to alternative systems [46].

In this chapter, a PCMA-based wireless system is considered as the test-bed and a new resource allocation scheme is proposed using GA. By exploiting the powerful search capability of GA, the proposed scheme is able to improve the attainable system capacity, and furthermore outperform both uniform resource allocation and the recently proposed Greedy-based '*min*' algorithm.

The organisation of this chapter is as follows. Section 4.2 introduces fundamentals of the PCMA-based system and presents the resource allocation problem. Section 4.3 describes the proposed GA based resource allocation scheme in details. In Section 4.4, the simulation environment is described. Section 4.5 gives the results and discussions. Finally, in Section 4.6 conclusions are drawn.

4.2 Resource Allocation Problem in PCMA

The packet-switching system model has been widely recognized as the most suitable model for traffic integration and variable bandwidth services and has received great support even in the cellular environment. PCMA is essentially a packet-switching TDMA access strategy, which eliminates the need for complex borrowing or locking algorithms, thus statistical multiplexing, soft handoff and macro-diversity can be managed efficiently [46]. Furthermore, a packet capture model is used in PCMA, which allows it to achieve much more intensive reuse of resources than Reuse Partitioning. Consequently, much greater system throughput can be achieved. In [45] and [46], the authors have studied non-uniform and uniform allocation in the PCMA-based system, respectively. However, due to the suboptimal characteristics of the proposed algorithms, they failed to achieve efficient system capacity utilisation. In this study, we aim to find an improved resource allocation scheme in the PCMA-based system by using GA approach.

We assume an idealized cellular system with one base station that can support m connections, and each connection made by a mobile user requires one unit of bandwidth (UB) per second. We also assume the system cells are constructed in a cluster of B cells with a single network call controller being responsible for call management. Such architecture has been proven to be optimal for micro- and pico-cellular networks [47]. These assumptions conform to those proposed in [45], [46].

For a packet-switching system, there are two important performance metrics, the probability of overload P_o, and the expected duration of overload θ, which is defined as the average time duration of the overload state that one cell experiences. From [45], [46] we have following equations

$$P_{o,m} = \left(\frac{B-1}{B}\right)^N \sum_{i=m+1}^{N} \binom{N}{i} \cdot \left(\frac{1}{B-1}\right)^i \tag{4.1}$$

and

$$\theta = T_m \cdot \left(\frac{B-1}{N-m} \cdot \frac{P_{o,m}}{P_m}\right) \tag{4.2}$$

and

$$P_m = \binom{N}{m} \cdot \left(\frac{1}{B-1}\right)^i \cdot \left(\frac{B-1}{B}\right)^N \tag{4.3}$$

where m is the maximum number of active connections that one base station can support, B is the number of base stations in the cluster, N is the connection admission threshold of the cluster, and T_{in} is the mean time between handoff events for any given mobile.

In this study, we consider a cluster of B=16 cells, in which there are D=8 possible coverage patterns including both uniform and non-uniform ones, and four of them are presented in Figure 4.1 as illustration. Note that the numbered 4 cells represent a subset case study which will be discussed in Section 4.5. In each pattern, the shaded cells can be granted some *UBs* simultaneously at a given time slot for servicing users. However, it should be noted that one *UB* under different pattern can service different amount of users, which is referred to as 'throughput'. Uniform coverage pattern (C_1) is illustrated in Figure 4.1(a), in which every cell can be allocated some *UBs* simultaneously for mobile users within the cell. The possible throughput is 0.4408, achieved by using two levels of virtual cells with reuse factors 1 and 3 according to [46]. Figure 4.1(b) and Figure 4.1(c) show two possible stripe patterns C_2 and C_3, in which half of the system cells are given some *UBs* for use. Furthermore, additional four new stripe patterns (C_4-C_7) can be obtained by rotating these two stripe patterns, respectively. The throughput achieved in each of these stripe patterns is 0.7164 [45]. Figure 4.1(d) illustrates the traditional 1/3 reuse pattern (C_8). This pattern can achieve the highest throughput which is 0.9186 [45]. However, only a third of system cells can enjoy this high throughput.

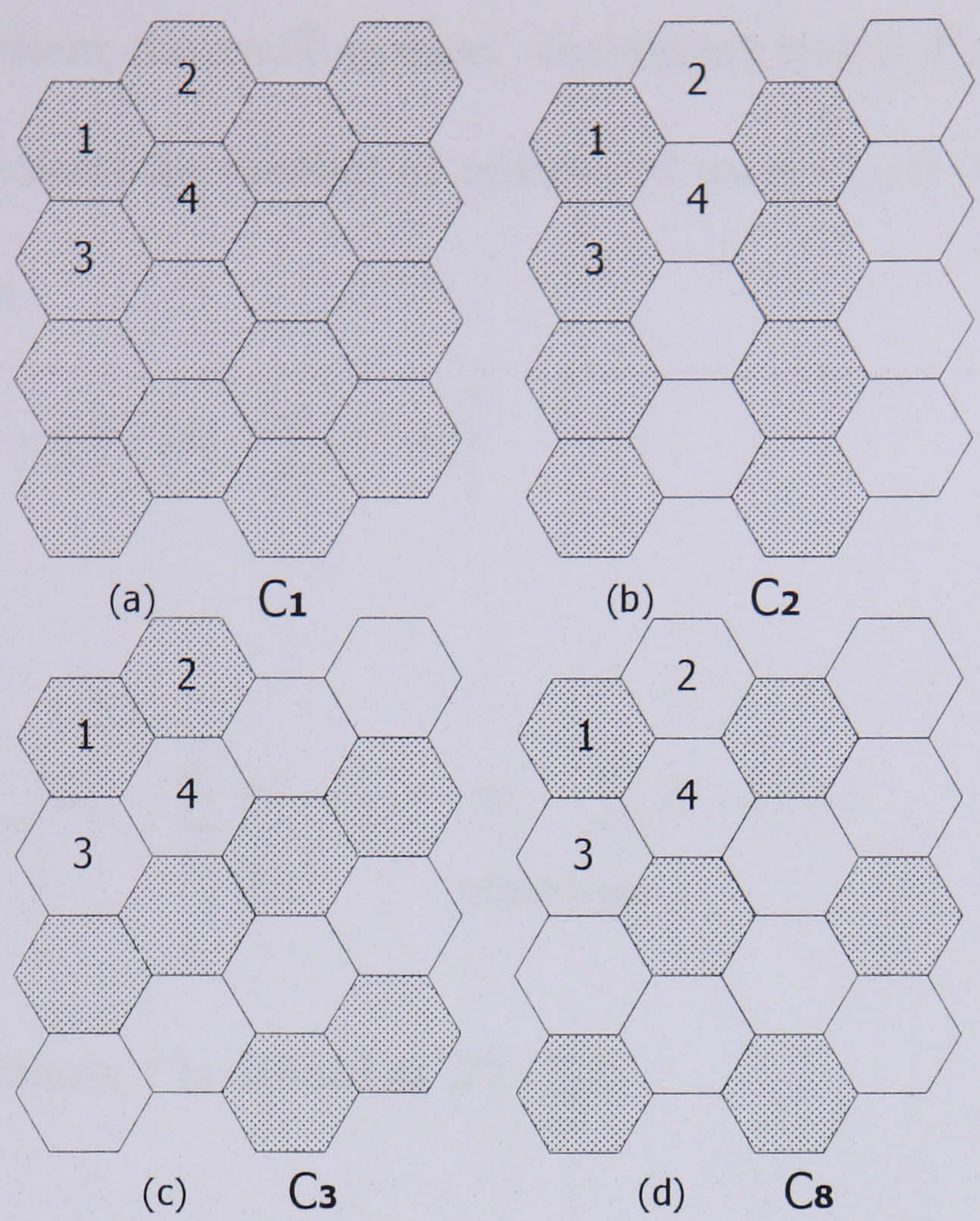


Figure 4.1: Four possible coverage patterns for a 16-cell cluster.

The main issue to consider for resource allocation is the number of *UBs* that should be allocated, in a given time period, to each of these possible coverage pattern. For a specific coverage pattern C_r, we define S_r as the corresponding throughput matrix. For example, each element S_{ij}^1 in matrix S_1 equals to 0.4408, while only half of the elements S_{ij}^2 in matrix S_2 equals to 0.7164 and the other half equals to zero [46]. According to the above definitions, if x *UBs* are allocated to pattern C_r, then cell (ij) can service $(x \cdot S_{ij}^r)$ user connections. We denote a resource allocation as a vector of variables $\boldsymbol{R} = [R^1, \cdots, R^D]$, where R_k is the number of *UBs* allocated to coverage pattern C_k. Note that a base station can support a maximum of m connections under uniform coverage, and hence a maximum of $Z = m / S_{ij}^1$ *UB*s being allocated, i.e. $\sum_{k=1}^{D} R^k \le Z$. The users' distribution in the cluster is mapped to an n by n matrix, $\boldsymbol{M}$. After applying an allocation to the system, the resulting users' distribution matrix $\boldsymbol{T}$ is then changed and the norm that represents the number of unserviced users N_{uu} is reduced. N_{uu} can be obtained as follows

$$N_{uu} = \|T\| = \left\| M - \sum_{k=1}^{D} R^{k} \cdot S_{ij}^{k} \right\| \tag{4.4}$$

Where

$$T_{ij} = \begin{cases} M_{ij} - \sum_{k=1}^{D} R^{k} \cdot S_{ij}^{k}, & M_{ij} - \sum_{k=1}^{D} R^{k} \cdot S_{ij}^{k} \geq 0 \\ 0 \quad , & \text{otherwise} \end{cases} \tag{4.5}$$

and the norm of a matrix T is defined as $\|T\| = \sum_{i,j} T_{ij}$.

An optimal way of resource (UBs) distribution among the coverage patterns is required so that the number of unserviced users is minimised, which is thereafter called the Minimum Unserviced Allocation (MUA) problem. We can find that it is necessary to enumerate explicitly or implicitly all non-isomorphic combinations of size D^{Z} at least in order to get the optimal solution; therefore, the problem space increases exponentially as Z increases. The proof below shows that the MUA problem belongs to the class of NP-hard combinatorial optimisation [3]. This suggests that there is no algorithm that solves the problem in polynomial time, and even a traditional heuristic search technique will find it too difficult to arrive at a good coverage of the space in an efficient manner. Hence, this study considers GA as an optimisation approach, which has been shown to be very well suited for finding approximate optimal solutions to difficult instances of combinatorial optimisation problems [5], [6].

Theorem 4.1: The problem of MUA is NP-complete.

Proof: When a decision version of a combinatorial optimisation problem is proved to belong to the class of NP-complete problems, then the optimisation version is NP-hard. Therefore, we show that the decision version of MUA problem is NP-complete, which can be formulated as follows:

Instance: Collection of D coverage patterns S_r, users' distribution in $\boldsymbol{M}$, positive integer Z of available bandwidth units, and a nonnegative integer F.

Question: Is there a *UBs* allocation of size Z or less chosen from S_r, such that unserved users in $\boldsymbol{M}$ is less than or equals to F ?

It's easy to see that the MUA decision problem belongs to NP class. To prove the MUA decision problem is NP-complete, we need prove that if it were possible to solve it in polynomial time, then it would be possible to solve in polynomial time a problem that is already known to be NP-complete, the so-called Minimum Set Cover (MSC) problem [3], which is formulated as follows:

Instance: Collection $\boldsymbol{X}$ of subsets of a finite set $\boldsymbol{Y}$, positive integer K.

Question: Is there a cover for $\boldsymbol{Y}$ of size K or less chosen from $\boldsymbol{X}$, i.e., a subset $X' \subseteq X$ with $|X'| \leq K$ such that every element in $\boldsymbol{Y}$ appears in at least one member of $\boldsymbol{X}'$?

If we choose $n \geq \sqrt{|Y|}$ and construct an n by n matrix $\boldsymbol{M}$ as the corresponding square matrix of $\boldsymbol{Y}$ representing the users' distribution in the MUA problem (filling zeros if necessary as shown below), then transform $\boldsymbol{X}$ to the set of coverage patterns S_r in the same way (and thus $D=|\boldsymbol{X}|$), and set $K=Z$ which is the available resource (*UBs*) in the MUA problem. Finally, set $F=0$. As a result, an instance of the MSC problem can be transformed to the corresponding instance of the MUA decision problem in polynomial time. As an illustrated example we consider a set cover instance with a ground set Y which consists of 7 elements. The collection of subsets be: X= {1000011, 1100101, 0010011, 0111101}, where '1' means the corresponding element of Y is in the subset and '0' otherwise, and K=2. In this case, the MSC problem is to find out whether two sets cover Y completely. The reduction works as follows: we choose $n \geq \sqrt{|Y|}$, i.e., n=3 in this case. The four coverage patterns transformed from $\boldsymbol{X}$ are {100;001;100}, {110;010;100}, {001;001;100}, {011;110;100}, the users' distribution matrix $\boldsymbol{M}$ is {111;111;100}. Note that in all patterns, as well as in $\boldsymbol{M}$, the last two entries are 0, which are only used to fill up the square matrix. Finally, we choose Z=K, which is 2 in this case. Now it is easy to see that if and only if we can satisfy all demands in the MUA problem, then the set cover instance has a solution. This suggests that if we had a polynomial time algorithm for the MUA problem, we would have a polynomial time algorithm for the MSC problem, which would imply that P=NP. So the MUA problem is NP-hard. Furthermore, we can readily calculate the cost value of each candidate allocation by substituting the $\boldsymbol{R}$ with it in the Equations (4.4) and (4.5), so the MUA problem is also in NP, and thus is actually NP-complete.

4.3 GA-based Resource Allocation Scheme for PCMA Systems

Real-coded GA has been chosen for the studied cases in this study. It should be noted that real value chromosomes should be converted into integer ones when they are evaluated in the fitness function. The following subsections outline the development of the GA-based allocation scheme, whose structure is depicted in Figure 4.2. The procedures in Figure 4.2 represent one profile of the whole allocation process at a certain time period. The users' distribution changes in time and the GA approach needs to be triggered again to find the new allocation solution. Therefore, the depicted GA procedures in Figure 4.2 are repeated at a time granularity determined by practical system requirements.

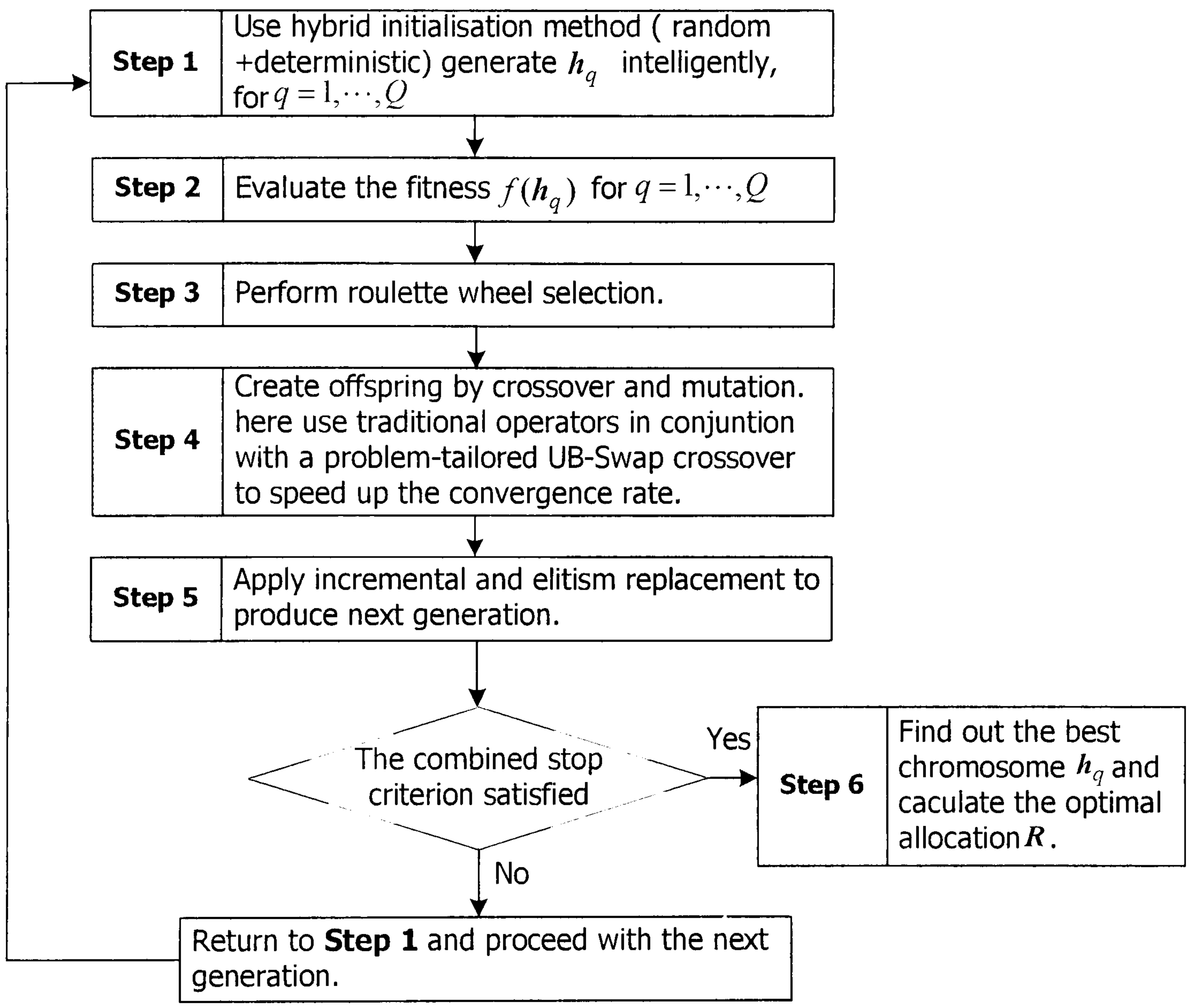


Figure 4.2: Flowchart depicting the structure of the proposed GA channel allocation approaches for PCMA system.

4.3.1 Chromosome Representation

The encoding scheme of chromosomes has a major impact on the performance because it can severely limit the search space observed by the system. For the proposed GA allocation approach, where the variable components are in real space, a real-valued encoding scheme is used in order to move the representation closer to the problem domain. The objective is to find a resource allocation vector $\boldsymbol{R}=[R^1,\cdots,R^D]$ that minimises the total number of unserviced users. However, there are difficulties if $\boldsymbol{R}$ were the variable vector to be determined by GA as a change in the value of element R^i will affect the value of another element R^j and this change is unclear in the GA search space. Therefore, the target needs to be modified into another problem space so that the GA approach is capable of searching for it. Since there are Z *UBs* available to the system, the problem can then be formulated as to which coverage pattern each of these available *UBs* should be allocated. We denote a Z-dimensional real-valued chromosome vector $\boldsymbol{h}=[h^1,\cdots,h^Z]$ as a pattern allocation vector, where each component corresponds to the number of coverage pattern C_r that the kth UB is allocated to. Thus, the number of *UBs* allocated to pattern C_k, i.e. R_k, can be obtained as

$$R^k = \sum_{i=1}^{Z} \begin{cases} 1, & \lfloor h^i \rfloor = k \\ 0, & \lfloor h^i \rfloor \neq k \end{cases} \qquad \text{for k = 1, 2 ...8.} \tag{4.6}$$

where $\lfloor h^i \rfloor$ is the integer truncation of the real-value h^i. The lower and upper bound for each variable h^i in the chromosome are denoted by a_k and b_k, respectively. We choose a_k=1 and b_k=8.999 so that each real valued element h^i in the chromosome can achieve any integer value in the range [1, 8], which represents pattern C_1 to C_8.

4.3.2 Initialisation

The population of real-coded chromosomes $\{\boldsymbol{h}_q=[h_q^1,\cdots,h_q^Z],\ q=1,\cdots,Q\}$ is initialised by employing a hybrid of random and deterministic approaches, where Q is known as the *population size*. The purpose of using a hybrid generation is to distribute the initial trial solutions intelligently. While a deterministic solution creates part of the population, which is allocated in the vicinity of the optimal solution, the random part of population maintains the diversity. To achieve this, the initialisation procedure produces the deterministic chromosomes as each variable h_q^k equals to one integer value in the range [1, 8], which means each of these possible solutions applies one type of eight possible coverage patterns exclusively.

4.3.3 Fitness Evaluation

By convention, the fitness function should be a positive value. Since N_{uu} is non-negative, Equation (4.4) provides the mechanism for evaluating the fitness of each chromosome (possible solution to the problem), and therefore serves as the fitness function (or energy function) of the proposed GA allocation approach. As the aim is to minimise the number of unserviced users, i.e., N_{uu}, then the lowest value in (4.4) corresponds to the best chromosome. The fitness value of each chromosome $f(\boldsymbol{h}_q)$ can be calculated as follows: For each possible pattern allocation vector $\boldsymbol{h}_q$, first calculate the resource allocation vector $\boldsymbol{R}$ by using (4.6); then calculate the corresponding fitness value of this chromosome from (4.4).

4.3.4 Genetic Operators

Based on the fitness function defined above, three basic types of genetic operators are required to modify the population: selection, crossover, and mutation. Selection is a process used for choosing parent chromosomes to participate in reproduction for the next generation, and among the many selection schemes available, the roulette wheel sampling scheme [5] is used.

Crossover is a crucial operator that combines two or more parent chromosomes to produce new offspring chromosomes. A suitably designed crossover can significantly accelerate the search process. The proposed GA allocation approach adopts a combination of four types of crossover operators, simple one-point crossover, arithmetic crossover, heuristic crossover and *UB*-swap crossover, which can lead to enhanced performance. The first three types of crossover work for any problem formulation [4], [5] and have been discussed in Chapter 3. However, *UB*-swap crossover is proposed in this study and only works for the *UBs* allocation purpose. Let $\boldsymbol{h}_p$ and $\boldsymbol{h}_q$ be two selected parent chromosomes. It firstly finds the most allocated coverage pattern in each chromosome, i.e. η_1 and η_2. Then *UB*-swap crossover creates two offspring chromosomes $\tilde{h}_p$ and $\tilde{h}_q$ by randomly selecting a crossover point δ and applying the two most allocated patterns in either segment as follows

$$\tilde{h}_q = \begin{cases} \tilde{h}_q^i = \eta_2, & \text{if } i < \delta \\ \tilde{h}_q^i = \eta_1, & \text{otherwise} \end{cases} \quad \tilde{h}_p = \begin{cases} \tilde{h}_p^i = \eta_1, & \text{if } i < \delta \\ \tilde{h}_p^i = \eta_2, & \text{otherwise} \end{cases} \tag{4.7}$$

As for the mutation operator, we consider multi-non-uniform mutation [34] for the proposed scheme which is the same as the technique that adopted in Chapter 3.

4.3.5 Replacement

The so-called incremental replacement and the elitist strategy are adopted in this study [5], [35].

4.3.6 Termination

Termination is a criterion by which the GA decides whether to continue searching or to stop the search. Since in the studied case, the number of iterations required to reach a predefined energy function is not known in advance, a combined termination strategy is adopted in this study. We consider that GA is exhausted in searching if it cannot find a better solution in 10 successive generations; therefore, it is combined with the predefined maximum generation G_{max} as the proposed GA approach's termination criterion, which means the GA will process to the next generation if $G_{cur} < G_{max}$ and a better solution has been found within 10 (empirically determined) previous generations, otherwise it will terminate. This strategy can not only ensure that the GA has enough time to converge, but also avoid excessively high complexity and processing time.

4.4 Simulation Environment

The performance of the proposed GA-based allocation scheme is evaluated by using portable-initiated discrete event simulation experiments [38]. The simulated cell cluster size B=16 and base station capacity m=50, giving a full cluster capacity of 800. A wraparound hexagonal topology is employed as shown in Figure 4.3; the white 16 cells are the actual simulated cells and the bordering shaded cells are used to create wraparound topology.

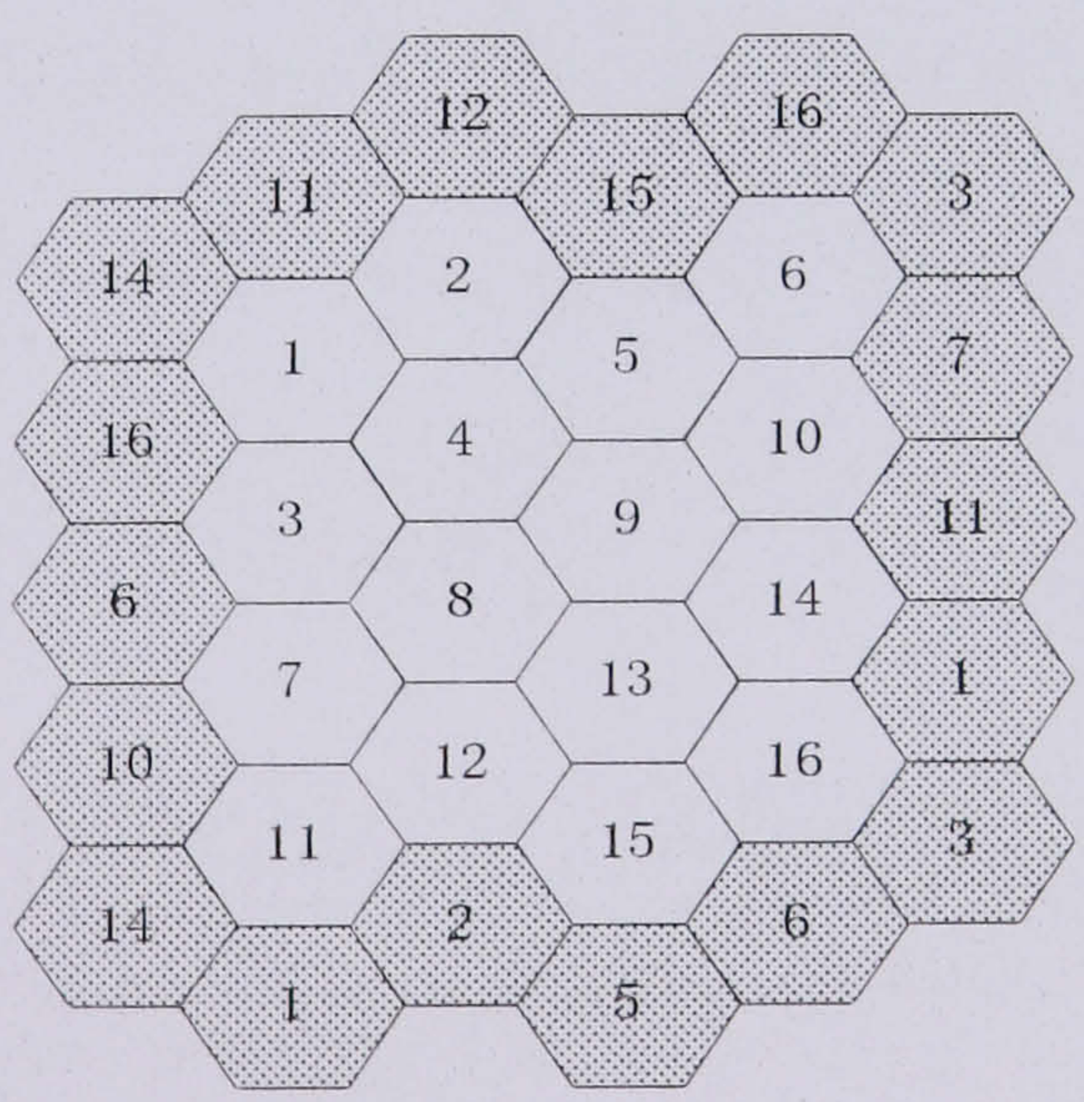


Figure 4.3: Wraparound topology of the simulated PCMA-based cellular network (with cell numbering).

This approach obviates the boundary effect that occurs in an unwrapped topology [36].

The mobility behaviour of mobiles in the simulation is described by a two-dimensional random walk [37]. In this model, a mobile user stays in the coverage area of a cell for a period of time (sojourn time) that has an exponential distribution with mean T_{in}. The mobile user then moves to one of the six neighbouring cells with the same routing probabilities of 1/6. Total number of mobile users in the system is varied from 400 to 800 in order to examine the system performance under different traffic load conditions.

4.5 Results and Discussions

In this section, computer simulation results are presented to evaluate the system performance of the proposed GA-based allocation scheme and alternative schemes. Various characteristics and parameters of the steady-state GA, which were determined by preliminary experiments and found to be robust and well suited for the studied cases, are given in Table 4.1.

Table 4.1: Summary of GA parameters used for the channel allocation simulations for PCMA system.

Parameter	Value/Type
Population size, Q	40, 60, 80
Generation	Combined stop criterion with G_{max}=100
Representation	Real-valued
Initialization	Hybrid (Random + Deterministic)
Generation selection	Roulette wheel
Crossover operators	Simple, Arithmetic, Heuristic, *UB*-Swap
Crossover probability, p_c	0.88
Mutation operator	Multi-non-uniform
Mutation probability, p_m	0.08
Replacement	Incremental + Elitism

Before presenting the P_o and θ of system capacity results, it is of interest to examine a case study, in which a subset of the cellular cluster is considered, to illustrate how the proposed GA approach is able to outperform alternative schemes. This case study has been considered in [45], where the authors proposed the '*min*' resource allocation algorithm to overcome the shortcoming of the original Greedy method. This study considers the same case to evaluate how the proposed GA-based scheme outperforms both the uniform and '*min*' algorithms. As in [45], we assume a subset system with a cluster consisting of B=4 cells (as shown in Figure 4.1 with numbered 4 cells) and each base station capacity is 10 connections. The uniform pattern, the traditional 1/3 reuse pattern, and each of the 6 stripe patterns provide throughputs of 0.4, 0.9 and 0.7, respectively. Consequently there are 10/0.4=25 *UBs* to be allocated in this case. According to [45], in using the original Greedy algorithm directly, it is found that

(35/2), (30/7), (45/14) *UBs* are allocated to pattern C_1, C_3 and C_5 respectively, which results in 2.75 unserviced users as shown in Figure 4.4(a) . However, if uniform pattern C_1 is used completely, there are only 2 unserviced users as shown in Figure 4.4(b). As a result, the Greedy algorithm was modified in [45] by comparing it with uniform allocation in each resulting user distribution in order to choose the better allocation. This was referred as the '*min*' algorithm. In our case we use the GA-based scheme to tackle this allocation problem. After a series of evolutional generations, GA will find an allocation vector $\boldsymbol{R}$. In this case, $\boldsymbol{R}$ has 25 (equal to the total *UBs* available) components with each corresponding to a pattern that one *UB* is allocated to. Specifically, there are 18, 3, 3, 1 *UBs* allocated to pattern C_1, C_3, C_5, C_7 respectively, which result in only 0.6 unserviced users as shown in Figure 4.4(c) It is clear that the GA-based scheme shows the best performance over the alternative schemes.

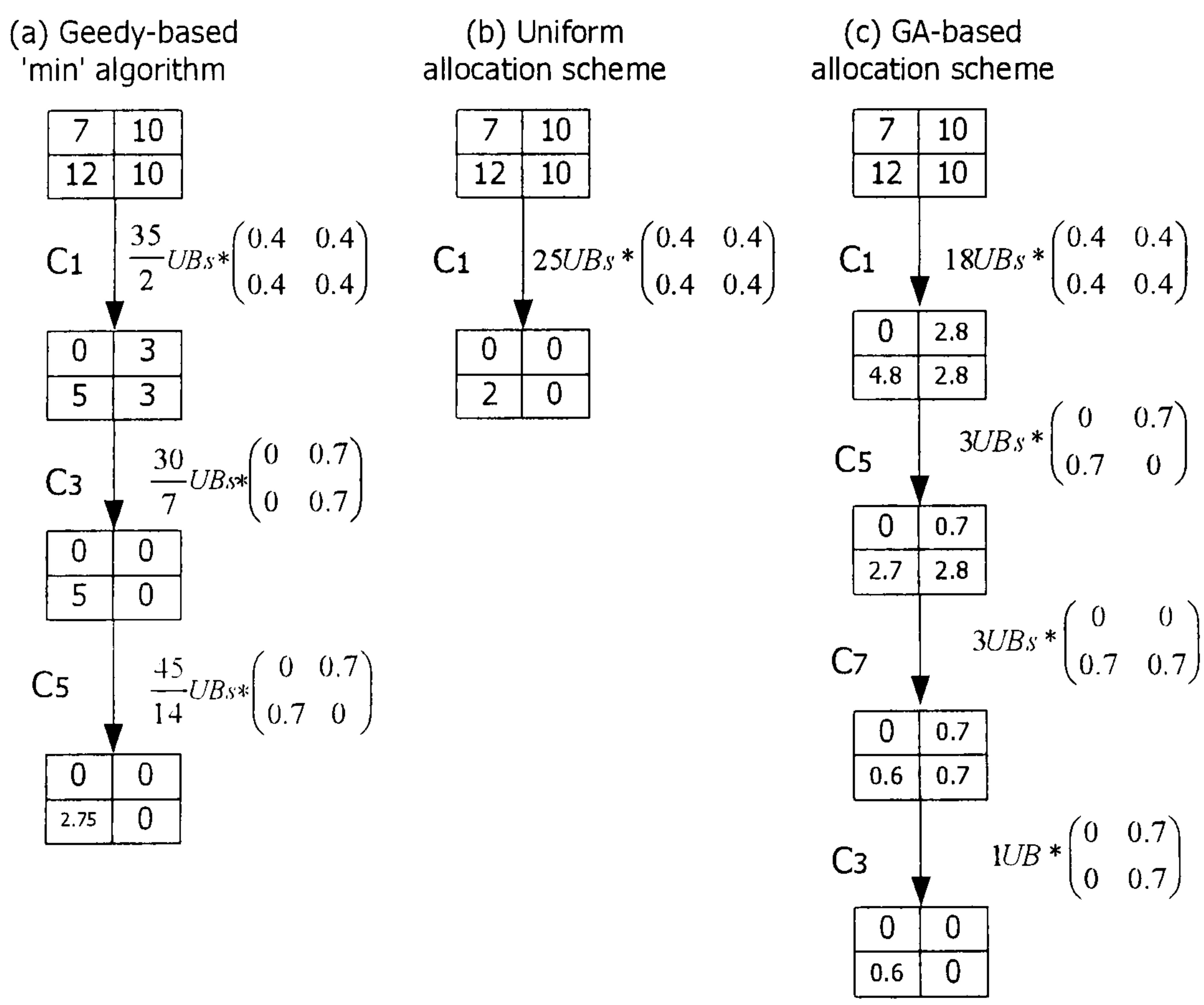


Figure 4.4: Comparisons of three allocation schemes for the subset case study in PCMA system.

Figure 4.5 demonstrates the evolution of energy function with respect to the studied subset case for the '*min*' algorithm and GA-based allocation schemes. For the GA-based approaches, we use a population size of Q=60 and the best chromosomes are used in energy evaluation. Note that we compare GA approaches in three different ways, namely, traditional GA, traditional GA with *UB*-Swap crossover and also with hybrid initialization. It is observed that all GA approaches were able to reach a lower energy as compared to the '*min*' algorithm. This is because the '*min*' algorithm allocates *UBs* using a greedy gradient approach, whereas the GA approaches maintain a multi-dimensional search for the solution. Therefore, it is not surprising to find that the GA methods have a slower convergence rate as compared to the '*min*' algorithm. However, it is also found that the GA approaches can speed up its convergence rate dramatically if using the problem-tailored crossover (*UB*-Swap crossover) and intelligent initialisation (hybrid of random and deterministic population).

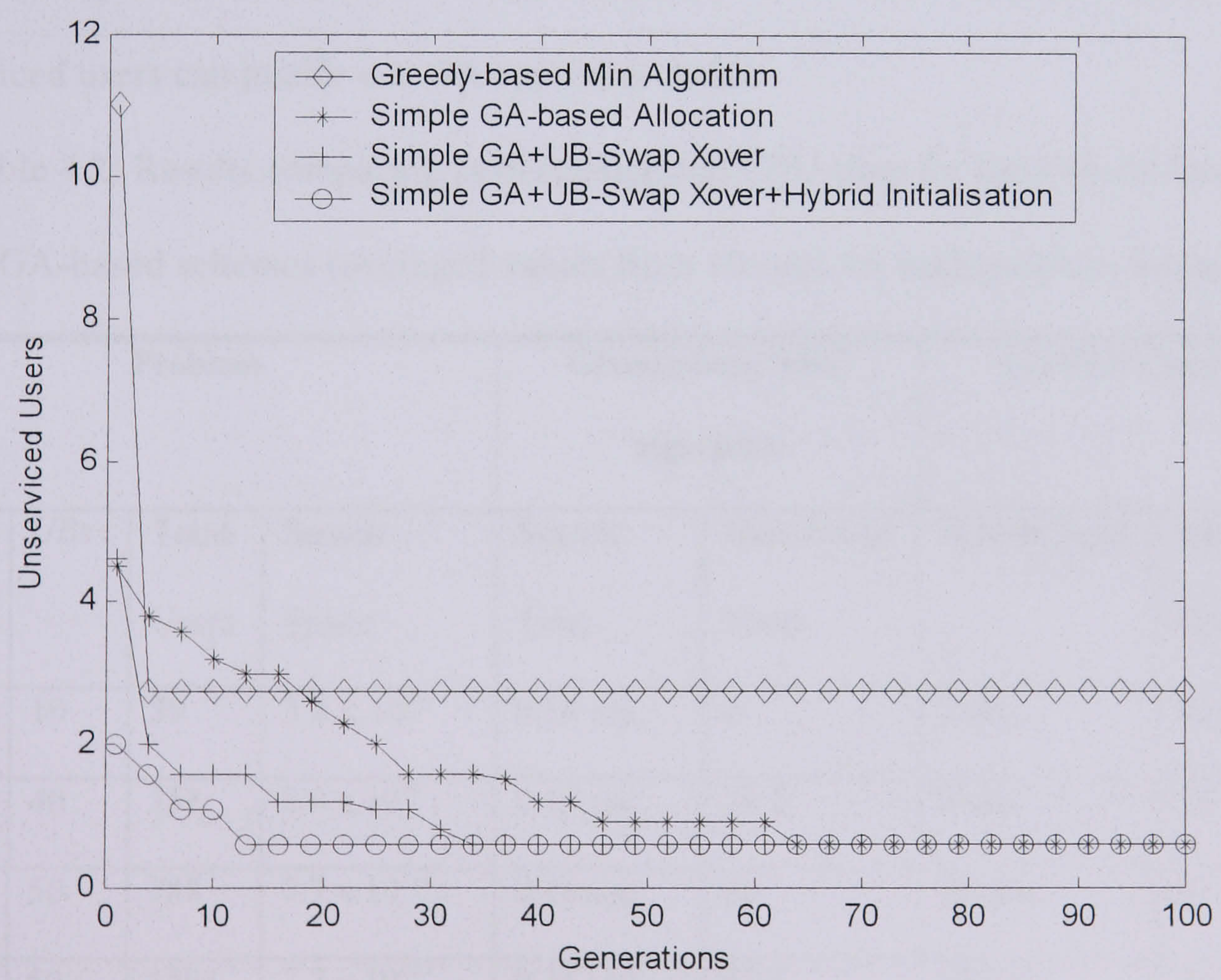


Figure 4.5: Evolution of energy functions (unserviced users) of the Greedy-based and GA-based allocation schemes with respect to the number of generations.

Table 4.2 gives the computation complexity comparison between the Greedy-based '*min*' algorithm and our proposed GA-based approach (Q=60). The users are distributed randomly in the comparative studied cases, and the GA approach has been implemented on a 1.2GHz Pentium III PC with 128MB SDRAM memory. For the proposed GA, the tables show the best results out of 10 runs for each problem instance and the average solution time over all 10 runs with the combined termination criterion as specified in the previous section. Clearly, GA-based approaches do get better solution than Greedy-based '*min*' algorithm at the expense of using more CPU-time. However, it should be noted that the search time only grows modestly, while the problem search size increases exponentially. Furthermore, we were primarily interested in finding high-quality solutions and as for the required CPU-time, we assume that the computation complexity can be accommodated by using much more powerful parallel computation implementation. Therefore, the improvement for the serviced users can justify our GA approach choice.

Table 4.2: Results comparing performance and CPU-time for the Greedy-based and GA-based schemes (averaged values from 10 runs for each problem instance).

Problem				**Greedy-base 'min' algorithm**		**GA-based approach**	
Cells	*UBs*	Total Users	Search Space	Search Time	Unserviced Users	Search Time	Unserviced Users
4	10	39	3.8×10^{22}	0.14 sec.	2	2 sec.	0.6
16	40	758	2.0×10^{90}	0.15 sec.	34.5	8 sec.	21
16	50	788	7.7×10^{112}	0.16 sec.	26	10 sec.	8
32	50	1594	7.7×10^{112}	0.16 sec.	65.5	10 sec.	34
32	60	1918	2.9×10^{135}	0.19 sec.	54	12 sec.	46

Results for the probability of overload P_o and the expected duration of overload θ are illustrated in Figs. 4.6 and 4.7, respectively. It is clear from the observation that the GA-based allocation scheme outperforms both uniform resource allocation and the Greedy-based '*min*' algorithm for both metrics. Furthermore, the performance of GA-based scheme improves as the population size Q increases, since the search space becomes larger. However, as Q is set to too large, excessive amount of processing time is required for each generation and the capacity improvement becomes unjustified. Hence, for the system under consideration, population size of 60 provides a good compromise between performance and complexity.

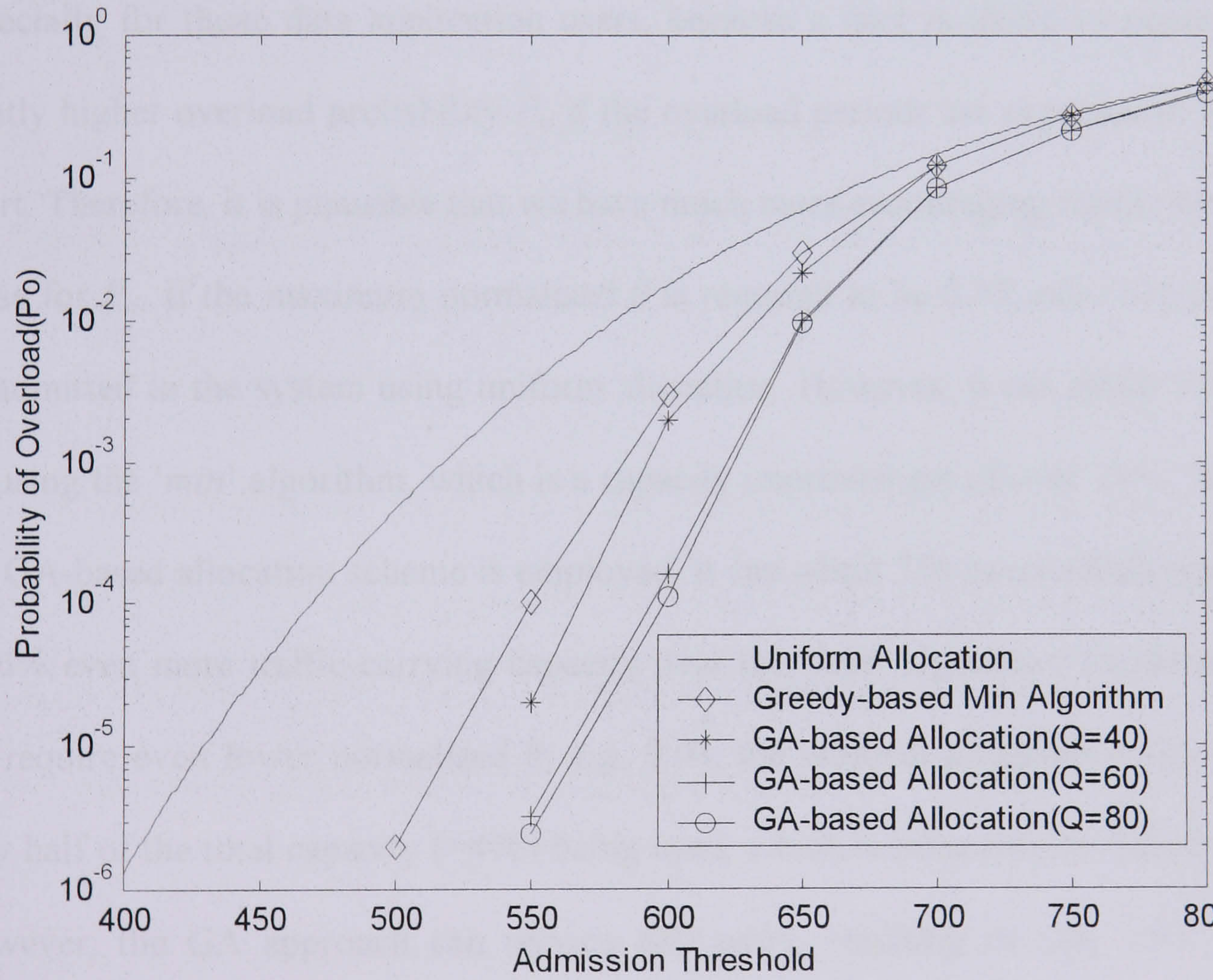


Figure 4.6: Probability of overload vs. admission threshold.

From Figure 4.6, we can see that in order to achieve P_o below 0.001 under uniform allocation, only 510 users can be admitted to the cell cluster. Therefore, approximately 37% of system capacity can not be fully utilised considering that the total cluster capacity is 800. On the other hand, the Greedy based '*min*' algorithm and GA-based scheme can permit 580 and 620 users, respectively. Hence, for the same overload probability of 0.001, the system admission threshold is increased by roughly 22% over uniform allocation, and 7% over the '*min*' algorithm respectively with the use of GA-based allocation scheme.

In Figure 4.7, we use normalization of θ for comparison, which is the ratio of θ to the mean time of handoff events T_{in}. This normalized value remains constant regardless of handoff rate as seen from (4.2). It should be noted that the expected duration of overload is more important than the probability of overload for the system users, especially for those data application users, because a user is likely to experience a lightly higher overload probability P_o if the overload periods are expected to be quite short. Therefore, it is plausible that we have much more encouraging results for θ than those for P_o. If the maximum normalised θ is required to be 0.05, only 510 uses can be admitted to the system using uniform allocation. However, it can admit 670 users by using the '*min*' algorithm, which is a capacity improvement of over 31%. While, if the GA-based allocation scheme is employed, it can admit 750 users which represents a 16% even more traffic-carrying capacity over the '*min*' algorithm. Furthermore, if we require even lower normalised θ, e.g. 0.03, the uniform allocation will result in only half of the total capacity (=400) being used, which is infeasible for practical use. However, the GA approach can service 680 users, resulting in only 15% unused cluster capacity.

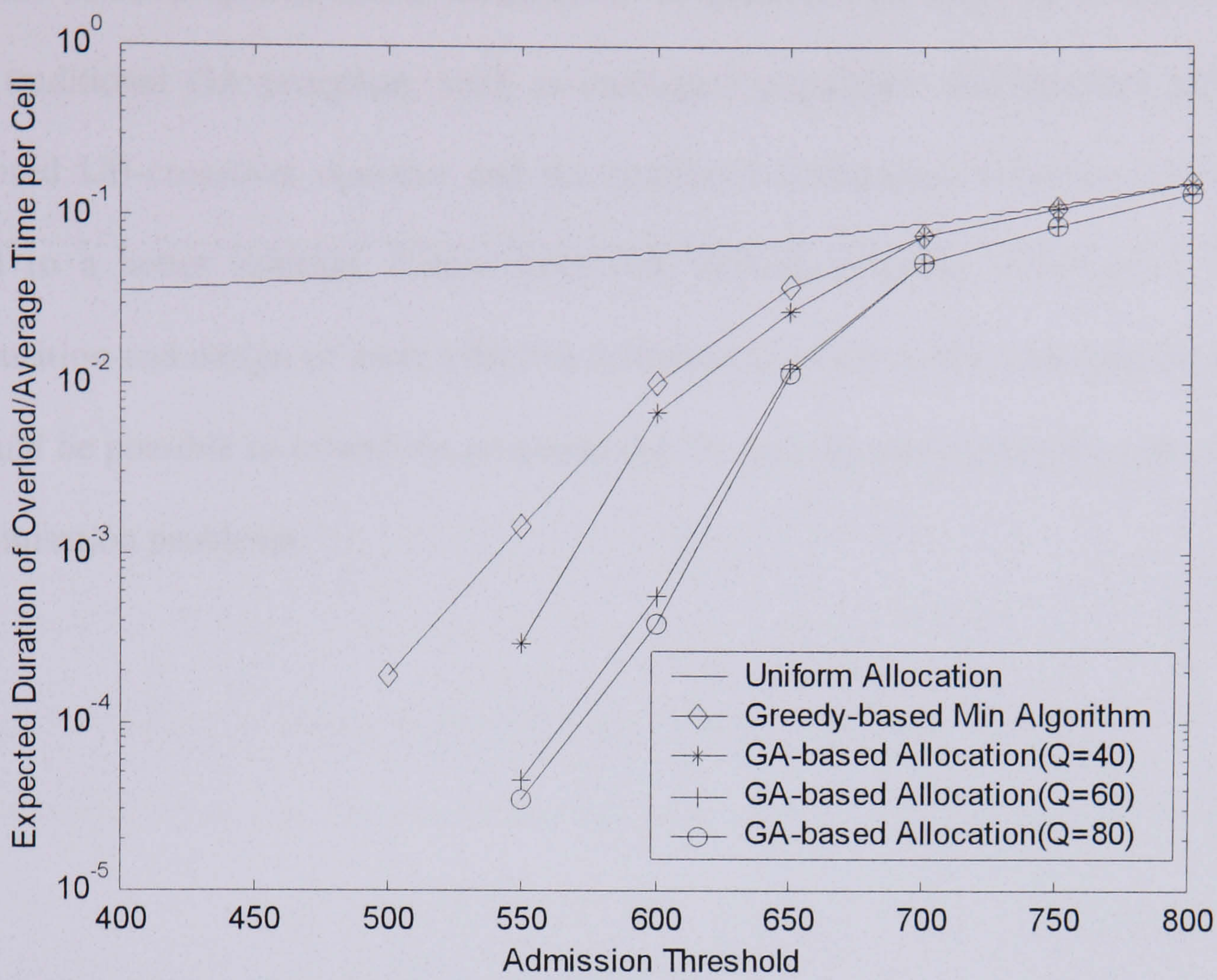


Figure 4.7: Normalised expected duration of overload vs. admission threshold.

4.6 Concluding Remarks

In this chapter, a new GA-based allocation scheme for PCMA systems is proposed. The resource allocation task has been proved to be in the class of NP-hard problem. Based on the users' traffic load at a specific time period, the proposed GA approach carries out a multi-point search by manipulating and maintaining a population of candidate solutions to find the number of *UBs* that need to be allocated in each cell. Simulation results demonstrate that the GA-based allocation scheme outperforms both uniform resource allocation and the recently proposed Greedy-based '*min*' algorithm in terms of serviced number of users. As a result, the system capacity utilisation can be improved and the PCMA strength be exploited. Also of importance is the fact that the research is taken under packet-switching architecture, which is the main backbone for future wireless applications.

In case of the proposed GA, it turned out to be essential that adopting modification to the traditional GA processes, such as intelligent population initialisation, problem-tailored UB-crossover operator and the combined termination criterion, which will lead to a better solution. Future work will include heuristic initialisation of the population and design of more effective operators to speed up the convergence rate. It should be possible to extend the proposed GA for solving more general combinatorial optimisation problems.

5 Application to Multiuser Detection in CDMA Systems

5.1 Introduction

To meet the ever-growing demand for wireless services, multiuser communication techniques, which make efficient use of precious resources are of utmost importance for system design. Direct-Sequence Code-Division Multiple Access (DS-CDMA) techniques are among the most promising candidates to fulfil this requirement. However, due to non-orthogonal signalling, multiple access interference (MAI) created by other users limits the capacity of CDMA systems. Multiuser detection [48] presents effective signal processing solutions for combating MAI and eliminating the near-far problem.

Despite considerable success in multiuser detection research over the last two decades, this work has been largely dominated by widespread use of the additive Gaussian noise assumption. However, it is well known that in many practical communication channels such as indoor and urban radio channels, the ambient noise is largely impulsive in nature, which can have an adverse effect on the performance of conventional receivers optimised for Gaussian noise [49], [50]. Therefore, practical implementation of multiuser detectors necessitates more accurate modelling of the ambient noise statistics along with the development of robust detection techniques [51].

Various multiuser detectors based on different perspectives and methodologies have been proposed in the literature. In this chapter, we propose a robust multiuser detector based on GA that can perform reliably under both Gaussian and non-Gaussian noise. Several GA-based multiuser detectors based on Gaussian noise models have also been reported. For example, Juntti et al. [52] conducted computer simulation of a GA for suboptimal multiuser detection. They showed that a good initial guess vector such as the output of the linear decorrelating detector (LDD) yields performance close to that of the single user system. Wang et al. [53] proposed a GA-based multiuser detector based on the maximum-likelihood decision rule combined with a modified Viterbi algorithm for sequential symbol detection. Ergun and Hacioglu [54] investigated a class of hybrid detectors that combine a GA and a multistage detector. Abedi and Tafazolli [55], [56] studied GA implementation of the optimum detector, and Yen and Hanzo [57] considered the problems of joint estimation of transmitted symbols and fading channel coefficients using GAs.

The GA-based detector proposed in this chapter is robust in a *minimax* sense as it utilises an objective function typical of the Huber's *M*-estimator [58]. The basic concept is to utilise a non-least-squares type penalty function, which attempts to assign a lesser weight to a small portion of outlying residuals that deviate considerably from the normal fit to avoid a catastrophic influence on the estimate. The proposed GA-based detector performs joint symbol detection and adaptive estimation of the cut-off parameter for the *M*-estimator's objective function by searching through a space of potential solutions using a GA optimisation strategy. It should be noted that the proposed GA-based detector implicitly implements a robust version of the LDD [59] based on the Huber's M-estimator proposed in [60], or its efficient recurrent neural network (RNN) solution [61]. In the sequel, we refer to the robust decorrelating detector as *M*-decorrelating detector (MDD). The MDD studied in [60], [61] uses a fixed cut-off parameter for the M-estimator's objective function, which is determined by a rough estimate of the noise scale. Clearly, this requires a separate channel estimator, and as will be seen later, fixing the cut-off parameter also imposes a constraint on the attainable performance. We will demonstrate how these problems can be overcome altogether to achieve better performance using a GA approach.

The remainder of the chapter is organised as follows. Section 5.2 describes the system model and briefly reviews both LDD and MDD. Details of the proposed GA-based multiuser detector are outlined in Section 5.3. Computer simulation results are presented in Section 5.4 to support the investigation. Finally, Section 5.5 concludes the chapter.

5.2 System Description

In the following, we consider a synchronous DS-CDMA system shared by K users. Each user is assigned a different spreading code of length N-chips. The modulation scheme is binary phase shift keying (BPSK) and demodulation is assumed to be coherent. The baseband output of the chip-matched filter corresponding to the jth chip of the ith symbol is given by

$$r_j(i) = \sum_{k=1}^{K} A_k b_k(i) s_j^k + n_j(i),\ j = 1, \cdots, N \tag{5.1}$$

or in vector form

$$\boldsymbol{r}(i) = \sum_{k=1}^{K} A_k b_k(i) \boldsymbol{s}_k + \boldsymbol{n}(i) = \boldsymbol{SA}\boldsymbol{b}(i) + \boldsymbol{n}(i) \tag{5.2}$$

where for the kth user, A_k, $b_k(i) \in \{\pm 1\}$ and s_k denote, respectively, the received amplitude, ith information bit and the normalised signature code.

$S = [s_1 | \cdots | s_K] \in \Re^{N \times K}$ is the matrix whose columns are the normalised spreading codes of all users. $A = \text{diag}\{A_1, \cdots, A_K\}$ and $b(i) = [b_1(i), \cdots, b_K(i)]^T$. n(i) contains the sequence of i.i.d. noise samples with a common univariate probability distribution function $f_n(\cdot)$ of zero-mean. In this study, we model the noise as symmetric α-stable random processes [62]. Due to synchronous assumption, the symbol index i can be dropped and we define $x_k = A_k b_k$, and Equation (5.1) can then be written as

$$r_j = \sum_{k=1}^{K} s_j^k x_k + n_j, \quad j = 1, \ldots, N \tag{5.3}$$

and Equation (5.2) can be expressed as a linear regression model

$$\boldsymbol{r} = \boldsymbol{S}\boldsymbol{x} + \boldsymbol{n} \tag{5.4}$$

where $x = [x_1, \cdots, x_K]^T = \boldsymbol{A}\boldsymbol{b}$.

5.2.1 Linear Decorrelating Detector

The conventional receiver consists of a matched filter (MF) bank with each filter matched to the signature waveform of a specific user followed by a threshold device to produce the symbol estimate

$$\hat{\boldsymbol{b}}_{\text{MF}} = \text{sign}(\boldsymbol{S}^T \boldsymbol{r}) = \text{sign}(\boldsymbol{S}^T \boldsymbol{S}\boldsymbol{x} + \boldsymbol{S}^T \boldsymbol{n}) \tag{5.5}$$

The normalised cross correlation matrix for all signature codes is given by $\boldsymbol{R} = \boldsymbol{S}^T \boldsymbol{S}$. The LDD [59] applies the inverse $\boldsymbol{R}^{-1}$ to the output of the MF bank

$$\hat{\boldsymbol{b}}_{\text{LDD}} = \text{sign}(\boldsymbol{R}^{-1} \boldsymbol{S}^T \boldsymbol{r}) = \text{sign}(\boldsymbol{x} + \boldsymbol{R}^{-1} \boldsymbol{S}^T \boldsymbol{n}) \tag{5.6}$$

It is seen from Equation (5.6) that the LDD does not depend on the signal powers of the interfering users and hence is near-far resistant.

5.2.2 Robust *M*-decorrelating Detector Using Recurrent Neural Network

Let $d = [d_1, \cdots, d_N]^T$ denote the residual vector where

$$d_j = r_j - \sum_{k=1}^{K} s_j^k x_k, \quad j = 1, \ldots, N \tag{5.7}$$

The *M*-estimate [58] for the vector $\boldsymbol{x}$ corresponds to minimising the following energy function

$$E(x) = \sum_{j=1}^{N} \rho\left(r_j - \sum_{k=1}^{K} s_j^k x_k \right) \tag{5.8}$$

where $\rho(\cdot)$ represents a specific penalty function that is generally convex [58]. Hence,

$$\hat{\boldsymbol{x}} = \arg \min_{x \in \Re^K} \sum_{j=1}^{N} \rho\left(r_j - \sum_{k=1}^{K} s_j^k x_k \right) \tag{5.9}$$

To make the minimisation problem meaningful $\rho(\cdot)$ should be non-negative with its minimum at zero. The resulting inferential technique is referred to as the *M*-estimator, where *M* refers to the maximum likelihood (ML) type estimator, where the choice $\rho_{\mathrm{ML}}(d_j) = -\log f_n(d_j)$ gives the ordinary ML estimate of $\boldsymbol{x}$. For example, the LDD uses the least squares (LS) penalty function, i.e., $\rho_{\mathrm{LS}}(d_j) = (d_j)^2 / 2$ which is optimal when the residuals d_j follow a Gaussian density. In the absence of this assumption the LS estimate is often vulnerable to outliers since each residual enters quadratically.

Huber aimed to achieve robustness in a *minmax* sense and considered the following penalty function that increases less rapidly than LS [58]:

$$\rho_H(d_j, h) = \begin{cases} \frac{1}{2}(d_j)^2, & \text{for } |d_j| \le h \\ h|d_j| - \frac{1}{2}h^2, & \text{for } |d_j| > h \end{cases} \tag{5.10}$$

with derivative

$$\psi_H(d_j, h) = \begin{cases} d_j, & \text{for } |d_j| \le h \\ h\,\text{sign}(d_j), & \text{for } |d_j| > h \end{cases} \tag{5.11}$$

where $h>0$ is a problem-dependent parameter for tuning the robustness of the estimator. The basic principle of Equation (5.10) is to use LS and penalise outliers with L_1 norm. The RNN-based MDD in [61] uses the logistic function as a smooth approximation to Equation (5.10). By applying the gradient method it is easy to show that the solution to Equation (5.9) is equivalent to solving the following equation [61]

$$\boldsymbol{S}^T \psi(\boldsymbol{r} - \boldsymbol{S}\boldsymbol{x}) = \boldsymbol{0}_K \tag{5.12}$$

where $\psi(\cdot) = \rho'(\cdot)$ and $\boldsymbol{0}_K$ denotes a K-vector zero. With the exception of the most elementary forms of ψ-function such as the LS, Equation (5.12) constitutes a set of nonlinear equations, and hence, iterative methods are required. Several iterative schemes which have been developed in statistics literature are summarised and compared in [63]-[66]. Basically they can be broadly categorised into three families: Newton's method, iteratively reweighted least squares, and the Huber's method as was used in [60]. In [61] it was shown that a RNN proposed in [67] could be used to implement the MDD without the need of the matrix inverse $[\boldsymbol{S}^T\boldsymbol{S}]^{-1}$ with a consequent saving in computational complexity, especially for a highly dynamic CDMA network. Mathematically, the RNN-based MDD iteratively estimates x as follows:

$$\hat{x}^{n+1} = \hat{x}^n + \mu S^T \psi(r - S\hat{x}^n) \tag{5.13}$$

where n is the iteration index and μ is the step size that controls the speed of convergence. The final symbol estimates are taken as the signs of the converged $\hat{x}$ in Equation (5.13). Unlike most of the multiuser detectors that process the matched filter bank output $S^T r$, the RNN-based MDD in Equation (5.13) and the GA-based detector proposed in this study take r, which is obtained from a chip-matched filter, as their observation vector.

5.3 GA-based Robust *M*-decorrelating Detector

Our investigation of a GA approach to tackle the robust detection problem formulated in Equation (5.9) was motivated by several factors. Conventional directed or gradient search methods such as the three existing algorithms for the *M*-estimator [63]-[66] and the RNN solution in Equation (5.13) are examples of strategies, which exploit the best solution for a given cut-off parameter h of the *M*-estimator's objective function. They are governed by differential/difference equations that use the iterative improvement technique. The technique is applied to a single point (the current point n) in a limited search space determined by h. During a single iteration, a new point is selected from the neighbourhood of the current point. Obviously, they are local search techniques in the sense that they provide local optimum values that are dependent on the selection of the cut-off parameter h. For example, the MDDs implemented in [60], [61] fix the cut-off parameter h to a value that is proportional to the noise variance/dispersion and only a single point in the search space is processed. Clearly, this requires a separate channel estimator and the fixed cut-off parameter h imposes a constraint on the attainable performance, since it does not explore the search space for other possible values of h.

On the other hand, GA can perform multi-directional search by manipulating and maintaining a population of potential solutions for different values of h and encouraging information formation and exchange between these directions. At each generation, the population undergoes a simulated evolution, in which better chromosomes reproduce, while inferior ones perish. Because of this, GA can strike a remarkable balance between exploration and exploitation of the search space by combining elements of directed and stochastic search, and subsequently lead to better solutions [34]. More remarkably, in addition to searching for the estimate of $\boldsymbol{x}$, the proposed GA-based multiuser detector also treats h as its optimisation parameter, and thus, eliminates the need of the channel estimator.

In the following subsections, we outline the development of the GA based MDD based on a real-coded GA, whose structure is depicted in Figure 5.1.

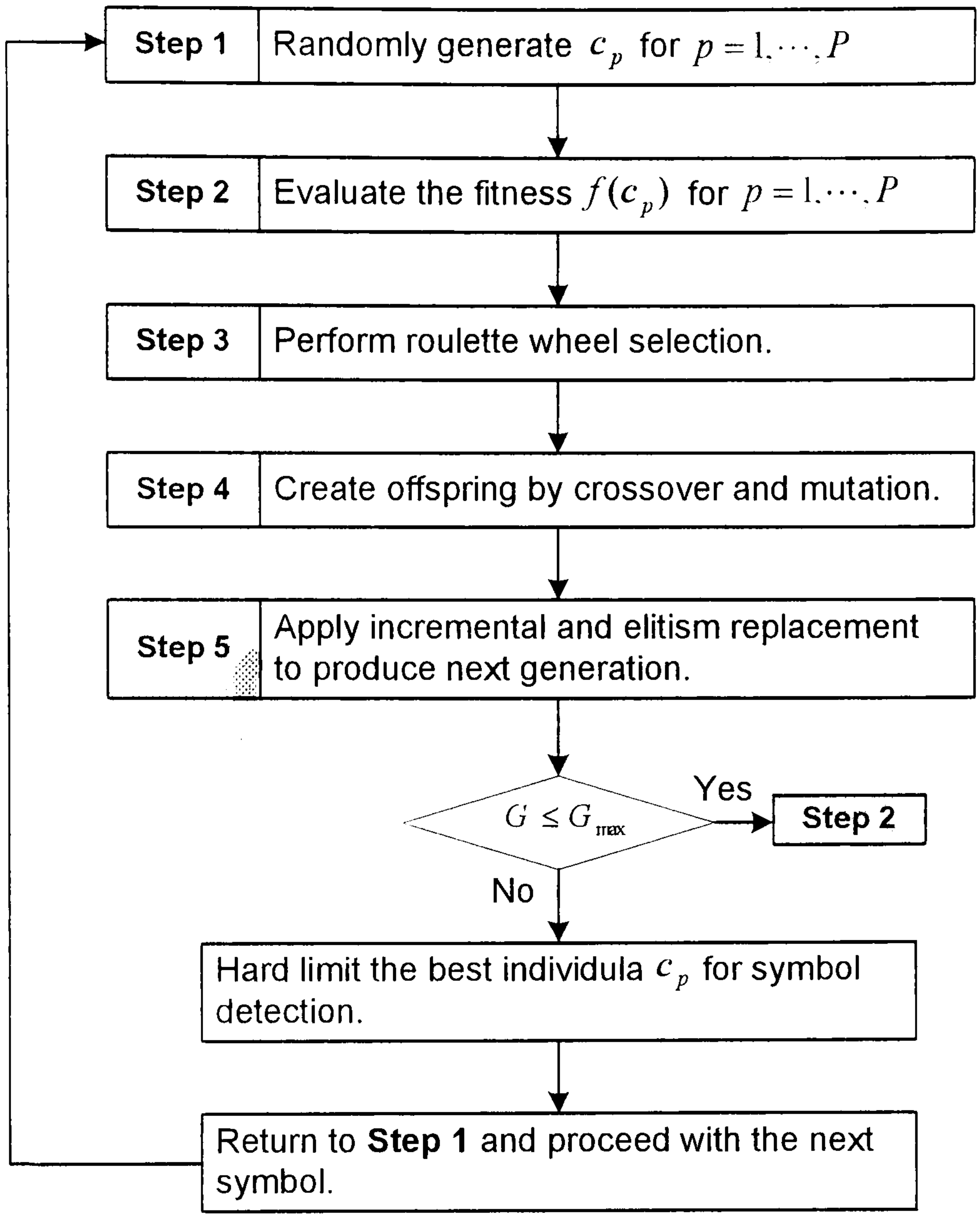


Figure 5.1: Flowchart depicting the structure of the proposed GA-based robust multiuser detector.

5.3.1 Chromosome Representation

The encoding scheme of chromosomes may vary according to the nature of the problem and has a major impact on the performance because it can severely limit the search space observed by the system. The most commonly used binary chromosomes are sometimes difficult and unnatural when applied to multidimensional, high precision optimisation problems and the need for richer data structures has been recognised for some time [68], [69]. Alternatively, real-coded chromosomes [70]-[71], which have received considerable interest produce better solutions in certain engineering problems [72]-[74]. The real-valued representation is capable of representing variables over continuous domains with higher precision and efficiency [34]. For the GA-based MDD where the user signal components in $\boldsymbol{x}$ and the cut-off parameter h are variables in real space, a real-valued encoding scheme is used in order to move the representation closer to the problem domain. Each chromosome is represented by a $(K+1)$-dimensional real-valued vector $c = [c^1, \cdots, c^{K+1}]$, where the first K components correspond to the signal components of all users $\{\hat{x}_1, \cdots, \hat{x}_K\}$, and the $(K+1)$th component is reserved for the estimator cut-off parameter h.

5.3.2 Initialisation

The population of real-coded chromosomes $\{c_p = [c_p^1, \cdots, c_p^{K+1}],\ p = 1, \cdots, P\}$ is initialised randomly, where P is known as the population size. The lower and upper bound for each variable $c_p^k \forall p$ in the chromosome is denoted by a_k and b_k, respectively. The purpose of using random generation is to distribute the initial trial solutions to a highly diversified search space (exploration).

5.3.3 Fitness Evaluation

The objective function in Equation (5.8) provides the mechanism for evaluating the fitness of each chromosome. By convention, the fitness function should be a positive value; otherwise we can use a scaling mechanism [5], [34]. We assume that the GA-based detector employs the Huber penalty function $\rho_{\mathrm{H}}(\cdot)$. Since $\rho_{\mathrm{H}}(\cdot)$ is non-negative, the fitness function (which is to be minimised) of each chromosome $f(c_p)$ is evaluated directly by substituting its elements into Equation (5. 8):

$$f(c_p)=\begin{cases} \sum_{j=1}^{N}\frac{1}{2}\left(r_j-\sum_{k=1}^{K}s_j^k c_p^k\right)^2, & \text{for } |d_j|\le c_p^{K+1} \\ \sum_{j=1}^{N}c_p^{K+1}\left|r_j-\sum_{k=1}^{K}s_j^k c_p^k\right|-\frac{1}{2}(c_p^{K+1})^2, & \text{for } |d_j|> c_p^{K+1} \end{cases} \tag{5.14}$$

Therefore, the GA only requires the objective function associated with each chromosome for implementation. By contrast, the aforementioned conventional gradient techniques need the derivatives of $\rho(\cdot)$ (calculated analytically or numerically) in order to be able to climb the next point, which could become infeasible if the derivatives do not exist.

5.3.4 Genetic Operators

Based on the fitness function defined above, three basic types of genetic operators are required to modify the population: selection, crossover, and mutation.

Selection is a process used for choosing parent chromosomes to participate in reproduction for the next generation. It models the survival-of-the-fittest mechanism observed in nature. The reproductive opportunity of an individual parent is normally granted in direct proportion to its fitness value so that highly fit chromosomes contribute more copies (with reselection permitted) to the mating pool than do those poor ones. Therefore, the best parents generate more copies, the average stay medium, and the worst die out. Among the many selection schemes available, we use the roulette wheel sampling [5].

Crossover is a crucial operator that combines two or more parent chromosomes to produce new offspring chromosomes. The key idea behind crossover is to generate new offspring by integrating good genetic material from highly fit parents. There are many ways to implement the crossover operation. In the proposed GA-based multiuser detector, we examine the use of three crossover operators: simple one-point crossover, arithmetic crossover, and heuristic crossover [34]. The details of each crossover technique have been discussed in Chapter 3.

As for the mutation operator, we consider multi-non-uniform mutation [34] for the proposed scheme, which is the same as the technique that adopted in Chapter 3.

5.3.5 Replacement

The so-called incremental replacement and the elitist strategy are adopted in this study [5], [35].

5.3.6 Termination

Termination is the criterion by which the GA decides whether to continue searching or stop the search. Typical termination criterion of the GA involves either satisfying a problem-specific success indicator or completing a specified number of generations to be run, G_{max}. Since in the proposed GA-based detector, the number of iterations required to reach a predefined cost function is not known in advance, we adopt the latter strategy to avoid excessively high complexity and detection delay. The final symbol estimate is taken as the sign of the best chromosome cp in the final generation, i.e.,

$$\hat{b}_k = \mathrm{sign}(c_p^k), \quad k = 1, \cdots, K \tag{5.15}$$

5.4 Results and Discussions

In this section, computer simulation results are presented to test the various aspects of the proposed GA structure for robust multiuser detector. A summary of the various

GA parameters used is given in Table 5.1.

Table 5.1: Summary of GA parameters used for the multiuser detection simulations.

Parameter	**Value/Type**
Population size, P	60, 80
Generation, G_{max}	50
Representation	Real-valued
Initialisation	Random
Generation selection	Roulette wheel
Crossover operators	Simple, Arithmetic, Heuristic
Crossover probability, p_c	0.88
Mutation operator	Multi-non-uniform
Mutation probability, p_m	0.08
Replacement	Incremental + Elitism

Throughout the simulations, a synchronous CDMA system with 10 equal-power users (K=10) is considered. Gold spreading codes of length 31 (N=31) are assigned to all users. For the RNN-based MDD, the convergence rate μ is chosen as 0.8 and the cut-off parameter h is set to h=3γ, where γ is the noise dispersion [64]. Due to the infinite-variance property of α-stable distributions leading to the inconsistency of standard signal-to-noise ratio (SNR), we use an alternative measure known as the Geometric SNR (GSNR) [75], which is defined as

$$GSNR = \frac{1}{2C_g}\left(\frac{A_k}{S_0}\right)^2 \qquad (5.16)$$

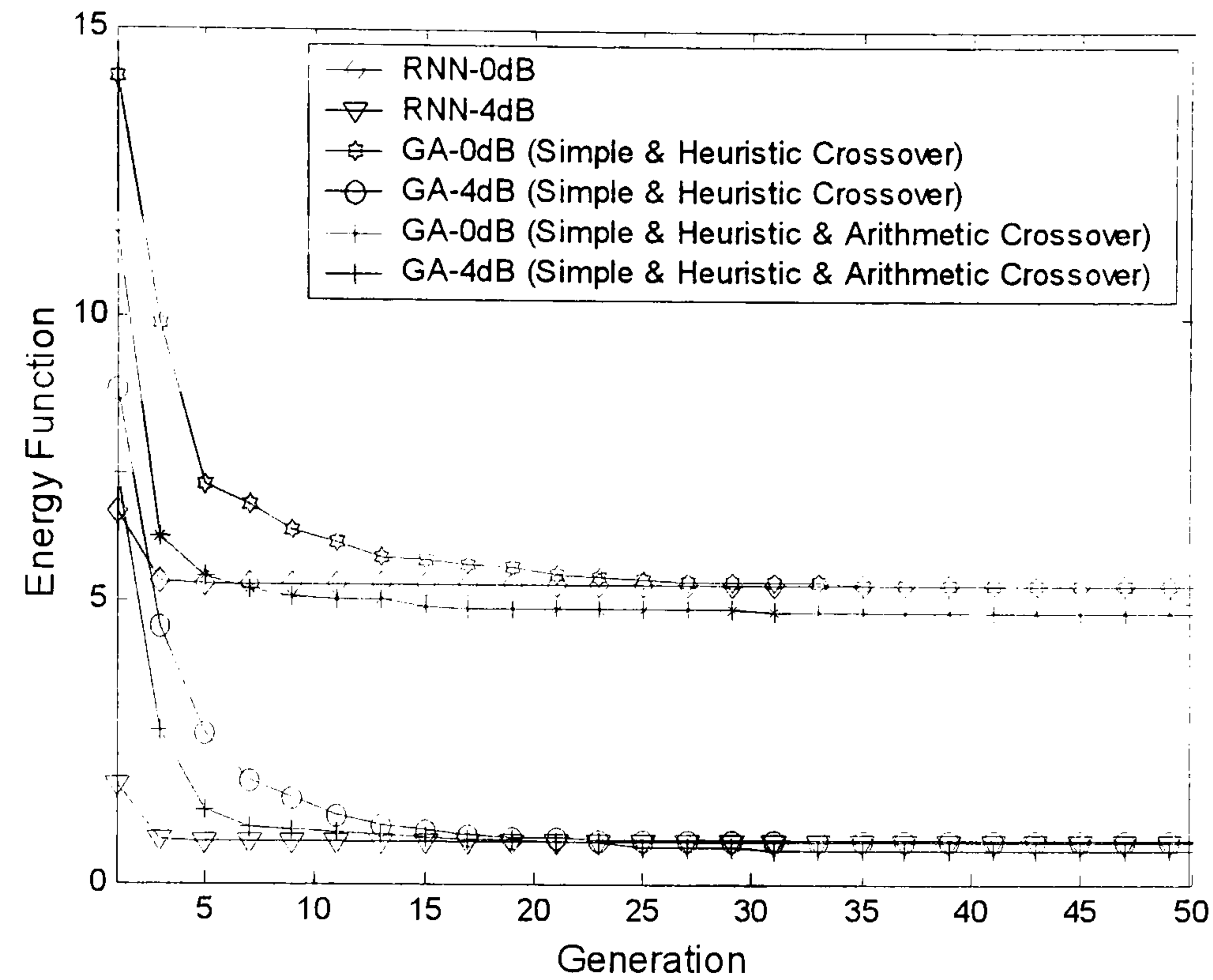
where A_k denotes the received amplitude kth user, and C_g is the exponential of the Euler constant. Before presenting the bit error rate (BER) results, it is of interest to examine the evolutionary behaviour of the GA detector in response to different noise scenarios.

5.4.1 Evolutionary Performance

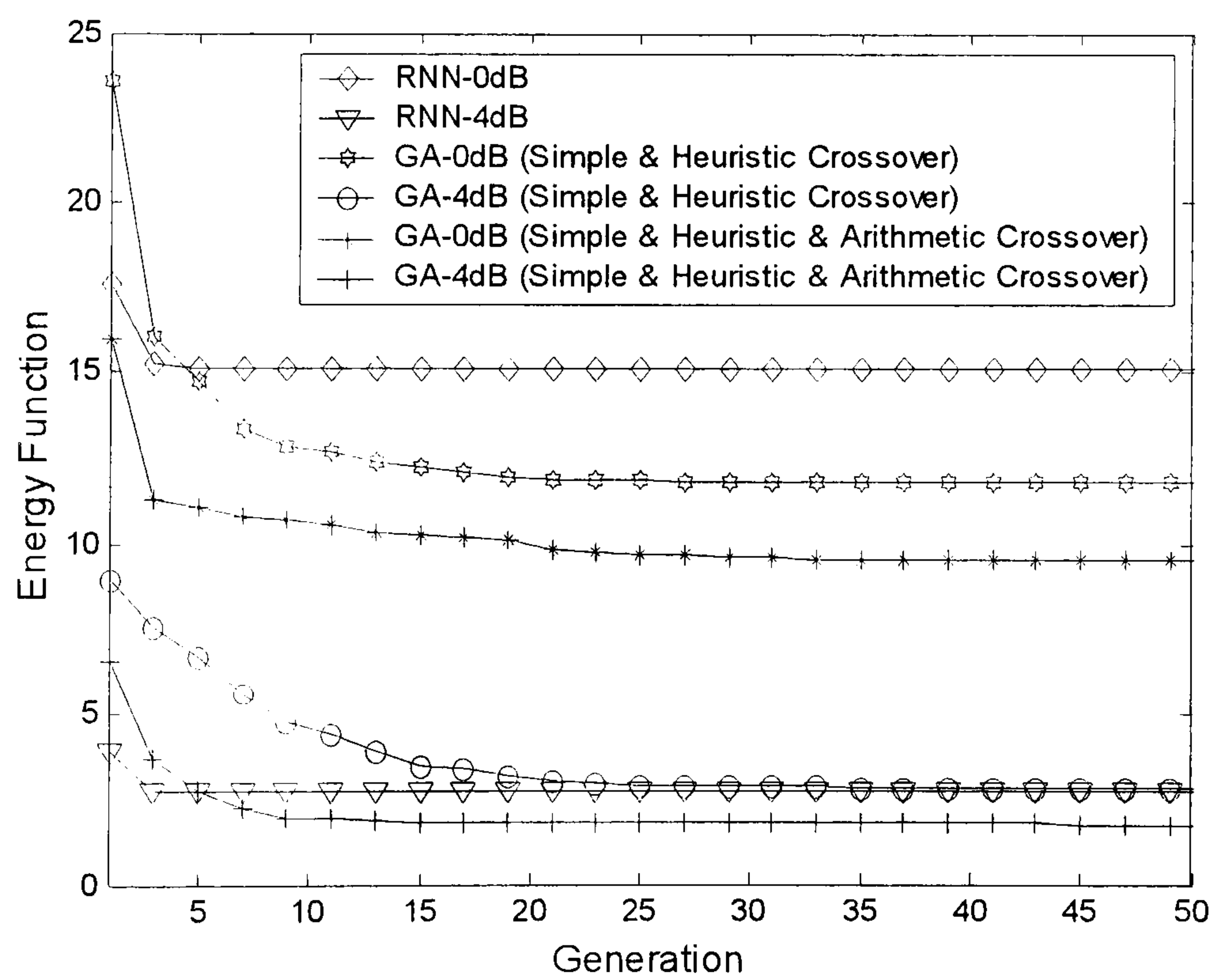
Figure 5.2 (a) and (b) demonstrate the evolution of energy function with respect to the number of generations for the RNN and GA-based MDDs in Gaussian (α=2) and impulsive noise (α=1.5), respectively. Note the ambient noise is modelled as the symmetric α-stable (SαS) random variables [75], which is best defined by its characteristic function

$$\varphi(t) = e^{-\gamma|t|^{\alpha}} \tag{5.17}$$

α is called the characteristic exponent. A small value of α corresponds to more impulsive behaviour and vice versa. Two different GSNRs, 0 and 4 dB, are considered. For the GA-based detectors, we use a population size of P=60 and the best chromosomes are used in energy evaluation. Note that we also compare GAs with and without the arithmetic crossover operator. It is observed that all GA detectors were able to reach a lower energy as compared to the RNN, particularly under hostile channel conditions, e.g. low GSNR and impulsive noise scenario. This is because the RNN implements a single point search using a gradient approach, whereas the GA detector evolves in multi-dimensional search while jointly adapting to find optimal values of the cut-off parameter h. Therefore it is not surprising to find that the GA methods have a slower convergence rate as compared to the RNN. Moreover, we also notice that the GA system without arithmetic crossover converges slower. In contrast, the system with arithmetic crossover converges faster and provides better solutions than those without arithmetic crossover.



(a)



(b)

Figure 5.2: Evolution of energy function of the RNN and GA-based detector with respect to the number of generations at two different GSNR (0 and 4 dB) in (a) Gaussian noise channel, (b) impulsive noise channel (α=1.5).

In order to examine the response of the GA-based detector to different noise environments, Figures 5.3-5.5 illustrate 3-dimensional plots between the cut-off parameter h, residual d_j (only the first chip d_1 is considered), and $\rho_H(d_j, h)$ given in Equation (5.10). The three crossover operations: simple, arithmetic, and heuristic crossover are used in combination. During the first generation (G=1), the (P=60) candidate solutions are randomly generated and distributed over the solution space as shown in Figure 5.3. Hence, it is highly explorative at the beginning.

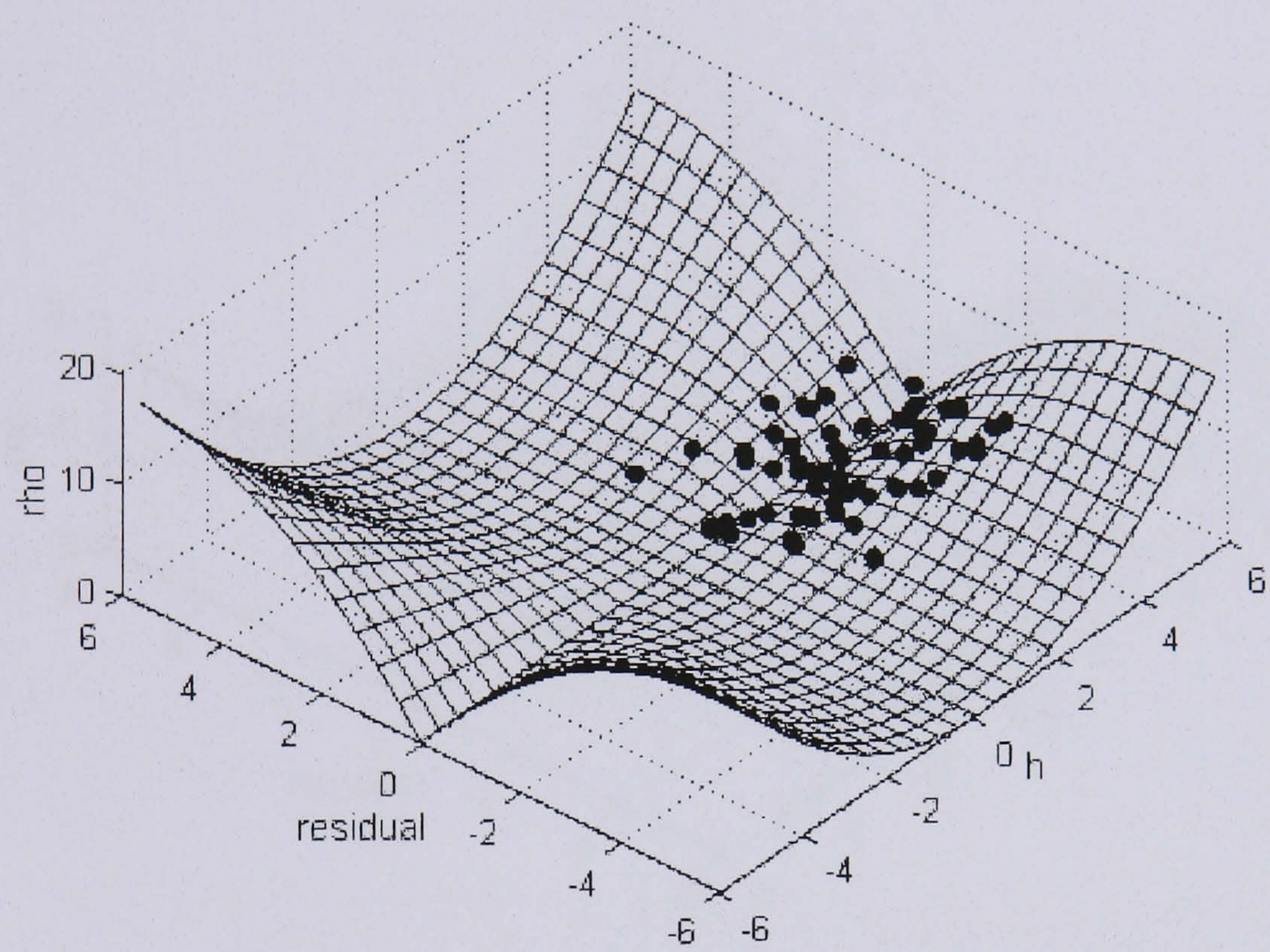


Figure 5.3: 3-dimensional plot of the relationship between the cut-off parameter h, residual d_j, and $\rho_H(d_j,h)$ during initialisation (G=1). Total population P=60.

By evolving this population of chromosomes over successive generations via probabilistic genetic operations, the GA detector quickly identifies and exploiting the subspaces in search for the best solution, while at the same time maintains the exploration of other parts of the solution space. This is illustrated in Figure 5.4 and Figure 5.5 after 50 generations in Gaussian and impulsive noise, respectively. The GSNR is 4dB. It is interesting to observe that when the noise is Gaussian, the cut-off parameter h is reasonably large (Figure 5.4). This is because the noise contains no impulsive components and ideally, h should approach infinity to yield the LS estimator (LDD), which is optimal in Gaussian noise. On the other hand, when the noise is impulsive (Figure 5.5), the parameter h has been reduced to a small value in an attempt to prevent the entry of impulsive noise. Therefore, the GA detector is able to adapt to statistical variations in noise distribution to achieve robust performance.

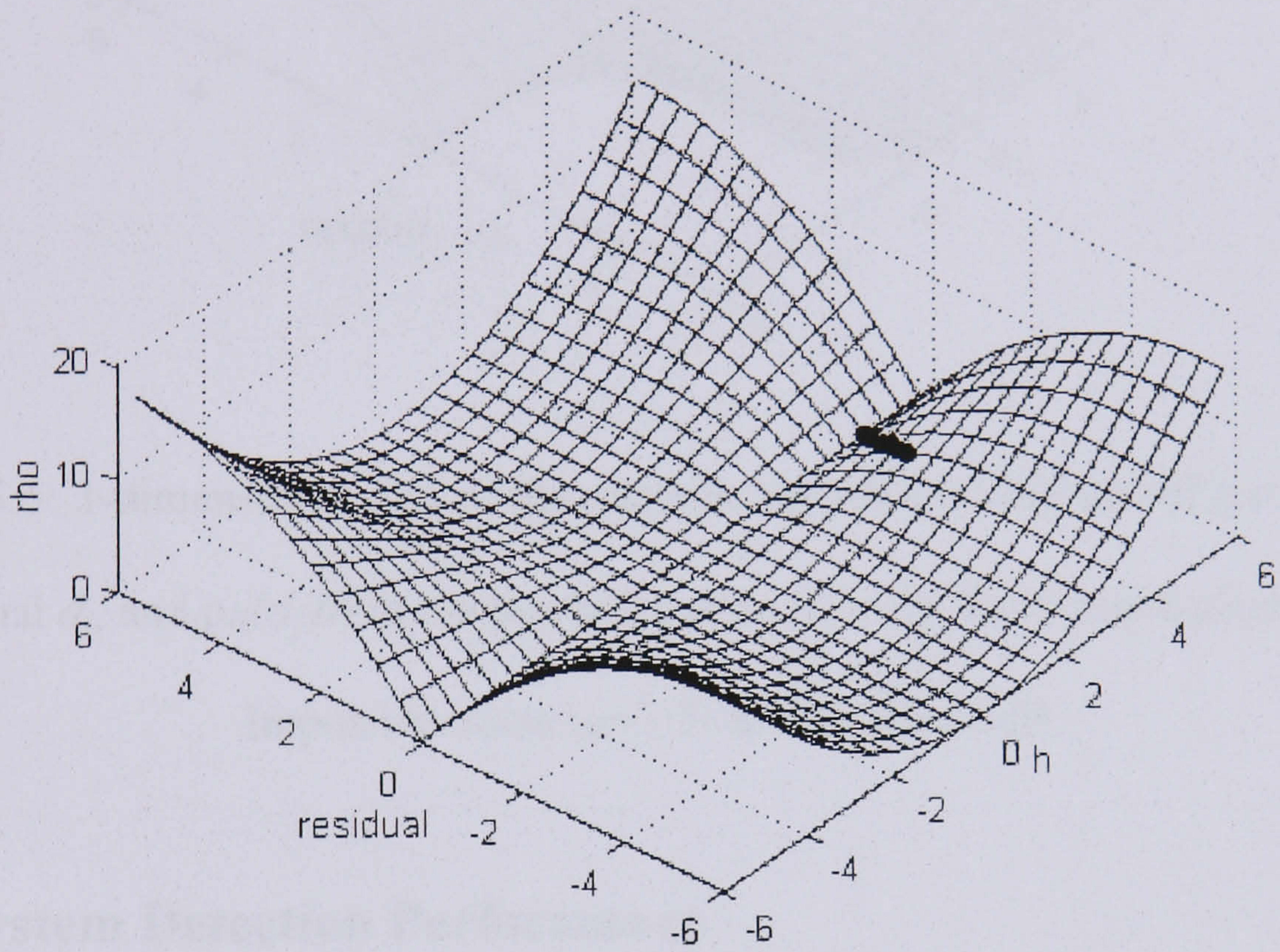


Figure 5.4: 3-dimensional plot of the relationship between the cut-off parameter h, residual d_j, and $\rho_H(d_j,h)$ in the last generation (G=50). Total population P=60, Gaussian noise at GSNR of 4dB.

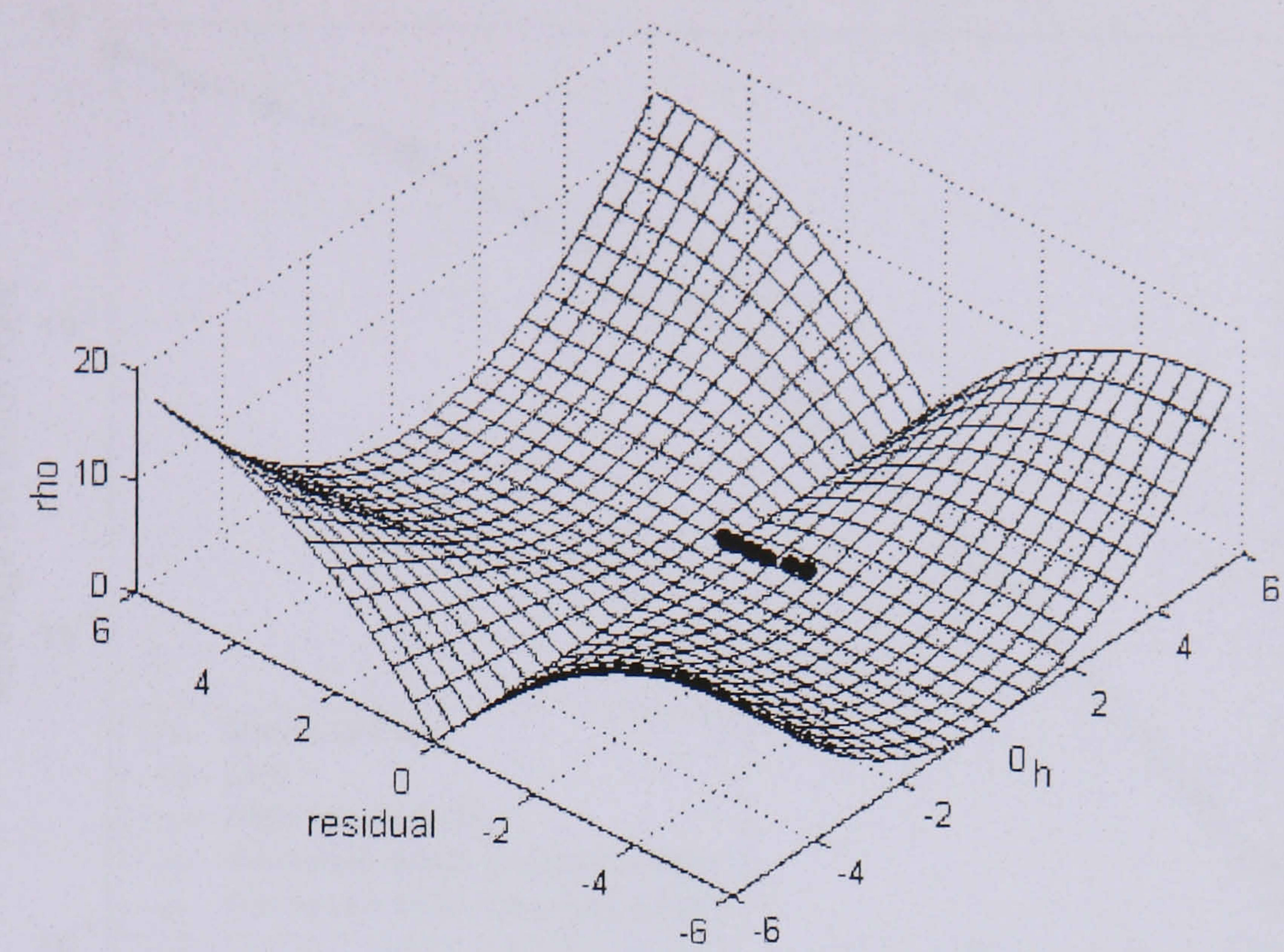


Figure 5.5: 3-dimensional plot of the relationship between the cut-off parameter h, residual d_j, and $\rho_H(d_j,h)$ in the last generation (G=50). Total population P=60, Impulsive noise (α=1.5) at GSNR of 4dB.

5.4.2 System Detection Performance

Ultimately, it is required to evaluate the BER performance of the proposed GA-based MDD. Comparison is made against the matched-filter receiver, the LDD, and the RNN-based MDD. For the GA detector, systems with population sizes of 60 and 80 are presented. The BER results of all 10 users are averaged and shown in Figure 5.6 and Figure 5.7, respectively, in Gaussian (α=2) and impulsive noise (α=1.5). Under the favourable Gaussian noise channel, it is observed that the RNN and GA-based MDDs have similar performance and they only incur negligibly small performance loss with respect to the LDD, which is the optimal estimator for $\boldsymbol{x}$ in Gaussian noise [48].

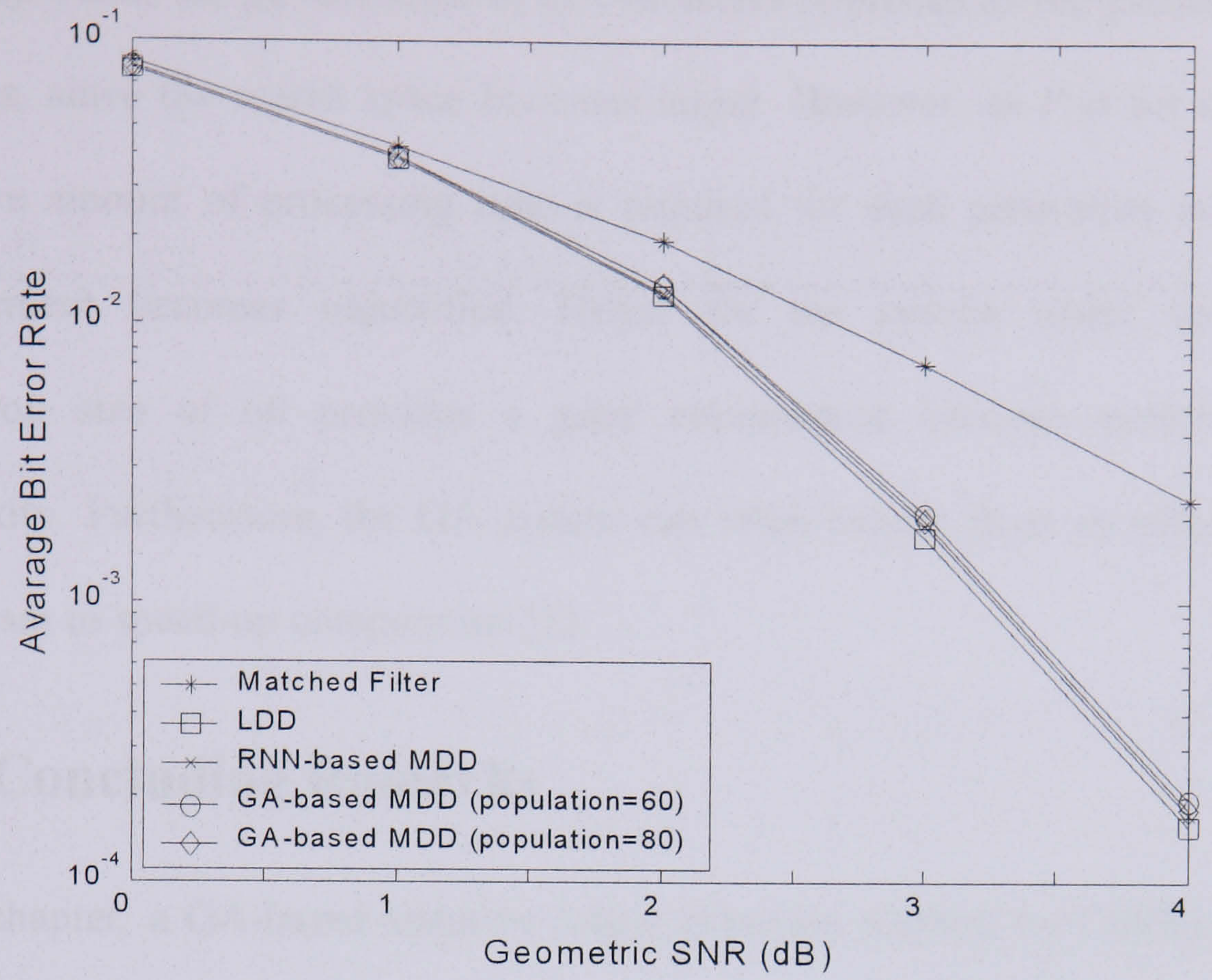


Figure 5.6: Average BER performance comparison in Gaussian noise.

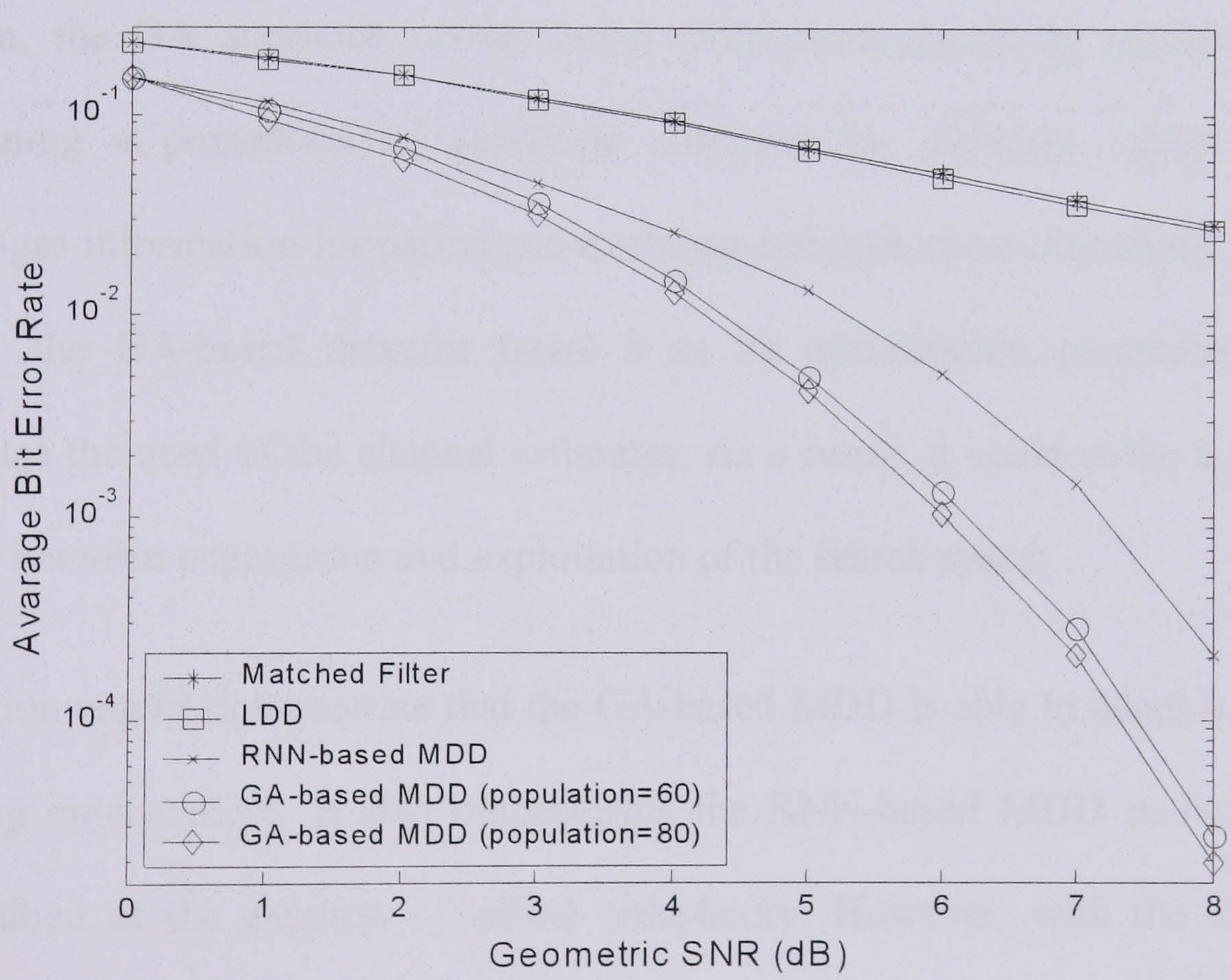


Figure 5.7: Average BER performance comparison in impulsive noise (α=1.5).

However, in the impulsive noise channel, the GA-based detectors clearly outperform all other detectors due to its multi-dimensional search and adaptive outlier rejection capability. Also, the performance of GA detectors improves as the population size P increases, since the search space becomes larger. However, as P is set to too large, excessive amount of processing time is required for each generation and the BER improvement becomes unjustified. Hence for the system under consideration, population size of 60 provides a good compromise between performance and complexity. Furthermore, the GA system can often benefit from an effective use of parallelism to speed-up computation [5].

5.5 Concluding Remarks

In this chapter, a GA-based adaptive robust detection method for CDMA systems is proposed. Unlike the recently proposed RNN-based MDD which performs local, single-point search for a given cut-off parameter h of the M-estimator's objective function, the GA approach carries out a multi-point search by manipulating and maintaining a population of candidate solutions for different values of h and encourages information formation and exchange between these directions. In addition to this, the GA-based detector treats h as its optimisation parameter and thus eliminates the need of the channel estimator. As a result, it could strike a remarkable balance between exploration and exploitation of the search space.

Simulation results demonstrate that the GA-based MDD is able to adapt h in a noise-changing environment. It also outperforms the RNN-based MDD in non-Gaussian noise, albeit at the expense of added complexity. However, with the advances in parallel architecture, it can be expected that the GA-detector will present a promising alternative to the RNN solution. Also of importance is the fact that the GA-based MDD totally overcomes the challenging problem of channel estimation, particularly in a time-varying wireless channel.

6 Application to Scheduling Problem in Mobile Ad Hoc Networks

6.1 Introduction

In the next generation of wireless communication systems, there will be a need for the rapid deployment of independent mobile users. Significant examples include establishing survivable, efficient, dynamic communication for emergency or rescue operations, disaster relief efforts, and military networks. Such network scenarios cannot rely on traditional centralised and organised connectivity and thus popularise the notion of infrastructure-less, or so-called MANET. MANET is an autonomous system of mobile hosts, which are free to move about arbitrarily and communicate with each other via multi-hop wireless connection in the absence of a fixed infrastructure. Ad hoc is a relatively new field in wireless networking area and a lot of issues remain open. In this chapter, we consider a class of problems known in literature as *scheduling problems* in MANET.

Since FDMA tends to become very inefficient in dense networks and CDMA is difficult to implement due to host mobility thus need to keep track of the frequency-hopping patterns and/or spreading codes for the time-varying neighbouring hosts, TDMA technology is envisioned to be widely used to provide collision-free packet transmission with QoS support for ad hoc networks [76]. Therefore, we will focus on the scheduling issue in TDMA-based ad hoc network in this study.

In a TDMA-based ad hoc network, time is partitioned into unit-length transmission slots and grouped into frames. Each slot is designed to accommodate a single packet to be transmitted and received between pairs of hosts in the network. It should be ensured that all the hosts must be allocated at least one time slot in every TDMA frame, which is referred to as no-transmission constraint. Furthermore, because every host operates in broadcast mode in packet transmission, this may cause two types of conflicts, i.e. the primary conflict and the secondary conflict [77]. The first one occurs when two or more directly connected hosts transmit at the same time, while the second one arises if two or more hosts at a distance of two hops transmit simultaneously. An ideal broadcast scheduling must guarantee that there is no any aforementioned conflict in an ad hoc network, and the no-transmission constraint is satisfied too. This kind of broadcast scheduling problem has been proven to be NP-complete [77], consequently there are no efficient algorithms guaranteed to give an optimal solution and run in polynomial time.

Various heuristic and other algorithms have been proposed to obtain the optimum scheduling solution over last two decades. Earlier work on the scheduling problem inclined to target different optimisation objectives, such as, [78], [79] aimed to minimise the length of TDMA frame to achieve lower delay, while [77], [80] expected to maximise the slot utilisation in TDMA frame to obtain higher throughput. However, after realising the fact that both objectives can affect the system performance simultaneously in [81], researchers intended to consider both optimisation objectives concurrently. Wang et al. [81] proposed an algorithm based on the mean field annealing (MFA) neural network, which has been shown to be able to find near-optimal solutions with reasonable computational complexity. A two-phase algorithm based on the sequential vertex colouring (SVC) method was presented by Yeo et al., which achieved comparatively better solutions than the MFA-based algorithm [82]. GA has also been considered as a means of solving scheduling problem by Chakraborty [83], which showed to be capable of finding the best solutions in the literature so far.

In this chapter, based on GA as well but with a different design strategy, a novel and competent scheduling algorithm is proposed. By employing a permutation-based encoding strategy, the proposed GA approach could effectively obviate the invalid solutions, and consequently reduce the problem search space to a great extent. Furthermore, the proposed GA approach exhibits very good search strength and obtains competitive solutions compared with the previous reported methods.

The organisation of this chapter is as follows. The formal formulation of the broadcast scheduling problem in a TDMA-based ad hoc network is given in Section 5.2. Section 5.3 discusses the proposed permutation encoded GA scheduling scheme in details. In Section 5.4, the results for different simulation scenarios are presented and discussed. Finally, in Section 5.5 conclusions are drawn.

6.2 Formulation of Optimum Broadcast Scheduling Problem

From a network point of view, a mobile ad hoc network can be represented by an undirected graph $\boldsymbol{G} = (V, \boldsymbol{E})$, where the vertices in $V = \{1, 2, \ldots, N\}$ represent the individual mobile host, and the set of undirected edges $\boldsymbol{E}$ characterizes the set of transmission links in the network. Note that N is the total number of the mobile hosts. As a result, there exists an undirected edge $e = (i, j) \notin \boldsymbol{E}$ if two hosts are within the range of each other, which is also known as one-hop apart. If $(i, j) \notin \boldsymbol{E}$, but there is an intermediate host k such that $(i, k) \in \boldsymbol{E}$ and $(k, j) \in \boldsymbol{E}$, then hosts i and j are two-hop apart. From above definitions, the topology of the ad hoc network can be described by an $N \times N$ symmetric connectivity matrix $\boldsymbol{C} = (c_{ij})$, which is defined as

$$c_{ij} = \begin{cases} 1, & \text{if hosts } i, j \text{ are one-hop apart} \\ 0, & \text{otherwise} \end{cases} \tag{6.1}$$

The corresponding compatibility matrix $\boldsymbol{F} = (f_{ij})$ can be obtained from matrix $\boldsymbol{C}$, and defined as

$$f_{ij} = \begin{cases} 1, & \text{if hosts } i, j \text{ are one-hop or two-hop apart} \\ 0, & \text{otherwise} \end{cases} \tag{6.2}$$

For the scheduling problem, it requires a conflict-free and constraints-satisfied TDMA frame for packet transmission and this frame is repeated over time. Thus, we assume that there are M time slots in each frame and use an $M \times N$ binary matrix $\boldsymbol{X} = \{x_{mj}\}$ to denote a TDMA frame, where the element

$$x_{mj} = \begin{cases} 1, & \text{if } m\text{th time slot be assigned to host } j \\ 0, & \text{otherwise} \end{cases} \tag{6.3}$$

Following this definition, the slot utilisation index for the whole network, ρ, is given by

$$\rho = \frac{1}{M \times N} \sum_{m=1}^{M} \sum_{j=1}^{N} x_{mj} \tag{6.4}$$

The objective is to get an optimum TDMA cycle that has the minimum frame length (M) and the maximum slot utilisation index (ρ), which is referred to as OBS problem in the following. More precisely, the OBS problem can be stated below

Minimise M and maximise ρ,

Subject to

$$\sum_{m=1}^{M} x_{mj} \geq 1, \ \forall j \tag{6.5}$$

$$\sum_{m=1}^{M}\sum_{i=1}^{N}\sum_{j=1}^{N} x_{mi}x_{mj}f_{ij} = 0, \tag{6.6}$$

Equation (6.5) reflects the no-transmission constraint, which guarantees that every mobile host should be assigned at least one time slot. Equation (6.6) characterises conflict-free constraint, which implies that every two hosts, which are one-hop or two-hop apart, must be scheduled to transmit in different time slots. For a mobile ad hoc network, the minimum TDMA frame length depends on the actual topology and generally is computationally intractable due to its NP-completeness nature [3]. However, a tight lower bound for the frame length M can be obtained easily by knowing the maximum degree of a host (G) in the considered network, which can be given by [82]

$$M \geq G + 1 \tag{6.7}$$

6.3 Permutation GA approach for OBS problem

GA has been successfully used to perform many sorts of scheduling optimsation problems, such as travelling salesman problem (TSP) [84], job shop [85], and flow shop [86]. Encouraging results from these problems drove the use of GA for OBS problem in this study.

In this chapter, a permutation encoded GA is proposed to solve the OBS problem, which involves many problem specific modifications appropriate for a given environment to suit the design requirements. The following subsections outline the development of the proposed GA scheduling scheme, whose structure is depicted in

Figure 6.1.

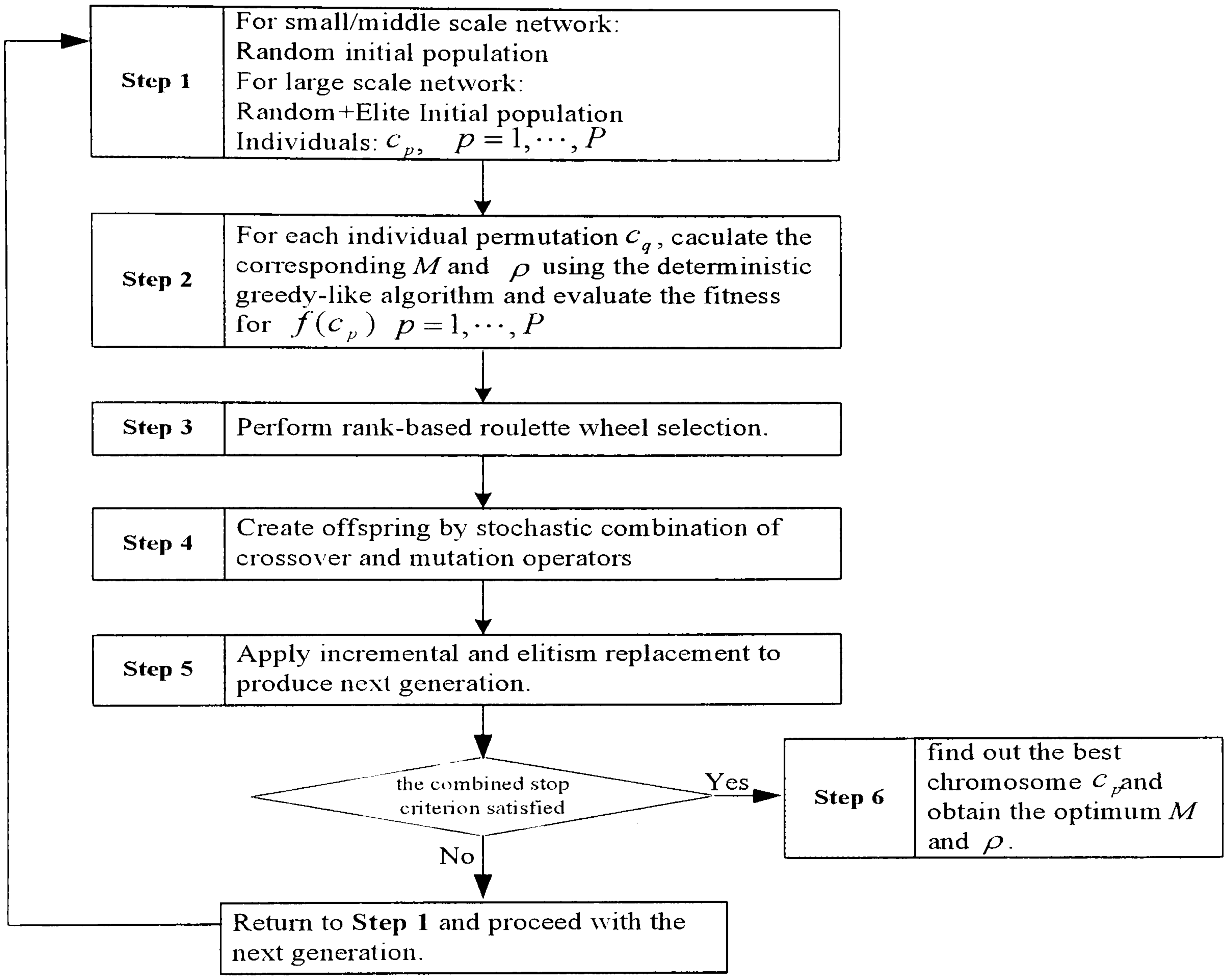


Figure 6.1: Flowchart depicting the structure of the proposed GA-based scheduling scheme for TDMA-based MANET.

6.3.1 Chromosome representation

The encoding scheme of chromosomes may vary according to the nature of the problem and has a major impact on the performance because it can severely limit the search space observed by the system. As for the OBS problem, Chakraborty adopted the traditional binary encoding strategy in [83], which mapped the target TDMA frame to a chromosome as an $N \times N$ binary matrix, and randomly initialised all the elements in chromosome by zeros and ones. Clearly, the search space has a huge size of 2^{NN} and contains many invalid solutions which violate the no-transmission and primary or secondary conflict constraints. Alternatively, due to the fact that time-slot assignment by an arbitrary sequence of hosts would result in different TDMA frames (details discussed in Section 5.3.3), a permutation-based encoding scheme is adopted in this study. Therefore, each chromosome is represented by an N-dimensional integer permutation vector $\mathbf{c} = [c^1, \cdots, c^N]$, where each component corresponds to the serial number of a mobile host. The advantages of this permutation encoding scheme are twofold. First, it is obvious that the size of the problem search space has been reduced from 2^{NxN} to $N!$, and all the chromosomes are valid because there is no constraint imposed on the host permutation sequence. Second, a permutation-based scheme eliminates the need for introducing the penalty function to penalise the infeasible solutions as encountered in [83], whose performance thus depends on tuning the parameters in the penalty function.

6.3.2 Initialisation

In initial experiments, the population of permutation chromosomes $\{\mathbf{c}_p = [c_p^1, \cdots, c_p^N],\ p = 1, \cdots, P\}$ is initialised randomly by generating different random permutations of 1 to N, where P is known as the population size. However, it is observed in this study that random initialisation works sufficiently only for small/middle scale networks ($N \leq 40$). As for the larger network, random initialisation appears to exhibit inefficiency in obtaining optimum solution (i.e. the lower bound of frame length). In that case, a rapid enumeration method was used to get an elite initial population. Details were discussed in Section 5.5.

6.3.3 Fitness Evaluation

Based on the permutation sequence, the time-slot assignment can be carried out using a deterministic greedy-like algorithm as follows:

Input: $\mathbf{c}_p = [c_p^1, \cdots, c_p^N]$

Output: M and ρ

Step 6.1) Initialise M=0.

Step 6.2) Get a host number in permutation c_p sequentially.

Step 6.3) Check all the already assigned time slot(s) to see if the host can be allocated without violating constraints.

Step 6.4) If yes, allocate the corresponding time slot(s) to the host, otherwise, allocate a new time slot to the host and increase M.

Step 6.5) If not end of permutation, go to Step 6.2).

Step 6.6) Check all the already assigned time slot(s) again to see if any host can be fitted into any time slot without violating constraints.

Step 6.7) Get M and calculate ρ by Equation (6.4).

By convention, the fitness value should be a positive value. Since the aim is to minimise the TDMA frame length M and to maximise the slot utilisation index ρ, we introduce Equation (6.8) for evaluating the fitness of each chromosome. Equation (6.8) serves as the fitness function of the proposed GA scheduling approach and the lowest value in Equation (6.8) corresponds to the best chromosome.

$$f(\mathbf{c}_p) = M - \rho \tag{6.8}$$

In reality, as for the two optimisation objectives M and ρ, it is desirable that frame length M could be given higher priority in consideration of the practical network requirements. It is trivial that the frame length M has the minimum step size of one, while the slot utilisation index ρ has the maximum value of one. Hence, without introducing any weight parameter the frame length M evidently has more influence than the slot utilisation index ρ in Equation (6.8).

6.3.4 Genetic operators

Based on the fitness function defined in Equation (6.8), three basic types of genetic operators are required to modify the population: selection, crossover, and mutation. In this subsection, we will present some genetic operators for the proposed GA scheduling approach. More discussion about permutation-coded genetic operators can be found in [87].

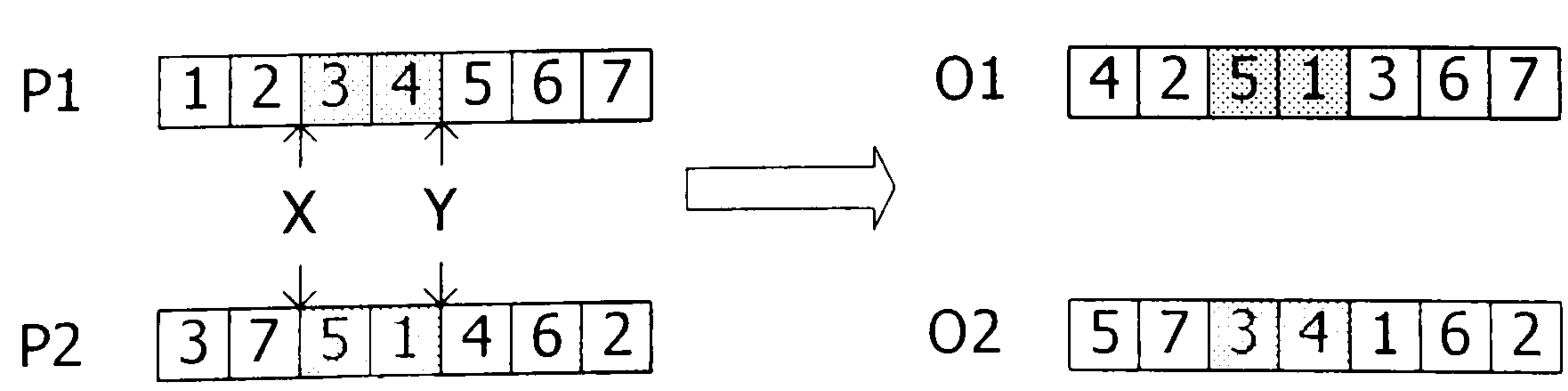
6.3.4.1 Selection

Selection is a process used for choosing parent chromosomes to participate in reproduction for the next generation. In most practices, a roulette wheel approach [5] is adopted as the selection procedure, in which the reproductive opportunity of an individual parent is granted a roulette wheel sector whose size is proportional to its fitness value so that highly fit chromosomes contribute more copies to the mating pool than do those poor ones. However, it is usual that super-fit individuals will be rapidly dominant in the population at the beginning of GA and thus a premature convergence occurs. Moreover, most individuals will be of similar fitness towards the end of GA and the search stagnates. In this study, a ranked-based selection scheme [88] is used in order to mitigate the above problems, in which the population is sorted according to the objective values and the fitness assigned to each individual depends only on its position in the individuals rank not on the actual objective value, thus providing a simple and effective way of controlling selective pressure.

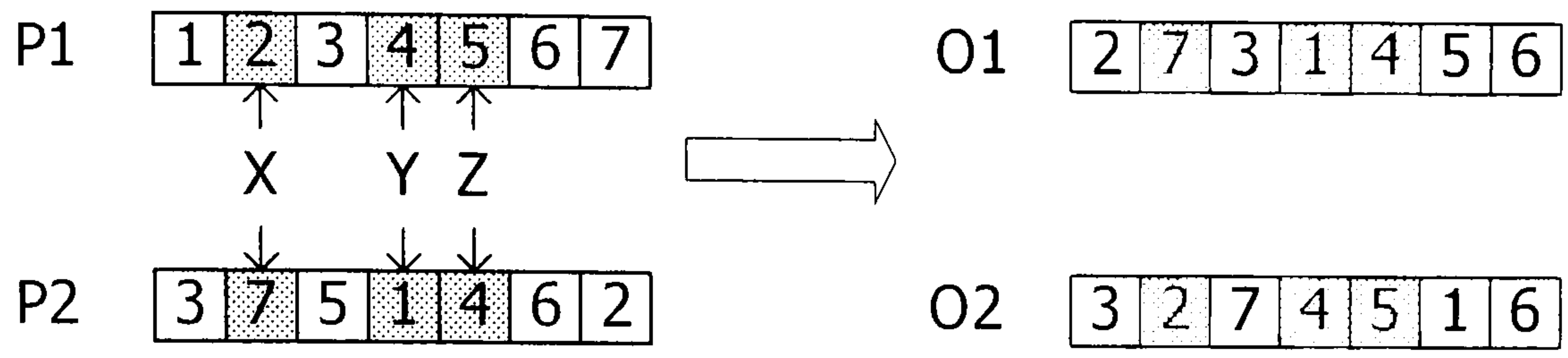
6.3.4.2 Crossover Operators

Crossover is a crucial operator that combines two or more parent chromosomes to produce new offspring chromosomes. There are many ways reported in literature to implement the permutation-based crossover [89], [91]. A suitably combined crossover scheme can make use of the advantages offered by each crossover technique and significantly accelerate the search process [92]. In the proposed GA scheduling approach, we examine the combined use of three crossover operators: PMX (partially mapped crossover) [89], PBX (position-based crossover) [90], and OX (ordered crossover) [91]. As will be seen later, the combined use of these crossover operators can lead to an enhanced performance.

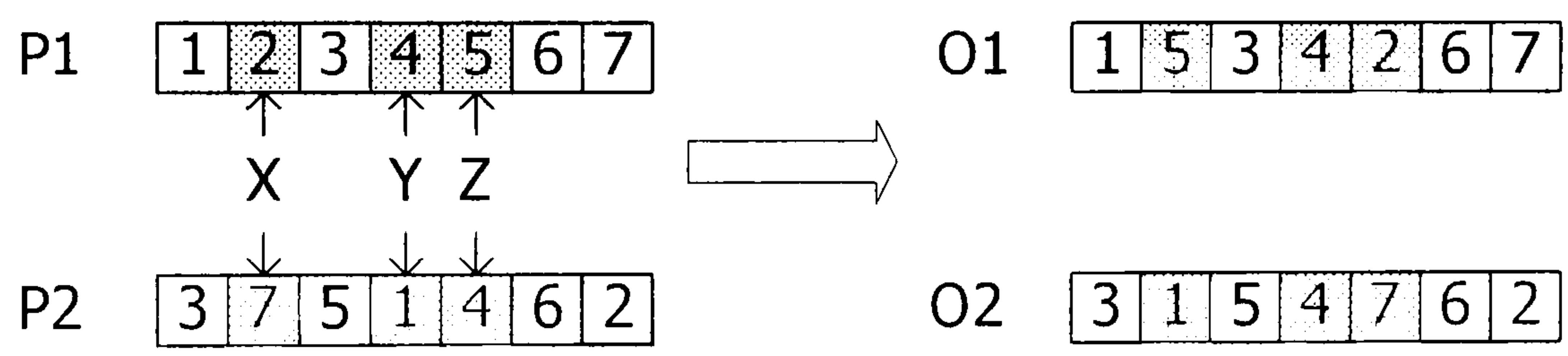
The PMX works as follows. Given two parent chromosomes, two crossover points are chosen uniformly at random along the length of chromosomes which define a matching section. The corresponding elements within the matching section exchange position-by-position to create two new offspring chromosomes. Thus in the following example, the crossover points X and Y define a matching section involves a mapping, in this case $\{3 \leftrightarrow 5,\ 4 \leftrightarrow 1\}$, and PMX gives



In PBX operation, several positions are chosen randomly and the absolute positions in which those elements appear in one parent are imposed on the other parent to produce two offspring. For instance, if choosing three positions (X, Y and Z) as follows in parent chromosomes, it might generate



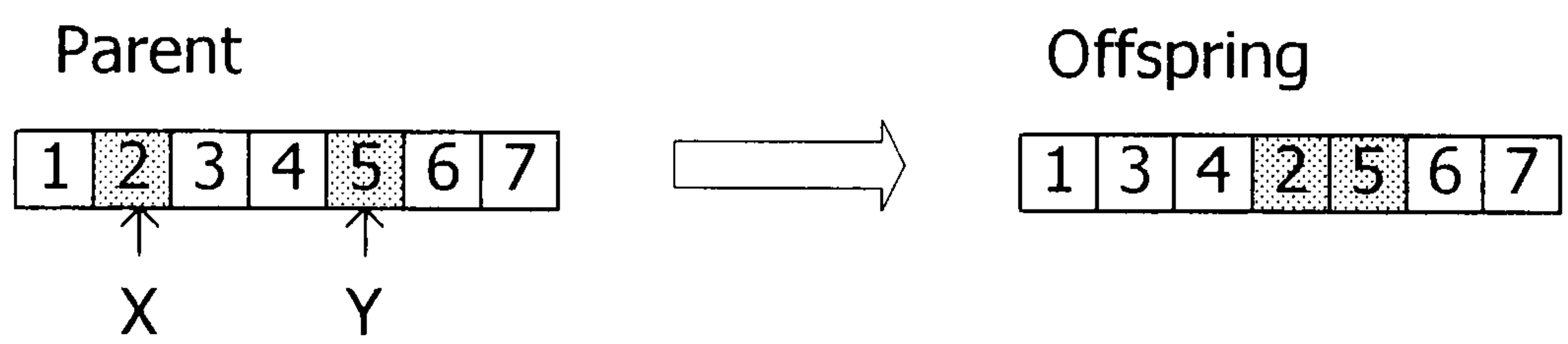
The rationale for such operators as PMX and PBX is that they preserve the absolute positions in the sequence of elements of one parent, and/or the relative positions of those from the other. On the other hand, OX operator tends to preserve the order of elements from parents to offspring. OX might create offspring as follows



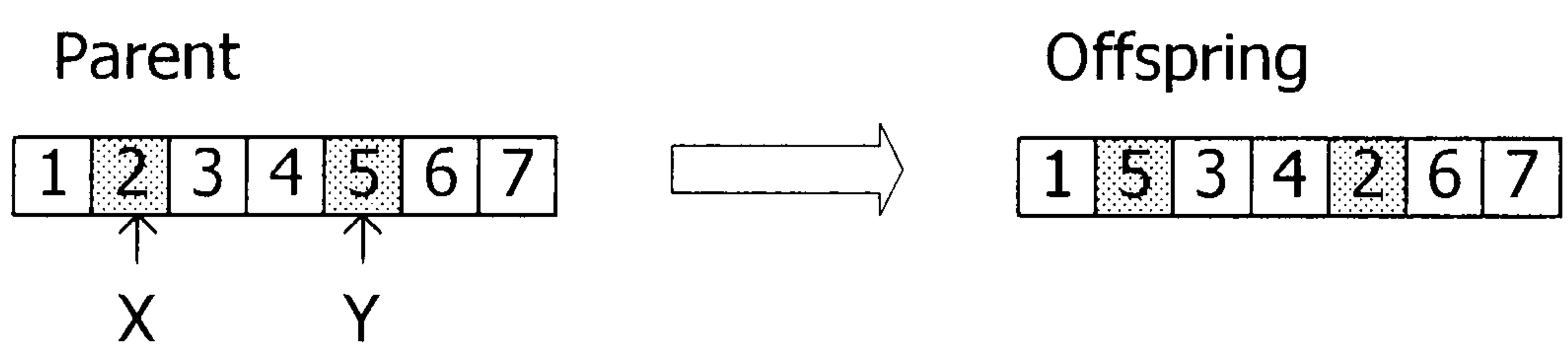
It should be noted that crossover operation is not always invoked and an empirical theoretical crossover rate p_c is usually used to determine the probability for performing crossovers in GA [5]. However, since in this study we adopted a combined crossover scheme, unlike the conventional method, we employed a stochastic crossover scheme proposed in [92] and used experimentation to choose the corresponding probabilities. This is discussed in Section 6.4.

6.3.4.3 Mutation Operators

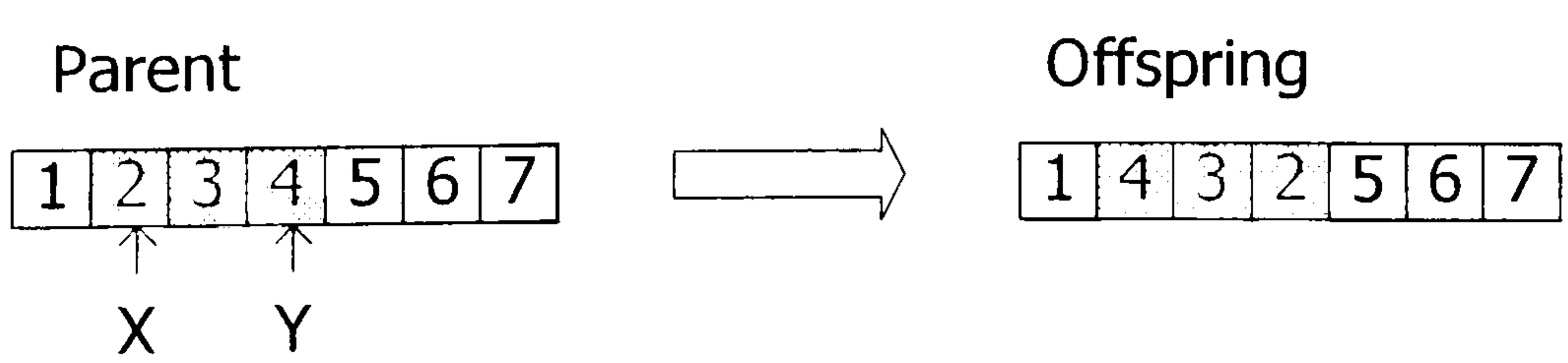
The mutation operator randomly alters some values in a chromosome to obtain entirely new offspring chromosomes. In order to achieve enhanced performance, a combined mutation scheme is adopted in this study, which incorporates position-based shift mutation operator (PBSM) [93], order-based swap mutation operator (OBSM) [93] and inversion mutation operator (IM) [5]. The idea of the PBSM is to push one randomly selected element in front of another randomly selected element while in OBSM are swapped two randomly selected elements in the parent chromosome. If we assume that X, Y are two random positions in the parent chromosome, PBSM operator works as follows



while OBSM operator might produce the offspring as follows



Clearly, both PBSM and OBSM operators retain most ordering information in the parent chromosomes. In comparison, the IM operator preserves the proximity information from the parent to offspring by randomly selecting a subsection in parent chromosome, and reversing the order of the elements in it. Therefore, in the following example, the subsection involves a re-order process where IM creates new offspring as follows



Mutation is potentially useful in restoring genetic diversity that may be lost in a population due to premature convergence. As with the crossover, the mutation operator is also generally associated with a mutation rate p_m to determine whether or not the mutation operator is to be applied to the chromosome [5]. In the same way, we determined the mutation probabilities by experimentation, which was discussed in Section 5.4.

To this end, all genetic operators utilised in the proposed GA scheduling approach have been described. We give a lemma as follows

Lemma 6.1: In the proposed GA scheduling approach, all the generated TDMA scheduling solutions in every generation are conflict-free and constraints-satisfied.

Proof: Because the proposed GA scheduling is based on permutation encoding scheme, the chromosomes produced by all genetic operators in every GA generation are arbitrary permutation of the mobile hosts. For every permutation, step 5.3) and step 5.6) of the deterministic greedy-like algorithm guaranteed that no primary or secondary conflict occurs, and step 5.4) assured that every host in the permutation is assigned at least one time slot which satisfies the no-transmission constraint. Therefore, the resulting TDMA scheduling solutions in every generation are conflict-free and constraints-satisfied.

6.3.5 Replacement

After a predefined number of offspring has been produced through the above genetic operators, a replacement strategy is required in order to modify the old population with the new generation [35]. The so-called incremental replacement is used in which children will have the chance to compete with some of the parent individuals. Besides, when creating a new population by crossover and mutation, the GA has a chance that it will lose the best chromosome. Thus, the elitist strategy is adopted to improve algorithm performance [5], which appends the best performing chromosome of the previous generation to the current population, and thus ensures that the chromosome with the best fitness value always survives to the next generation.

6.3.6 Termination

Termination is a criterion by which the GA decides whether to continue searching or stop the search. Although for the fitness value calculation in Equation (6.8) a lower bound for the frame length M exists, no upper bound for slot utilisation index ρ is known in advance. Therefore, a combined termination strategy is adopted in this study. The GA will terminate if it had reached the predefined G_{max} or it had not improved over the last 50 (empirically determined) successive generations. This strategy cannot only ensure that the GA has enough time to converge, but also avoid excessively high complexity and processing time.

6.4 Results and Discussions

In this section, computer simulation results are presented to demonstrate the various aspects of the proposed GA scheduling scheme. A summary of the various GA parameters used is given in Table 6.1. Before presenting the optimum scheduling results, i.e. TDMA frame length M and slot utilisation index ρ, it is of interest to examine the evolutionary behaviour of the GA scheduling approach including the effects of different genetic operators and choice of GA parameters.

Table 6.1: Summary of GA parameters used for the scheduling simulations in MANET system.

Parameter	Value/Type
Population size, P	50 (small/middle networks), 100 (larger networks)
Representation	Permutation-encoded
Initialisation	Random(small/middle networks), Elite (larger networks)
Generation selection	Rank-based roulette wheel
Crossover operators	PMX;PBX;OBX (20%;40%;40%) in stochastic roulette wheel selection
Crossover probability, p_c	0.6
Mutation operator	PBSM;OBSM;IM (20%;40%;40%) in stochastic roulette wheel selection
Mutation probability, p_m	0.1
Replacement	Incremental + Elitism
Generation	Combined stop criterion with G_{max}=200

6.4.1 Evolutionary Performance

In the proposed GA scheduling scheme, the permutations found by the GA will be examined by the deterministic greedy-like algorithm to determine the corresponding schedules, which implies that the results from the GA have indirect impact on the final scheduling results. From this observation, it is interesting to find that: First, the complementary process of the GA and deterministic greedy-like algorithm mitigates the search burden of the GA and thus enables a relatively smaller crossover rate in the GA. We found that setting crossover probability as 0.6 would provide a good compromise between performance and complexity. Second, the mutation operator can have a relatively larger rate because the complementary process moderates the difference between crossover and mutation operators, and the main objective of the mutation operator is to achieve the exploration area that the crossover operators may not be able to reach. Empirically, the mutation probability is set to 0.1 in this study.

As we discussed in Section 6.4, a stochastic combination technique for both the crossover and mutation operators is adopted in this study. The stochastic combination technique is very similar to the process of proportional selection of reproduction candidates from a mating pool [92]. That is, crossover or mutation is selected from a biased roulette wheel where different crossover or mutation operators are placed. In the biased roulette wheel for stochastic crossover or mutation, each operator has a roulette wheel sector sized according to its performance to find the best solution.

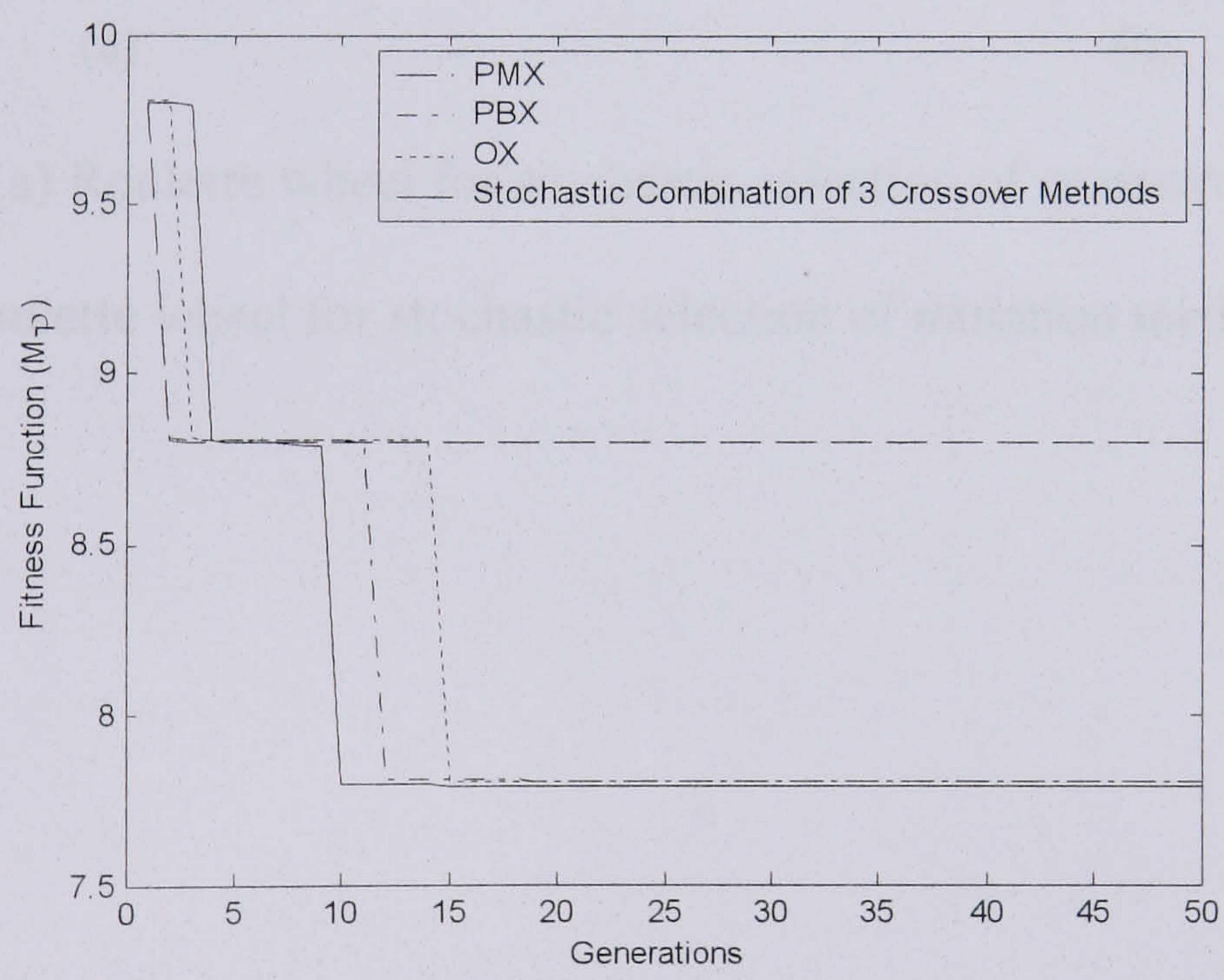


Figure 6.2: Evolutionary performance comparison of different crossover methods for 40-host network (N=40).

Figure 6.2 shows the convergence characteristics of each crossover technique for scheduling in an ad hoc network (N=40). Among the three considered crossovers, the PMX crossover showed the worst performance, and the other two methods showed similar better performance. Based on these empirical studies, the weight of each crossover was determined in a biased roulette wheel as shown in Figure 6.3(a). By combining the three crossover techniques in a stochastic manner, the convergence rate becomes faster and the performance is also further improved, as can be seen in Figure 6.2. Same experiments were conducted for mutation techniques, and it is found that PBSM mutation is the worst one and the other two are better with almost same performance. Thus, the stochastic mutation is chosen as illustrated in Figure 6.3(b).

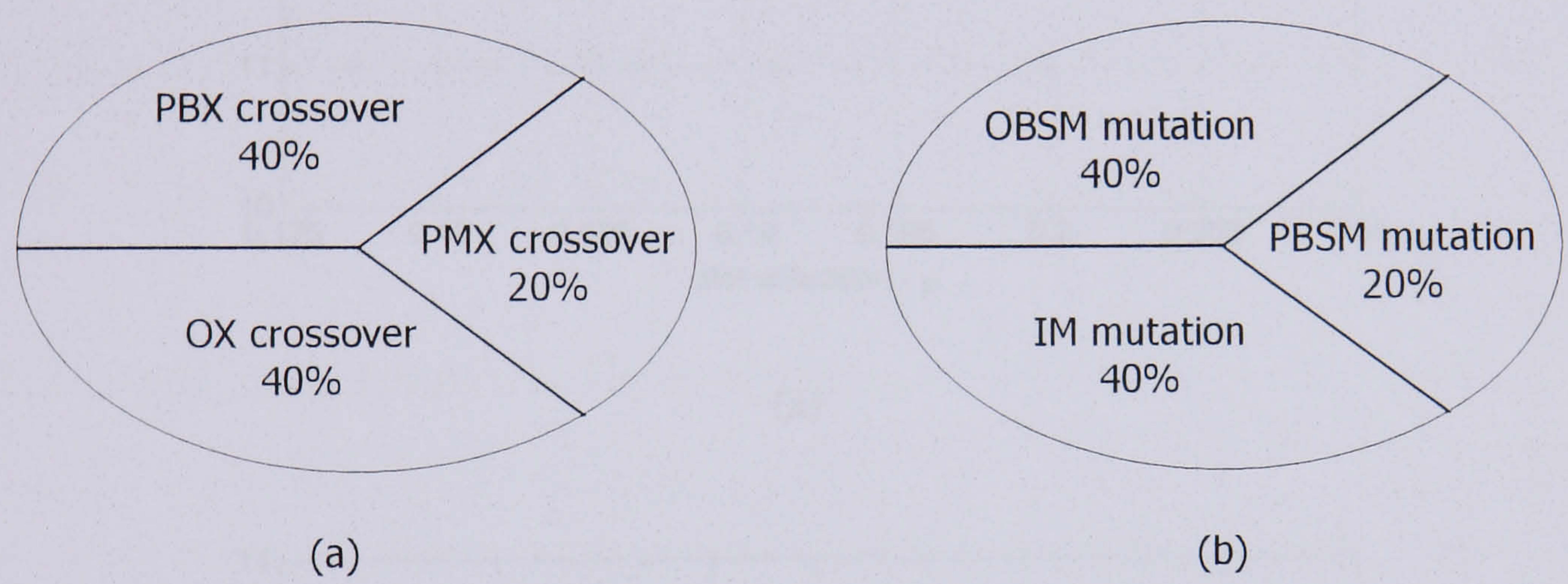
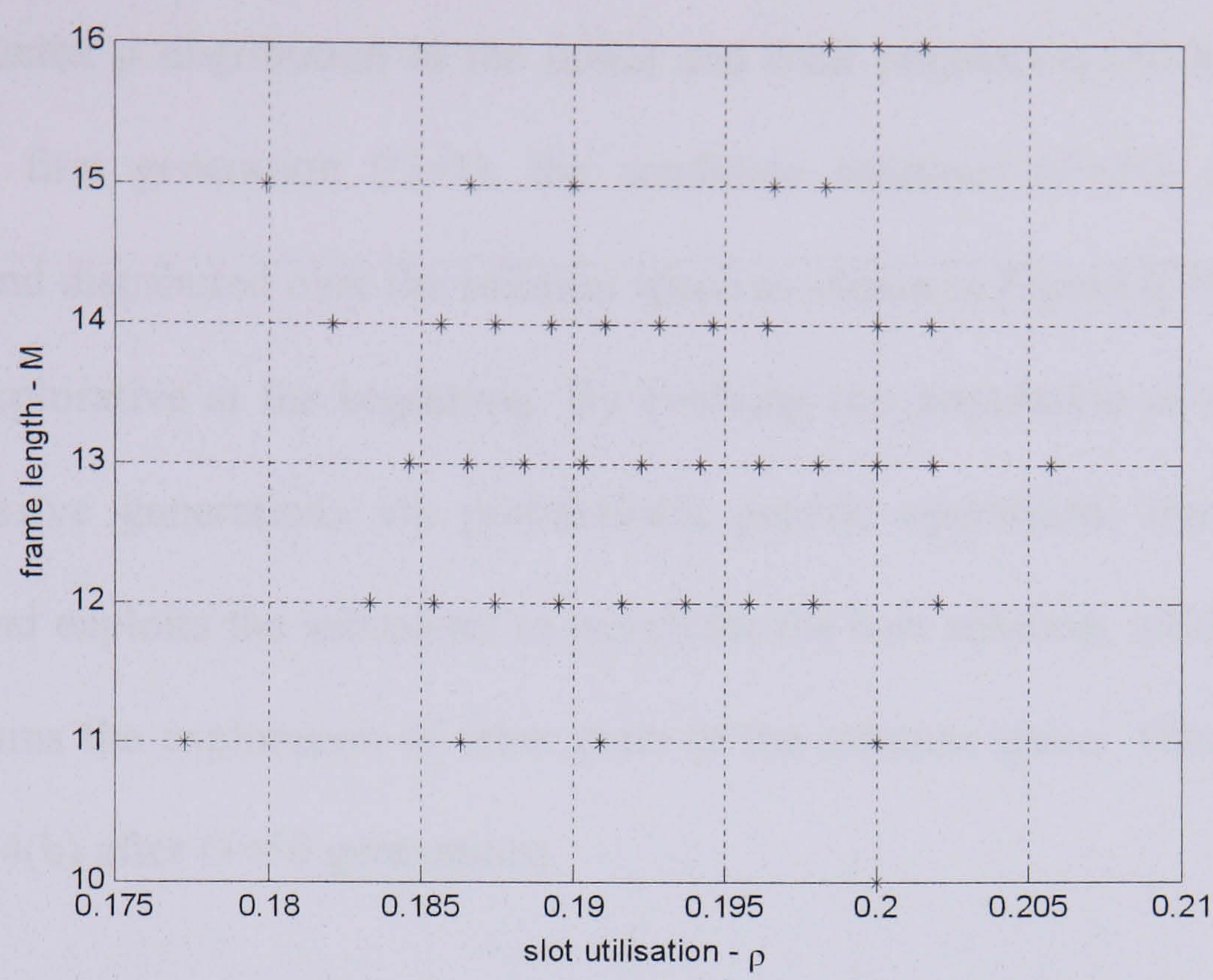
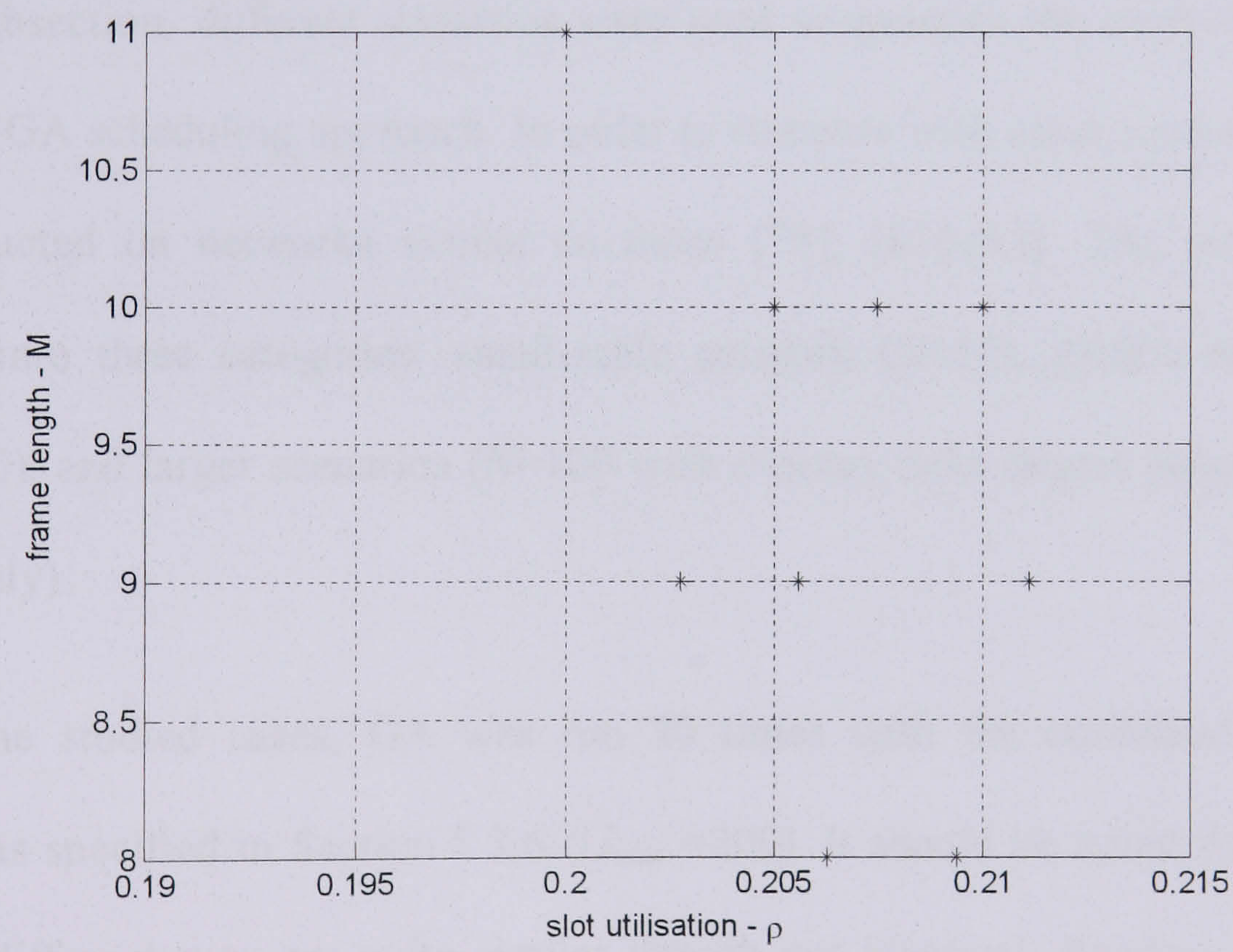


Figure 6.3: (a) Roulette wheel for stochastic selection of crossover methods. (b) Roulette wheel for stochastic selection of mutation methods.



(a)



(b)

Figure 6.4: (a) 2-dimensional plot of the distribution of initial population (b) 2-dimensional plot of the distribution of end population. (P=50 and N=40).

In order to examine the evolutionary performance of the GA scheduling scheme,

Figure 6.4 illustrates two dimensional plots with the frame length M and slot utilisation index ρ distribution in the initial and final population (40-host network). During the first generation (G=1), the candidate solutions (P=50) are randomly generated and distributed over the solution space as shown in Figure 6.4(a). Clearly, it is highly explorative at the beginning. By evolving this population of chromosomes over successive generations via probabilistic genetic operations, the GA quickly identifies and exploits the subspaces in search for the best solution, while at the same time maintains the exploration of other parts of the solution space. This is illustrated in Figure 6.4(b) after G=50 generations.

6.4.2 Optimum Schedule Performance

In this subsection, different scenarios were used to quantify the performance of the proposed GA scheduling approach. In order to compare with other works, simulations are conducted on networks similar to those [78], [81]-[83]. The scenarios were grouped into three categories: small-scale network (N=15), middle-scale network (N=30, 40), and larger scenarios (N=100 with average links degree equal to 4 and 6, respectively).

For all the studied cases, GA was run 20 times with the combined termination criterion as specified in Section 5.3.6 (G_{max} =200). It should be noted that the results from 20 different runs are quite similar though not identical. Random initialisation and stochastic crossover/mutation scheme were used. Considering different computation requirements, we choose population size P=50 and P=100 for small/middle and large scale ad hoc network, respectively.

Table 6.2: Simulation results with random initialisation. Population size P=50 was used for cases 1-3 and P=100 was used for cases 4-5. Note that the CPU time measured on a 1.2GHz Pentium III PC with 128MB SDRAM memory.

Cases	hosts, links, av. degree, G+1	Frame length M in no. of gen.	Slot utilisation ρ in no. of gen.	Corresponding transmissions improvement	Comp. time for 100 gen. in CPU time
1	15. 29. 3.8667, 8	8 in 1	0.1667 in 1	15 $\rightarrow$ 20	15 sec.
2	30, 70. 4.6667, 9	10 in 9	0.1233 in 18	30 $\rightarrow$ 37	28 sec.
3	40, 66, 3.3, 8	8 in 10	0.2 in 15	40 $\rightarrow$ 64	2.6 min.
4	100. 200, 4.0. 9	13 in 25	0.1577 in 85	100 $\rightarrow$ 205	6.8 min.
5	100, 300, 6.0, 11	14 in 40	0.1100 in 120	100 $\rightarrow$ 154	7.3 min

Table 6.2 shows the best results out of 20 runs for each problem instance and the average solution times over all 20 runs are presented for demonstration. It is observed that the proposed GA approach exhibits very good search strength for small/middle cases as it always finds the optimum or near-optimum TDMA frame length. However, GA appears inefficient in finding the optimum solutions for larger scenario. This problem will be tackled by using elite initial population as discussed in Section 5.4.3. Furthermore, it can be noticed that the GA always found the lower frame length first, and then tried to futher improve the slot utilisation. Moreover, the search time only grows modestly while the problem search size increases considerably.

6.4.3 Comparison with Other Methods

As for small/middle scenarios, we compare our approach with the methods reported in [81] and [82]. Comparison of results is summarized in Table 6.3. Clearly, the proposed GA shows the best performance for all considered cases. More precisely, for the 30-host middle scale network, the GA approach improves the TDMA frame length by 2 time slots over MFA method [81] and 1 time slot over SVC algorithm [82]. The slot utilisation index has been increased 13.8% more than [81] and 10.1% more than [82] by using the proposed GA approach.

Table 6.3: Comparison of results with other methods for small and middle scale networks.

Cases	hosts, links, av. degree, G+1	Frame length / slot utilisation (M / ρ)		
		The proposed GA	reported in [81]	reported in [82]
1	15, 29, 3.8667, 8	8 / 0.1667	8 / 0.15	8 / 0.15
2	30, 70, 4.6667, 9	10 / 0.1233	12 / 0.1083	11 / 0.112
3	40, 66, 3.3, 8	8 / 0.2	9 / 0.1972	8 / 0.188

As for larger scenarios, we first used random initialisation and the same stochastic crossover/mutation scheme as above to perform the experiments. It is found that the proposed GA scheme achieves better performance compared with the binary GA approach with random initialisation proposed in [83]. However, both of methods failed to obtain optimum solutions, i.e. the lower bound of frame length as defined in Equation (6.7). To cope with this problem, a rapid enumeration method was used in [83] to get a heuristic initial population to improve the performance. In this study, elite initial population is also adopted in order to obtain enhanced performance. In [83] it was found that around 5000 different random permutations are required to be generated in order to create an elite population of 50 heuristic initial solutions. It is should be noted that the these rough figures are based on the probabilistic model, therefore it may require generating more than 100 times of random permutations to arrived at the required elite initial population. After using elite initialisation, our approach achieved marginal improvement on slot utilisation compared with [83]. Comparison of results is shown in Table 6.4.

Table 6.4: Comparison of results with other methods for small and middle scale networks.

Cases	Hosts, links, av. degree, G+1	M/ ρ for the proposed GA		M/ ρ reported in [83]	
		Random initialisation	Elite initialisation	Random initialisation	Elite initialisation
3	40, 66, 3.3, 8	8 / 0.2	8 / 0.2094	9 / 0.2	8 / 0.203
4	100, 200, 4.0, 9	13 / 0.1577	9 / 0.1511	16 / 0.155	9 / 0.148
5	100, 300, 6.0. 11	14 / 0.1100	11 / 0.1091	18 / 0.114	11 / 0.104

In summary, the three simulated networks are shown in Figure 6.5 with N=40 and N=100 including link degree equal to 4 and 6.

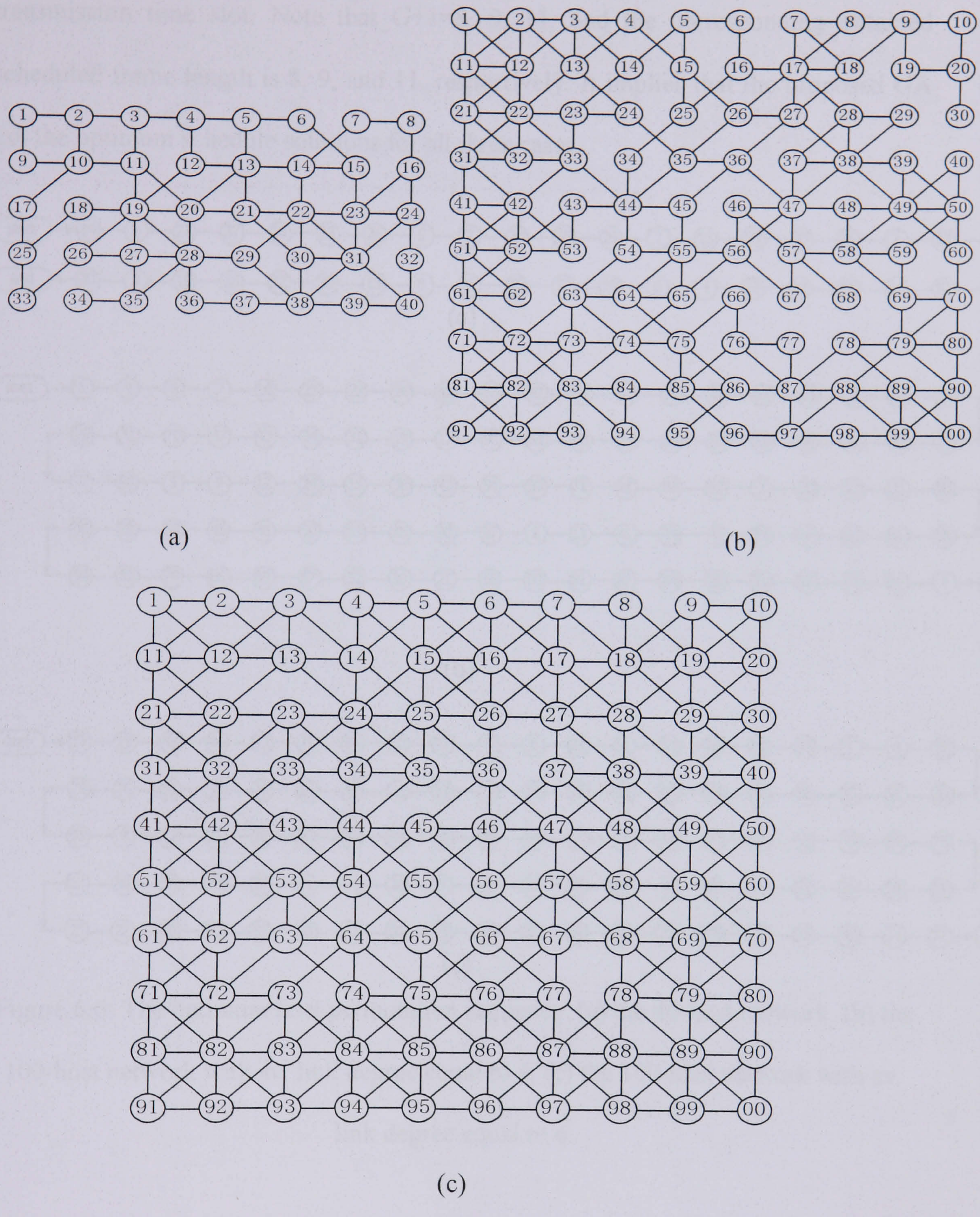


Figure 6.5: The simulated ad-hoc network scenario (host with numbering). (a) the 40-host network, (b) the 100-host network with av. link degree equal to 4, (c) the 100-host network with av. link degree equal to 6.

The optimum host sequences for each case found by GA are shown in Figure 6.6 and the resulting schedules are shown in Figure 6.7, where a black box represents a transmission time slot. Note that G+1=8, 9, 11, and the corresponding obtained scheduled frame length is 8, 9, and 11, respectively. It implies that the proposed GA got the optimum schedule solutions for all three cases.

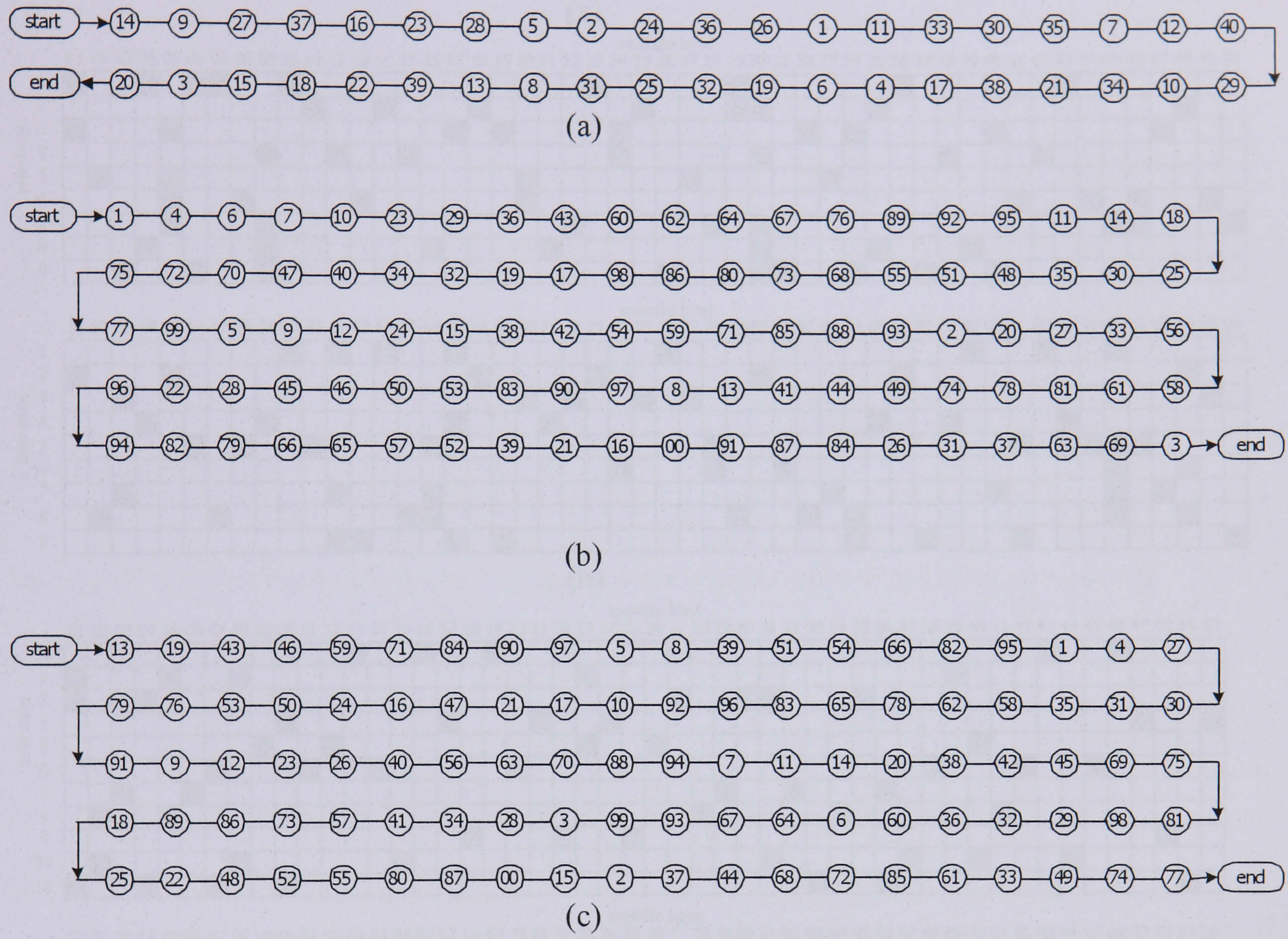


Figure 6.6: The optimum host permutation sequence. (a) the 40-host network, (b) the 100-host network with av. link degree equal to 4, (c) the 100-host network with av. link degree equal to 6.

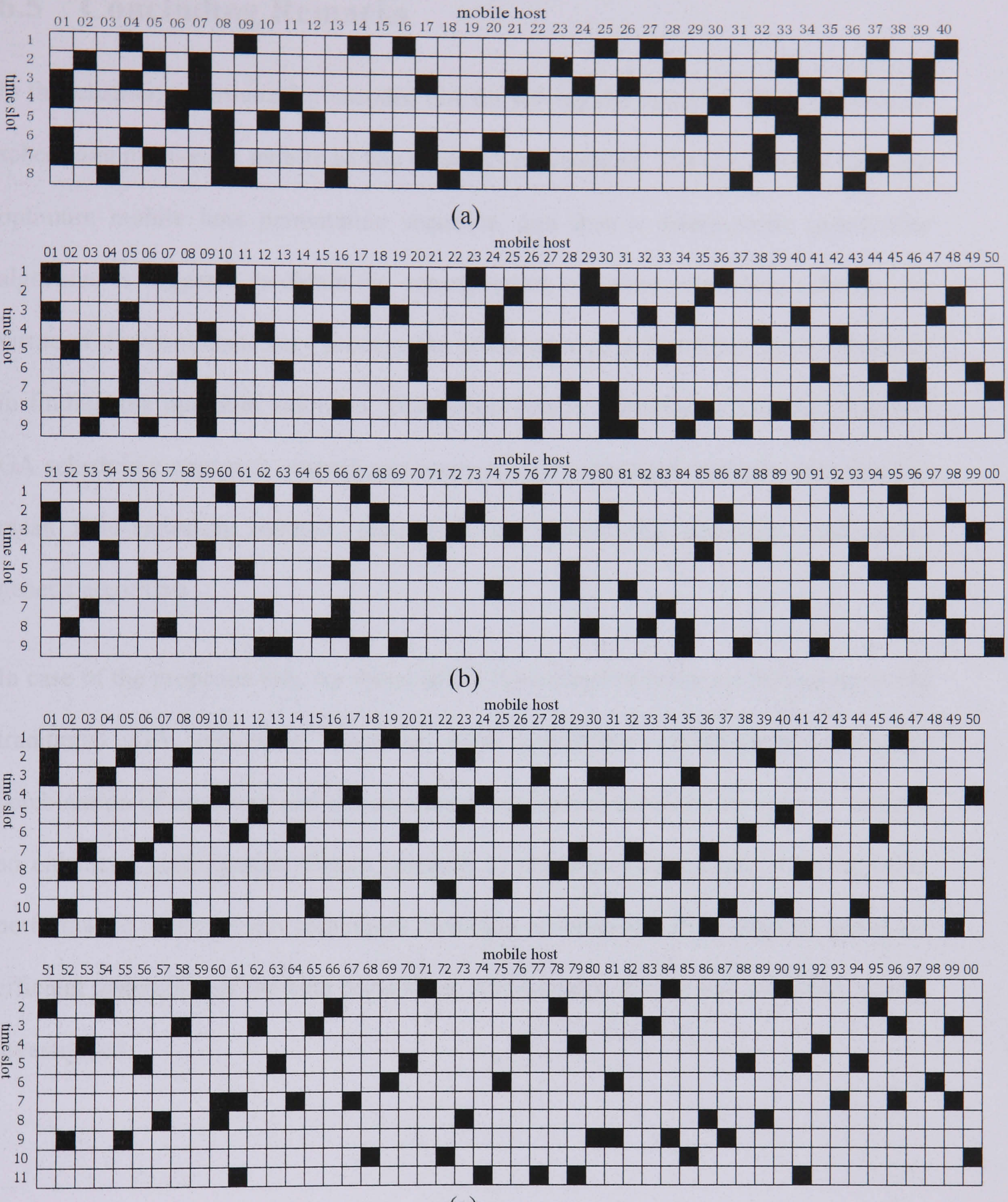


Figure 6.7: The optimum scheduling results. (a) the 40-host network, (b) the 100-host network with av. link degree equal to 4, (c) the 100-host network with av. link degree equal to 6.

6.5 Concluding Remarks

In this chapter, a permutation encoded GA for solving the optimum TDMA broadcast scheduling problem in mobile ad hoc networks is presented. The GA initially finds the optimum mobile host permutation sequence, and then a deterministic greedy-like algorithm is triggered to obtain the corresponding optimum scheduling solution. By virtue of this complementary process, the proposed approach exhibits great efficiency in finding the optimum solutions. Simulation results demonstrate that the proposed GA scheduling approach outperforms many recently proposed methods including the mean field-annealing method, sequential vertex colouring algorithm, and binary genetic algorithm.

In case of the proposed GA, we found that it is essential to adopt modifications to the traditional GA processes, such as elite population initialization, stochastic combination of operators and the combined termination criterion, in order to achieve an enhanced performance. Future research in this area will include studying other performance improvement techniques including problem-specific operators and more efficient generation of the elite population. Additionally, further test problems will be investigated.

7 Application to Minimum Power Problem in Wireless Ad Hoc Networks

7.1 Introduction

A wireless ad hoc network is an autonomous system of mobile nodes, which are free to move about arbitrarily and communicate with each other via multi-hop wireless connection in the absence of a fixed infrastructure. Due to ad hoc nature and mobile environment, broadcasting is a fundamental operation in ad hoc networks to resolve many issues, such as paging a particular host or sending an alarm message to all the nodes in the network [94].

Unlike the wired counterpart, mobile nodes are usually equipped with omni-directional antennas in wireless networks, thus multiple nodes can be reached by a single transmission. Therefore, it is straightforward that the source node can broadcast the messages using its maximum power to reach all the other nodes in the network, or alternatively broadcast the messages to some intermediate nodes using medium power and those intermediate nodes relay the messages to other nodes in the network. In either case, a broadcast tree needs to be formed and some nodes in the broadcast tree contribute the power consumption. Since the mobile nodes in networks are power-constrained in nature, a power efficiency broadcast is required so that the lifetime of network can be prolonged. This kind of problem is first formulated by Wieselthier et al. as the MPB problem in [95]. The objective of the MPB is to assign transmission powers to the nodes in such a way that the network is fully-connected and the total consumption power is minimised. The MPB problem is shown to be NP-complete by Cagalj et al. [96], consequently there are no efficient algorithms guaranteed to give an optimal solution and run in polynomial time. Various heuristic and other algorithms have been proposed to solve MPB problem recently. For example, Wieselthier et al. [95] proposed the BIP algorithm, which is based on greedy heuristic method and provides a very fast sub-optimal solution. An evolutionary approach using stochastic search is proposed by Marks et al. [97], which suggests that improved results can be achieved over BIP algorithm at a higher computational cost. A swarm based Ant Colony System (ACS) algorithm is presented by Das et al. [98], which is shown to be able to find better solutions in very little computation time. Moreover, the r-shrink algorithm and SA-based approaches are proposed by Das et al. in [99] and [100] respectively, both of which are shown to be capable of further improving the solutions obtained using BIP algorithm and achieving the best results for MPB problem thus far in the literature.

In this chapter, a permutation-based GA approach is proposed for solving MPB problem. In the proposed approach, a deterministic greedy-like algorithm is developed to cooperate with GA in order to obtain the optimum solutions. By virtue of the powerful search capability of GA, the proposed approach is shown to be able to find the broadcast tree with minimum power consumption.

The organisation of this chapter is as follows. The mathematical formulation of the MPB problem in ad hoc networks is given in Section 7.2. Section 7.3 discusses the proposed permutation-encoded GA approach in details. In Section 7.4, the results are presented and discussed. Finally, in Section 7.5 conclusions are drawn.

7.2 Formulation of Minimum Power Broadcast Problem

A fixed N-node wireless ad hoc network is considered in this study, in which every node is equipped with omni-directional antennas and randomly distributed over a specified region. One particular node is chosen to be the source node, which has to broadcast a message to all other nodes in the network. It is well known that path loss accounts for the signal attenuation due to physical distance between transmitter and receiver in a wireless scenario. Without loss of generality, the required signal power for supporting a transmission from node u to node v can be expressed as

$$p_{uv} = d_{uv}^{\alpha} \tag{7.1}$$

where d_{uv} is the Euclidean distance between nodes u and v, α is a constant referred to as path-loss exponent, which is between 2 and 4 depending on the wireless environment. By default, the algorithms in this study are conducted by setting α=2 and it is assumed that there is no constraint on the maximum transmitter power p_{max}. These assumptions comply with the previous works [95], [97]-[100].

A graph representation for the multicast tree in an ad hoc network is proposed in [101]. The same concept is applied to the MPB problem here in order to model the broadcast tree in the ad hoc network. We represent an ad hoc network by a node-weighted directed graph $\boldsymbol{G} = (\boldsymbol{N}, \boldsymbol{E}, \boldsymbol{p})$, where the vertices in $\boldsymbol{N} = \{1, 2, \ldots, N\}$ represent the individual mobile node, E characterises the set of transmission links in the network, and p_{uv} is defined by (1) when $u \neq v$. It should be noted that a dummy edge (u, u) with $p_{uu} = 0$ is inserted to $\boldsymbol{E}$ for each $i \in N$. According to the above definitions, a broadcast tree is defined as a directed tree which is composed of a source node s with no incoming links and all other nodes with exactly one incoming link. A node with no outgoing links is known as the leaf node, and all other nodes except the source node are intermediate nodes. More precisely, a broadcast tree is modelled by a node-weighted directed tree $\boldsymbol{T} = (\boldsymbol{N}', \boldsymbol{E}', \boldsymbol{q})$ rooted at source node s, with a node set $\boldsymbol{N}'$ including all the nodes in network (i.e., $\boldsymbol{N}'=\boldsymbol{N}$) and an edge set $\boldsymbol{E}' \subseteq \boldsymbol{E}$. Following this definition, the corresponding decision matrix $\boldsymbol{X}$ for a broadcast tree $\boldsymbol{T}$ is given by

$$x_{uv} = \begin{cases} 1, & \text{if edge}\,(u,v) \in \mathrm{E}' \\ 0, & \text{otherwise} \end{cases} \tag{7.2}$$

The element q_u in $\boldsymbol{q}$ is defined in (7.3), which denotes the transmission power of the node u required for its transmission in the broadcast tree T.

$$q_u = \begin{cases} \max_v (x_{uv} p_{uv}), & \text{if edge}\,(u,v) \in \mathrm{E}' \\ 0, & \text{otherwise} \end{cases} \tag{7.3}$$

The corresponding consumption power for a broadcast tree is calculated as

$$P_T = \sum_{u \in N'} q_u \tag{7.4}$$

The objective of the MPB is to extract a sub-graph T^* from the original graph G, such that T^* is a broadcast tree with minimum power consumption P_{T^*}. Therefore, the MPB problem can be stated as

$$\text{Minimise } P_T = \sum_{u \in N'} q_u = \sum_{u \in N'} \max_v (x_{uv} p_{uv}), \tag{7.5}$$

Subject to

$$\sum_{v \in N} x_{sv} \geq 1, \tag{7.6}$$

$$\sum_{v \in N} x_{vs} = 0, \tag{7.7}$$

$$\sum_{u \in N} x_{uv} = 1, \qquad \forall v \in N/\{s\} \tag{7.8}$$

$$N' = N \tag{7.9}$$

Equations (7.6) and (7.7) reflect the constraints on the source node, which guarantee that the source node should have at least one outgoing link and no incoming links. Equation (7.8) characterises the constraint on the non-source nodes, which implies that every non-source node must have exactly one incoming link. Equation (7.9) represents the wireless broadcast constraint, which means there is no unconnected node in the broadcast tree.

7.3 Permutation GA approach for MPB problem

GA has been successfully used to perform a wide range of optimisation problems [5], and encouraging results drove the use of GA for the MPB problem in this study. In this chapter, a permutation encoded GA is proposed to solve the MPB problem, which involves many problem specific modifications appropriate for a given environment to suit the design requirements. The following subsections outline the development of the proposed GA scheduling scheme, whose structure is depicted in Figure 7.1.

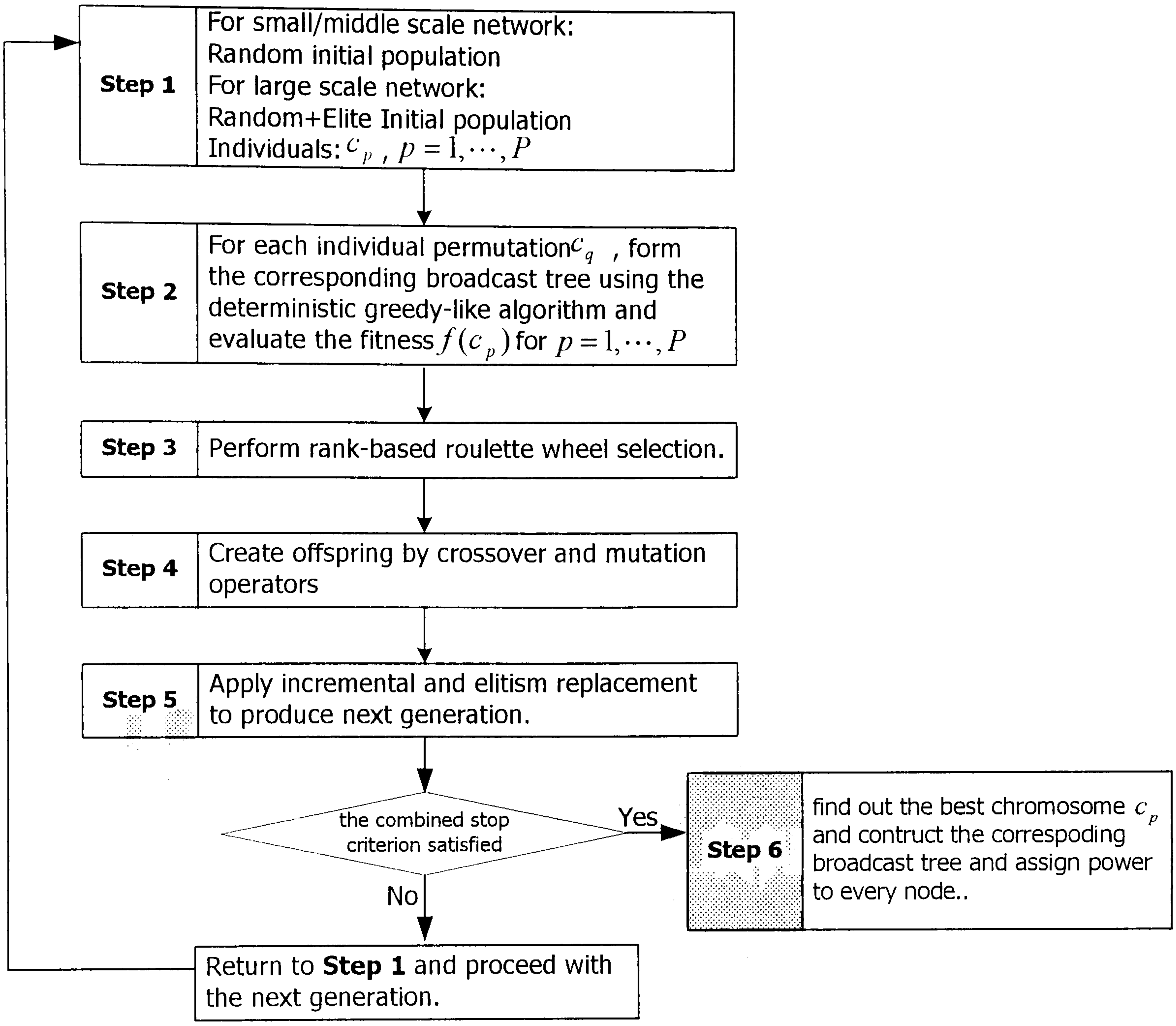


Figure 7.1: Flowchart depicting the structure of the proposed GA-based approach for solving the MPB problem.

7.3.1 Chromosome Representation

The encoding scheme of chromosomes may vary according to the nature of the problem and has a major impact on the performance because it can severely limit the search space observed by the system. As for the MPB problem, due to the fact that assigning power to an arbitrary sequence of mobile nodes would produce different broadcast tree (details discussed in Section 7.3.3), a permutation-based encoding scheme is adopted in this study. In the proposed GA, each chromosome is represented by an N-dimensional integer permutation vector $c = [c^1, \cdots, c^N]$, where each component corresponds to the serial number of a mobile node. To make the algorithm more general, the source node is also encoded in the chromosome and would be marked out in the fitness evaluation procedure as discussed in Section 7.3.3.

7.3.2 Initialisation

The population of permutation-based chromosomes $\{c_p = [c_p^1, \cdots, c_p^N], p = 1, \ldots, P\}$ is initialised by employing a hybrid of random and deterministic approaches, where P is known as the population size. The purpose of using a hybrid generation is to distribute the trial solutions intelligently to produce an elite initial population. While a deterministic solution creates part of the population which is allocated in the vicinity of the optimal solution, the random part of population maintains the diversity. In order to produce the deterministic solution, the BIP algorithm is run first and the corresponding node subsequently added to the broadcast tree is recorded into a node sequence, which serves as the deterministic population for the proposed GA approach (example illustrated in Section 7.4.1).

7.3.3 Fitness Evaluation

Based on the node sequence, the construction of the broadcast tree can be carried out using a deterministic greedy-like algorithm as follows:

Input: source node s and chromosome $c_p = [c_p^1, \cdots, c_p^N]$

Output: broadcast tree $\boldsymbol{T}$ and the corresponding consumption power P_T

Step 7.1) Initialise P_T=0, mark the source node s in the permutation c_p to avoid the subsequent selections.

Step 7.2) Obtain the first unmarked node z in permutation c_p, and construct a link between source nodes s and z and assign the required power p_{sz} to the source node.

Step 7.3) Check all the remaining unmarked nodes in the permutation to see if any other node is already covered by the above transmission between nodes s and z, mark those nodes to avoid the subsequent selections.

Step 7.4) Go through the unmarked node z' in permutation sequentially. Employ BIP algorithm the rules to add node z' to the existing tree and assign a power $p_{s'z'}$ to the corresponding source node (assuming it is s').

Step 7.5) Check all the remaining unmarked nodes in the permutation to see if any other node is covered by the above transmission between nodes s' and z', mark those nodes to avoid the subsequent selections.

Step 7.6) If not end of permutation, go to Step 7.5).

Step 7.7) Calculate P_T by Equation (7.5), and form a broadcast tree $\boldsymbol{T}$ in which all constructed links correspondent the edges in the $\boldsymbol{T}$.

By convention, the fitness value should be a positive value. Since the aim is to minimise the consumption power P_T of a broadcast tree, the fitness value of each chromosome is equivalent to the consumption power. It is worthwhile to rewrite the fitness function as follows

$$f(\boldsymbol{c}_p) = P_T = \sum_{u \in N} \max_v (x_{uv} p_{uv}) \qquad (7.10)$$

where T is the corresponding broadcast tree to the permutation c_p, and the lowest value in (7.9) corresponds to the best chromosome. From the above description, it is evident that the above algorithm constructs the transmission from the source node to the first appeared node compulsorily and excludes the already covered nodes subsequently, but treats the remaining nodes based on greedy rules as adopted in BIP algorithm. This is an important difference from the BIP algorithm, which can produce an interesting improvement in the final performance as illustrated in Section 7.4.1. Note the final optimum result achieved in the proposed GA approach is guaranteed by the population-based GA search.

7.3.4 Genetic Operators

Based on the fitness function defined above, three basic types of genetic operators are required to modify the population: selection, crossover, and mutation. In this subsection, we will present some genetic operators for the proposed GA approach.

Selection is a process used for choosing parent chromosomes to participate in reproduction for the next generation. In order to mitigate the premature problem encountered by the traditional roulette wheel approach, a so-called ranked-based selection scheme is used in this study [88].

Crossover is a crucial operator that combines two or more parent chromosomes to produce new offspring chromosomes. In the proposed GA approach, we examine the combined use of three crossover operators: PMX (partially mapped crossover) [89], PBX (position-based crossover) [90], and OX (ordered crossover) [91]. The details about these crossover techniques are referred to Chapter 6. As will be seen later, the combined use of these crossover operators can lead to an enhanced performance.

It should be noted that we employ a stochastic crossover scheme proposed in [92] and use experimentation to choose the corresponding probabilities, since in this study we adopt a combined crossover scheme, unlike the conventional method. This is discussed in Section 7.4.2.

The mutation operator randomly alters some values in a chromosome and results in entirely new offspring chromosomes. Same as the crossover strategy, a combined mutation scheme is adopted in this study in order to achieve the enhance performance, which incorporates position-based shift mutation operator (PBSM) [93], order-based swap mutation operator (OBSM) [93] and inversion mutation operator (IM) [5]. The details about these mutation techniques are referred to Chapter 6. We determine the mutation probabilities by experimentation, which is discussed in Section 7.4.2. To this end, all genetic operators utilised in the proposed GA approach have been described. We give a lemma as follows

Lemma 7.1: In the proposed GA approach, all generated solutions in every generation are constraints-satisfied broadcast tree.

Proof: Because the proposed GA approach is based on a permutation encoding scheme, the chromosomes produced by all genetic operators in every GA generation are arbitrary permutation of the mobile nodes. For every permutation, step 7.2) and step 7.3) of the deterministic greedy-like algorithm guaranteed that source node must have at least one outgoing link and no incoming links, which satisfies the source-node constraints (7.6) and (7.7). The marking mechanism used in the deterministic greedy-like algorithm assured that the already covered nodes do not involve the sequent selection, which means every node can be covered only once, and thus has exactly one incoming link, i.e. constraints (7.8). Furthermore, it is obvious that every node is included in each permutation, thus each node will be scanned and added to the corresponding tree, which result in a fully-connected tree, i.e. constraints (7.9). Therefore, the resulting solutions in every generation are constraints-satisfied broadcast tree.

7.3.5 Replacement

The so-called incremental replacement and the elitist strategy are adopted in this study [5], [35].

7.3.6 Termination

Termination is the criterion by which the GA decides whether to continue searching or stop the search. Since for the fitness value calculation in Equation (7.10), a combined termination strategy is adopted in this study. The GA will terminate if it had reached the predefined maximum generation G_{max} or it had not improved over the last 30 (empirically determined) successive generations. This strategy cannot only ensure that the GA has enough time to converge, but also avoid excessively high complexity and processing time.

7.4 Results and Discussions

In this section, computer simulation results are presented to demonstrate the various aspects of the proposed GA approach. A summary of the various GA parameters used in the simulation is given in Table 7.1, which is obtained from the preliminary experiments. Before presenting the system broadcast results, it is of interest to examine a case study to illustrate how the proposed GA approach is able to outperform the BIP algorithm.

Table 7.1: Summary of GA parameters used for the MPB simulations.

Parameter	Value/Type
Population size, P	50 (small/middle networks). 100 (larger networks)
Representation	Permutation-encoded
Initialisation	Hybrid (Random + Deterministic)
Generation selection	Rank-based roulette wheel
Crossover operators	PMX;PBX;OBX (30%;50%;20%) in stochastic roulette wheel selection
Crossover probability, p_c	0.6
Mutation operator	PBSM;OBSM;IM (50%;20%;30%) in stochastic roulette wheel selection
Mutation probability, p_m	0.1
Replacement	Incremental + Elitism
Generation	Combined stop criterion with G_{max}=50

7.4.1 Illustration Case Study

A 10-node ad hoc network is considered in this case study as shown in Figure 7.2. Assume the 10th node is the source node and the path-loss exponent α=2. According to BIP algorithm as proposed in [95], the order in which the nodes are added to the broadcast tree is $10 \rightarrow 9$, $10 \rightarrow 6$, $6 \rightarrow 7$, $6 \rightarrow 8$, $6 \rightarrow 5$, $9 \rightarrow 1$, $9 \rightarrow 3$, $9 \rightarrow 4$, $9 \rightarrow 2$, and the resulting broadcast tree is shown in Figure 7.2(a). Note the notation $10 \rightarrow 9$ means that the transmission from node 10 to node 9. In this example, nodes 10, 6, and 9 transmit, while the other nodes, which are leaf nodes, do not. The overall transmitter power is therefore

$$P_T = \max\{d_{10,6}^{\alpha}, d_{10,9}^{\alpha}\} + \max\{d_{6,5}^{\alpha}, d_{6,7}^{\alpha}, d_{6,8}^{\alpha}\} + \max\{d_{9,1}^{\alpha}, d_{9,2}^{\alpha}, d_{9,3}^{\alpha}, d_{9,4}^{\alpha}\}$$

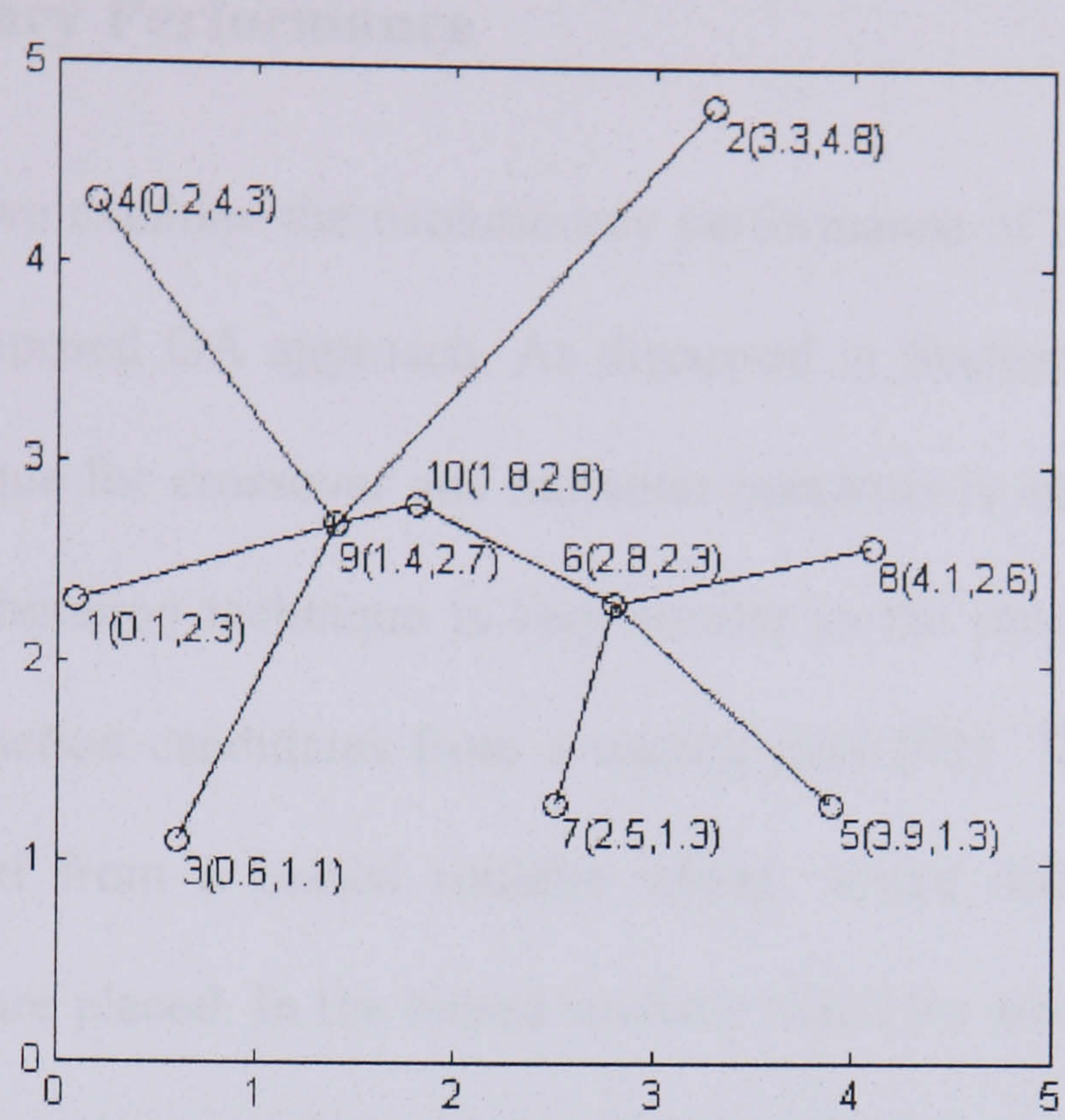
$$= d_{10,6}^{\alpha} + d_{6,5}^{\alpha} + d_{9,2}^{\alpha} = 11.48$$

It should be noted that the corresponding node sequence for the above BIP solution is recorded as {10, 9, 6, 7, 8, 5, 1, 3, 4, 2}, which can be used as the deterministic part of population in the proposed GA approach. For this example, the proposed GA approach is carried out to search the minimum power solution. We choose the population size P=30 and the maximum generation G_{max}=50. After a series of evolutional generations, GA finds an optimal node sequence as {5, 3, 9, 6, 4, 10, 2, 8, 7, 1}, which produces a broadcast tree as 10 → 1, 10 → 2, 10 → 3, 10 → 4, 10 → 5, 10 → 6, 10 → 7, 10 → 8, 10 → 9 as shown in Figure 7.2(b). Since the 5th node which is the longest range node from the source node appeared first in the above sequence, according to the rules of the deterministic greedy-like algorithm in the proposed GA, the source node would construct a link to it directly. Following that, all other nodes in the sequence are already covered by the above link. Therefore, in the final broadcast tree, only source node transmits while all the other nodes are leaf nodes with no power consumption. Hence the corresponding overall transmitter power is

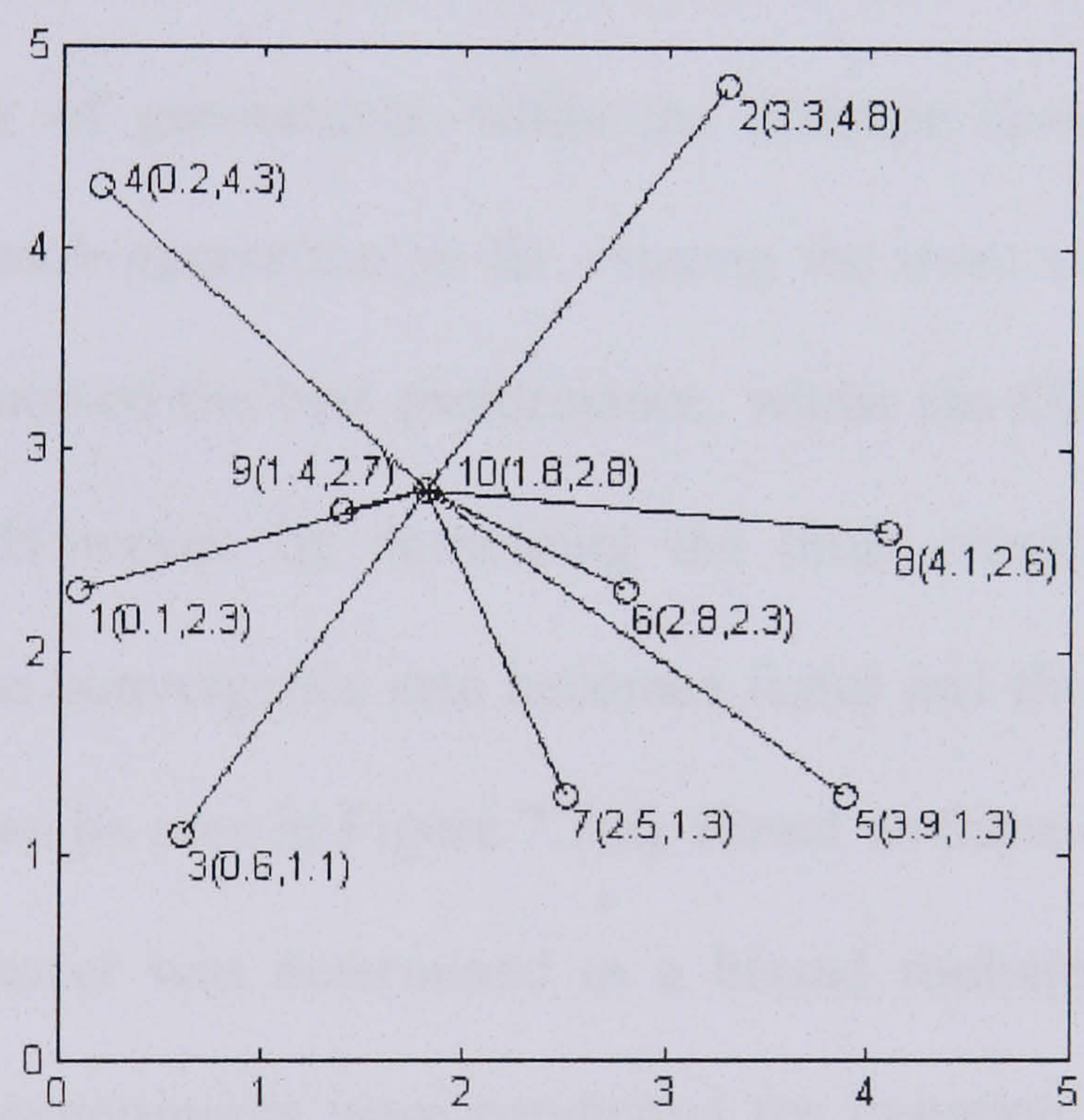
$$P_T = \max\{d_{10,1}^{\alpha}, d_{10,2}^{\alpha}, d_{10,3}^{\alpha}, d_{10,4}^{\alpha}, d_{10,5}^{\alpha}, d_{10,6}^{\alpha}, d_{10,7}^{\alpha}, d_{10,8}^{\alpha}, d_{10,9}^{\alpha}\}$$

$$= d_{10,5}^{\alpha} = 6.66$$

It is clear that the proposed GA approach achieves considerably lower power consumption than the BIP algorithm. Actually, for the considered example, as long as the 5th node (the longest range node) is appeared in the first place in the permutation, the GA approach would produce the same optimum broadcast tree. Therefore, it is quite easy for the GA approach to find such a permutation in its evolutionary search.



(a)

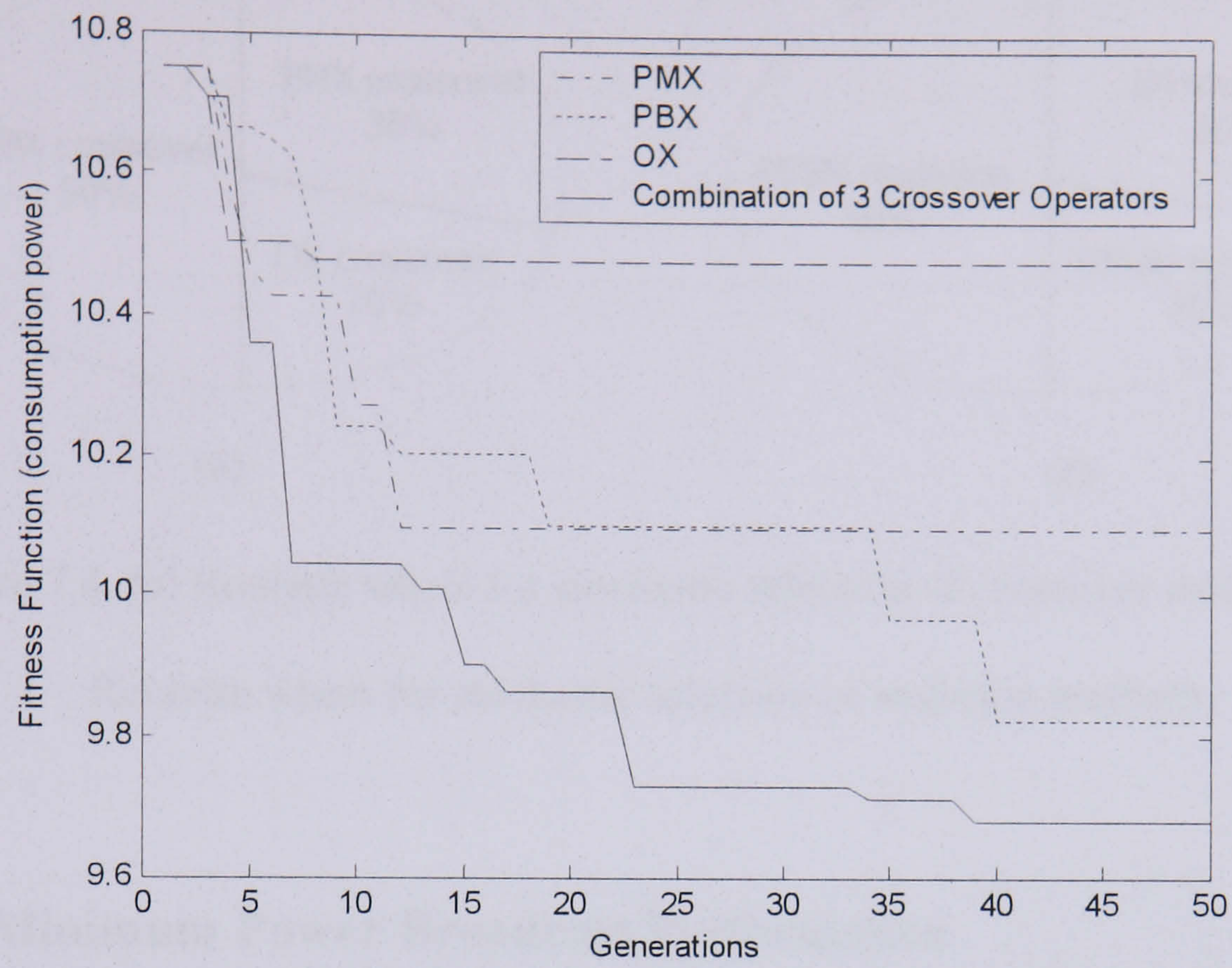


(b)

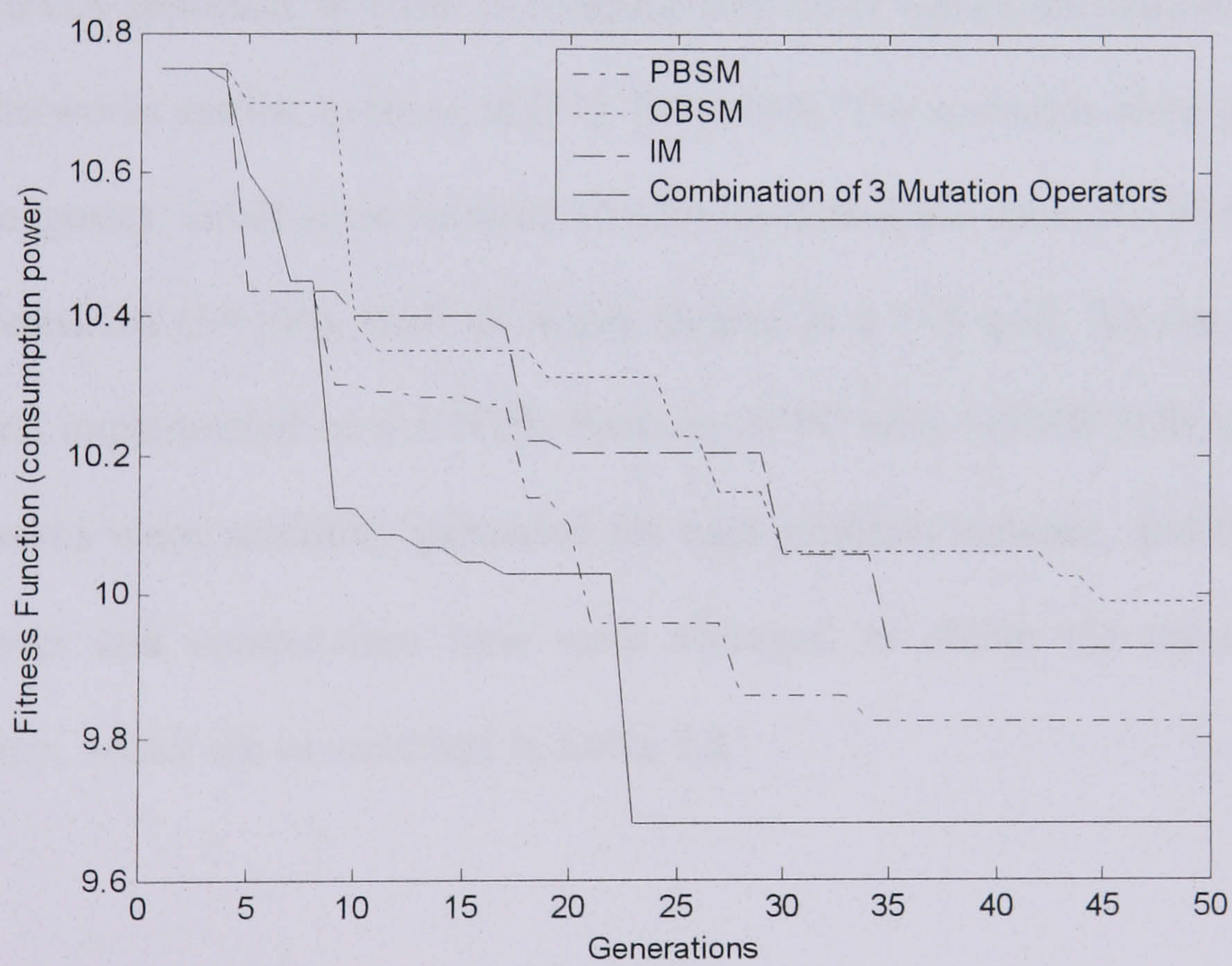
Figure 7.2: Example of tree construction for 10-node network (nodes with numbering and coordinates, α=2). (a) the BIP algorithm (b) the GA approach.

7.4.2 Evolutionary Performance

In this subsection, we examine the evolutionary performance of the different genetic operators in the proposed GA approach. As discussed in Section 7.3.4, a stochastic combination technique for crossover and mutation operators is adopted in this study. The stochastic combination technique is very similar to the process of proportional selection of reproduction candidates from a mating pool [92]. That is, crossover or mutation is selected from a biased roulette wheel, where different crossover or mutation operators are placed. In the biased roulette wheel for stochastic crossover or mutation, each operator has a roulette wheel sector sized according to its performance to find the best solution. Figure 7.3(a) shows the convergence characteristics of each crossover technique for a randomly distributed 50-node network. The abscissa indicates the number of generations, while the ordinate shows the best function evaluation found in each generation so far. Among the three considered crossovers, the PBX crossover showed the best performance, whilst the OX crossover gives the worst performance. However, by combining the three crossover techniques in a stochastic manner, the convergence rate becomes faster and the performance is also further improved as can be seen in Figure 7.3(a). Based on these empirical studies, the weight of each crossover was determined in a biased roulette wheel as shown in Figure 7.4(a). Same experiments were conducted for mutation techniques, and it is found that PBSM mutation is the best one, whilst OBSM mutation is the worst one, and the combined scheme provides even better performance than any individual one as illustrated in Figure 7.3(b). Thus the stochastic mutation is chosen as Figure 7.4(b).



(a)



(b)

Figure 7.3: Evolutionary performance comparison of different genetic operators for a 50-host network. (a) Crossover operators. (b) Mutation operators.

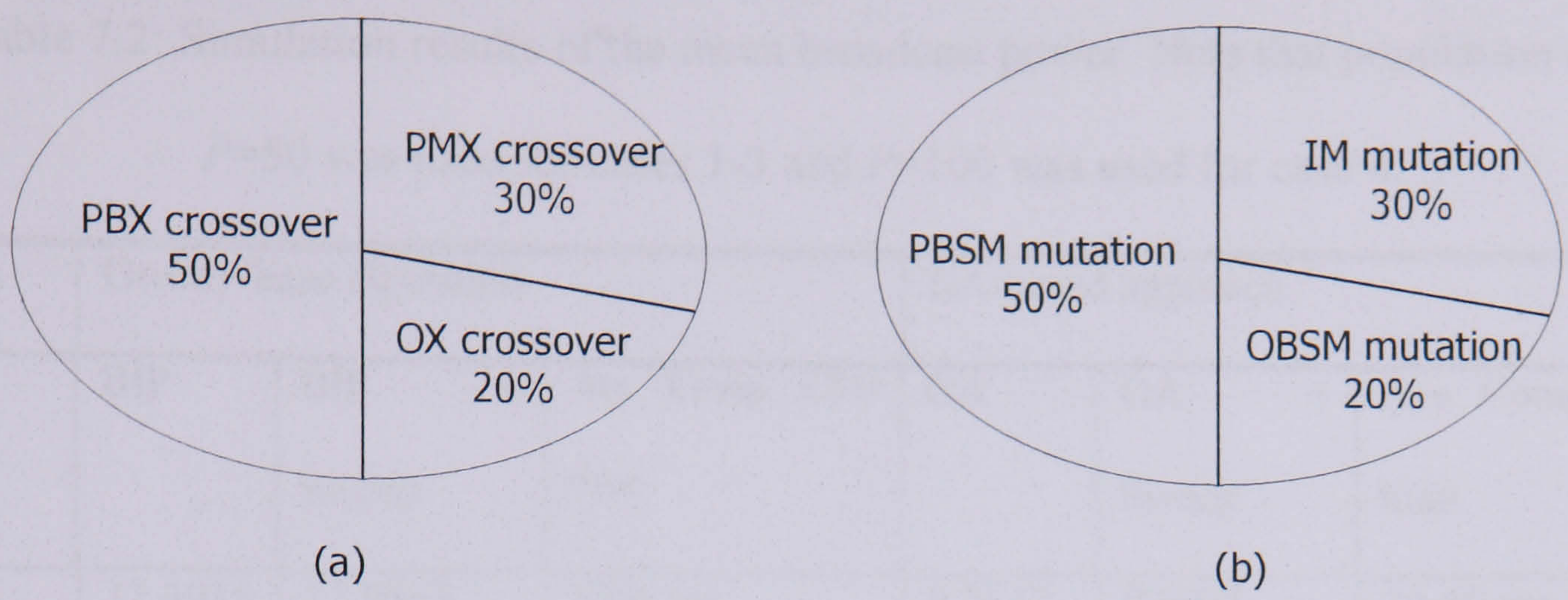


Figure 7.4: (a) Roulette wheel for stochastic selection of crossover methods. (b) Roulette wheel for stochastic selection of mutation methods.

7.4.3 Minimum Power Broadcast Performance

Different scenarios were used to quantify the system power performance of the proposed GA approach. In order to compare with other works, simulations are carried out on networks similar to those in [95], [97]-[100]. The scenarios were grouped into three categories: small-scale network (N=25), middle-scale network (N=50, 75), and larger scenarios (N=100), with all nodes located in a 5×5 grid. All the simulations have been implemented on a 1.2GHz Pentium III PC with 128MB SDRAM memory. 50 networks were randomly generated for each problem instance, and the resulting tree power and computation time were averaged to obtain the mean value for illustration, which are summarised in Table 7.2.

Table 7.2: Simulation results of the mean broadcast power. Note that population size P=50 was used for cases 1-3 and P=100 was used for case 4.

Cases	Greedy-base algorithm			GA-based approach		
	BIP	BIP + Sweep	Ave. Comp. CPU time	GA	GA + Sweep	Ave. Comp. CPU time
N=25	12.4015	12.0047	0.04 sec.	9.9842	9.9031	20.49 sec.
N=50	11.9818	11.7062	0.22 sec.	9.8757	9.8275	46.42 sec.
N=75	11.9075	11.6265	0.66sec.	9.9770	9.9243	82.05 sec.
N=100	11.7485	11.5171	1.47 sec.	10.0401	9.9335	120.06 sec.

Considering different computation requirements, we choose population size P=50 and P=100 for small/middle and large scale ad hoc network, respectively. Note that the 'Sweep' algorithm proposed in [95] is also adopted for the proposed GA approach in order to further improve the attainable system performance. The '*Sweep*' algorithm requires almost negligible computation and can be invoked after the GA terminates. The results show that the GA approach exhibits very good search strength in obtaining the lower power broadcast tree than the BIP algorithm, but at the expense of using more CPU-time. However, we were primarily interested in finding high-quality solutions, and as for the required CPU-time we assume that the computation complexity can accommodated by optimising the codes or by using a powerful computing machine. Therefore, the improvement for the system consumption power can justify our GA approach choice.

7.4.4 Comparison with Other Methods

In this subsection, we compare our approach with other recently reported methods [95], [99]-[100]. Due to simulations running on the different platform, percentages of the improvement instead of the absolute values over BIP algorithm are chosen for comparison, which is summarised in Table 7.3.

Table 7.3: Comparison of improvements of the mean broadcast power over BIP algorithm with other competitive methods.

Cases	Percentage improvement (%) in mean tree power over BIP algorithm					
	BIP + Sweep	r-shrink reported in [99]	SA reported in [100]	GA	SA + Sweep reported in [100]	GA + Sweep
N=25	3.20	17.90	19.90	19.49	20.14	20.15
N=50	2.30	15.17	17.14	17.58	17.93	17.98
N=75	2.36	15.05	15.39	16.21	16.25	16.66
N=100	1.97	14.91	14.31	14.54	15.34	15.45

It is observed that the proposed GA shows very competitive performance for all considered cases. More precisely, the proposed GA approach outperforms the BIP algorithm significantly in both small/medium and lager networks, and is marginally better than the r-shrink and SA methods. Besides, the 'Sweep' algorithm can also be used to eliminate the unnecessary transmissions and further improve the performance of GA approach.

7.5 Concluding Remarks

In this chapter, we showed how GA can be applied to solve minimum power broadcast problem in wireless ad hoc networks. The permutation-encoded GA is employed to search for the optimum mobile node sequence and a deterministic greedy-like algorithm is triggered to obtain the corresponding broadcast tree. By virtue of this complementary process, the proposed approach exhibits great strength in obtaining the broadcast tree with minimum consumption power. Simulation results demonstrate that the proposed GA approach significantly outperforms the BIP algorithm, and is comparable to the best reported results in the literature so far.

In case of the proposed GA, we found that it is essential to adopt modifications to the traditional GA processes, such as elite population initialization, stochastic combination of operators and the combined termination criterion, in order to achieve an enhanced performance. Future research in this area will include studying other performance improvement techniques including problem-specific operators and more efficient generation of the broadcast tree. Additionally, further test problems will be investigated.

8 Conclusions

This dissertation has been an investigation into the application of GA to solve complex optimisation problems arising in wireless communication systems. Most of these problems are NP-complete and involve numerous constraints, which are intractable to solution using analytical approaches or simple deterministic algorithms that cannot terminate in polynomial time. Hence this study has considered GA as an alternative optimisation approach to achieve the research objective which is the application of GA to solve realistic problems in wireless systems. In this chapter, the completed work is summarised and some suggestions are presented for future research directions.

8.1 Summary of Completed Work

The theoretical foundation for the methods adopted in this thesis is presented in Chapter 2 the optimisation problem is mathematically formulated and various available optimisation methods are investigated. Advantages and shortcomings of different methods are assessed and GA is chosen as the main approach due to its superior strength in a wide range of applications.

Chapter 3 investigates the problem of channel allocation in PCS networks using GA when the traffic is possibly non-uniform and co-channel interference must be avoided. The quality measure of network performance is expressed in terms of new call and handoff call blocking probabilities, the weighted sum of which is to be minimised. The system is modelled using a continuous-time Markov process, and the steady-state distribution of the number of channels in use in each cell can be derived as functions of the arrival rates. The optimal channel allocation in each cell, including the number of reserved channels for handoff calls, is searched for by introducing randomness in the searching procedure using GA. The results show that the channel allocation schemes using GA outperforms the fixed channel allocation schemes under different traffic loads. The comparison is performed for their new call blocking probabilities, handoff call blocking probabilities and a weighted sum of the two.

In Chapter 4 the resource allocation scheme associated with a novel access method (PCMA) is investigated. The allocation task is formulated to a constrained optimisation problem, which is proven to be NP-complete. Therefore, the GA approach is developed to in order to obtain near-optimal solutions. Compared with the greedy-based '*min*' algorithm, the proposed GA approach is shown to exhibit excellent search strength, and outperforms the '*min*' algorithm in both considered performance metrics: the probability of overload and the expected duration of overload. This study is significant due to the fact that the research is taken under packet-switching architecture which is the main backbone for future wireless applications.

Chapter 5 is concerned with the multi-user technique in the DS-CDMA system. A GA-based MUD was proposed, which performs joint symbol detection and adaptive estimation of the cut-off parameter for the Huber's M-estimator's objective function, thus eliminates the need for a separate channel estimator, and performs reliably under both Gaussian and non-Gaussian noise. The proposed GA-based MUD has been compared with the RNN. Under the favourable Gaussian noise channel, the RNN and GA-based MUD have similar performance. However, in impulsive noise channels, the GA-based detectors clearly outperform other detectors due to its multi-dimensional search and adaptive outlier rejection capability.

Chapter 6 has considered the so-called scheduling problem in MANET. The problem is widely known as NP-complete and diverse heuristic algorithms including the binary encoded GA were reported to solve this problem. However, many of them experienced difficulty in obtaining the optimum solution. Therefore, a novel permutation-encoded GA scheme is considered in this study, which is shown to be able to reduce the problem search space to a great extent, and to achieve considerably improved performance. Moreover, the GA approach is developed to cooperate with a deterministic greedy-like algorithm in order to obtain the optimum schedules. Simulation results suggested that the proposed GA approach obtained the best results thus far in literature.

Chapter 7 is devoted to the minimum power problem in wireless ad hoc networks. Also, based on the ad hoc network scenario, a different problem (broadcasting) was highlighted, which has many applications in practice. This problem is NP-complete too. The broadcast problem is mathematically formulated as a constrained optimisation problem using a graph representation. The objective is to extract a sub-graph from the original graph such that a broadcast tree with minimum power consumption is constructed. GA approach is developed to cooperate with a deterministic greedy-like algorithm to obtain the minimum power broadcast tree. The results indicate that the proposed GA approach significantly outperforms the greedy-based BIP algorithm and is comparable to the best results obtained by the recent proposed Simulated Annealing method.

8.2 Future Work

Although this research has investigated many optimisation problems in wireless systems, a lot still remains to be done in this area. From all the work in this study, several promising areas for further work have arisen concerning the designing of the problem-specific GA for solving the encountered optimisation problems.

The GA approach usually took much more CPU time compared with other sub-optimal methods, especially when dealing with large-scale problems. Therefore, the efficiency is the most significant issue in the GA design. Since the objective functions sometimes involve intensive computation, such as the OBS and MPB problems studied in Chapter 6 and Chapter 7 respectively, an advanced computing mechanism should be considered for practical use. A parallel implementation of GA so that the fitness of each individual can be computed separately would reduce the computational time significantly.

The encoding scheme of GA approaches has a major impact on the performance of the algorithm, because it can not only severely limit the search space but also determine the subsequent method used to map the encoded chromosomes to the desired solutions. For instance, as for OBS problem, the permutation-based scheme adopted is able to reduce the search space to a great extent compared with the binary-based counterpart. However, the permutation-based method also involved the deterministic greedy-like algorithm to map the chromosome to the required problem solutions. In contrast, the binary-based method used complicated crossover to constrain the chromosome within the valid solutions. Thus, a trade-off must be made. Future work will consider other efficient problem mapping schemes such as tree encoding scheme and problem-specific genetic operator to enhance the performance.

The fitness function is necessary to drive the evolution of the population toward the optimal solutions. As the MUD problem considered in Chapter 5, the fitness function successfully involved the desired signal components and the estimator cut-off parameter, thus, a combined optimisation is achieved and the need for a separate channel estimator is thus eliminated. However, this combination is very simple and future work will consider other cost functions, which would hopefully produce more promising results

The MUA problem studied in Chapter 4 has been proved to be NP-complete. Therefore, the GA approach became a first choice for the solutions and also produced very encouraging results. However, if the problem is formulated to a linear programming model when the resource (*UB*) is allowed to be real value instead of integer value, other methods such as Mixed Integer Programming can possibly solve the problem in polynomial time. This could be considered in the future work.

Abbreviations and Acronyms

ACS	Ant Colony System
AI	Artificial Intelligence
ARCS	Adaptive Reservation Channel Scheme
ASA	Adaptive Simulated Annealing
BIP	Broadcast Incremental Power
BPSK	Binary Phase Shift Keying
CAS	Channel Allocation Scheme
CDMA	Code Division Multiple Access
CDPA	Capture Division Packet Access
CP	Compact Pattern
FCA	Fixed Channel Allocation
GA	Genetic Algorithm
GCS	Guard Channel Schemes
GoS	Grade of Service
HQS	Handoff Queuing Scheme
IM	Inversion Mutation
LDD	Linear Decorrelating Detector

MAI	Multiple Access Interference
MANET	Mobile Ad Hoc Network
MDD	*M*-Decorrelating Detector
MF	Matched Filter
MFA	Mean Field Annealing
ML	Maximum Likelihood
MPB	Minimum Power Broadcast
MSC	Minimum Set Cover
MUA	Minimum Unserviced Allocation
MUD	Multiuser Detection
NN	Neural Network
NP	Nondeterministic Polynomial
NPS	non-prioritized scheme
OBS	Optimum Broadcast Scheduling
OBSM	Order-based Swap Mutation
OX	Ordered Crossover
PBSM	Position-based Shift Mutation
PBX	Position-based Crossover
PCS	Personal Communication Services

PCMA	Plane Cover Multiple Access
PMX	Partially Mapped Crossover
RCS	Reserved Channel Scheme
RCS1	Reserved Fixed One Channel Scheme
RCS4	Reserved Fixed Four Channel Scheme
RNN	Recurrent Neural Network
RP	Reuse Partitioning
SA	Simulated Annealing
SVC	Sequential Vertex Colouring
TS	Tabu Search
TDMA	Time Division Multiple Access
TSP	Travelling Salesman Problem
VFSR	Very Fast Simulated Re-annealing

Bibliography

[1] Prasad, R. and Ruggieri, M.: "Future strategy for the new millenium wireless world", *Wireless Personal Communications*, 2001, **17**, pp. 149–153

[2] Aazhang, B. and Cavallaro, J.: "Multitier wireless communications", *Wireless Personal Communications*, 2001, **17**, pp. 323–330

[3] Garey, M.D. and Johnson, D.S.: "Computers and Intractability: A Guide to the Theory of NP-Completeness" (Freeman, San Francisco, 1979)

[4] Holland, J.H.: "Adaptation in Natural and Artificial Systems: an introductory analysis with applications to biology, control, and artificial intelligence" (Michigan Press, 2nd edition, 1975)

[5] Goldberg, D.E.: "Genetic Algorithms in Search, Optimsation, and Machine Learning" (Addison-Wesley, Reading, MA, 1989)

[6] Man, K.F., Tang, K.S. and Kwong, S.: "Genetic algorithms: Concepts and applications", *IEEE Trans. Industrial Electronics*, 1996, **43**(5), pp. 519-534

[7] Bi, Q., Zysman, G.I. and Menkes, H.: "Wireless mobile Communications at the start of the 21st century," *IEEE Communications Magazine*, 2001, **39**(1), pp. 110-116

[8] Adachi, F.: "Wireless past and future – Evolving mobile communications Systems," *IEICE Transactions on Fundamentals*, 2001, **E84**(1), pp.55-60

[9] Pintér, J.D.: "Global Optimsation in Action" (Kluwer Academic Publishers, London, 1996)

[10] Fletcher, R.: "Practical Methods of Optimsation" (Wiley, New York, 1987)

[11] Metropolis, N., Rosenbluth, A.W., Rosenbluth, M. N., Teller, A.H. and Teller, E., "Equations of State Calculations by Fast Computing Machines," Journal of Chemical Physics, 1958, **21**, pp. 1087- 1092

[12] Kirkpatrick, S., Gerlatt, C. D. Jr., and Vecchi, M.P.: "Optimsation by Simulated Annealing," Science, 1983, **220**, pp. 671-680

[13] Ingber, L.: "Very Fast Simulated Reannealing," Mathematical Computational Modeling, 1989, **12**, pp. 967-973.

[14] Glover, F.: "Future Paths for Integer Programming and Links to Artificial Intelligence," *Computers and Operations Research*, 1986, **13**, pp. 533-549

[15] Haykin, S.: "Neural Networks: A Comprehensive Foundation" (Prentice Hall International Inc., 1998)

[16] Horowitz, E., Sahni, S., and Rajasekar, S.: "Fundamentals of Computer Algorithms" (Galgotia Publications, 1998)

[17] Mitchell, M.: "An Introduction to Genetic Algorithms" (MIT Press, 1996)

[18] Vivek D.P., William M.P., William T.: "A Survey of Optimsation Techniques Being Used in the Field," in Proc. of the Third International Meeting on Research in Logistics (IMRL 2000). Quebec, Canada, May, 2000

[19] Sidi, M., Starobinski, D.: "New call blocking versus handoff call blocking in cellular networks," *Wireless Networks*, 1997, **3**(1), pp. 15-27

[20] Lipper, E.H., Rumsewicz, M.P.: "Teletraffic considerations for widespread deployment of PCS," *IEEE Network*, 1994, **8**(5), pp. 40-49

[21] Hou, J.K. and Fang, Y.G.: "Mobility-based call admission control schemes for wireless mobile networks," *Wireless Communications and Mobile Computing*, 2001, **1**, pp. 269-282

[22] Ramanathan, P., Sivalingam, K.M., Agrawal, P. and Kishore, S.: "Dynamic resource allocation schemes during handoff for mobile multimedia wireless networks," *IEEE Journal on Selected Areas in Communications*, 1999, 17(7), pp. 1270-1283.

[23] Katzela, I. and Nagshineh, M: "Channel assignment schemes for cellular mobile telecommunication systems: A comprehensive survey," *IEEE Personal Communications*, 1996, **3**, pp. 10-31

[24] Huberman R., et al.: "Multihour dimensioning for a dynamically routed network," in ITC'11, Kyoto, Japan, 1985, pp. 730–736

[25] Boumerdassi S. and Beylot A.L.: "Adaptive channel allocation for wireless PCN," *Mobile Networks and Applications*, 1999, **4**, pp. 111-116

[26] Ortigoza-Guerrero, L. and Aghvami, A.H.: "A prioritized handoff dynamic channel allocation strategy for PCS," *IEEE Transactions on Vehicular Technology*, 1999, **48**(4), pp. 1203-1215

[27] Hu L.R. and Rappaport S.S.: "Personal communication systems using multiple hierarchical cellular overlays," *IEEE Journal on Selected Areas Communications*, 1995, 13, pp. 406-415

[28] Zeng H.L., Fang Y.G., and Chlamtac I.: "Call blocking performance study for PCS networks under more realistic mobility assumptions," *Telecommunication Systems*, 2002, 19(2), pp. 125-146

[29] Yeung, K.L. and Yum, T.S.: "The optimisation of nominal channel allocation in cellular mobile systems," in Proc. of IEEE ICC, Geneva, 1993, pp. 915–919.

[30] Yeung, K.L. and Yum, T.S.: "Compact pattern based dynamic channel assignment for cellular mobile systems," *IEEE Transactions on Vehicular Technology*, 1994, **43**, pp. 892-896

[31] Zhang, M. and Yum, T.P.: "The non-uniform compact pattern allocation algorithm for cellular mobile systems", *IEEE Transactions on Vehicular Technology*, 1991, **40**, pp. 387-391

[32] Mathar, R. and Mattfeldt, J.: "Channel assignment in cellular radio networks," *IEEE Transactions on Vehicular Technology*, 1993, 42, pp. 647-656

[33] Haupt, R.I. and Haupt, S.E.: "Practical Genetic Algorithms" (John Wiley & Sons Ltd., 1998)

[34] Michalewicz, Z.: "Genetic Algorithms + Data Structures = Evolution Programs" (Springer-Verlag Telos, Berlin, 1992)

[35] Davis, L.: "Handbook of Genetic Algorithms" (Van Nostrand Reinhold, New York, 1991)

[36] Lin, Y.B., Mohan, S., and Noerpel, A.: "The Sub-rating channel assignment strategy for PCS hand-offs," *IEEE Transactions on Vehicular Technology*, 1996, 45, pp. 122-130

[37] Lin, Y.B.: "Reducing location update cost in a PCS," *IEEE/ACM Trans. Networking*, 1997, 5, pp. 25-33

[38] Carothers, C.D., Fujimoto, R.M., and Lin, Y.B.: "A case study in simulating PCS networks using time warp," in Proc. of 9th Workshop on Parallel and Distributed Simulation, New York, 1995, pp. 87-94

[39] Ingber, L.: "Adaptive simulated annealing (ASA): Lessons learned," Control and Cybernetics, 1996, 25(1), pp. 33-54

[40] Ingber, L.: "Adaptive simulated annealing (ASA) software," 1993-1998, http://www.ingber.co.com/#ASA-CODE.

[41] Acampora, A.: "Wireless ATM: A perspective on issues and prospects," *IEEE Personal Communications Msg.*, 1996, **3**, pp. 8-17

[42] Schwartz, M.: "Network management and control issues in multimedia wireless networks," *IEEE Personal Communications Msg.*, 1995, **2**, pp. 8-16

[43] Borgonovo, F., Zorzi, M., Fratta, L., Trecordi, V., and Bianchi, G.: "Capture-Division Packet Access for Wireless Personal Communications," *IEEE Journal on Selected Areas in Communications*, 1996, 14(4), pp. 609-622

[44] Zande, J. and Frodigh, J.: "Capacity Allocation and Channel Assignment in Cellular Radio Systems using Reuse Partitioning," *Electronic Letters*, 1992, 28, pp. 438-440

[45] Blair, P. and Polyzos, G.C.: "Packet Cellular System Optimsations for Non-Uniform Traffic," in Proc. of the IEEE International Conference on Communications (ICC), Helsinki, Finland, June 2001

[46] Blair, P., Polyzos, G.D. and Zorzi, M.: "Plane Cover Multiple Access: a new approach to maximizing cellular system capacity," IEEE *Journal on Selected Areas in Communications*, 2001, **19**, pp. 2131-2141

[47] Acampora, A.S. and Naghshineh, M.: "An Architecture and Methodology for Mobile-Executed Handoff in Cellular ATM Networks", *IEEE Journal on Selected Areas in Communications*, 1994, **12**, pp. 1365-1374

[48] Verdu, S.: "Multiuser Detection" (Cambridge Univ. Press, 1998)

[49] Blackard, K.L., Rappaport, T.S., and Bostian, C.W.: "Measurements and models of radio-frequency impulsive noise for indoor wireless communications," *IEEE J. Select. Areas Commun.*, 1993, **11**, pp. 991-1001

[50] Middleton, D.: "Non-Gaussian noise models in signal processing for telecommunications: New methods and results for class A and class B noise models," *IEEE Trans. Inform. Theory*, 1999, **45**, pp. 1129-1149

[51] Delic, H. and Hocanýn, A.: "Robust detection in DS/CDMA," *IEEE Trans. Veh. Technol.*, 2002, **51**(1), pp. 155-170

[52] Juntti, M.J., Schlosser, T., and Lilleberg, L.O.: "Genetic algorithm for multiuser detection in synchronous CDMA," in Proc. IEEE Int. Symp. Inform. Theory, Ulm, Germany, 1997, pp. 492

[53] Wang, X.F., Lu, W.S., and Antoniou, A.: "A genetic algorithm-based multiuser detector for multiple-access communications," in Proc. IEEE Int. Symp. Circuits and Systems, Monterey, CA, 1998, Vol. 4, pp. 534-537

[54] Ergun, C. and Hacioglu, K.: "Multiuser detection using a genetic algorithm in CDMA communications systems," *IEEE Trans. Commun.*, 2000, **48**, pp. 1374-1383

[55] Abedi, S. and Tafazolli, R.: "Genetic multiuser receiver for code division multiple access communications," *Electron. Lett.*, 2000, **36**, pp. 1957-1958

[56] Abedi, S. and Tafazolli, R.: "GMD: A new CDMA multiuser detection technique using an evolutionary algorithm," *IEE Proc.-Commun.*, 2001, **148**, pp. 393-399

[57] Yen, K. and Hanzo, L.: "Genetic algorithm assisted joint multiuser symbol detection and fading channel estimation for synchronous CDMA systems," *IEEE J. Sel. Areas Commun.*, 2001, **19**, pp. 985-998

[58] Huber, P.J.: "Robust Statistics" (Wiley, 1981)

[59] Lupas, R. and Verdu, S.: "Linear multiuser detectors for synchronous code-division multiple-access channels," *IEEE Trans. Inform. Theory*, 1989, **35**, pp. 123-136

[60] Wang, X. and Poor, H.V.: "Robust multiuser detection in non-Gaussian channels," *IEEE Trans. Signal Process.*, 1999, **47**, pp. 289-305

[61] Chuah, T.C., Sharif, B.S., and Hinton, O.R.: "A neural network approach to DS-CDMA multiuser detection in impulsive channels," in Proc. Int. Conf. Inform., Commun., Signal Processing, Singapore, Oct. 2001

[62] Nikias, C.L. and Shao, M.: "Signal Processing with Alpha-Stable Distributions and Applications" (Wiley, 1995)

[63] Byrd, R.H.: "Algorithms for robust regression," in Nonlinear Optimisation, (Academic Press, 1982), pp. 79-84

[64] Clark, D.I. and Osborne, M.R.: "Finite algorithms for Huber M-estimator," SIAM J. Sci. Stat. Comput., 1986, 7, pp. 72-85

[65] Dutter, R.: "Numerical solution of robust regression problems: Computational aspects, a comparison," *J. Statist. Comput. Simul.*, 1977, **5**, pp. 207-238

[66] Shanno, D.F. and Rocke, D.M.: "Numerical methods for robust regression: Linear models," *SIAM J. Sci. Stat. Comput.*, 1986, 7, pp. 86-97

[67] Cichocki, A. and Unbehauen, R.: "Neural networks for solving systems of linear equations and related problems," *IEEE Trans. Circuits Syst. I-Fundam. Theor. Appl.*, 1992, **39**, pp. 124-138

[68] Booker, L.B.: "Improving search in genetic algorithms", in Genetic Algorithms and Simulated Annealing, (Morgan Kaufmann Publishers, 1987), pp. 61-73

[69] De Jong, K.A.: "Learning with genetic algorithms: An overview," *Maching Learning*, 1988, **3**, pp. 121-138

[70] Janikow, C.Z. and Michalewicz, Z.: "An experiment comparison of binary and floating point representation in genetic algorithms," in Proc. 4th Int. Conf. Genetic Algorithms, 1991, pp. 31-36

[71] Wright, A.H.: "Genetic algorithms for real parameter optimisation," in Foundations of Genetic Algorithms, Rawlins, G.J.E., Ed., (Morgan Kaufmann Publishers, 1991), pp. 205-218

[72] Thomas, G.M., Gerth, R., Velasco, T., and Rabelo, L.C.: "Using real-coded genetic algorithms for Weibull parameter-estimation," *Comput. Ind. Eng.*, 1995, **29**, pp. 377-381

[73] Qing, A. and Lee, C.K.: "Microwave imaging of a perfectly conducting cylinder using a real-coded genetic algorithm," *IEE Proc.-Microw. Antennas Propag.*, 1999, **146**, pp. 421-425

[74] Yan, W. and Zhu, Z.: "Maximum likelihood DOA estimation by real-valued genetic algorithm," in Proc. IEEE Nat. Aerospace and Electronics Conf., Dayton, OH, 2000, Vol. 1, pp. 633-636

[75] Chuah, T.C., Sharif, B.S., and Hinton, O.R.: "Robust adaptive spread-spectrum receiver with neural-net preprocessing in non-Gaussian noise," *IEEE Trans. Neural Networks*, 2001, **12**, pp. 546-558

[76]Kleinrock, L. and Silvester, J.: "On the behaviour of packet radio networks," in Proc. IEEE, 1987, **75**, pp. 156-167

[77] Ephremides, A. and Truong, T.V.: "Scheduling broadcasts in multihop radio networks," *IEEE Trans. Commun.*, 1990, **38**, pp. 456–460

[78] Funabiki, N. and Takefuji, Y.: "A parallel algorithm for broadcast scheduling problems in packet radio networks," *IEEE Trans. Commun.*, 1993, **41**, pp. 828-831

[79] Cidon, I. and Sidi, M.: "Distributed assignment algorithms for multihop packet radio network," *IEEE Trans. Comput.*, 1989, **38**, pp, 1353-1361

[80] Wang, G. and Ansari, N.: "'A neural network approach to broadcast scheduling in multihop radio networks," in Proc. Int. Conf. Neural Networks, pp. 4699-4704, Orlando, USA, 1994

[81] Wang, G. and Ansari, N.: "Optimal broadcast scheduling in packet radio networks using mean field annealing," *IEEE J. Select. Areas Commun.*, 1997, **15**, pp. 250-260

[82] Yeo, H.L. and Kim, S.: "An efficient broadcast scheduling algorithm for TDMA ad-hoc networks," *Computers & Operations Research*, 2002, **29**, pp. 1793-1806

[83] Chakraborty, G.: "Genetic algorithm to solve optimum TDMA transmission schedule in broadcast packet radio networks," *IEEE Trans. Commun.*, 2004, **52**, pp. 765-777

[84] Whitley, D., Starkweather, T., and Fuquay, D.: "Scheduling problems and traveling salesmen: The genetic edge recombination operator," in Proc. of the 3rd Int. Conf. on Genetic Algorithms, pp. 133-140, Morgan Kaufmann, 1989

[85] Wang, L. and Zheng, D.: "An effective hybrid optimisation strategy for job-shop scheduling problems," *Computers & Operations Research*, 2001, **28**, pp. 585-596

[86] Iyer, S.K. and Saxena, B.: "Improved genetic algorithm for the permutation flowshop scheduling problem," *Computers & Operations Research*, 2004, **31**, pp. 593-606

[87] Fox, B.R. and McMahon, M.B.: "Genetic operators for sequencing problems," Foundations of Genetic Algorithms (Morgan Kaufmann, San Mateo, 1991), pp. 284–300

[88] Wang, L., Siegel, H.J., Roychowdhury V.P., and Maciejewski A.A.: "Task Matching and Scheduling in Heterogeneous Computing Environments Using a Genetic-Algorithm-Based Approach," *J. of Parallel and Distributed Comput.*, 1997, **47**, pp. 8-22

[89] Goldberg, D.E. and Lingle, J.R.: "Alleles, loci and the travelling salesman problem," in Proc. of an Int. Conf. on Genetic Algorithms and their applications, Carnegie Mellon publishers, 1985, pp. 154-159

[90] Starkweather, T., McDaniel, S., Mathias, K., Whitley, D. and Whitley, C.: "A comparison of genetic sequencing operators," in Proc. of 4th Int. Conf. on Genetic Algorithms, Morgan Kaufmann, pp. 69–76, San Mateo, CA, 1991

[91] Davis, L.: "Job Shop Scheduling and Genetic Algorithms," in Proc. of 1st Int. Conf. on Genetic Algorithms and their Applications, Lawrence Erlbaum Associates, pp. 136-140, Hillsdale, NJ, 1985

[92] Park, Y.M., Park, J.B., and Won, J.R.: "A hybrid genetic algorithm/dynamic programming approach to optimal long-term generation expansion planning," Electrical Power & Energy Systems, pp. 295-303, 1998

[93] Murata, T. and Ishibuchi, H.: "Performance evaluation of genetic algorithms for flow shop scheduling problems," in Proc. of the 1st IEEE Conf. on Evolutionary Computation, 1994, 2, pp. 812-817

[94]Ni, S.Y., Tseng, Y.C., Chen, Y.S., and Sheu, J.P.: "The Broadcast Storm Problem in a Wireless Mobile Ad Hoc Network," in Proc. of the International Conference on Mobile Computing and Networking, pp. 151- 162, Aug. 1999

[95] Wieselthier, J., Nguyen, G. and Ephremides, A.: "On the construction of energy-efficient broadcast and multicast trees in wireless networks," in Proc. of the IEEE Infocom 2000 Conference, Tel-Aviv, Israel, Mar. 2000, pp. 585-594

[96] Cagalj, M., Hubaux, J.P. and Enz, C.: "Minimum-energy broadcast in all-wireless networks: NP-completeness and distribution issues," in Proc. of the Mobicom 2002 Conference, Atlanta, GA, Sep. 2002

[97]Marks, R.J., Das, A.K. El-Sharkawi, M., Arabshahi, P. and Gray, A.: "Minimum power broadcast trees for wireless networks: optimizing using the viability lemma," in Proc. of the IEEE International Symposium on Circuits & Systems, Scottsdale, AZ, May 2002

[98]Das, A.K., Marks, R.J., El-Sharkawi, M., Arabshahi, P. and Gray, A.: "The minimum power broadcast problem in wireless networks: an ant colony system approach," in Proc. of the IEEE CAS Workshop on Wireless Communications and Networking, Pasadena, CA, Sep. 2002

[99]Das, A.K., Marks, R.J., El-Sharkawi, M., Arabshahi, P. and Gray, A.: "r-shrink: A heuristic for improving minimum power broadcast trees in wireless networks," in Proc. of the IEEE Globecom 2003 Conference, San Francisco, CA, Dec. 2003

[100] Montemanni, R., Gambardella, L.M. and Das, A.K.: "The minimum power broadcast problem in wireless networks: a simulated annealing approach," submitted for publication.

[101] Guo, S. and Yang, O.: "Optimal Tree Construction for Minimum-Energy Multicast in Wireless Ad Hoc Networks," in 22nd Biennial Symposium on Communications, Ontario, Canada, May 2004