Approximate Algorithms for the Global Planning Problem of UMTS Networks

by

Shangyun Liu, B.Eng.

A thesis submitted to the Faculty of Graduate Studies and Research
in partial fulfillment of the requirements for the degree of
Master of Applied Science in Electrical Engineering

The Ottawa-Carleton Institute for
Electrical and Computer Engineering (OCIECE)

Department of Systems and Computer Engineering
Carleton University
Ottawa, Ontario, Canada
April 2009

The Undersigned Recommend to
the Faculty of Graduate Studies and Research
Acceptance of Dissertation

Approximate Algorithms for the Global Planning Problem of UMTS Networks

Submitted by Shangyun Liu, B.Eng.
In partial fulfillment of the requirements for the degree of
Master of Applied Science in Electrical Engineering

Marc St-Hilaire, Thesis Supervisor

Chair, Department of Systems and Computer Engineering

Carleton University
2009

Abstract

In this thesis, a detailed and comprehensive study is presented on the Universal Mobile Telecommunications System (UMTS) network planning problem. This problem has been shown to be NP-hard. Therefore, approximate algorithms are necessary to build planning tools. Three planning tools, based respectively on genetic algorithm, simulated annealing and a novel cooperative method, are designed and implemented to solve the global planning problem of UMTS networks.

Using the optimal solutions as references, numerical results are compared amongst the proposed planning tools and a previously designed tool based on tabu search. The cooperative method shows its superiority over the three other planning tools with 90 percent confidence that the true mean solution gap from the optima is within the interval of [0.01%, 0.33%]. Moreover, this solution closeness to optima is not necessarily accompanied with long computation time. These observations make the cooperative method more appropriate for global planning of UMTS networks.

Acknowledgments

First I would like to thank Prof. Marc St-Hilaire for giving me the opportunity that led me to this journey. His knowledge, help and encouragement are the key in the accomplishment of my study and the present of this thesis. My eternal thanks, appreciation and respect go to him for not letting me give up when I was going through the toughest moments of my master study.

I would like to express my deep gratitude to Prof. Halim Yanikomeroglu for the encouragement and advices, helping me focus on the study. I also want to express my sincere appreciation to Dr. Jim Yan. His support and trust on my effort give me the courage to keep pursuing my degree. My truly thanks also goes to Ms. Anna Lee for the administrative help.

Particularly I wish to thank my group mate, Mr. Mohammd Reza Pasandideh, for the great help of finding the useful research related materials. I also would like to greatly thank all professors and friends for all courses and research related discussions and help in the related areas: Prof. Amir H. Banihashemi, Mr. Hongqing Zeng, Mr. Baohua Zhang, Mr. Jiesheng Zhu, Mr. Liang Xu, Ms. Jie Xiao, Mr. Yifan Zhao, Mr. Benjamin Feng and Ms. Wen Dai from the Department of Systems and Computer Engineering, Mr. Qi Liu from ARS Lab, Carleton University; Mr. Qing Wang from School of Information Technology and Engineering, University of Ottawa.

Last, but certainly not least, I would like to thank my husband, Gang, and my son, Victor, for their tremendous support and understanding during my whole study period.

Table of Contents

List of Figures

List of Tables

List of Appendixes

List of Acronyms

List of Acronyms

Acronym	Explanation
1G	The First-Generation
1X	One-point Crossover
1xRTT	1x (single-carrier) Radio Transmission Technology
2G	The Second-Generation
2.5G	A transition step between 2G and 3G network
3G	The Third-Generation
3.5G	A transition step between 3G and 4G network
4G	The Fourth-Generation
AMPS	Advanced Mobile Phone System
BSC	Base Station Controller
CAT	Combination Algorithm for Total Optimization
CDMA	Code Division Multiple Access
CN	Core Network
CPICH	Common Pilot Channel
CS	Circuit-Switched

ECPT	European Conference of Postal and Telecommunications Administrations
EM	Electromagnetic
ETSI	European Telecommunications Standards Institute
FDMA	Frequency Division Multiple Access
GA	Genetic Algorithm
GE	Gigabit Ethernet
GGSN	Gateway GPRS Support Node
GMSC	Gateway MSC
GPRS	General Packet Radio Service
GSM	Global System for Mobile Communications
HLR	Home Location Register
HSDPA	High Speed Downlink Packet Access
HSPA+	Evolved High Speed Packet Access
HSUPA	High-Speed Uplink Packet Access
ILP	Integer Linear Programming
IP	Integer Programming
IS-95	Interim Standard 95
LP	Linear Programming
LS	Local Search
MGw	Media Gateway
MSC	Mobile Switching Centre
NMT	Nordic Mobile Telephone
node B	node for Broadband access
PS	Packet-Switched
PSTN	Public Switched Telephone Network

RAN	Radio Access Network
RNC	Radio Network Controller
ROI	Return On Investment
SA	Simulated Annealing
SAL	Simulated Allocation
SGSN	Serving GPRS Support Node
SHO	Soft-Handover
SIR	Signal-to-Interference Ratio
SMS	Short Message Service
TDMA	Time Division Multiple Access
TL	Time Limit
TN	Traffic Node
TP	Test Point
TS	Tabu Search
TSSA	Two-stage Simulated Annealing
UMTS	Universal Mobile Telecommunications System
UTRAN	UMTS Terrestrial Radio Access Network
UX	Uniform Crossover
VLR	Visitor Location Register
W-CDMA	Wideband Code Division Multiple Access

Chapter 1

Introduction

Nowadays, the Universal Mobile Telecommunications System (UMTS) takes a very important role in the wireless communication market. To offer a high-quality network and stay ahead of the competition, network operators need to invest a large portion of their budget in their network infrastructures. Network planning is then the key to reach a delicate balance between network investment and performance in order to maximize the return on investment (ROI).

The primary task of the overall network planning process is the topology planning, which describes the network infrastructure and the required initial investment. Since the planning of UMTS networks involves many tunable variables, efficient planning tools are essential for a successful network planning.

In this thesis, we are interested in the global planning of UMTS networks, focusing on the development of efficient automatic planning tools based on approximate algorithms for the deployment of new UMTS networks. This first chapter starts by presenting some background information that will be useful for the remaining chapters. Then, the problem statement is exposed followed by the research objectives and the proposed methodology. Finally, the main contributions of this thesis are outlined and we conclude with an overview of the remaining chapters.

1.1 Background

In this section, we will introduce some concepts that will be useful to better understand the remaining chapters. We will first describe the cellular network evolution and the network planning process. Then, different planning techniques will be exposed.

1.1.1 Evolution of Cellular Networks

The cellular network industry has evolved phenomenally in the past few decades: transmission technologies changed from analog to digital; various data services gained a rapid growth to compliment the original single voice service; data transmission speed rose tremendously; and network coverage increased from being national to being international. People can now get regular phone and multimedia services without the constraint of wires from landline networks. With these changes, cellular network operators are now facing the challenges of the rapid growth of regular voice services, as well as the fast expansion of wireless internet based multimedia services. This pushes fast evolution of technologies, as well as the demand for more efficient networks.

As cellular networks evolve, the term ‘generation’ is used to differentiate significant technology improvements. In fact, we saw the first generation (1G) of cellular networks in the early 80’s followed by the second generation (2G) in the early 90’s. A few years later, the third generation (3G) was launched. To date, researchers are already focusing on the fourth generation (4G).

1.1.1.1 1G Cellular Networks

1G networks, started in the 1980s, brought out the boom of the mobile communication development. The cell technology was used in 1G networks to provide radio signal coverage and to enable users to move from one cell to another. Because of this, 1G networks were also called cellular networks. With analog transmission techniques, 1G networks could only provide limited services, such as voice and voice related services. Standards like Nordic Mobile Telephone (NMT) used in Nordic countries and Eastern Europe and Advanced Mobile Phone System (AMPS) used in the United States are examples of 1G standards. User mobility was very limited due to standard incompatibility. In fact, there was no concept of worldwide wireless communications, nor a coordination of worldwide technical standards. Mobile users could not roam like users do nowadays. At the same time, in standards like AMPS, each phone call occupied a separate radio frequency during the call even when there was no conversation in process. This caused a waste of the spectrum utilization, especially with the rapid growth of mobile users.

1.1.1.2 2G Cellular Networks

The increasing needs of the mobility in wireless communication require more network compatibility. 2G networks, using digital signal transmission technologies, are built to realize global compatibility with better services. Compression and multiplexing technologies are applied on the digital signals to enhance spectrum utilization efficiency and increase the network capacity. In addition to voice and voice related services, 2G networks also provide mobile users with data services, such as short message service (SMS).

Based on signal multiplexing techniques, 2G standards are divided into two groups: Time Division Multiple Access (TDMA) based standards, such as Global System for Mobile Communications (GSM), and Code Division Multiple Access (CDMA) based standards, like CDMA. These digital signal transmission technologies greatly increase the network capacity with the same available spectrum as in 1G networks.

The GSM standard was originally created by European Conference of Postal and Telecommunications Administrations (ECPT) in 1982. Later, GSM responsibility was transferred to the European Telecommunications Standards Institute (ETSI). GSM networks were commercially launched in 1991, operating at either the 900MHz or 1800MHz spectrum. It is the most popular standard of mobile phone services, accounting for the major part of the global wireless communication market. The popularity of the GSM standard also makes the global roaming feasible between different mobile network operators with roaming agreements. The CDMA standard, which is the competitor of GSM, was pioneered by Qualcomm. It is also called CDMA IS-95 (Interim Standard 95).

2G networks provide non-differentiated voice and data services in the circuit switched manner, which delivers excellent voice services and low transmission rate data services. However, 2G networks still do not fulfill the standard unification globally.

1.1.1.3 2.5G Cellular Networks

2.5G networks are a transition step between 2G and 3G networks. They enable faster data transmission for 2G phones. Besides the circuit switched core network, 2.5G networks have also implemented a packet switched core network. By such upgrades, 2.5G networks provide partial benefits of 3G networks, such as higher data transmission speed than 2G networks. The General Packet Radio Service (GPRS), with up to 180.4Kbps data transmission speed in the downlink direction, is an example of 2.5G networks for GSM operators. While CDMA2000 1xRTT (1 times Radio Transmission Technology) is an example of 2.5G CDMA networks, which provides up to 307.2Kbps downlink data transmission rate.

1.1.1.4 3G Cellular Networks

The growing needs for wireless internet access require a universal standard for wireless communications. Comparing with 2G networks, which mainly provide voice services, 3G networks are expected to support higher speed data services with rates up to 2Mbps. There are two main standards in 3G networks: UMTS/W-CDMA (Wideband Code Division Multiple Access) and CDMA2000.

UMTS networks, based on GSM, are the European version of 3G networks. W-CDMA standard is taken in the air interface, using a pair of 5MHz bandwidth carriers. The first national customer UMTS network was launched in 2002. The CDMA2000 standard, based on CDMA IS-95, is the American 3G variant. It is evolving to support new services in a standard 1.25MHz bandwidth.

Both W-CDMA and CDMA2000 use coding schemes to differentiate users and base stations. However, these two standards are still incompatible. There are several reasons for this incompatibility. The most significant one, as stated previously, is that W-CDMA takes a pair of 5MHz bandwidth carriers, while CDMA2000 occupies a pair of 1.25MHz carriers. CDMA2000 3x, the evolution of CDMA2000, will take three pairs of 1.25MHz bandwidth carriers and construct a super channel structure.

Since GSM is, by far, the most popular 2G standard, UMTS are expected to get the biggest market share as they are backward compatible with GSM networks. Therefore, this thesis will focus on developing automatic planning tools for UMTS networks. Figure 1.1 summarizes the cellular network evolution from 2G to 3G.

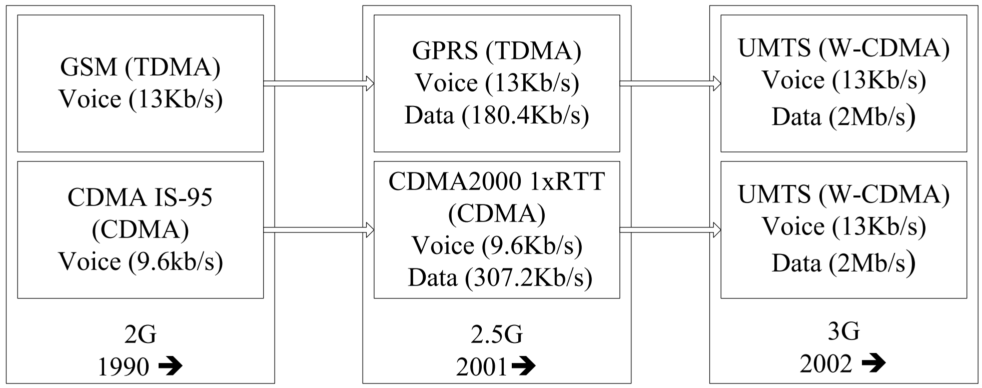


Figure 1.1: The cellular network evolution

From the network infrastructure point of view, UMTS networks are composed of two parts: the Radio Access Network (RAN), also called the Universal Terrestrial Radio Access Network (UTRAN), and the Core Network (CN). The RAN, which is based on the W-CDMA technology, is composed of node Bs (node for Broadband access) and Radio Network Controllers (RNC). Node B, formerly known as base station in 2G networks, houses the radio transceiver and provides the interface between the radio link and the network itself. The RNC, previously known as Base Station Controller (BSC) in 2G networks, provides connectivity between node Bs and the core network. It is also responsible for the call and mobility management and takes the full charge of radio resource management without involving the core network signaling. The CN includes two domains: a circuit-switched (CS) domain and a packet-switched (PS) domain. On one side, the CS deals with real-time traffic, like voice, and provides connectivity to the Public Switched Telephone Network (PSTN). On the other side, the PS handles other types of traffic, such as time non-sensitive services, and ultimately provides a connection to the public IP network. The CN definitions are based on the 2G/2.5G network specifications. In fact, the CN makes use of the existing GPRS infrastructure, such as the Mobile Switching Center (MSC), the Gateway MSC (GMSC), the Home Location

Register (HLR) and the Visitor Location Register (VLR) for the CS domain, and the Serving GPRS Support Node (SGSN) and the Gateway GPRS Support Node (GGSN) for the PS domain. There is a network element, called Media Gateway (MGw), which is as an intermediate node to provide connectivity between the access network and the core network. A typical UMTS network infrastructure is provided in Figure 1.2.

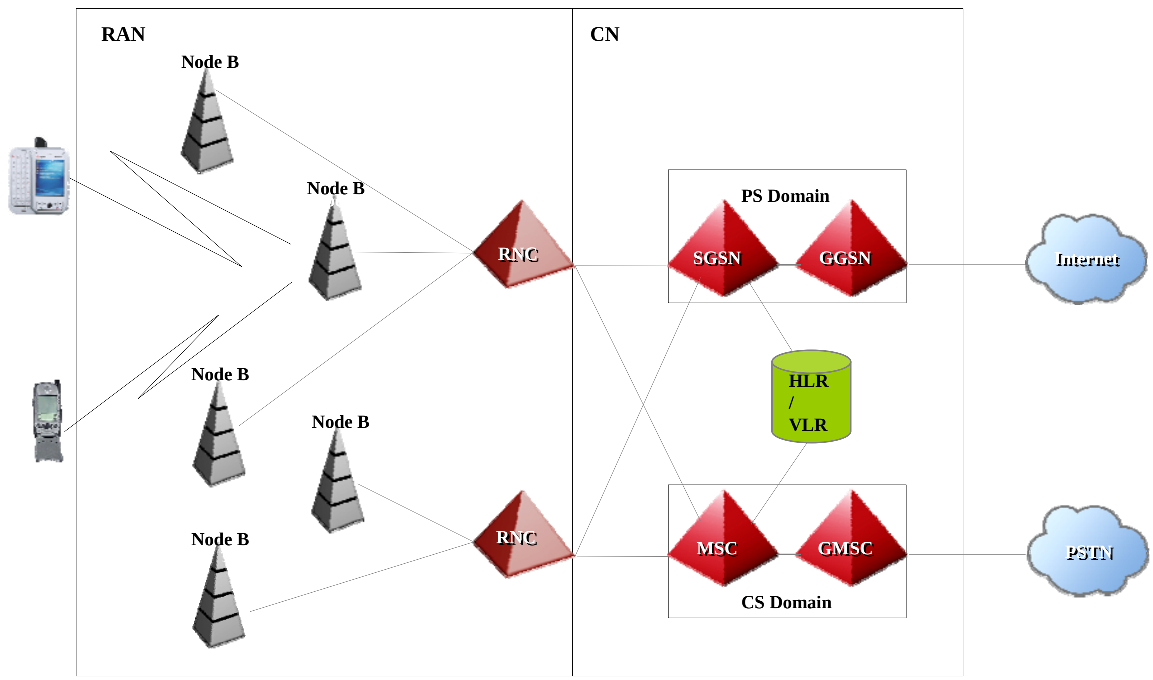


Figure 1.2: A typical UMTS network architecture [14]

1.1.1.5 3.5G/4G Cellular Networks

Starting from 2006, many countries began to upgrade their UMTS networks with High Speed Downlink Packet Access (HSDPA), in order to enhance downlink data transmission speeds up to 7.2Mbps. Further speed increases were achieved by using Evolved High Speed Packet Access (HSPA+), in which the data transmission speed could reach up to 42Mbps. In the uplink direction, the High-Speed Uplink Packet Access

(HSUPA) can be applied to improve data transmission speeds. These techniques are usually called 3.5G.

The 4G standard has not yet been defined. It is characterized as ubiquitous, mobile, and broadband [1]. The overall objective for 4G networks is to build an All-IP network. This will enable all existing different wireless technologies to have a common platform to communicate with each other. With guaranteed quality and security, the 4G networks are expected to support speeds of 100Mbps to 1Gbps for wireless multimedia services. Using the UMTS networks as an example, the fundamental difference between 3G networks and 4G networks is that the function of RNCs in 3G networks is taken by node Bs, a set of servers and gateways. The investment in the 4G network infrastructure is supposed to be less, while the data transmission speed goes higher.

1.1.2 Network Planning Process

Network planning is an iterative process composed of three main steps as shown in Figure 1.3.

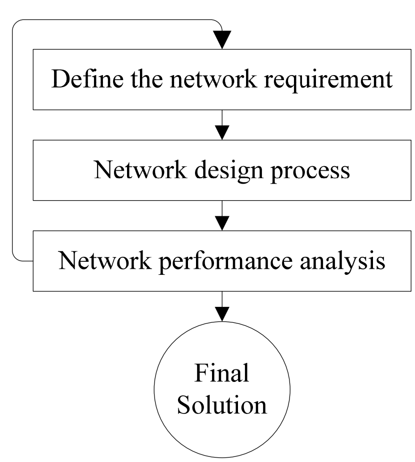


Figure 1.3: The network planning process [2]

- *Define the network requirement*: The network requirement need to be defined before planning the network. This is one of the most time-consuming step which includes the clarification of traffic requirements, such as the traffic type, volume and distribution; the equipment (network nodes and links) costs; the network design parameters, such as technology specific requirements; the operational and utilization constraints, and so on. For an existing network, these requirements can be collected. However, for a completely new network, the unavailable data (such as the traffic distribution) have to be predicted or generated.
- *Network design process*: This process involves exploring as many solutions as possible to decide the node placement, the link connection, the node and link sizing, as well as the traffic routing. Usually, due to the large number of possible combinations, this task cannot be done manually. As a result, different planning tools are developed and applied to help the network planners to make their decisions. A network topology will be developed as the primary output of this process.
- *Network performance analysis*: Once the overall network topology has been developed, it will then be evaluated according to certain criteria such as cost, reliability, the network coverage, and capacity, etc. A good solution will be useful for further network configuration refinements (fine tuning) either manually or by using additional techniques.

The above three-step process can be repeated by changing the input information or using different planning techniques to produce alternative network topologies. The one with the best performance can be selected as the final solution.

1.1.3 Network Planning Techniques

In the early stages, network planning was solved manually. However, since network planning involves several tunable parameters and a huge amount of computations, manual processes obviously limited efficiency and are prone to errors. Over the past several decades, with the development of computer hardware and software, automatic planning tools were developed in order to improve network planning accuracy and efficiency. Automatic planning tools make use of a technique, called algorithm, to perform those tedious work previously done manually. An algorithm is a well-defined procedure for solving a problem in a finite number of steps [2]. After building a model, represented by objectives, variables and constraints for characterizing the problem, the algorithm follows predefined procedures to solve the problem using the representation in the model.

Developing an efficient network planning tool, based on the algorithms, is an ongoing concern amongst network planning researchers. There are many choices of algorithms, where two main branches are generally classified: exact algorithms and approximate algorithms.

1.1.3.1 Exact Algorithms

Exact algorithms search all potential solutions in the search space to obtain an optimum solution for the problem. Constraints of the problem help to discard those infeasible solutions. However, when the solution space is very large, exact algorithms will take too much time for finding a solution, which makes them inefficient for large size problems. In fact, for some problems, the CPU time may increase exponentially with respect to the problem size. Moreover, even the computer memory may be insufficient. Linear programming (LP) and integer programming (IP) are examples of exact algorithms. Please refer to [3] for more details about exact algorithms.

1.1.3.2 Approximate Algorithms

Approximate algorithms, also called heuristics, aim to provide relatively “good” solutions in reasonable amount of computation time. Basically, it is a tradeoff between solution quality and execution time. According to their needs, network planners can adjust the parameters in approximate algorithms to make them focus on either finding better solutions or finding faster solutions. There is a general class of approximate algorithms, called meta-heuristics (please refer to [4] for detailed information on meta-heuristics), which are usually used to solve combinatorial optimization problems. A combinatorial optimization problem is a minimization/maximization problem with three elements: a set of instances; a finite set of candidate solutions for each instance; and a cost function [5].

Before introducing the approximate algorithms, several concepts need to be clarified:

- NP-hard problems: problems that can not be solved with an exact solution in polynomial time;
- Neighborhood: a set of solutions obtained by applying a move or a transformation to the current solution;
- Global optimum vs. local optimum: let $\boldsymbol{X}$ be the set of variable x, where $x \in \boldsymbol{X}$. The global optimum minimizes/maximizes $f(x)$ over all $x \in \boldsymbol{X}$, while the local optimum is better than all solutions in its neighborhood as shown in Figure 1.4.

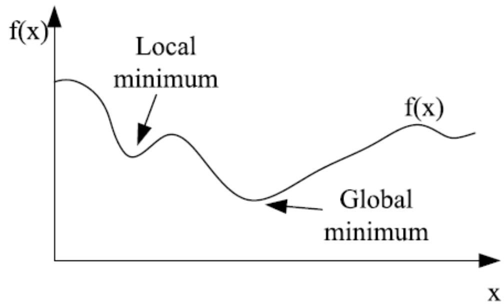


Figure 1.4: Global optimum vs. local optimum

The following four approximate algorithms are commonly used to solve network planning problems.

- **Greedy Algorithm**

A greedy algorithm is governed by the following rule: always find the local optimum in the neighborhood of the current solution. As such, the greedy algorithm can also be defined as a local search (LS) procedure. At each iteration, the greedy algorithm makes greedy choice and keeps reducing the solution set into a smaller one. It usually commits to certain choices too early to search the entire solution space. Typically, the greedy algorithm fails to find the global optimum. Examples of greedy algorithms include Kruskal and Prim algorithms (for finding the minimum spanning tree) and Dijkstra's algorithm (for finding the shortest path).

As stated above, the characteristic of the iterative improvement of the greedy algorithm makes it easily being trapped into local optimum. Different methods have been proposed to try to solve this problem, such as applying the greedy algorithm multiple times with different initial solutions and then choosing the best result as the final optimum. However, the proposed methods still do not guarantee to find the global optimum. As the problem size increases, especially for NP-hard problems, the greedy algorithm becomes even less feasible. Nevertheless, greedy algorithms are proven to be useful for finding a "good" solution as an initial solution for further improvement methods [6]. A group of approximate algorithms is proposed in order to overcome the drawbacks of greedy algorithms in this thesis.

- **Tabu Search**

The tabu search, denoted as TS, is derived from the best improvement local search, aiming to find the global optimum. The tabu search uses the local search to iteratively move from the current solution to a better one within its neighborhood. By allowing the temporary solution degradation, tabu search avoids the search process being trapped into the local optima. Two mechanisms, short term memory and long term memory, can be used to keep track of attributes of solutions previously visited and guide the search direction. The short-term memory contains a tabu list and the aspiration criteria. A tabu list is a storage space for solutions being recently visited. In order to avoid cycling, solutions in the tabu list are prevented to be revisited for a time period. This time period, determined by the tabu size, defines how long a solution will be a member of tabu list. At the same time, the aspiration criteria works to avoid missing good solutions in the tabu list and to make them available for the search. Tabu search may also make use of a long-term memory, which operates when there is no solution improvement for a given number of iterations. To avoid some solutions/moves being selected more frequently than the others, the occurrence frequency of solutions will be memorized in the long-term memory. Solutions with occurrence frequency over a certain threshold will be penalized. Tabu search may make use of one or both of these two memories to finally find a good solution within an acceptable computation time. For more detailed information on tabu search, please see [4] and [7].

- **Simulated Annealing**

Simulated annealing, denoted as SA, simulates the physical annealing process, which starts from a high enough temperature T (the analog to a control parameter) and then the temperature decreases according to a cooling schedule. During cooling process, which makes reference to the solution search process, solution transforms from current state to a random new state in its neighborhood at every temperature *T*. New state selection is done according to a certain probability, which allows temporary solution deterioration in order to avoid the final solution being trapped into the local optimum. The control parameter, *T*, guides the problem to its final state. If the temperature decreases too fast, the solution will be trapped into the local optimum with higher cost function value. On the other hand, if the temperature decreases too slowly, it will take too much time for finding the final solution, which greatly decreases the algorithm efficiency. A well designed cooling schedule for the temperature is the key to make the simulated annealing successfully find the global optimum. For more information about SA, please refer to [4] and [8].

- **Genetic Algorithm**

The genetic algorithm, also known as GA, is a class of evolutionary algorithms based on Darwin theory of natural selection. The main idea of the algorithm is to start with an initial population. Then, some individuals from the population (called parents) are selected in order to generate new individuals (also referred to as offspring). The choice of the parents is based on the fitness of the individuals, which is evaluated by the objective function. The higher the level of the fitness, the higher the probability that the individual will be selected to produce offspring. Offspring are generated by applying different recombination operators, such as crossover or mutation (or both). The new generated offspring will then be integrated into the current population to create a new generation. The whole process stops when some predefined conditions (such as the maximum number of generations) are reached. As we can see, this process favors the ”mating” of the more fit individuals and allows exploration of the promising area in the search space. Please refer to [9] and [10] for more detailed information about the genetic algorithm.

Tabu search, simulated annealing and genetic algorithm are common examples of meta-heuristics. More details about these algorithms will be given in Chapter 3. After having presented the background information, we will now look at the problems that could occur when planning UMTS networks.

1.2 Problem Statement

The primary goal of the UMTS network planning is to generate an optimum topology for the network. To a great extent, it is decided by the node location selection. For a normal cellular network, there is a huge amount of nodes to be installed. At the same time, a great number of factors have to be taken into consideration for choosing proper node locations, such as traffic distribution, network node features, network management issues, and so on. Geographical factors also play an important role for node location selection in wireless communication networks.

Many models and algorithms have been proposed to solve the UMTS network planning problem. However, due to the problem complexity, most of them only focus on a portion of the overall network. In fact, the whole planning problem is usually decomposed into three subproblems: the cell planning subproblem, the access network planning subproblem and the core network planning subproblem. Each of them has already been proven to be NP-hard [11, 12, 13].

In order to find a solution for the whole UMTS network, these three subproblems need to be solved sequentially. Unfortunately, such an approach doesn't consider the interconnections between the subproblems. In fact, the combination of the solution of each subproblem may provide a local optimum. In other words, combining partial solutions in order to obtain a solution to the global planning problem may provide suboptimal solutions, rather than the optimal solution.

A different way of solving the planning problem of UMTS networks is to use a global approach, where the three subproblems are solved simultaneously. Since all interconnections among the subproblems are taken into consideration, a global approach has the advantage of providing the global optimum.

A mathematical programming model has been proposed for the global planning problem of UMTS networks [14]. However, due to the complexity of the global planning problem, it cannot be solved within a reasonable amount of time by a commercial solver, such as CPLEX [15]. Approximate algorithms, aiming to provide relatively "good" solutions in reasonable amount of computation time, are more suitable to build planning tools. In fact, they are a tradeoff between solution quality and execution time.

Furthermore, the efficiency of the planning tool is problem dependent. A local search algorithm and a tabu search algorithm have already been proposed in [14] and [16] for solving the global planning of UMTS networks. However, besides these two algorithms, other approximate algorithms, such as genetic algorithm and simulated annealing, have been widely and successfully implemented for solving network planning problems. For example, in [23], the simulated annealing was proposed to solve the UMTS site selection problem, while in [25] the genetic algorithm was used to solve the base station positioning problem. In [37], the performance of the simulated annealing, the genetic algorithm and another approximate algorithm were compared when solving the UMTS base station location planning problem. Therefore, it is interesting to investigate how they will perform on the global planning of UMTS networks. Moreover, combining two algorithms may also provide interesting results since different algorithms have different strength.

Based on the problem statement, we will now formulate the research objectives in the following subsection.

1.3 Research Objectives

The main objective of this thesis is to develop efficient automatic planning tools based on different meta-heuristics to solve the global planning problem of UMTS networks. More specifically, we will address the following topics:

- Develop a first automatic planning tool based on the genetic algorithm;
- Evaluate the performance of the previous planning tool by comparing its solution with the optimal solution;
- Propose a second planning tool based on simulated annealing;
- Evaluate the performance of the second planning tool by comparing its solution with the optimal solution;
- Design a third planning tool based on a combination of tabu search and genetic algorithm;
- Evaluate the performance of the third planning tool by comparing its solution with the optimal solution;
- Compare the three previous planning tools with results obtained from the tabu search algorithm.

1.4 Methodology

UMTS network planning can initially be viewed as a very involved task since a large number of variables are tunable in both the UMTS network and the automatic planning tool. In order to achieve the objectives stated in the previous section, we will firstly make a thorough study on each heuristic. To simplify the planning process, a step by step approach is proposed in Figure 1.5.

- *Study meta-heuristics*: The UMTS network planning problem is a NP-hard combinatorial optimization problem. In this phase, two meta-heuristics, the genetic algorithm and simulated annealing, are selected for study because of their capability to solve this kind of problem.
- *Algorithm implementation*: C/C++ will be used to implement the planning tools. Three planning tools are proposed to solve the global UMTS network planning problem. The first one is based on the genetic algorithm, while the second one is

using the simulated annealing and the third planning tool is a combination of tabu search and genetic algorithm. The components in the algorithms are well-designed aiming to find the best solution for this specific planning problem. A series of tests are made in order to work out the optimal parameter settings.

- *Solution comparison with the optimal solution*: In order to find the optimal solutions, a commercial solver called CPLEX will be used. The solutions obtained from CPLEX will be applied to evaluate the quality of the solutions obtained from the genetic algorithm, the simulated annealing and the cooperation of the tabu search and the genetic algorithm respectively in terms of the objective function value and the CPU time.
- *Solution comparison with tabu search*: Using the optimal solution as the reference, the solution obtained from the three algorithms will be compared with the tabu search in order to find out the most efficient planning tool.

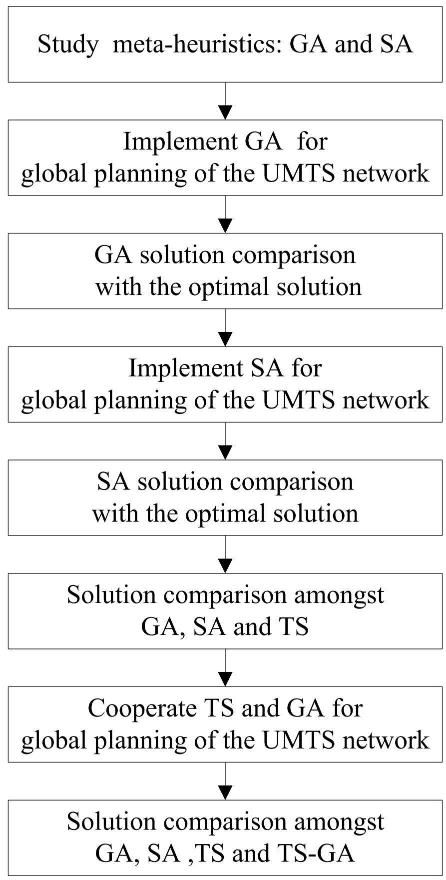


Figure1.5: The proposed methodology

1.5 Main Contributions

The main contributions of this thesis can be summarized as follows:

- A detailed study/analysis on the genetic algorithm. A first automatic planning tool is built based on the genetic algorithm and the performance evaluation is made with the reference of the optimal solution;
- A detailed study/analysis on the simulated annealing. A second automatic planning tool is developed based on the simulated annealing and the performance evaluation is made with respect to the optimal solution;
- A third automatic planning tool is proposed based on the cooperation of the genetic

algorithm and the tabu search. Performance evaluation is made with the reference of the optimal solution;.

- A comparative study is made amongst the different algorithms in order to analyze the advantages and disadvantages of each of them to make better use of the meta-heuristic in solving the global planning problem of UMTS networks.

1.6 Thesis Overview

The remainder of this thesis is organized as follows: Chapter 2 presents a selective review on the UMTS network planning problem. The three subproblems are explained in details in this Chapter as well as the two approaches commonly used to solve the problem. Then, Chapter 3 provides a brief description of the mathematical model used to represent the global planning problem. Different meta-heuristics (genetic algorithm, simulated annealing, tabu search and a combination) are also studied and implemented to build planning tools for solving the global UMTS network planning problem. The simulation results and analysis are presented in Chapter 4, where the solution quality and the CPU time are compared with respect to CPLEX. Finally, conclusions are drawn in Chapter 5.

Chapter 2

Related Work on the UMTS Network Planning

UMTS network planning has recently been a subject of great interest. It is a complex but necessary step towards building an efficient network. Typically, a planned area is divided into cells, where each cell is covered by a node B through radio interface. Then, node Bs are connected to RNCs to construct the access network. Through such connections, mobile traffic is sent to the network to get corresponding services. After that, a group of RNCs is connected to a MSC/SGSN in a core network. Based on the service type, the traffic, after being served, is further transmitted to external networks. In this case, a “bottom-up” approach is an efficient network planning hierarchy. Therefore, the planning of UMTS network can be divided into three planning areas [18, 19]:

- The cell planning for mobile terminals and node Bs;
- The access network planning for node Bs and RNCs;
- The core network planning for RNCs and core network elements.

The following sections will cover all three planning areas of UMTS networks in terms of the different tools, methods and algorithms proposed so far.

2.1 Sequential Approach

Solving the UMTS network topology planning problem is very complex. A popular way to simplify the planning complexity is to decompose the problem into subproblems by using, for example, a sequential approach. When using such an approach, three planning subproblems are defined: the cell planning subproblem, the access network planning subproblem and the core network planning subproblem. These three subproblems are solved step by step to finally solve the whole planning problem as shown in Figure 2.1. Besides the input to each subproblem itself, the output of the previous step turns out to be the input for the next step, until the whole network problem is solved.

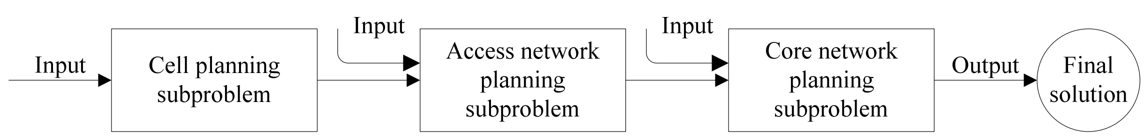


Figure 2.1: The sequential approach for UMTS network planning

Each subproblem has been extensively researched in previous studies. They will be explained in details in the following subsections.

2.1.1 Cell Planning

Cell planning is a process to connect all mobile terminals through the air interface that node Bs provide, as shown in Figure 2.2. As mentioned previously, UMTS networks deploy the W-CDMA technique in the air interface. All mobile connections in a UMTS network share the same frequency bandwidth. This makes simultaneous mobile connections in the neighborhood the main cause for the noise level at the receiver of mobile terminals. The coverage of a node B is not decided solely by the signal strength level. In fact, traffic distribution, power control mechanism, transmission power limits and quality constraints may all be considered for the coverage prediction of a node B [11]. Furthermore, link direction has to be decided before the cell planning stage: the network planner must choose to focus on uplink direction (from mobiles to node Bs), downlink direction (from node Bs to mobiles) or both. Uplink direction is suitable for predicted symmetric traffic, such as voice services. However, if the network is predicted to provide more data services, such as web-browsing, where downloading is more prevalent than uploading [20], downlink direction would be more appropriate for the consideration. The studies on the fulfillment of cell planning with uplink direction, downlink direction and both directions are shown in references [11, 21, 22] respectively.

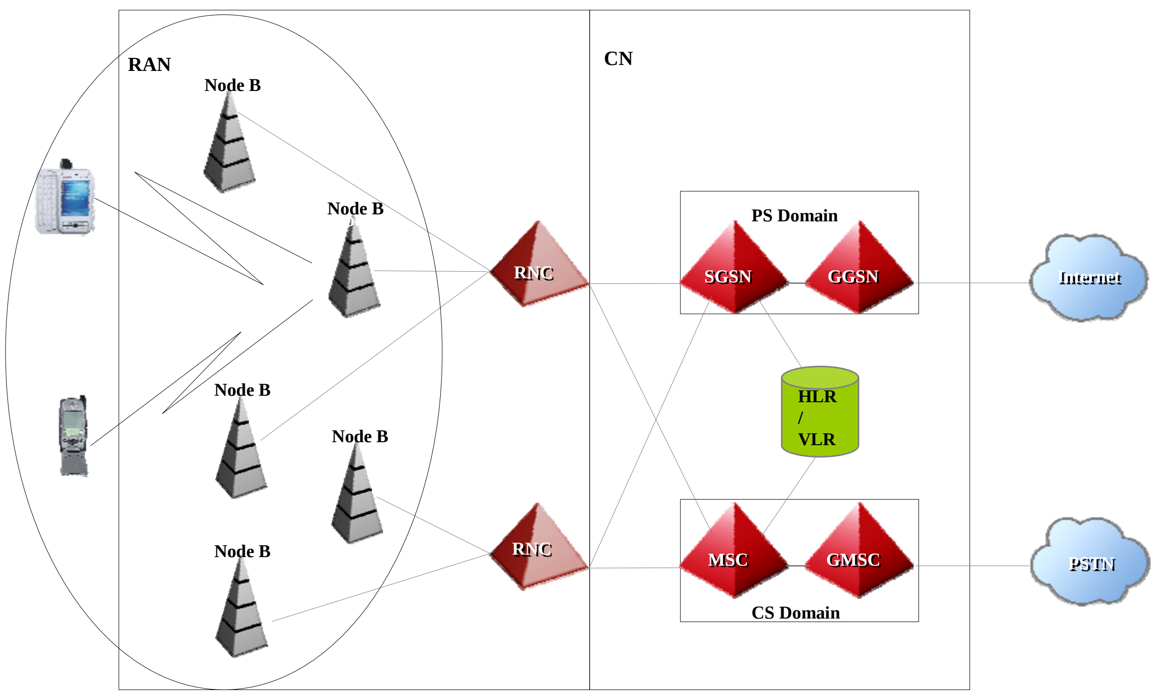


Figure 2.2: The scope of the cell planning

2.1.1.1 Cell Planning Objectives

From the operator point of view, the minimum investment with the best performance and long-term profitability is the ultimate goal for the network planning. For the cell planning, this can be targeted into several objectives:

- Minimizing the network cost;
- Maximizing the coverage;
- Maximizing the capacity;
- Maximizing the signal quality;
- Minimizing electromagnetic field levels.

As we can see, the above criteria might be contradictory with one another. For example, to maximize the coverage, the network planner may need to deploy extra node Bs, thus increasing the network cost. In UMTS networks, all mobile terminals share the same frequency bandwidth. It makes the UMTS network a self-interference network. The coverage and capacity can be antagonistic to each other too.

The above concerns bring out a multi-objective planning strategy. Previous studies proposed two ways to represent a multi-objective function. One way was to use a linear combination of different objective criteria [23, 24, 25] to form a single objective function, where different objectives were given a certain weight between 0 and 1. In the second method, the problem was formulated by a set of decision variables (parameter space/vector) and a set of objective functions (objective vectors). These objective functions could be any of the above stated objective criteria [23, 24]. When there was no solution that could improve one objective criterion without degrading the other objective criteria, it could be said that the optimum solution was found. This method is referred to the Pareto optimal solution. The objective functions in the set can also be assigned with a weight correspondingly, known as weighting objectives, which is similar to the first method. The weighted multi-objective functions give more flexibility to the network planner by assigning higher (lower) weight to put more (less) emphasis on a given objective.

Besides the objectives that have been stated at the beginning of this section, some other objective criteria were also proposed to evaluate the solution quality. In [24], a downlink UMTS omni-cell planning problem was studied. Wu et al. built a model aiming to maximize the transmission power, antenna height, and the assignment between mobile terminals and node Bs. A combinatorial objective function was formulated with three sections: minimizing the total cost of node Bs, minimizing the total emitted power by active mobile terminals, and maximizing the total number of active connections. Two constant weight parameters were applied on the second and third objective criteria. The interference from inside and outside the cell, the maximum required power of mobile terminals in a given cell and Signal-to-Interference Ratio (SIR) threshold at mobile terminals were considered in the constraint set. Crainic et al. [17] dealt with cell planning from the electromagnetic (EM) field level point of view. With the goal of minimizing EM field level, the radio protection constraints, handover and downlink capacity constraints were taken into considerations. Five objective functions, scaling five electrical field levels, were formulated to model the field level as well as explore different solution spaces.

2.1.1.2 Cell Planning Input

To solve cell planning problem, the following information is required as input [18, 28]: traffic modeling; node B potential location information; node B model specification; and node B coverage/propagation prediction.

- **Traffic modeling**

Mobile traffic distribution is a decision factor for the network topology planning. Since the UMTS network provides both voice and data services, traffic distribution should be differentiated based on the service type.

Several traffic models were studied in this subsection based on different design requirements. For the purpose of determining a network topology, traffic intensity model is preferred [29]. In this model, mobile terminals were represented by the amount of traffic (traffic intensity) requested from a given area during a fixed time interval, where they were clustered or agglomerated to simplify the traffic description instead of representing every single mobile terminal. These agglomerations were called traffic nodes (TN) [30] or test points (TP) [28] for the purpose of measurement, such as signal strength, quality of service, capacity requirement, and so on.

Classification of the area to be planned will also be done at this stage. It decides what kind of area the planning will work on: dense urban, urban, suburban, rural, and so on [19].

- **Node B potential location information**

Potential locations where node Bs can be installed need to be defined. In theory, node Bs can be installed anywhere. However, constraints like geographical issues may make it not possible in practice. Some locations are naturally good choices as potential sites, such as the top of a building. However, some other locations cannot be installed with node Bs because of, for example, block from other high-rise buildings. As a result, a discrete set of possible locations should be provided.

Usually, the number of potential sites is more than the actual needed number of node Bs. In [24], the potential sites were randomly generated with the uniform distribution with a certain probability. In [19], the probability of whether a sub-traffic area would be installed with a base station was decided by four factors: traffic density, building height, terrain height and if there was a GSM site in the area. The bigger the probability is, the more chance the base station would be installed in the analyzed area.

- **Node B model specification**

A series of parameters/features of available node B models need to be specified at this stage [6]. One of the most important factors is the antenna type. Shall a directed or an omni-directional antenna be used? Antenna type decides the signal radiation degree, which is then affecting the interference scope. Typical parameters that need to be considered include the following: antenna height, tilt, azimuth, transmission power, sensitivity, switch fabric capacity and the cost.

- **Node B coverage/propagation prediction**

The coverage/propagation prediction can be used to approximately estimate the number of node Bs needed in a given area. In W-CDMA, since radio frequencies are shared by all node Bs [31, 32], it is not enough to predict the coverage simply based on the signal level. Traffic distribution, signal quality (usually measured by SIR) and power control also need to be taken into considerations [28].

Gould [33] described some challenges that radio network design engineers would face when planning urban areas. When dealing with radio frequencies, many aspects, such as signal propagation, attenuation and interference must be considered. Signal propagation parameters can be obtained using actual measurement, which is very complex. That is why different models have been developed in the literature. From experimental results and statistical data, Okumura [34] developed several practical charts in order to predict signal propagation. Later, on the basis of Okumura curves, Hata [35] proposed an empirical formulation for propagation loss. This model, called the Hata model, is widely used in telecommunications networks. Other models such as COST 231 [36], extended the model proposed by Hata to the upper frequency band (1500 MHz$\leq f \leq$ 2000 MHz).

As mentioned previously, the W-CDMA technique in the UMTS network air interface constrains interference for mobile terminals mainly from the neighborhood mobiles sharing the same frequency bandwidth. The transmission power of the mobile terminal is limited. If far away from the node B and surrounded by the high level interference, the mobile terminal may not be able to get the minimum acceptable SIR

[28]. That is, cell coverage is heavily affected by the traffic distribution and interference. On one side, a cell can cover a large number of users if they are relatively close to the node B. On the other side, the cell will only be able to cover a few users if they are located far away from the node B. This phenomenon makes reference to the cell breathing effect. As shown in Figure 2.3, the cell breathing effect can be defined as the constant change in the coverage area with respect to the amount of traffic. When a cell becomes overloaded, the interference will increase and therefore, the cell size will decrease. Users that are excluded from a cell will usually be redirected to neighborhood cells. It is important to keep the transmission power of the node Bs and mobile terminals at the minimum levels while ensuring adequate quality at the receiver [37]. Serious power control mechanism has been adopted in the UMTS network. References [20, 21, 28, 38] provide detailed studies of power control for the UMTS cell planning.

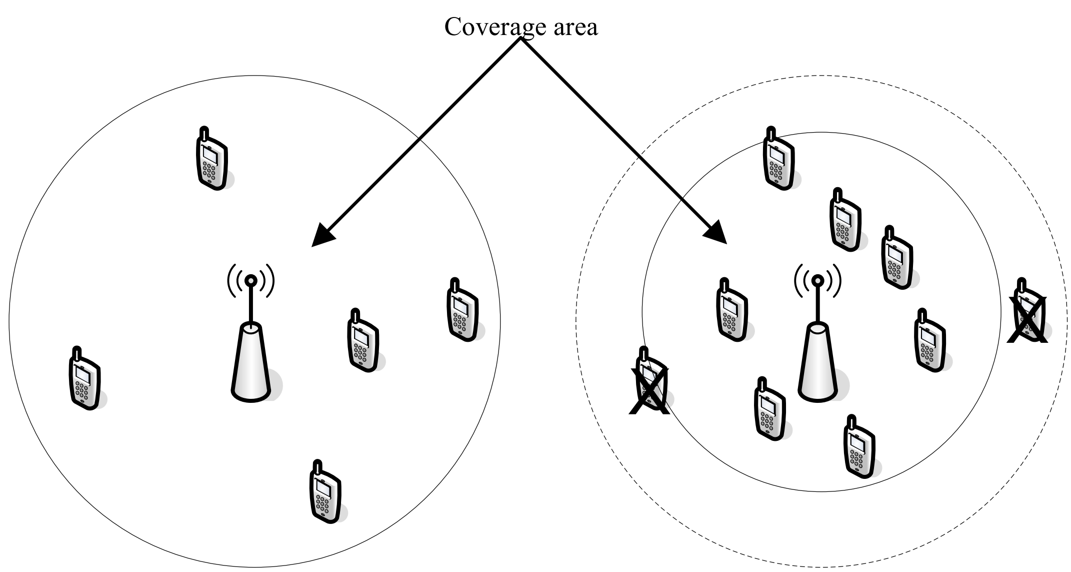


Figure 2.3: Cell breathing effect

2.1.1.3 Cell Planning Output

Following a "bottom-up" planning scheme, cell planning is the first step in order to connect subscriber traffic to the UMTS network, as shown in Figure 2.2. The general idea behind the cell planning problem is to cover all mobile terminals in a given region with the minimum number of node Bs. More precisely, the cell planning problem usually deals with one (or more) of the following item(s):

- The optimal number of node Bs;
- The best locations to install node Bs;
- The types (or models) of node Bs;
- The configurations (height, orientation, tilt, power, etc.) of node Bs;
- The assignment of mobile terminals to node Bs.

2.1.1.4 Cell Planning Tools

The cell planning problem has been proven to be NP-hard [28]. As a result, most planning tools are based on approximate algorithms.

Downlink omni-cell planning task solved by greedy algorithm, tabu search and simulated annealing were presented in [24]. The result showed that the tabu search had the best performance. In [39], the problem of locating node Bs was studied. Greedy algorithm, genetic algorithm and a combination algorithm for total optimization (CAT) were proposed to solve the problem, where CAT was superior in terms of the computation time and solution quality. An integer programming formulation was proposed in [40] to solve the node B placement problem in the uplink direction. Randomized greedy, reverse greedy heuristics and the combined randomized add and remove algorithm were applied to the cell planning problem. For medium to large size problems, randomized add and remove algorithm showed its capability to find good solutions with an acceptable amount of computation time. Taking into account of fast power control, soft handover, and pilot signal power, a model was built for node B location selection with the consideration of both uplink and downlink direction [37],. In their paper, the simulated annealing, evolutionary simulated annealing, genetic algorithm and greedy search were compared in terms of the computation time and solution quality as a reference for heuristic selection. In [41], on the basis of greedy algorithm search result, tabu search was used to optimize the node B location and configuration problem in an uplink direction model. The genetic algorithm, simulated annealing, tabu search and greedy algorithm were compared in [42] for node B location selection. The power control, soft-handover (SHO) and common pilot channel (CPICH) power were considered in the model. Tabu search demonstrated its superiority in finding a good quality solution within reasonable computation time.

Different from only using one meta-heuristics, Crainic et al. [17] proposed to cooperate two meta-heuristics, tabu search and genetic algorithm, in parallel to fulfill the task of automatic planning. The model was designed to solve not only node Bs' location and emission power but also the antenna height, tilt and orientation. It was proven that the tabu search was good at deep search of solution space but with relative small configuration parameters. The genetic algorithm could explore the whole set of the configuration parameters but with high computing cost. In the proposed method, the tabu search worked independently on different parts of the solution spaces with different objectives. The result from the tabu search would then be combined together as the initial solution of the genetic algorithm. The result from GA would then further increase the TS diversification. The result proved that this cooperative method was able to deal with the high volume of configuration parameters to get an accurate planning.

Once we have the location and configuration of node Bs, the next step is the access network planning. The output of the cell planning will be treated as the input for the access network planning, along with some other input information.

2.1.2 Access Network Planning

The access network is used to concentrate connections and trunk them to the upper level core network [43], as shown in Figure 2.4.

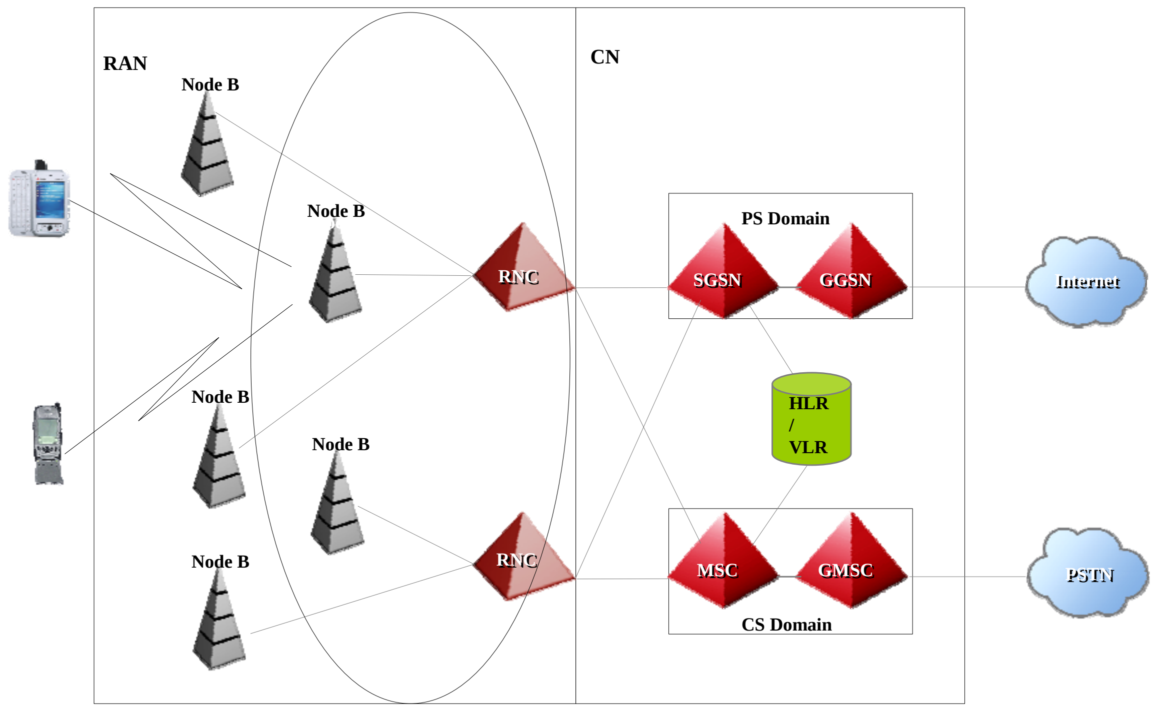


Figure 2.4: The scope of the access network planning

2.1.2.1 Access Network Planning Objectives

Currently, the objective of the access network planning focuses on two aspects: the cost-based planning and the reliability-based planning.

- **Cost-based planning objective**

In [18], the equipment cost was composed of the RNC cost and the access concentrator cost, represented as a stepwise function with respect to the expected traffic. The link cost consisted of links between different node Bs, from RNCs to node Bs, from RNCs to core networks nodes, as well as links between different RNCs for signaling messages and handover traffic. The total link cost was a piecewise linear function of link length and a stepwise function of expected traffic on the link. The access network cost was a tradeoff between the sum of the equipment cost and link cost. Apparently, using high capacity RNCs would decrease the total number of RNCs needed, however, at the cost of increasing the link usage. The goal of the access network topology planning was to find the minimal equipment and link cost while satisfying the traffic requirement.

The RNC type was not differentiated in [43]. Their objective function included the cost of RNCs and links. The total cost of RNCs was calculated as a linear function of RNC number. The total link cost was a step-wise function with respect to the link capacity, with the consideration of inter-node B links, which were arbitrary in the model, and links between node Bs and RNCs.

The handover, also called handoff, is the process where mobile terminals maintain communication with the system when moving from one coverage area to another one. Related studies appeared in [44, 45, 46, 47], where handover was differentiated into two types: simple handover (two cells connected to a same switch) and complex handover (two cells connected to two different switches). The decision-making process for handover type is done at RNC. It takes 10-80ms, which is roughly two times the air-interface capacity used by a mobile terminal under non-handover situation [20]. As we can see, the handover is an important aspect when planning the radio access network capacity. Thus, in [44, 45, 46, 47], when assigning cells to switches, the objective function consisted of not only link cost, but also a virtual cost generated by the handovers. In [44, 45, 46], the handover cost only included complex handovers and the cost of simple handover was neglected. However, in [47], the handover cost was defined as the cost from both simple and complex handovers.

Wu et al. [48] built a model to find the RNC locations and assigned node Bs to RNCs. The objective function of their model contained hardware cost, including RNC cost and link cost, as well as the handover cost. Only complex handovers were taken into consideration in this paper.

- **Reliability-based planning objective**

Besides building an access network with minimum cost, the reliability of a network is also an important aspect that needs to be considered by network planners. Different from wired networks, small failure in one part of the network may cause serious consequences on neighbor networks. In fact, mobile terminals that are disconnected from a network will try to re-connect to neighbor networks, thus increasing the target network load and degrading their performance or even getting the networks down.

Since reliability comes with cost, network planners need to find a balance between network investment and reliability/survivability. Szlovencsak et al. studied this tradeoff by building a network with reasonable cost as well as an acceptable traffic loss. In the first phase of their two-phase method, the objective function was a linear combination of two types of cost [49]: structural cost (including RNC and node B cost as well as the link cost) and penalty cost (caused by a network failure). The parameters in the cost function could be adjusted to change the weight of reliability related penalty cost. Based on the topology obtained from the first phase, the second phase tried to add new link in order to increase reliability. The objective function of the second phase was composed of the cost, node availability, and new added link length to finally find the lower cost solution. The final result would match the required reliability with an acceptable cost.

Charnsripinyo et al., in [50], also proposed a two-phase method to design a network topology with optimal cost. The first phase aimed to build a minimum cost network. The objective function only contained the cost generated by links. The second phase objective function consisted of the total cost of the new links needed to reach reliability requirement.

2.1.2.2 Access Network Planning Input

A lot of information is needed in order to plan the access network. Typical input information can be summarized as follows:

- The physical location and type of the node Bs that are installed (can either be given or obtained by solving the cell planning subproblem);
- The traffic demand going through each node B (can either be given or obtained from the cell planning subproblem);
- The set of potential locations to install the RNCs;
- The different types/capacities of RNCs (this can include the number of ports, the switch fabric capacities, and so on);
- The different types/capacities of links available to connect the node Bs to the RNCs;
- The handover frequency between adjacent cells.

Wu et al. mainly focused on the constraint-based optimization model for the access network design in [48, 51]. The limited traffic capacity and available ports of a RNC for connecting node Bs were specified as corresponding constraints to reduce the problem complexity.

2.1.2.3 Access Network Planning Output

Based on the cell planning result, the access network planning will work on clustering the node Bs into RNC areas [18, 19]. It will deal with one or more of the following aspects:

- The optimal number of RNCs;
- The best location to install the RNCs;
- The type (or model) of RNCs;
- The link topology and type between node Bs and RNCs;

- The link topology and type between RNCs;
- The link topology and type between node Bs.

2.1.2.4 Access Network Planning Tools

Most of the time, a star or a tree topology will be selected to build an access network. In a star topology, all node Bs have their own link(s) directly connected to the RNC. A tree topology is implemented when a node B provides connectivity for other node Bs to the RNC. The topology built in [48] was a typical star topology. In their model, only link cost between node Bs and RNCs was considered. Also, there was no notion of degree constraint for the node Bs. More information about the star topology can be found in [14, 51].

In [43], Harmatos et al. utilized simulated annealing and greedy algorithm to build a tree topology for the access network. The initial state (the number and locations of RNCs) was randomly generated for the given node Bs. The simulated annealing algorithm was used for finding the optimal state of the RNCs. Then, the greedy algorithm was applied for determining the access links from the RNCs to the node Bs to build a minimum cost tree. The tree construction was later proven to be the bottleneck of their planning algorithm. In a later work of Harmatos et al. [52], an access network was built for only one RNC using a new algorithm, which was closely related to the spanning tree problem. This new algorithm was said to be capable of solving those multi-constrained capacitated tree optimization problems with non-linear objective function. Afterward, Juttner et al. [13] improved this tree topology planning with a two-step method. Firstly, the overall UMTS access network was planned using a “global algorithm”, which was a combination of the simulated annealing and the b-matching algorithms. Then, a “local algorithm”, based on the branch-and-bound algorithm with Lagrangian lower bound, was applied to plan or improve the single tree in the trees obtained from the previous step. Test results demonstrated further reduction of network cost.

Lauther et al. [18] handled the access network topology planning as a clustering problem. Given the location of node Bs, they provided two methods to construct the cluster. The first method was based on the tree generation and cutting. Multidimensional binary search tree algorithm (or k-d tree) was applied to generate a proximity graph [3, 53, 54]. This proximity graph was then calculated using Prim's or Kruskal's algorithm to find the minimum number of links needed to connect all nodes. Then linear tree partitioning algorithm, proposed by Kundu and Misra [55], was used to cut the tree into subtrees (clusters). RNCs were assumed to be located at the potential locations near the cluster center (based on the traffic distribution). The second method made use of Kruskal-like algorithm, starting by considering each node B as a cluster. At each iteration, two clusters were merged and it kept doing this merge as long as the total cost could be reduced during the merge.

Providing network reliability is not a trivial issue in wireless communication, especially in 3G network, which is expected to support high-speed multimedia services with guaranteed quality of service (QoS). Network reliability/survivability need to be incorporated with the network infrastructure. It is known that a tree topology does not provide reliability in case of a failure. Szlovencsak et al. [49] proposed a two-phase method to build an economical and reliable network. The first phase was based on the minimum cost tree proposed in [43, 52], which was using the simulated annealing algorithm to modify (add or remove links while keeping tree topology) the trees considering the reliability issues. The first phase solution was good enough for the planning with less strict reliability requirement. The second phase would add redundant links between the node Bs to secure the weakest part of the network. Both greedy and randomized methods were proposed for this phase. The randomized method was finally preferred to select which node Bs would be connected.

Instead of using tree topology, Charnsripinyo and Tipper [50] proposed a two-phase method to build a mesh topology for access network, aiming to finding the minimal network cost, as well as satisfying QoS and survivability requirements. A shortest-path routing algorithm with link-cost metric was presented in both phases to minimize the routing cost.

2.1.3 Core Network Planning

The core network is the center of the whole network. It fulfills main functions of UMTS networks and provides access to external networks, as shown in Figure 2.5. In the core network, real time services, like voice, are routed to the PSTN, while high-speed data traffic is directed to the public IP networks. Besides traffic switching, the core network also provides QoS, mobility management, network security, and billing [56]. In the sequential approach, the core network planning is the final but not trivial step because of its nuclear position in the whole network.

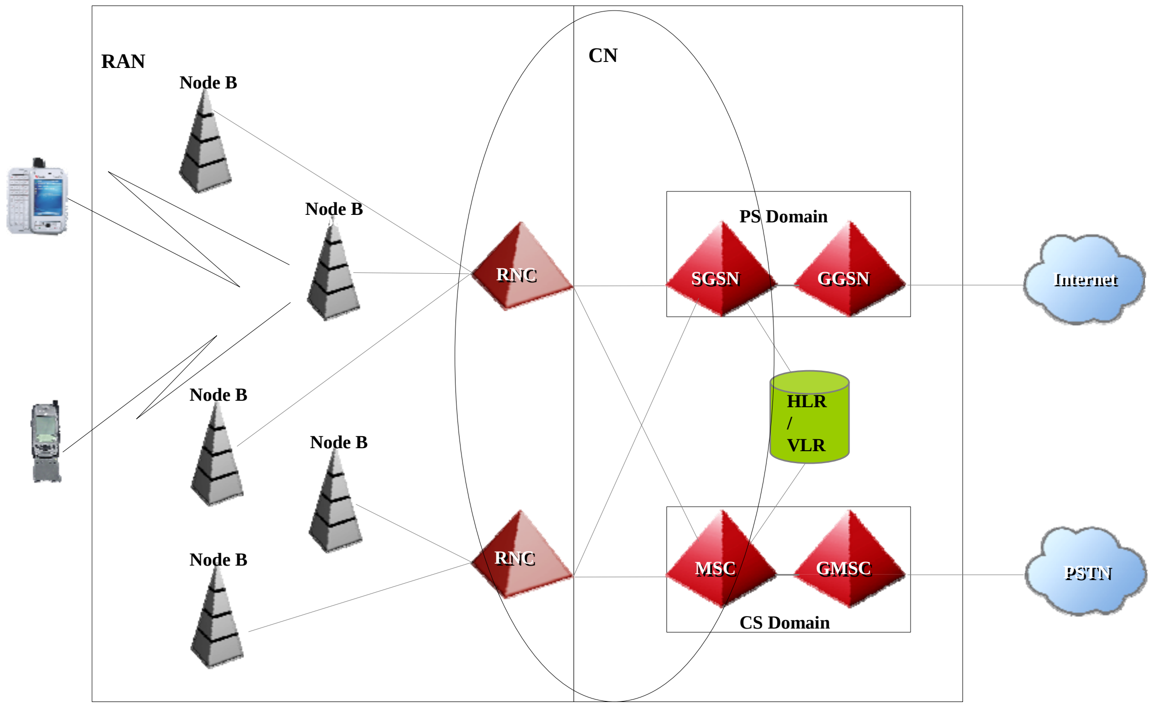


Figure 2.5: The scope of the core network planning

2.1.3.1 Core Network Planning Objectives

The objective of core network planning is to build a cost-efficient network while respecting the QoS [56]. Not many studies have been dedicated done on the UMTS core network planning problem. This can be explained by the fact that this subproblem is similar to wired network planning problems.

Harmatos [12] formulated the objective function with a linear combination of equipment and link costs. The equipment cost was composed of hardware cost, installation cost, and port/interface cost. It was applied for RNCs, MGws, and transport nodes, which represented core network nodes. Link cost included the cost of links between different RNCs, RNCs and MGws, MGws and transport nodes, as well as between different transport nodes. In the objective function, the link cost was represented by a step-wise function of link capacity and a piece-wise linear function of link length between repeaters.

Ricciato et al., working on GPRS network, focused on finding the optimal assignment of RNCs to SGSNs on the basis of measured data [57]. The objective was set to balance the number of RNCs per SGSN and minimize the inter-SGSN routing, which was represented by minimizing the peak number of RNCs that could be connected to SGSN, and minimizing the number of corresponding complex handovers. A parameter was used to tune the weight of these two objectives.

2.1.3.2 Core Network Planning Input

Based on the output of the access network planning (UTRAN topology and corresponding traffic distribution), the following inputs are used to plan the core network:

- The physical location of the RNCs (can be obtained from the access network planning or measured from real data);
- The traffic demand (volume and type) going through each RNC (can be obtained from the access network planning or measured from real data);
- The potential location of core network nodes;
- The different types/capacities of core network nodes;
- The different types/capacities of links available to connect RNCs to core network nodes.

The location and the load of each RNC have a direct influence on the core network topology planning.

2.1.3.3 Core Network Planning Output

The core network planning mainly deals with the assignment of RNCs to MSCs/SGSNs. The output of the core network planning can be generalized with the following items:

- The best location of nodes (MGw, MSC and SGSN etc.);
- The optimal number of nodes;
- The type/characteristic of nodes;
- The link topology and link type between RNCs (optional);
- The link topology and link type between RNCs and the core network nodes.

2.1.3.4 Core Network Planning Tools

Because of a greater routing diversity and reliability, a mesh topology will usually be used in the core network [58].

In [12], Harmatos divided the planning problem into two steps: based on a randomly generated initial solution, they first used the simulated annealing to decide the location of MGws, links between MGws and RNCs, as well as links between MGws and core network nodes. On the basis of the topology obtained from the first step, the second step aimed to find the optimal path for the traffic. In this step, the simulated annealing and simulated allocation (SAL), proposed by Pioro in [58], were applied and compared to find the cost optimal routing path for the required traffic. The result showed that the simulated annealing was good for small size problems, while SAL is superior for large size problems.

Ricciato et al. [57] implemented an integer linear programming (ILP) model to solve the linear combination of two objective functions. They first obtained the optimal solutions for the primary objective. Then the solutions obtained from the first step, together with other original problem constraints, worked as the constraints to solve the secondary objective.

2.1.4 Sequential Approach Summarization

As stated before, the goal of using a sequential approach to solve the UMTS network planning problem is to reduce the problem complexity. In fact, we end up with three different subproblems, where solving each one of them is easier than solving the whole problem. As a result, more details can be considered for each subproblem.

The sequential approach also has drawbacks. The major disadvantage of this approach is that each subproblem is considered independently from one another, which easily leads the overall planning to a local optimum. Most of the time, the combination of subproblem optima does not provide an optimal solution to the global problem. Moreover, there is no integration strategies being developed yet to incorporate all partial solutions in order to obtain a global solution. Integration techniques are very difficult to be developed because we need to have a global view of the network.

A different way of planning the UMTS network is to use a global (also called integrated) approach.

2.2 Global Approach

As mentioned previously, a sequential approach breaks down the whole network planning problem into three subproblems and solves them in sequence. A global approach, however, considers at least more than one subproblem simultaneously. Since interconnections between the subproblems are taken into consideration, the global approach has the advantage of providing solutions that are closer to the global optimum. In Figure 2.6, all three subproblems are considered simultaneously. As a result, the optimal solution can be obtained at the expense of computational complexity. As stated in section 1.2, all three subproblems are NP-hard. Consequently, the global planning problem of UMTS networks is also NP-hard.

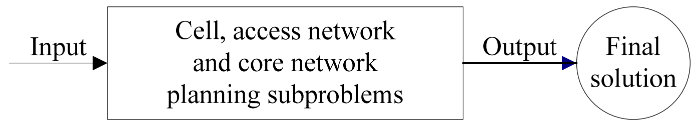


Figure 2.6: The global approach for UMTS network planning

The network planning objective in a global approach is more general than its counterparts in sequential approaches due to the fact that the global approach deals with two or three subproblems together. Existing studies mainly focus on minimizing the network cost, while satisfying network performance requirements. There are three research directions: the UTRAN planning, the access and core network planning, and the whole network (the cell, the access network and the core network) planning.

Zhang et al. [19] proposed to use a global approach to solve the UTRAN planning problem, aiming to position node Bs and RNCs, decide the number of the nodes, and the link connections between them with the minimal network cost.

Chamberland and Pierre [59] focused on GSM access and core network planning subproblems. With given locations of base stations, nodes of access network and core network were located, along with node types. The interconnections between the access network and the core network, as well as link types were also decided. They designed an automatic planning tool based on tabu search to find the minimum cost. With certain modifications, the network model and implemented algorithm proposed in this paper can also be applicable to the UMTS network.

St-Hilaire et al. [14] developed a mathematical programming model for global planning of UMTS network in the uplink direction. Within an acceptable amount of time, a local search heuristic was implemented to find a good solution for the node location, link connections between network nodes, and the type of the network nodes and links. In their later work [16], a planning tool based on the tabu search is proposed to solve this global planning problem with improved performances.

2.3 Section Remarks

The primary task of network planning is the topology planning, which demonstrates the network node placement and the link connection. A well designed network topology provides operators a proper investment budget on the network infrastructure and a good basis for further network configuration refinement.

There are two main branches commonly applied on the UMTS network planning problems: the sequential approach and the global approach. In the sequential approach, the planning problem of the overall UMTS network is divided into three subproblems: the cell planning subproblem, the access network subproblem and the core network subproblem. Thus, the overall network planning problem is solved by tackling the three subproblems sequentially.

Since each subproblem is relatively easier to solve than the whole network planning problem, the sequential approach can be time efficient. Moreover, more details can be considered in each subproblem. However, since each subproblem is considered independently, the combination of the optimal solution of each subproblem might not be able to construct an optimal solution for the overall network planning problem.

A global approach consists of solving more than one subproblem simultaneously, where all interactions among the subproblems are taken into consideration. As a result, the global planning gains the advantage of finding solutions that are closer to the global optimum. If all three subproblems are considered simultaneously, the global optimal solutions can be obtained. However, since each subproblem has already been proved to be NP-hard, we can easily assume that the global planning problem is also NP-hard. In fact, the global approach becomes much more complex than the sequential approach due to the consideration of subproblem interactions. This urges the development of efficient automatic planning tools to finally build a cost-effective network.

Chapter 3

Network Model and Planning Tools

As mentioned previously, meta-heuristics are usually used to solve combinatorial optimization problems. In this chapter, we will firstly describe the mathematical model for the global planning problem of UMTS networks. This model will be used to evaluate the quality of the proposed planning tools. Then, the next two subsections provide the detailed design strategy of the new planning tools based on genetic algorithm and simulated annealing respectively. A planning tool based on tabu search will also be briefly described in the following subsection for performance comparison purpose. Finally, a novel planning tool based on the cooperation between genetic algorithm and tabu search is designed in order to obtain better solution quality for the network planning problem.

3.1 Exact Mathematical Model

In the UMTS network topology planning problem, finding out the network topology while minimizing the cost is the ultimate goal. Several constraints are used to define the solution space (or guarantee the feasibility of potential solutions), which will be explored during the search process of the implemented meta-heuristics. The model for the global planning problem has already been studied and proposed in [14]. However, for the sake of completeness, the model is summarized in Figure 3.1. It is only formulated in words for readability reasons.

Objective function: min($C_N(\mathbf{x})+C_L(\mathbf{x})$ **)**
Subject to:

- Uniqueness constraints
- Assignment constraints
- Equipment capacity constraints
- Link capacity constraints
- Traffic flow conservation constraints

Figure 3.1: The UMTS network planning model

As we can see, the objective function is composed of two items: the cost of nodes (C_N) and the cost of links and interfaces (C_L). These two items are represented by equations 3.1 and 3.2 respectively.

$$C_N = \sum_{t\in T_1} b_1{}^t \sum_{i\in S_1} x_1{}^{it} + \sum_{t\in T_2} b_2{}^t \sum_{j\in S_2} x_2{}^{jt} + \sum_{t\in T_3} b_3{}^t \sum_{k\in S_3} x_3{}^{kt} + \sum_{t\in T_4} b_4{}^t \sum_{l\in S_4} x_4{}^{lt} \qquad \text{(eq. 3.1)}$$

$$C_L = \sum_{i\in S_1}\sum_{j\in S_2}\sum_{m\in M_{12}} a_{12}{}^{ijm} v_{12}{}^{ijm} + \sum_{j\in S_2}\sum_{k\in S_3}\sum_{m\in M_{23}} a_{23}{}^{jkm} v_{23}{}^{jkm} + \sum_{j\in S_2}\sum_{l\in S_4}\sum_{m\in M_{24}} a_{24}{}^{jlm} v_{24}{}^{jlm} \qquad \text{(eq. 3.2)}$$

In these equations, S_1, S_2, S_3 and S_4 are respectively the set of potential sites to install the node Bs, the RNCs, the MSCs and the SGSNs. T_1, T_2, T_3 and T_4 are sets of node B types, RNC types, MSC types and SGSN types. Similarly, M_{12}, M_{23} and M_{24} are respectively the set of link and interface types that can be used to connect node Bs to RNCs, RNCs to MSCs and RNCs to SGSNs. Variables a (a_{12}, a_{23}, a_{24}) and b (b_1, b_2, b_3,

b_4) respectively represents the cost of a given link type between two locations and the cost of a given type of network equipment. Finally, v (v_{12}, v_{23}, v_{24}) and x (x_1, x_2, x_3, x_4) are two decision variables. The first one corresponds to the number of links of a given type between two locations and the second one simply indicates if a piece of equipment of a given type is installed (or not) at a given location.

As shown in Figure 3.1, the solution space is defined by a series of constraints. The uniqueness constraints limit the number of node that can be installed at each potential site to one. The assignment constraints state that each TP must be covered by one node B (while respecting the minimum receiving power and SIR_{min}), each node B must be connected to one RNC which must be linked to one MSC and one SGSN. The constraints for the equipment capacity guarantee that the total volume of outgoing traffic is smaller than the switch fabric capacity and the total number of outgoing links is smaller than the total number of available interfaces on the node. Link capacity constraints make sure that the total traffic on all the links between two locations is smaller than the total capacity of these links. Finally, the traffic flow constraints ensure that for each node installed, the total outgoing traffic will be equal to the total incoming traffic. For more details about the mathematical model and the mathematical formulations of the constraints, please refer to [14].

The previous mathematical model has been shown to be NP-hard. As a result, approximate algorithms are expected to build network planning tools. In the following subsections, planning tools based on genetic algorithm, simulated annealing, tabu search and a new cooperative method will be described.

3.2 Planning Tool Based on Genetic Algorithm

The genetic algorithm is a well-known meta-heuristic and is frequently used to solve network planning problems [25, 26]. It is a class of evolutionary algorithm as it is an analogue of the natural selection from Darwin's theory.

The genetic algorithm starts the reproduction process by generating an initial population. This initial population represents a subset of possible solutions to a particular problem. Each individual in the population is then evaluated by the given objective/cost function to obtain a corresponding fitness. For maximization (minimization) objective function, the higher (lower) the objective function value is, the higher (higher) the individual fitness will be. The parent individual selection is based on the individual fitness. The higher the fitness level is, the higher the probability that the individual will be selected to reproduce. By applying different recombination operators such as crossover or mutation (or both), offspring are generated. This reproduction process shares some characteristics from both parent individuals. Finally, offspring will be integrated into the current population to create the next generation by either replacing the whole current population or part of it. This reproduction process favors the "mating" of the more fit individuals, such that the promising search areas are explored. With a good design, the genetic algorithm will usually converge to "acceptably good" solutions to the problem "acceptably quickly" [62]. The whole process stops when the predefined conditions are reached. There are several basic components in the genetic algorithm. The following subsections provide a close look at each of these components to find a good design scheme for the global UMTS network planning problem.

3.2.1 Initial Population

As mentioned previously, the initial population is a group of potential solutions for the problem. There are two factors that need to be taken into consideration for the initial population: the population generation and the population size.

3.2.1.1 Initial Population Generation

The first step of applying the genetic algorithm on a particular problem is to decide the population representation scheme. There are two classes of encoding schemes: the binary encoding scheme and the non-binary encoding scheme. The choice of the encoding scheme depends on the problem.

In the binary encoding scheme, the decision variable is represented by 0 or 1. For example, in the UMTS network planning problem, the binary code could be used to indicate if the corresponding site is installed with a network node or not. Figure 3.2 is an example of 0-1 binary encoding scheme, where 1 means the corresponding site is installed with a network node and *l* is the solution string length [63].

Column (site)	1	2	3	4	5	…	*l*-1	*l*
Bit string	0	1	1	0	1	…	0	1

Figure 3.2: The binary encoding scheme

However, in some other problems, it is not efficient or suitable to use the binary encoding scheme. For instance, in the UMTS network planning problem, different types of network nodes are available for each site. In face, we need to decide not only if a network node will be installed on a specific site, but also which node type will be selected. For example, if three types of nodes are available for each site, then the value for each site will be chosen from four defined values: 0, 1, 2, and 3 (0 means there is no piece of equipment installed). Figure 3.3 is a solution string example for a UMTS network with five potential node B sites, three potential RNC sites, two potential MSC sites, and two potential SGSN sites.

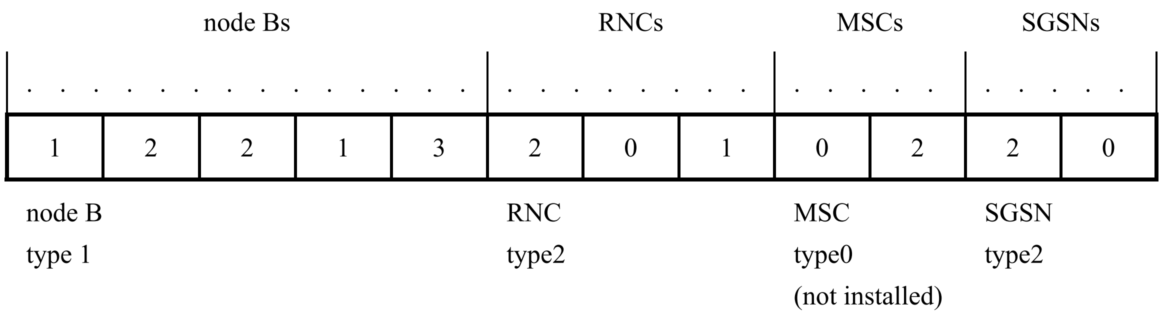


Figure 3.3: A GA coding scheme for UMTS network planning

Genetic algorithm is stochastic in nature [63]. For non-binary encoding, random values uniformly distributed between 0 and 3 must be generated. Once we have this structure, it is easy to evaluate the feasibility and the cost of a given solution.

Some studies proposed to seed the genetic algorithm an initial population with known good solutions, which might be obtained from other heuristic methods. As a result, this would speed up the search and hopefully find a better solution. However, some other studies argued that this could lead to premature convergence with a poor solution [64, 65].

3.2.1.2 Population Size

After deciding an appropriate way to generate the initial population, the next step deals with the population size. A population is a group of potential solutions for the problem. On one side, if the population size is too small, the search space will not be sufficient and will lead the search to premature convergence. On the other side, if it is too big, the search will be inefficient and the solution will not be found within a reasonable computation time [66]. Choosing an appropriate population size is always a trade-off between solution quality and execution time.

Intuitively, for a given solution string length, there should exist a corresponding optimal population size. Goldberg [67] proposed in his early research work that the population size should increase as an exponential function of the string length. Some empirical results indicated that a population size of 30 is adequate in many cases [68, 69]. The later research work of Goldberg and his colleague showed the linear dependency relationship between the population size and solution string length [70].

Based on the assumption that all values are presented in the initial population, Reeves [71] worked on finding the minimum size for a meaningful search, which meant that every point in the search space should be reachable from the initial population by crossover only. For binary strings, with string length of l, the minimum population size (N) can be expressed as: $N \approx \left| 1 + \log(-l / \ln P_2^*) / \log 2 \right|$, where P_2^* takes the value of 99.9%, which means that the calculated minimum population size N will provide the meaningful search with the probability of 99.9% [71]. For the non-binary / q-ary alphabets (q possible values for a position in a string, where $q>2$) encoding scheme, the derived expressions can be converted numerically into graphs for specified confidence levels. Figure 3.4 gives an example for 99.9% confidence level.

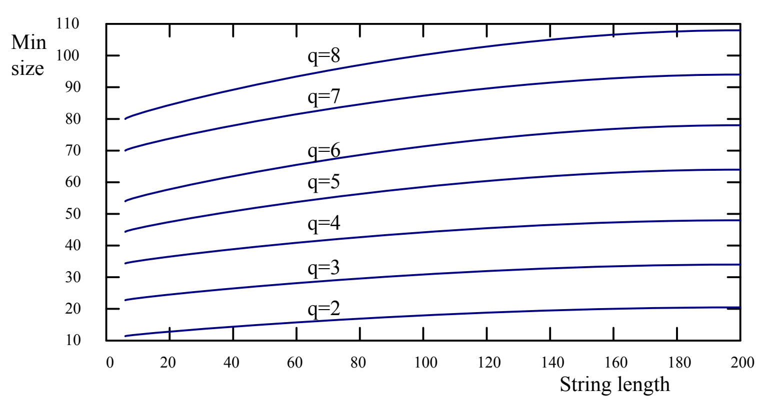


Figure 3.4: Solution string length vs. the minimal meaningful population size [71]

After building the initial population, each solution string (individual) in the population will be evaluated by the given objective functions (i.e. eq. 3.1 and 3.2) in order to find the fitness. Based on the fitness of each individual, the more fit individuals will be selected as parents to reproduce and finally form the next generation of population.

3.2.2 Selection

Genetic algorithm is a population-based search method [72]. The search process starts by selecting better individuals from the initial population according to the individual fitness and reproducing them to generate offspring. The offspring will be improved generation by generation in terms of fitness.

During the iterative searching process, the selection plays a critical role as it determines the search direction. There are two factors that need to be considered during the selection process. The first one is the selection pressure, which is the degree that the better individuals are favored to. The second concept is the convergence rate [73]. On one hand, high selection pressure will speed up the convergence rate over the search process, which will increase the premature chance and lead to a sub-optimal solution. On the other hand, low selection pressure will slow down the convergence rate. It will make the search process take a longer time to find the solution or it may not be able to find a solution within an expected computation time. There are a number of selection methods. The following subsections will briefly introduce several popular selection methods.

3.2.2.1 Proportional Selection

A typical example of the proportional selection method is the roulette-wheel selection. Giving a maximization objective example, the circumference of a wheel is divided into a number of parts (the number equals to the population size). The length of each part is proportional to the proportion of the objective function value of an individual with respect to the sum of objective function values all individuals [26]. The individual landed on when the wheel is spun will be selected for reproduction. This process will be repeated until the parent individuals with predefined number are selected.

Let $\{x_1^{(t)}, x_2^{(t)}, ..., x_N^{(t)}\}$ denote the population of size N at generation t and $f(x_i^{(t)})$ be the fitness of individual $x_i^{(t)}$, which is corresponding to the actual objective function value of $x_i^{(t)}$ in the roulette-wheel selection. Then, the probability of selecting individual i at generation t is given by equation 3.3 [72]:

$$p(x_i^{(t)}) = \frac{f(x_i^{(t)})}{\sum_{f=1}^{N} f(x_i^{(t)})} \qquad \text{(eq. 3.3)}$$

Since the roulette-wheel selection is based on the actual value of the objective function of an individual, "super" individuals will have more opportunity to be reproduced.

3.2.2.2 Tournament Selection

In tournament selection, firstly a group of individuals is randomly chosen from the ranked population. Then, the individuals in this group (tour) are ordered based on their ranks. The best individual in the group will be selected to reproduce. The process will stop when the parents with the expected number are chosen. In tournament selection, the tournament size influences the selection pressure. Usually, the tournament size, denoted q, is set to 2. Then, the best one will be selected to reproduce in this 2-individual tour. Increased selection pressure can be provided by simply increasing the value of q, because, on average, the individuals selected from a larger group will have better fitness than from a smaller group [73].

For the q-tournament selection, the probability of selecting individual i at generation t is given by equation 3.4 where the best individual has the lowest index (i) [72].

$$p(x_i^{(t)}) = \frac{1}{N^q}((N-i+1)^q - (N-i)^q) \qquad \text{(eq. 3.4)}$$

3.2.2.3 Linear Ranking Selection

In linear ranking selection, individuals are ordered according to their fitness. Each individual is assigned a rank according to the objective function value. For example, the worst individual in the population of size N will have the rank value of 1, while the best individual gets rank N [74]. The probability of selecting individual i at generation t is given by equation 3.5, where $max+min=2$, $1 \le max \le 2$, $p(x_i^{(t)}) \ge 0$ for (i=1, 2,…, N), and $\sum_{i=1}^{N} p(x_i^{(t)}) = 1$ for each generation t [72].

$$p(x_i^{(t)}) = \frac{1}{N}(\min + \frac{(\max - \min)(rank(x_i^{(t)}) - 1)}{N-1}) \qquad \text{(eq. 3.5)}$$

The performance comparison of the three selection methods is presented in Tables 3.1 and 3.2 in terms of solution quality and convergence rate respectively where N is the problem size [72]. Table 3.1 shows that the linear ranking selection and the tournament selection are superior to proportional selection in terms of average fitness. Table 3.2 provides the standard deviation of the convergence time of the three selection methods. The linear ranking selection has the best convergence rate, which is 3-4 times the convergence rate of the tournament selection. The tournament size of $q = 2$ for the tournament selection and *max*=1.1 for the linear ranking selection are recommended.

Table 3.1: Comparison of solution quality (greater is better) [72]

N	**unit**	**Proportional selection**	**Tournament selection**	**Linear ranking selection**
10	$\times 10^{-2}$	6.277 ±1.159	6.476 ±1.093	6.459 ±1.095
20	$\times 10^{-3}$	8.394 ±1.317	9.368 ±1.018	9.384 ±1.065
30	$\times 10^{-3}$	2.608 ±0.291	2.858 ±0.235	2.850 ±0.230
40	$\times 10^{-3}$	1.143 ±0.115	1.231 ±0.098	1.229 ±0.095
50	$\times 10^{-4}$	5.958 ±0.504	6.416 ±0.496	6.412 ±0.499

Table 3.2: Comparison of convergence time [72]

N	**unit**	**Proportional selection**	**Tournament selection**	**Linear ranking selection**
10	$\times 10^{5}$	2.778 ±0.928	1.510 ±0.924	0.524 ±0.239
20	$\times 10^{5}$	8.274 ±3.504	3.953 ±3.850	0.917 ±0.334
30	$\times 10^{5}$	13.097±5.434	6.254 ±6.072	1.603 ±0.537
40	$\times 10^{5}$	19.443±5.350	8.052 ±8.001	2.456 ±0.529
50	$\times 10^{5}$	27.611±9.206	12.713±1.696	4.038 ±0.959

After the parent individual selection, the next task is to reproduce them by applying two operators: crossover and mutation.

3.2.3 Crossover

Integer or binary valued recombination strategies are usually termed as crossover methods [26]. Different representations need different forms of crossover. For the UMTS network planning problem, for example, linear crossover will be applied.

The linear crossover operator can be represented as a binary string/mask $m \in \{0,1\}^l$, where l is the string length [10]. For example, a two-point crossover for the string with $l = 6$ can be represented as: 1 1 0 0 0 1. Then, one offspring from string a (a_1, a_2, a_3, a_4, a_5, a_6) and string b (b_1, b_2, b_3, b_4, b_5,b_6) is calculated using the following operation: $m \otimes a \oplus \overline{m} \otimes b$, where $\overline{m}$ is the complement of m and $\oplus, \otimes$ denote the component-wise addition and multiplication respectively. As a result, the offspring will be composed of the following components: (a_1, a_2, b_3, b_4, b_5, a_6).

There are three main linear crossover methods: one-point crossover (1X), *m*-point crossover and the uniform crossover (UX). Davis compared the three crossover methods and concluded that the uniform crossover outperforms the other two methods [75]. In uniform crossover, by generating the pattern of 0's and 1's stochastically (using a Bernoulli distribution), we obtain the uniform crossover mask like 1 0 1 0 0 1. Bernoulli parameter *p* can either be defined as *p*=0.5 or bias one parent by choosing an appropriate value of p.

3.2.4 Mutation

There are two modes for the recombination operator to generate new individuals: crossover $-AND-$ mutation or crossover $-OR-$ mutation. Some studies insist that $-AND-$ mode will finally find better solutions, while others believe that the search process should always do something, either the crossover or the mutation, but not both [10].

If $-AND-$ mode is applied, the next step will be mutation. A mutation operator can be generated by using a Bernoulli distribution. An operator against a string such as 0

1 0 0 0 1, means that the value of positions 2 and 6 will be changed. For the binary encoding scheme, the position value is changed either from 1 to 0 or vice versa. For the non-binary encoding, changing a position value will be accompanied with deciding the new value. This decision can be either stochastic or sequential.

There is a crucial parameter for the mutation operator: mutation rate. Some practitioners suggest the optimal mutation rate of $1/l$, where l is the string length. It means to randomly mutate one bit per string. Other researchers believe that this fixed mutation rate is too small to be efficient, especially when the GA has converged. Since crossover works on combining good individuals at the initial search stage, the role of mutation is not apparent. With the progress of the GA search, the crossover operator becomes less productive, while mutation begins to take more responsibility for the search process. In fact, when the search starts to converge, mutation becomes the main factor for the search. This means that the adaptive crossover and the mutation rate will outperform the fixed rate for these two operators. Davis [75] proposed the use of the adaptive crossover and the mutation operator fitness to improve genetic algorithm search efficiency.

3.2.5 New Generation Construction

Once operators like selection, crossover and mutation are applied to some individuals in the population, new offspring can be generated. We have already seen that considerable efforts have been spent to obtain a good solution. If we simply replace the whole parent population with a newly generated population, good ones in the parent population will be thrown away with no opportunity of further reproduction. To avoid this, some studies worked on finding an appropriate way to construct the new generation of the population. One of them is the steady-state reproduction [75], where the number of offspring to be generated is an important parameter in order to have a steady-state reproduction. In the steady-state reproduction, *n* offspring are generated through the reproduction operations (as mentioned above). Then, the generated offspring will replace the worst *n* individuals from the parent population. Instead of generating a full size (*N*) new population, the number of offspring that needs to be generated will be decided first. The steady-state reproduction can be done in three steps [75]:

- Generate *n* offspring through reproduction;
- Delete the *n* worst individuals from the parent population;
- Evaluate and insert the children into the parent population to construct a new generation.

Typically, practitioners generate and insert just one or two offspring at a time [75].

It is important to note that the steady-state reproduction doesn't really perform better than replacing the whole previous generation if the offspring that are going to be integrated to the parent population are duplicated to the parent individuals. Davis [75] proposed the steady-state without duplication to solve this problem. The child individuals that are duplicates with current parent individuals will be discarded, so that the population members will always be different from each other. According to Davis's study, the overhead caused by duplication comparison is negligible compared to the time spent in optimization. In many optimization problems, there will be a set of complicated constraints for the problem. The infeasible children probably are generated from two feasible parents. Usually, the infeasible individuals can be ignored to prevent them to be inserted to the population.

3.2.6 Genetic Algorithm Design

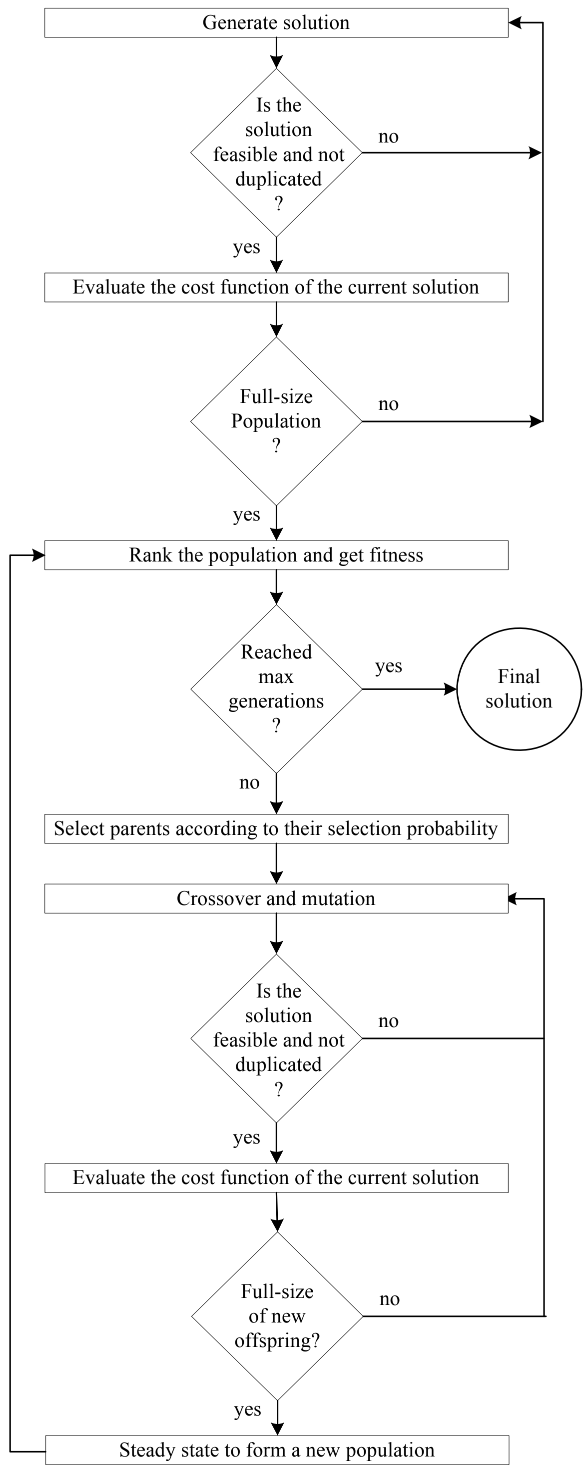


Figure 3.5: Genetic algorithm

Figure 3.5 is the proposed genetic algorithm to solve the global planning problem of UMTS networks. Individuals in the initial population are randomly generated. Each individual can be represented as a one-dimensional array, where each element of the array represents the status of a given location (i.e. whether or not a network node is installed and which type of node is installed).

Once an individual is feasible and not duplicated, the next step is to evaluate the "quality" of the solution by computing its cost. The latter is obtained by using equations 3.1 and 3.2. New members/solutions will be generated until the given population size is reached. The population size is set to 50.

The following step is to rank all individuals from the population according to their cost function values. The individual with the lowest cost will have the highest rank (such as 50) or the highest fitness. Individuals will be selected using the linear ranking selection, with *max*=1.3, as the parents to reproduce.

Crossover and mutation are both taken as reproduction operators. The uniform crossover with the crossover rate of 0.5 is applied without the bias of one parent. Since mutation takes more charge of increasing diversity rather than crossover when the solution space starts to converge, the mutation rate is set to $1/l$ (i.e., only one position value is changed) for the first three quarters of generations and $1/2$ (i.e., half of position values are changed) for the remaining generations.

Steady state reproduction is applied to construct new generations, where the number of new generated offspring takes the value of 2. After calculating the cost, new offspring are integrated to the parent population to form the next generation. The search process will stop when the maximum number of generations (set to10,000) is reached.

It is important to point out that all the parameters mentioned above were selected after several trials. The best results were obtained with these parameters.

3.3 Planning Tool Based on Simulated Annealing

Annealing is a technique from statistical mechanics [76]. In the annealing process, a solid material is initially heated over the melting point. At this point, the solid material turns into liquid with randomly dispersed particles. Then the material is cooled down. During the cooling process, the material structural property highly depends on the cooling rate. If the material is cooled properly into low energy states and is controlled to stay in each state for certain duration, all particles re-crystallize to a more ordered state. Finally, the solid stable state is reached with the most ordered state and the minimum energy. However, if the material initial temperature is too low and/or if the material is cooled too fast, there could be imperfection in the crystals with higher energy.

Firstly proposed by Kirkpatrick et al. [8], simulated annealing analogizes the annealing process, which is also the reason that this algorithm is called "simulated annealing". Simulated annealing is a powerful algorithmic approach for general combinatorial minimization problems. It makes use of the cost function to evaluate the solution (state) quality during the solution search process (annealing). The temperature is the analogue of the control parameter guiding the solution search to find the global optimum. If the temperature decreases too fast, the final solution may be trapped in a local optimum with higher energy level. Meanwhile, if the algorithm convergence rate is too fast, the final solution may also be trapped in a local optimum. Therefore, gradually lowering the temperature in a well-controlled way and accepting temporary deteriorations are necessary approaches for finding the final solution that significantly approaches the global optimum in polynomial time. The deteriorations are controlled by the configuration parameters to guarantee their acceptances. Table 3.3 shows the mapping relationship between physical annealing and simulated annealing (cost-oriented) [77]:

Table 3.3: Physical annealing vs. simulated annealing [77]

Physical Annealing	Simulated Annealing
Material States	Feasible Solutions
Material Energy	Cost Function Value
Material State Changes	Neighboring Solutions
Temperature	Control Parameter
Material Frozen State	Heuristic Solution

The annealing process starts at a high enough temperature point. Along with the cooling phases, the solid will reach the thermal equilibrium at every temperature *T*, where no further state improvement is expected with high probability, characterized by the Boltzmann distribution as shown in equation 3.6 [78], where *Z(T)* is a normalization factor, k_B is the Boltzmann constant and $exp(-\frac{E}{k_B T})$ is the Boltzmann factor.

$$P_r\{\mathrm{E} = E\} = \frac{1}{Z(T)} * \exp(-\frac{E}{k_B T}) \qquad \text{(eq. 3.6)}$$

When the well-controlled decreasing temperature approaches zero, the Boltzmann distribution will concentrate on the minimum energy state with non-zero probability.

In 1953, Metropolis et al. [79] proposed Monte Carlo method (also called Metropolis algorithm) to generate a solid state sequence, which simulates the solid state changing to the thermal equilibrium at a fixed temperature *T*. Given the solid current state (current positions of all particles), a particle is randomly chosen to do a small random perturbation. The energy difference (ΔE) between the current state and the new state will be calculated. If the new state has a lower energy ($\Delta E \leq 0$), the current state will change to the new state. Otherwise ($\Delta E \geq 0$), this acceptance probability of the new state will be evaluated by the Boltzmann factor. The evaluation rule is also called Metropolis criterion [78]. Following the rule, solid state will finally transit to thermal equilibrium. For the whole process, the distribution of state acceptance probability approaches Boltzmann distribution [78].

Simulated annealing can be treated as a sequence of Metropolis algorithms along with decreasing temperature *T*. Thus, lowing the temperature *T* in a well-controlled way, global optimum can be obtained by randomly sampling the neighborhood and accepting the deteriorate solution according to Metropolis criteria. There are three key processes of simulated annealing, which affect the quality of the final state: the state initialization, state transformation/perturbation and temperature cooling schedule.

3.3.1 State Initialization

Simulated annealing starts with a random non-optimal initial state/solution. The same coding scheme as in genetic algorithm can also be used in simulated annealing. Improving the algorithm efficiency is always an interesting topic. Instead of randomly generating an initial solution, some studies proposed to start the algorithm with a good quality initial solution. An easy way to obtain relatively good solutions is to use a local search heuristic. This strategy is called the two-stage simulated annealing (TSSA) [80], which is proven to be able to improve the final solution quality, as well as decrease the computation time. Please see [80] for more information on TSSA. Obviously, the initial solution must be feasible. In this thesis, TSSA is applied by implementing a greedy algorithm to obtain a good initial solution.

3.3.2 State Transformation/Perturbation

Based on the initial solution, the SA search process will transform/perturb the current state into a new state in its neighborhood. This transformation is also known to be the Markov chain transformation, because the current state only depends on the previous state. There are two common algorithms used for choosing another state j from the neighborhood of the current state i: Metropolis algorithm and Glauber algorithm [81].

In Metropolis algorithm, if the difference of the cost functions between two states ($\Delta C_{ij} = C(j) - C(i)$) is smaller than or equal to zero (i.e. a better or same quality solution is found), the acceptance probability of state j for the next state is set to 1. On the other side, if the difference is greater than zero (i.e. a worse solution is found), then the acceptance probability of the new state will be decided according to the Metropolis criterion. The latter is a two-step process that works as follows:

Step 1: Generate a random number $\in [0,1)$;

Step 2: If the generated random number is less than $\exp(-\frac{\Delta C_{ij}}{T})$, then set the acceptance probability to 1. Otherwise, reject the solution.

In summary, the Metropolis algorithm will accept all downhill moves (better solutions) and part of uphill moves (worse solutions) according to the Metropolis criteria to avoid being trapped into the local optimum. Glauber algorithm, on the other side, evaluates the acceptance of all new states (downhill and uphill moves) according to a certain decision criteria [81]. As we can see, the Metropolis algorithm is greedier than the Glauber algorithm. For more information of Glauber algorithm, please refer to [81].

3.3.3 Cooling Schedule

As we can see from the decision criteria, such as Metropolis criteria, the acceptance probability of uphill moves strongly depends on the temperature *T*. Initially, when *T* is very large, a big percentage of uphill moves will be accepted. This guarantees the diversity of the search space. With the decrease of the temperature, the acceptance of the uphill moves decreases. Finally, when the temperature decreases to a certain value, no uphill moves will be accepted. Through the controlled uphill moves, simulated annealing can avoid the solution being trapped in a local optimum and hopefully find the global optimum.

A well-designed cooling schedule is the key for a successful simulated annealing. Several aspects need to be defined for the cooling schedule [78]: the initial temperature, the decrement rule of temperature, the termination condition, and the number of iterations at each temperature (i.e. the length of the Markov chain).

3.3.3.1 Initial Temperature

Since the Metropolis algorithm will iteratively visit the states/configurations at a given fixed temperature with a certain probability, there is a chance to miss the best state. To reduce this chance, the initial temperature (T_0) has to be set high enough to be able to visit almost any neighborhood state. At the same time, it cannot be set too high. Otherwise, the search will become completely random and will not act as the defined simulated annealing algorithm. Finding an appropriate starting temperature is a crucial step.

Kirkpatrick et al. [8] proposed to choose a high value of T_0 and set the acceptance decision criteria of the worse solution as $X_0 = \exp(-\frac{\Delta C_{ij}}{T})$. Then a number of moves will be made. After that, the initial temperature can be calculated according to $T_0 = \frac{\Delta C_{ij}}{\ln(X_0^{-1})}$.

Define the acceptance ratio X as the ration of the number of accepted moves over the total number of moves. Given an empirical fixed value $X_0 = 0.8$, double the value of T_0 if $X < X_0$. Keep doing this until acceptance ratio $X > X_0$. $X_0 = 0.9$ is suggested by Das et al. [81].

3.3.3.2 Decrement Rule of Temperature

Once we got the initial temperature, the temperature decrement rule needs to be decided. According to Das et al. [81], two basic types of cooling schedules are commonly used. The first one is called the exponential cooling schedule, which was first proposed by Kirkpatrick et al. [8]. It is given by equation 3.7, where α is a constant and takes the value of 0.95 empirically.

$$T_{k+1} = \alpha * T_k \text{ , } (k = 0, 1, 2, \ldots) \qquad \text{(eq. 3.7)}$$

Some empirical studies said that α should take the value ranging from 0.8 to 0.99 [8]. The higher value of α, the longer the search process. Therefore, a better solution can be obtained. Instead of the constant ratio (T_{k+1} / T_k) of the temperature, some studies proposed a fixed number of decrement steps, say K. Then the calculation of temperature T_k at step k (k = 1, …, K) is shown in equation 3.8 [83, 84] :

$$T_k = \frac{(K - k)}{K} * T_0 \qquad \text{(eq. 3.8)}$$

3.3.3.3 Termination Condition

It is usual to set the final temperature to zero as the stop criteria. However, this might cause longer execution time of the algorithm. In practice, when the temperature approaches zero, the acceptance ratio of worse move usually already reaches zero. Therefore, the stop criterion can either be a suitable final temperature or when the equilibrium is reached. At this point, the state probability distribution approaches the Boltzmann distribution.

Das et al. [81] suggested that if $\frac{C_f * T_f}{C_0 * T_0}$ is less than a value, such as 10^{-5}, the search process should be terminated. C_f and C_0 are respectively the final and initial value of the cost function while T_f and T_0 are the final and initial value of temperature respectively.

3.3.3.4 Iterations at Each Temperature

To obtain the thermal equilibrium at each temperature *T* in the cooling sequence, a sufficient amount of transformation/perturbation iterations should be done. A constant number of iterations at each temperature is one of the choices. Lundy [85] proposed a scheme with one iteration at each temperature *T*. The temperature is lowered at a very slow speed according to equation 3.9, where β is a suitable small value.

$$T = \frac{T}{(1 + \beta T)} \qquad \text{(eq. 3.9)}$$

3.3.4 Performance Improvement Strategies

The simulated annealing gains its popularity because of its ability to solve NP-hard combinatorial minimization problems. However, when the temperature becomes low, many moves may be rejected before the search process actually moves to the next state. This leads to a long computation time, which is an obstacle for the application of simulated annealing. Many efforts have been done to overcome this drawback. There are mainly two effort directions. The first one focuses on a parallel implementation of simulated annealing [5]. The other one works on optimizing the cooling schedule. In this thesis, cooling schedule optimization will be used to improve the algorithm efficiency [5, 86, 87, 88].

Various adaptive cooling schedules have been proposed to improve the performance of simulated annealing [88]. One significant branch of them is Lam schedule, proposed by Lam et al. [5] in 1988. As already mentioned, to finally get the material in the minimum energy state, the material has to approach the thermal equilibrium at each temperature point, which needs infinite temperature decrement [5] or infinite computation time. By applying the Lam schedule to "cool down" the temperature, the balance between the final solution quality and the computation time was reached, where the temperature changing rate is also called Lam rate. The simulation results showed that the computation could be sped up to 24 times of the general schedule. Please refer to [5] for details of Lam schedule.

However, in Lam schedule, the user cannot predict the total number of moves for the whole search process and when the process should end. Swartz et al. [89] observed Lam schedule results from a large amount of experiments and worked out a default number of total moves with a high probability to find the optimum solution as shown in equation 3.10, where N_c is the number of variables that needs to be solved.

$$total\_moves = 1500\,N_c^{4/3} \qquad \text{(eq. 3.10)}$$

Swartz also summarized that the acceptance ratio (defined as the ratio of the number of accepted moves over the total number of moves) and the new/remaining moves in Lam schedule follow the pattern shown in Figure 3.6 [86].

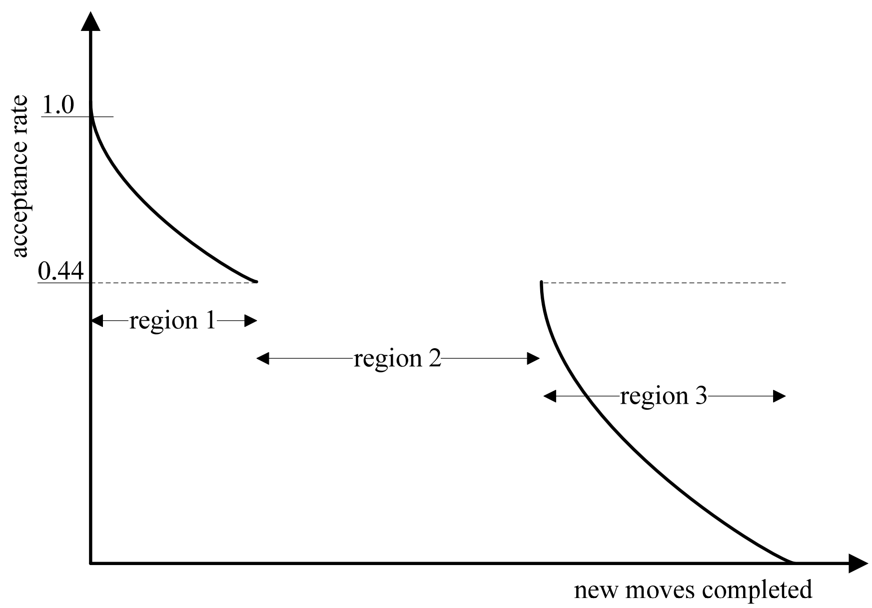


Figure 3.6: Pattern of acceptance ratio vs. new moves completed in Lam schedule [86]

Figure 3.6 is, in fact, the Lam rate changing process, where three periods were presented. The effect of the Lam schedule was the basis of the Swartz schedule. Without changing the effect of Lam schedule, Swartz et al. [89] simplified the temperature update schedule in the Lam schedule and obtained almost the same solution quality, as well as the same acceptance ratio curve as shown in Figure 3.6. Firstly, the acceptance rate starts at almost 100% and decreases in exponential pattern. During this period, about 15% of the total moves are finished. Then the acceptance rate stabilizes at about 44%. With this desired value of 0.44, the temperature decreases most rapidly, while satisfying the equilibrium condition. About 50% of total moves are done at this period. After that, the acceptance rate decreases again following an exponential pattern until the end of the whole process.

Based on Swartz's research results, Boyan [88], in 1998, further simplified the temperature update strategy as shown in equation 3.11, where the cooling rate α $(\alpha \in \{0.9999, 0.99999, 0.999999\})$ was recommended.

$$T_{+} = T * \alpha^{i} \qquad \text{(eq. 3.11)}$$

Boyan schedule keeps using the Metropolis criteria to decide if a worse solution will be accepted. In Metropolis criteria, assuming a constant ΔC_{ij}, the smaller the temperature *T* is, the smaller the acceptance probability will be. Since Boyan schedule is based on Swartz's modified Lam schedule, to follow the pattern in Figure 3.6, when acceptance rate is greater than Lam rate, *i* takes the value of +1 (decrease the temperature), and -1 (increase temperature) otherwise. For more information about the Boyan schedule, please refer to [88].

3.3.5 Simulated Annealing Design

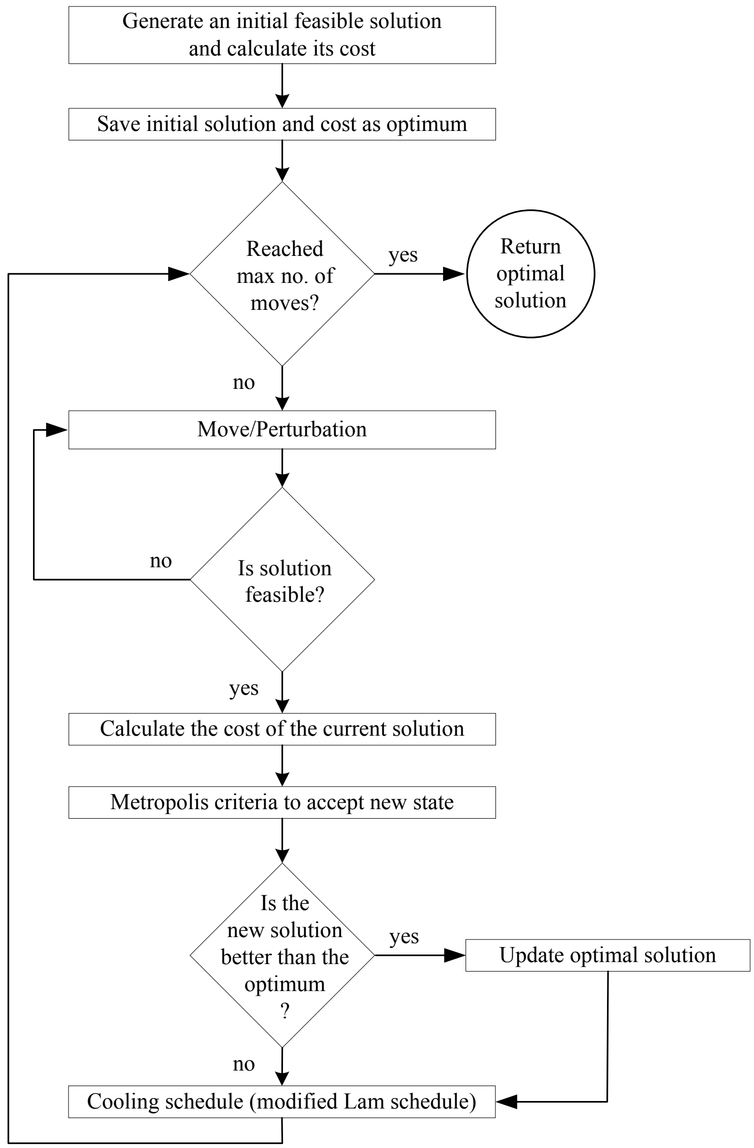


Figure 3.7: Simulated annealing algorithm

Figure 3.7 is the designed simulated annealing algorithm for the global UMTS network planning problem. Instead of randomly generating an initial solution, TSSA is used to get the search process a better and feasible start point. The initial temperature, T_0, is set to be 10 times the highest cost of candidate network nodes, which is 5,000,000. Equations 3.1 and 3.2 are used to evaluate the solution quality during “annealing”.

Based on a series of literature review and several experiments, the Boyan schedule is used to schedule the temperature cooling process. The total number of moves, proposed by Swartz et al., is given by equation 3.10. The decrement rate $\alpha = 0.999$, together with the initial temperature $T_0 = 5{,}000{,}000$, has been proved to be able to provide the best solutions. One iteration at every temperature T is implemented in the search process.

3.4 Planning Tool Based on Tabu Search

In this section, we briefly present the tabu search algorithm, as proposed in [16], for the global planning problem of UMTS networks. The latter will only be used for comparison purposes.

Tabu search is an adaptive search technique [82], using the best improvement local search as the basic ingredient. By allowing temporary solution degradation, tabu search avoids the search process being trapped into the local optimum. Two mechanisms, the short term memory and long term memory, can be applied to keep track of attributes of previously visited solutions and guide the tabu search process. The main steps of the tabu search algorithm, as proposed in [16], are outlined in Figure 3.8.

Step 1 Initial solution

Find an initial solution using a basic local search algorithm as proposed in [14].

Step 2 Tabu Search

Repeat Steps 2.1 to 2.2 for certain iterations

Step 2.1 Explore the neighborhood

2.1.1 Determine the best move, while taking into consideration the tabu moves and the aspiration criteria. The cost of the solution is the total cost of the network.

2.1.2 Determine the number of iterations (according to a uniform distribution) for which the chosen site is tabu

Step 2.2 TS best solution update

If the cost of the current solution is less than the cost of the best solution found so far, update this best solution.

Step3 Multi-start

Step 3.1 Update the solution

If the cost of the current solution is less than the cost of the best solution found so far, update this best solution.

Step 3.2 Stop condition

If the number of start is smaller than the maximum allowed number of starts, go to step 2. Otherwise, return the best cost found so far.

Figure 3.8: Tabu search algorithm as proposed in [16]

As shown in Figure 3.8, the tabu search algorithm initiates the search process based on the local search result, which provides tabu search a good start point. Then, the potential solutions in the neighborhood are explored. 100 search iterations in total are defined for tabu search. At each iteration, the neighbor with the lowest cost, while considering the tabu moves and the aspiration criteria, will be chosen to be the current solution. This solution will then be memorized as tabu for a given number of iterations, which is randomly generated between 5 and 9.

A multi-start tabu search is proposed in [16], where the number of the multi-start is set to 2. Since the duration of a solution staying in the tabu is randomly determined, the multi-start scheme may help the tabu search find different solutions for each search process.

At each iteration during the overall search process, the current solution will be compared with the best solution found so far in order to keep the best solution up to date. When the whole search process ends, the best solution over the whole search process will be the final output.

3.5 Planning Tool Based on the Cooperative Method

Amongst the three planning tools presented in the previous sections, an outstanding characteristic of the genetic algorithm is that it is a population-based algorithm. Since many potential solutions are visited at each generation, it is capable to explore a comparatively wide area solution space. On the other side, both simulated annealing and tabu search start the search process based on a single initial solution. The simulated annealing search process randomly moves to a new solution with respect to the acceptance criteria while the tabu search always finds the best solution in the neighborhood of the current solution while considering the tabu list. Tabu search is well known for its capability of intensive solution search with good performance [17]. To improve the performance of the planning tool and make better use of the superiority of different meta-heuristics, a cooperation between the tabu search and the genetic algorithm is proposed to solve the global planning problem of UMTS networks. Figure 3.9 is the implementation flow chart for the cooperative method.

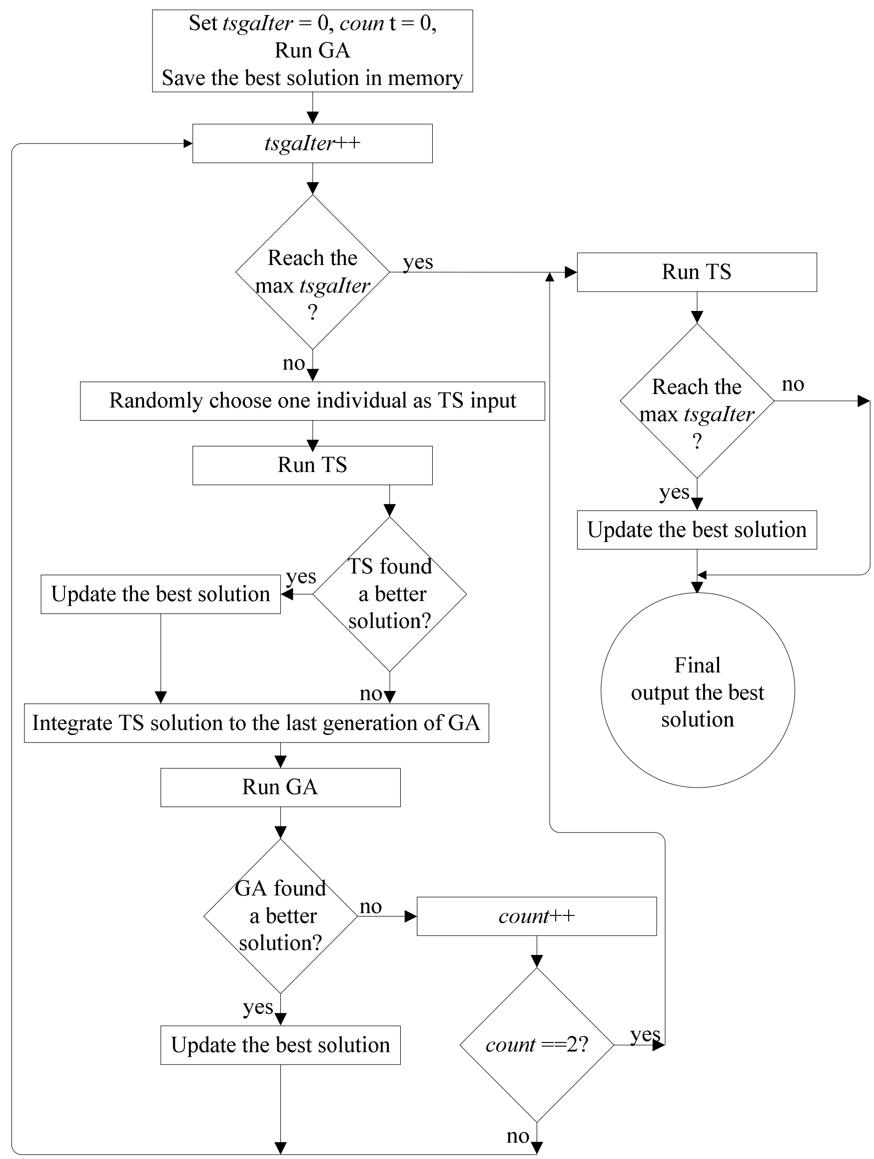


Figure 3.9: Cooperative method

In this cooperative method, at first, the genetic algorithm finds a group of solutions, using the same process as stated in section 3.2. After that, the tabu search and genetic algorithm search the solution space alternatively. A randomly chosen individual from the last generation of the genetic algorithm is used as the initial solution for the tabu search. The solution obtained from the tabu search is then integrated to the last generation of the genetic algorithm. As a result, a new generation is formed and the genetic algorithm can keep searching the solution space based on it. This alternative search process will stop when two consecutive cycles of .the algorithms do not have solution quality improvement (*count*=2) or the maximum cycle iteration number (*tsgaIter* = 3) is reached. A randomly chosen solution from the last generation of the genetic algorithm in the cycle will be sent to the tabu search to make the last solution search.

In the TS-GA cycle, the maximum generation number in the genetic algorithm is decreased to half of the one in the original genetic algorithm (5,000) and the multi-start is set to 1 in tabu search. Each time an algorithm finishes the search process, the final result will be compared with the best solution found so far and update it if necessary. Finally the best solution of the overall solution search process will be returned.

In the next chapter, simulation results based on these algorithms will be presented and analyzed.

Chapter 4

Experiment Design and Result Analysis

In this section, we will firstly outline how the simulations were developed, the simulation environment and the data used to run the simulations. Results are then presented and followed by a complete analysis.

4.1 Experiment Design

In our simulation, only the uplink direction is considered. The latter is very important when the amount of traffic is balanced between the uplink and the downlink direction. To model the traffic, the notion of test point is used. Each TP represents the traffic from several co-located mobile users in a given area. The behavior of the signal propagation is simulated by the model proposed in [25].

Three node B types, three RNC types, three MSC types and two SGSN types are available for the network design. Their features are respectively presented in Tables 4.1 to 4.4. Moreover, OC-3 and OC-12 links can be used to connect the node Bs to RNCs. DS-3 links are used to connect RNCs to MSCs and gigabit ethernet (GE) links are used to connect RNCs to SGSNs (see Table 4.5). The costs of various types of interfaces (ports) are also presented in Table 4.5.

Table 4.1: Node B characteristics

	Type 1	Type 2	Type 3
Capacity (circuits)	100	200	400
Capacity (Mbps)	120	240	480
Number of interfaces	1	2	2
Sensitivity (dBm)	-90	-100	-110
Cost ($)	20,000	30,000	50,000

Table 4.2: RNC characteristics

	Type 1	Type 2	Type 3
Switch fabric capacity (Mbps)	2000	5000	10,000
Number of node B interfaces	10	20	40
Number of MSC/SGSN interfaces	15	30	60
Cost ($)	50,000	90,000	120,000

Table 4.3: MSC characteristics

	Type 1	Type 2	Type 3
Switch fabric capacity (circuits)	100,000	200,000	300,000
Number of interfaces	50	100	150
Cost ($)	200,000	350,000	500,000

Table 4.4: SGSN characteristics

	Type 1	Type 2
Switch fabric capacity (Mbps)	20,000	40,000
Number of interfaces	16	32
Cost ($)	40,000	60,000

Table 4.5: Links and interfaces characteristics

Type	Capacity	Link cost ($/km)	Interface cost ($)
DS-3	2688 circuits	2,500	1,500
OC-3	155Mbps	1,500	2,000
OC-12	622Mbps	4,000	4,500
GE	1Gbps	4,000	2,000

As mentioned previously, each meta-heuristic has several tunable parameters and components, which will eventually have impact on the quality of the final solution. Here are the selected parameters and our implementation of those components for the proposed meta-heuristics:

Genetic algorithm:

- Maximum number of generations: 10,000;
- Population size: 50;
- Selection: Linear ranking selection with *max*=1.3;
- Crossover: Uniform crossover with crossover rate of 0.5;
- Mutation rate: $1/l$ for the first 3/4 generations and 1/2 for the rest;
- Steady-state reproduction with 2 offspring generated.

Simulated Annealing:

- Initial solution: Two-stage simulated annealing. A basic local search is used to find an initial solution;
- Initial temperature: 5,000,000, which is 10 times the value of the highest candidate equipment cost;
- Fixed number of total moves: $1500 * N_c^{4/3}$;

- Acceptance/rejection criteria: Metropolis criteria;
- Cooling schedule: Swartz and Boyan modified Lam schedule, with a cooling rate of 0.999.

Cooperative method:

- Genetic algorithm with maximum number of generations of 5,000;
- Tabu search with 100 iterations and multi-start=1.

These parameters and component realization were selected based on many trials. The best results were obtained from those empirical values.

4.2 Result Analysis

In this section, we present the experiment results to assess the performance of the proposed algorithms. Four instances of 42 different problem sizes were randomly generated within a $4km^2$ area. Table 4.6 shows the 42 problem sizes. The first column in the table represents the problem number. Column 2 shows the number of TPs that need to be covered. The next four columns present respectively the number of potential node B locations, the number of potential RNC locations, the number of potential MSC locations and the number of potential SGSN locations. The 42 problems are divided into six groups. Each group has seven different problems and the number of TPs increases from 10 to 40. The number of node Bs increases from 10, 20 and 30 for group 1, 2 and 3 respectively, as well as for group 4, 5 and 6. The first three groups have 5 potential locations for RNC, MSC and SGSN. Finally, the next three problem groups have 10 potential locations for these three equipments.

Table 4.6: Problem sizes

	Problem numbers	Number of TPs	Number of node Bs	Number of RNCs	Number of MSCs	Number of SGSNs
Group 1	1	10	10	5	5	5
	2	15	10	5	5	5
	3	20	10	5	5	5
	4	25	10	5	5	5
	5	30	10	5	5	5
	6	35	10	5	5	5
	7	40	10	5	5	5
Group 2	8	10	20	5	5	5
	9	15	20	5	5	5
	10	20	20	5	5	5
	11	25	20	5	5	5
	12	30	20	5	5	5
	13	35	20	5	5	5
	14	40	20	5	5	5
Group 3	15	10	30	5	5	5
	16	15	30	5	5	5
	17	20	30	5	5	5
	18	25	30	5	5	5
	19	30	30	5	5	5
	20	35	30	5	5	5
	21	40	30	5	5	5
Group 4	22	10	10	10	10	10
	23	15	10	10	10	10
	24	20	10	10	10	10
	25	25	10	10	10	10
	26	30	10	10	10	10
	27	35	10	10	10	10
	28	40	10	10	10	10
Group 5	29	10	20	10	10	10
	30	15	20	10	10	10
	31	20	20	10	10	10
	32	25	20	10	10	10
	33	30	20	10	10	10
	34	35	20	10	10	10
	35	40	20	10	10	10
Group 6	36	10	30	10	10	10
	37	15	30	10	10	10
	38	20	30	10	10	10
	39	25	30	10	10	10
	40	30	30	10	10	10
	41	35	30	10	10	10
	42	40	30	10	10	10

The computing platform used to obtain the results is a Linux PC with a 3 GHz CPU and 1024 MB RAM.

Table 4.7 shows the results for the first instance of the 42 problems, referred to as problem set 1 hereafter. The first column shows the problem number which corresponds to the first column of Table 4.6. The following two columns contain the results obtained by solving the mathematical model as proposed in [14], where CPLEX 10.1.1 is employed to find the optimal solutions. These values are used to assess the performance of the proposed algorithms. A CPU time limit (TL) of 30 hours is set for CPLEX. This means that if CPLEX cannot find the optimal solution within 30 hours, it will return the best solution found so far. Also, since the problem is NP-hard, even the memory of the computer may be insufficient. In this case, CPLEX will return the best solution found before it runs out of memory (OM). Columns 4 and 5 provide, respectively, the best results and the corresponding CPU time obtained with the tabu search. Column 6 shows the gap (expressed as a percentage) between the solution value obtained with the tabu search and the value of the optimal solution. The following three columns show the best solutions, the corresponding computation time as well as the gap for the genetic algorithm. The next three columns are the results for the simulated annealing algorithm. Finally, similar information for the cooperative method is provided in the last three columns. The results for the other three instances (problem set 2, 3 and 4) are presented in Appendix A1, A2 and A3.

Table 4.7: Simulation results (problem set 1)

#	CPLEX		TS			GA			SA			TSGA		
	Cost ($)	CPU (sec)	Cost ($)	CPU (sec)	Gap %	Cost ($)	CPU (sec)	Gap %	Cost ($)	CPU (sec)	Gap %	Cost ($)	CPU (sec)	Gap %
1	381004	13	381004	13	0.00	385711	56	1.24	410351	97	7.70	381004	93	0.00
2	464926	13	464966	16	0.01	480292	86	3.31	464966	395	0.01	464926	136	0.00
3	433388	72	433903	23	0.12	443954	110	2.44	444722	521	2.62	433388	197	0.00
4	499261	28	499261	32	0.00	500562	153	0.26	545180	657	9.20	499261	264	0.00
5	505226	856	505226	43	0.00	506679	168	0.29	506413	857	0.23	505226	366	0.00
6	514474	74	514474	42	0.00	516431	204	0.38	515185	1234	0.14	514474	411	0.00
7	555559	3	555559	51	0.00	556665	250	0.20	555559	992	0.00	555559	496	0.00
8	410286	29	434973	76	6.02	411382	112	0.27	448974	930	9.43	410286	316	0.00
9	408274	61	408274	74	0.00	428785	141	5.02	408274	1107	0.00	408274	345	0.00
10	434191	256	443487	112	2.14	444858	181	2.46	444113	1358	2.29	434191	555	0.00
11	467535	507	480789	173	2.83	472081	248	0.97	558670	1973	19.49	479164	770	2.49
12	481116	6385	494884	208	2.86	516321	292	7.32	514427	2657	6.92	481151	893	0.01
13	514925	42731	519875	271	0.96	524435	393	1.85	554237	2690	7.63	514925	1189	0.00
14	522172	81681	541990	310	3.80	524425	303	0.43	522441	2928	0.05	522172	1231	0.00
15	389682	66	389682	243	0.00	393443	141	0.97	411235	1856	5.53	391061	493	0.35
16	407005	149	407005	255	0.00	413036	230	1.48	407005	2728	0.00	407005	1098	0.00
17	426377	313	443511	409	4.02	445530	293	4.49	448817	3348	5.26	443511	1160	4.02
18	449794	20653	463676	506	3.09	489855	427	8.91	489852	4722	8.91	449794	1669	0.00
19	461927	108000	461927	628	-	477324	498	-	481173	5260	-	462762	2564	-
20	484759	62458	499805	842	3.10	533254	677	10.00	505504	6358	4.28	484759	3376	0.00
21	542147	108000	541473	1030	-	574526	748	-	677322	8379	-	539657	3900	-
22	378325	58	378325	32	0.00	383486	53	1.36	409034	1130	8.12	378325	91	0.00
23	443421	611	443421	40	0.00	446659	67	0.73	460854	1017	3.93	443421	137	0.00
24	443900	463	443900	49	0.00	468211	89	5.48	446532	1203	0.59	443900	202	0.00
25	444452	107	444825	57	0.08	481383	112	8.31	445055	1284	0.14	444825	220	0.08
26	514312	8822	514312	72	0.00	516142	147	0.36	535041	2063	4.03	514312	308	0.00
27	513898	1858	513898	74	0.00	517325	137	0.67	513898	1798	0.00	513898	327	0.00
28	520429	31616	520888	87	0.09	522244	165	0.35	521001	2012	0.11	520429	406	0.00
29	352890	466	353231	89	0.10	356120	80	0.92	353384	3464	0.14	352890	250	0.00
30	406914	4173	406914	130	0.00	446168	134	9.65	406914	3027	0.00	406914	404	0.00
31	434800	2692	434800	155	0.00	448740	180	3.21	463200	3157	6.53	434800	547	0.00
32	443135	23544	443135	206	0.00	460164	195	3.84	461332	3652	4.11	443135	645	0.00
33	484236	25826	498502	285	2.95	540607	275	11.64	537555	4249	11.01	484236	853	0.00
34	632398	68280	694478	377	9.82	696689	599	10.17	705342	5114	11.53	694478	1484	9.82
35	529214	108000	538541	424	-	555407	425	-	594745	7439	-	529122	1300	-
36	352420	242	352420	192	0.00	354182	138	0.50	355169	6015	0.78	352420	489	0.00
37	408448	10314	408448	300	0.00	427620	207	4.69	408448	6067	0.00	408591	804	0.04
38	443841	108000	443841	395	-	462014	320	-	476502	7361	-	448982	1260	-
39	449020	20028	449974	561	0.21	502844	436	11.99	473682	7580	5.49	449020	1469	0.00
40	469826	108000	479988	658	-	492278	457	-	524147	8934	-	476465	1859	-
41	554890	108000	554877	1030	-	584539	766	-	583270	12261	-	554877	2815	-
42	585527	108000	594314	1415	-	597437	959	-	610201	12005	-	585527	3626	-
Average				285	1.21		277	3.60		3616	4.18		977	0.48

From all four problem sets (168 different problems), 126 problems gained optimal solutions by using CPLEX (75%). The tabu search was able to provide the optimal solutions for 82 different problems (48.81%). The genetic algorithm found 2 optimal solutions (1.19%) and simulated annealing obtained the optima for 11 different problems (6.55%). The cooperative method, which combines the tabu search and the genetic algorithm, showed the superiority to all three other meta-heuristics. In fact, 63.10% of the total problems (106 out of 168) had obtained the optimal solutions.

The comparison in terms of the solution quality amongst CPLEX, tabu search, genetic algorithm, simulated annealing and the cooperative method for problem set 1 is provided in Figure 4.1. Please refer to Appendixes B1, B2 and B3 for the solution comparison for problem sets 2, 3 and 4.

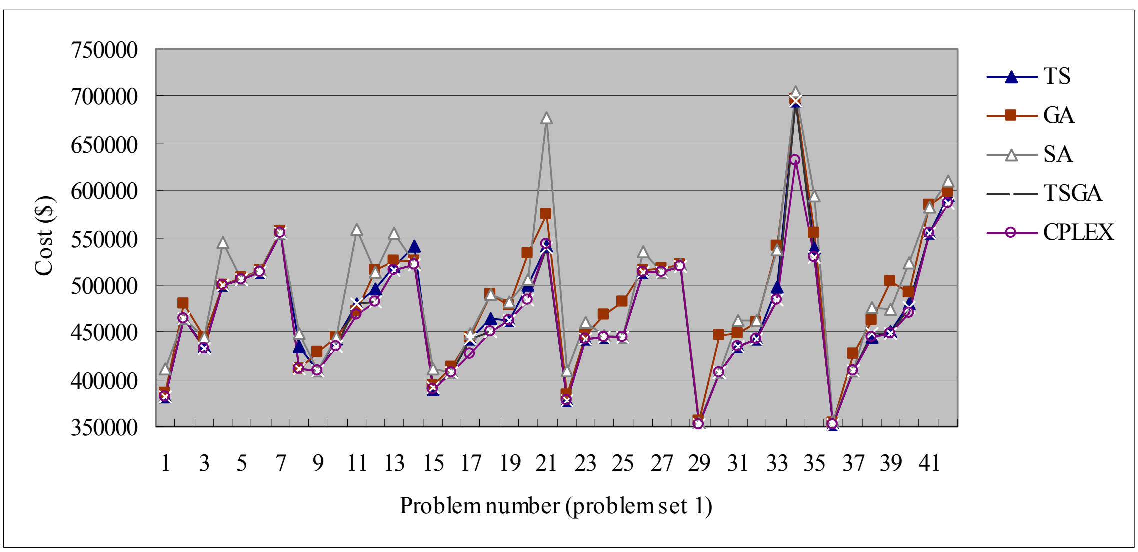


Figure 4.1: Solution comparison (problem set 1)

Figure 4.1 shows six groups of solutions. Using CPLEX solutions as the reference, we can say that the four planning tools are able to find solutions that are relatively close to the optimal solutions. We can also notice that for each group of problems, solutions of the five planning tools increase with the increase of the TP number. This complies with the fact that if more mobile users are present, more equipment will be needed in order to cover all the users.

Figure 4.2 shows a clear graph for the 3rd solution group (problem 15 to 21). Solutions from all planning tools increase as the TP number increase from 10 to 40. Table 4.8 shows the result for problem 18. As we can see, TSGA found the optimal solution. TS returns a higher solution than the optimal one, but lower than the GA and SA, which return the similar results. In fact, for the 4 problem sets, TSGA shows its superiority over the other three planning tools. This can be shown from the statistical comparison in Table 4.9

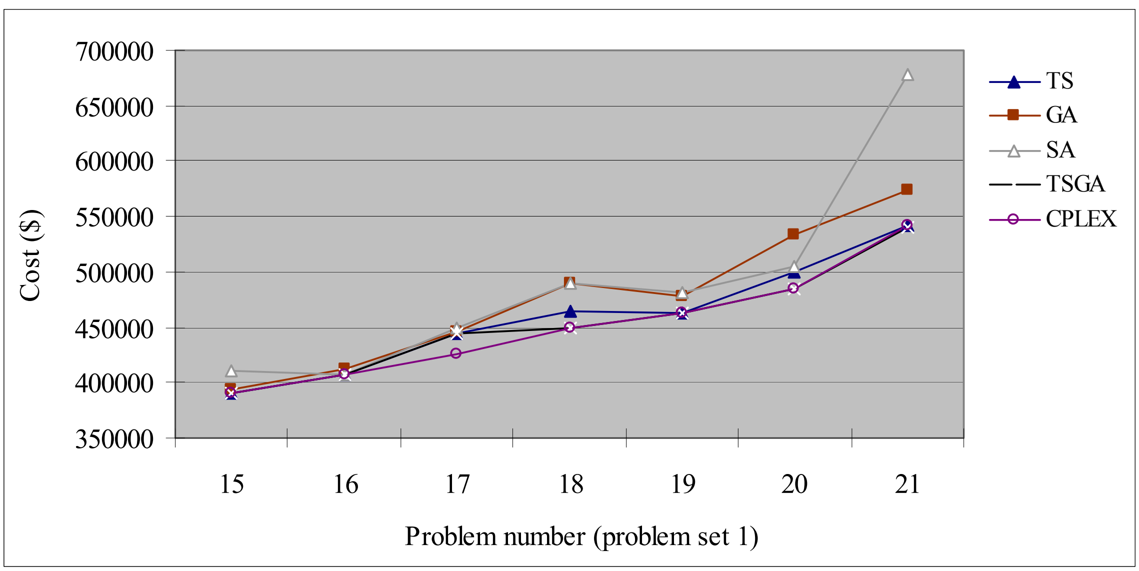


Figure 4.2: Solution comparison for problem 15 to 21 (problem set 1)

Table 4.8: Solution of problem 18 (problem set 1)

	Problem number 18 (2000_25_30_5_5_5)				
Algorithms	CPLEX	TS	GA	SA	TSGA
Cost ($)	449794	463676	489855	489852	449794

Table 4.9 shows the statistical results over the four problem sets. The first three columns represent the minimum, the maximum, and the average solution gaps. The average solution gaps were computed only if CPLEX was able to find the optimal solutions. The last two columns are respectively the standard deviation and confidence interval (C. I.) for the average solution gaps. It shows cooperative method has 90 percent confidence that the true mean solution gap is within the interval of [0.01%, 0.33%].

Table 4.9: Solution gap comparison (over four problem sets)

Algorithms	Min. gap (%)	Max. gap (%)	Ave. gap (%)	Std. dev. (%)	90-percent C. I. (%)
TS	0.00	9.82	0.71	1.55	0.71 ± 0.23
GA	0.00	12.68	3.31	3.52	3.31 ± 0.53
SA	0.00	25.61	4.59	4.41	4.59 ± 0.67
TSGA	0.00	9.82	0.17	1.06	0.17 ± 0.16

The comparison in terms of the CPU time amongst CPLEX and the four planning tools for problem set 1 is provided in Figure 4.3. Appendixes C1, C2 and C3 are corresponding information for problem sets 2, 3 and 4. The statistical comparison of CPU time over four problem sets are presented in Table 4.10.

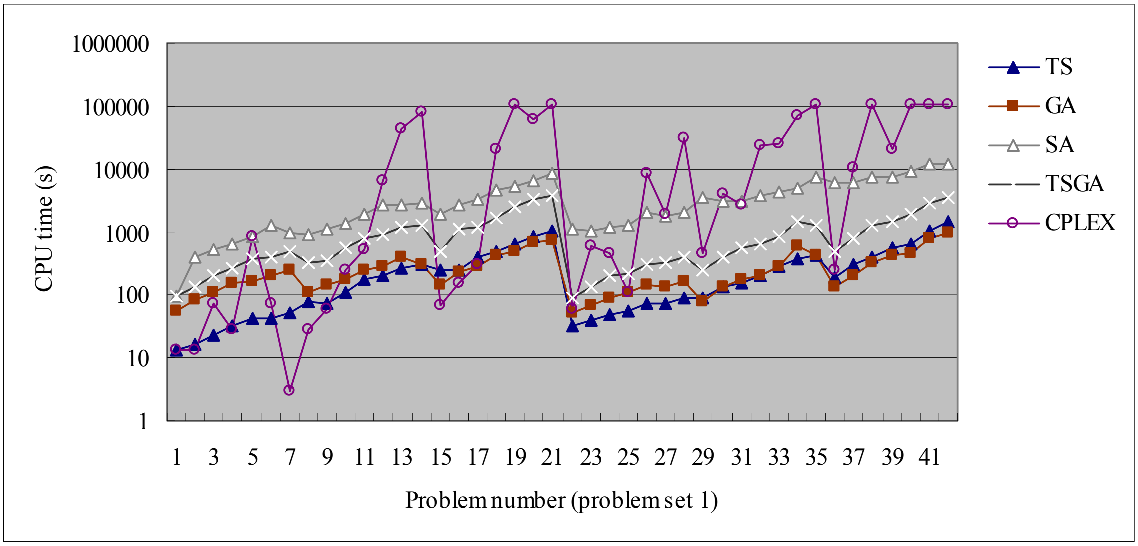


Figure 4.3: CPU time comparison (problem set 1)

As we can see from Figure 4.3, the four algorithms can provide relatively good solutions in a reasonable amount of time. In fact, the CPU times of four algorithms increase at a constant rate with respect to the problem size. We can also notice that most problems were solved within 10,000 seconds. Even if CPLEX is faster for small size instances, its CPU time is almost increasing exponentially with respect to the problem size. It is important to note that a time limit of 30 hours was used for CPLEX. The results in Table 4.7 implies that the required CPU times for problems 19, 21, 35, 38, 40, 41 and 42 may be much longer for CPLEX to find the optimal solution. This behavior is as expected since, as mentioned before, the global UMTS network planning problem is NP-hard. Finally, the small variation in the CPLEX execution time can be explained by the fact that CPLEX is using the branch and bound algorithm.

Table 4.10: CPU time comparison (over four problem sets)

Algorithms	Min. CPU (sec)	Max. CPU (sec)	Ave. CPU (sec)	Std. dev. (sec)	90-percent C. I. (sec)
TS	12	1415	297	332	297 ± 50
GA	53	1376	290	214	290 ± 32
SA	9	14765	3060	3063	3060 ± 464
TSGA	73	4202	923	864	923 ± 131
CPLEX	3	108000	35425	44339	35425 ± 6711

From the solution gap comparison shown in Table 4.9 and the CPU time comparison provided in Table 4.10, we can summarize that the tabu search and the genetic algorithm are able to use less CPU time to generate better solutions than simulated annealing. At the same time, the cooperative method can provide a significant solution quality improvement. In fact, an average gap of 0.17% is obtained which is roughly an improvement of 76% over the gap obtained from the tabu search (0.71%). The computation time of the new method increases slightly, but stays within an acceptable range with 90 percent confidence that the true mean CPU time is within the interval of [792, 1054] seconds as shown in Table 4.10.

The main idea behind these algorithms is to find an acceptable tradeoff between the quality of a solution and the computation time required to find the solution. During the simulation process, we noticed that, based on the same initial solution, tabu search may search different solution space with the hope to find a better solution by increasing the number of multi-start. This obviously increases the computation time, however, without certainty of working out a better result. Similar for the genetic algorithm, with the increase of the total generation number, the solution quality increases. However, after a certain number of generations, the rate of solution quality improvement decreases. To achieve a small amount of solution quality improvement, the total number of generations needs to be greatly increased, which corresponds to a significant growth of the computation time.

In the cooperative method, the improvement of the solution quality can be explained by the meaningful information exchanged between the tabu search and the genetic algorithm. Comparing to the pure implementation of each meta-heuristic, the proposed cooperative method makes use of the superiority of both the tabu search and the genetic algorithm. The genetic algorithm is a population-based algorithm. It is capable to explore a wide solution space during the search process. The result from the genetic algorithm obviously provides the tabu search a good search basis. Then, the tabu search keeps track of this input solution attribute to do an intensive search and find a better result. After that, this better result is integrated into the last generation of the genetic algorithm. Based on this new formed population, the genetic algorithm will be directed to a more promising solution space. This iterative information exchange affects both meta-heuristic search process and finally better results can be obtained. The most significant point is that this great improvement of the solution quality is not necessarily accompanied with a sharp increase of the computation time as its counterparts in the implementation of the independent meta-heuristics. Such observations make the cooperative method be the most efficient planning tool among those competitors.

In the real world planning, UMTS networks can be very large. Using CPLEX to solve the UMTS network planning problem will be time-consuming. As a result, the proposed algorithms are more appropriate in this situation. The new designed cooperative method with the best solution quality and reasonable computation time is especially recommended for solving the UMTS network planning problem.

Chapter 5

Conclusions and Future Work

The primary goal of the UMTS network planning is the topology planning. To help network operators gain a long term profit, efficient planning tools are necessary in order to reach a delicate balance between network investment and performance. Since the global UMTS network planning problem has been shown to be NP-hard, approximate algorithms (based on meta-heuristics) must be used to find the balance between the final solution quality and the computation time. The efficiency of the algorithm is problem dependent. An appropriate selection and design of the meta-heuristics is the key for the success of a planning tool.

In this thesis, we used two independent meta-heuristic algorithms (genetic algorithm and simulated annealing) and applied them to our specific problem. On top of that, we also proposed a cooperative method based on the tabu search and the genetic algorithm. These three algorithms aim to solve the global planning problem of UMTS networks. More especially, they consider simultaneously the cell, the access network, and the core network planning subproblems.

During the simulation, four instances of 42 different problem sizes were randomly generated and solved with the proposed algorithms. In order to assess the performance of those algorithms, we compared the results with the optimal solutions obtained from a commercial solver named CPLEX. The results demonstrate the genetic algorithm and the simulated annealing are able to find solutions with 90-percent confidence that the true mean gap is within the intervals [2.78%, 3.84%] and [3.92%, 5.26%] from the optimal solutions respectively. Even if these results are relatively good, they are not better than the algorithm based on the tabu search which returns solutions with a mean gap in the interval of [0.48%, 0.94%] with 90-percent confidence. However, when combining the tabu search and the genetic algorithm in a cooperative manner, average solution gap has 90-percent confidence to be in the interval of [0.01%. 0.33%] from the optima. Comparing to the exact algorithm, the main advantage of these proposed algorithms is the speed up in terms of CPU execution time. Therefore, larger instances of the problem can be tackled with the new proposals.

This research work has successfully shown that the proposed algorithms are able to find good solutions within reasonable computation time for the global planning problem of UMTS networks. Nevertheless, some aspects of it could be further improved. In our simulation, the user traffic distribution is randomly generated for this network planning project. There will be a certain difference from the real world network planning. Furthermore, besides the network infrastructure costs, the operational costs, like maintenance, reparation, rental, etc., are also important aspects concerned by the network operators. Unfortunately, at this time, our algorithms do not consider these costs. Another limitation is that CPLEX is not very efficient for solving medium/large size problems. As we saw from the simulation results, several problems were not solved within the time limit. As a result, it is difficult to evaluate the quality of the proposed algorithms for medium/large size problems since there is no comparison value.

Some future works of this project may consist of investigating the parallel implementation of these meta-heuristics and performance analysis between the parallel and the cooperative implementation. The performance comparison of different planning tools can also be made under fair amount of computation time. Another avenue could be a comparative study amongst the proposed algorithms for larger size problems. We could also implement the algorithms for the downlink direction or even both directions simultaneously. Finally, building a network cost model with the consideration of different operational costs, and modifying the model to satisfy the network expansion planning problem may be potential research directions as well.

References

[1] Y. K. Kim and R. Prasad, “4G Roadmap and Emerging Communication Technologies”, Artech House, 2006

[2] T. C. Piliouras, “Network Design: Management and Technical Perspectives, 2nd Edition”, Routledge, USA, 2005

[3] T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein, “Introduction to Algorithms”, The MIT Press, 2001

[4] J. Dréo, A. Pétrowski, P. Siarry, E. Taillard, “Metaheuristics for Hard Optimization: Methods and Case Studies”, Springer, 2003

[5] J. Lam, “An Efficient Simulated Annealing Schedule,” Ph.D. dissertation, Yale University, 1988

[6] R. M. Whitaker, S. Hurley, “Evolution of Planning for Wireless Communication Systems”, Proceedings of the 36th Hawaii International Conference on System Sciences (HICSS’03), vol. 9, IEEE Computer Society, 2003, pp. 10

[7] F. Glover and M. Laguna, “Tabu Search”, Springer, 1998

[8] S. Kirkpatrick, C. D. Gelatt and M. P. Vecchi, “Optimization by Simulated Annealing”, Science, vol. 220, no. 4598, 1983, pp. 671-680

[9] D. Beasley, D. R. Bull and R. R. Martin, “An Overview of Genetic Algorithm: Part 1, Fundamentals, University Computing”, vol. 15, no. 2, 1993, pp. 58-69

[10] C. R. Reeves and J. E.ROWE, “Genetic Algorithms: Principles and Perspectives - A Guide to GA Theory”, Kluwer Academic Publishers, 2002

[11] E. Amaldi, A. Capone, F. Malucelli, “Optimizing Base Station Sitting in UMTS Networks”, In 53rd Vehicular Technology Conference (VTC), Spring 2001, pp. 2828–2832

[12] J. Harmatos, “Planning of UMTS Core Networks”, 13th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, 2002, pp. 740-744

[13] A. Juttner, A. Orban and Z. Fiala, “Two New Algorithms for UMTS Access Network Topology Design”, European Journal of Operational Research, vol. 164, 2005, pp. 456-474

[14] M. St-Hilaire, S. Chamberland and S. Pierre, “Uplink UMTS Network Design - An Integrated Approach”, Computer Netwroks, vol. 50, no. 15, 2006, pp. 2747-2761

[15] ILOG Inc., Incline Village, NV, http://www.ilog.com/products/cplex

[16] M. St-Hilaire, S. Chamberland and S. Pierre, “A Tabu Search Algorithm for the Global Planning Problem of Third Generation Mobile Networks”, Computers and Electrical Engineering, vol. 34, 2008, pp. 470-487

[17] T. G. Crainic, B. D. Chiara, M. Nonato and L. Tarricone, “ Tackling Electrosmog in Completely Configured 3G Networks by Parallel Cooperative Meta-heuristics”, IEEE Wireless Communications, vol. 13, 2006, pp. 34-41

[18] U. Lauther, T. Winter and M. Ziegelmann, “Proximity Graph Based Clustering Algorithms for Optimized Planning of UMTS Access Network Topologies”, IEEE International Conference on Telecommunications, vol. 2, 2003, pp. 1329-1334

[19] J. Zhang, L. Guo and J. U. Wu, “An Integrated Approach for UTRAN Planning

and Optimization”, IEEE 59th Vehicular technology Conference (VTC), vol. 4, May 2004, pp. 2360-2364

[20] A. R. Mishra, “Fundamentals of Cellular Network Planning and Optimisation 2G/2.5G/3G... Evolution to 4G”, John Wiley & Sons Ltd, 2004

[21] E. Amaldi, A. Capone, F. Malucelli and F. Signori, “Optimization Models and Algorithms for Downlink UMTS Radio Planning”, IEEE Wireless Communications and Networking Conference (WCNC), vol. 2, 2003, pp. 827-831

[22] E. Amaldi, A. Capone, F. Malucelli and F. Signori, “A Mathematical programming Approach for W-CDMA Radio Planning with Uplink and Downlink Constraints”, IEEE 58th Vehicular Technology Conference (VTC), vol. 2, 2003, pp. 806-810

[23] S.U. Thiel, P. Giuliani, L.J. Ibbetson and D. Lister, “An Automated UMTS Site Selection Tool”, 3rd International Conference on 3G Mobile Communication Technologies, 69-73, 2002

[24] Y. Wu and S. Pierre, “Base Station Positioning in Third Generation Mobile Networks”, IEEE Canadian Conference on Electrical and Computer Engineering (CCECE), vol. 1, 2003, pp. 31-34

[25] Y. S. Choi, K. S. Kim and N. Kim, “The Positioning of Base Station in Wireless Communication with Genetic Approach”, 12th IFIP International Conference on Personal Wireless Communications, vol. 245, 2007, pp. 217-229

[26] C. Maple, L. Guo and J. Zhang, “Parallel Genetic Algorithms for Third Generation Mobile Network Planning”, International Conference on Parallel Computing in Electrical Engineering (PARELEC), 2004, pp. 229-236

[27] S. B. Jamaa, Z. Altman, J.M. Picard and B. Fourestie, “Multi-Objective Strategies for Automatic Cell Planning of UMTS Networks”, IEEE 59th

Vehicular Technology Conference (VTC), vol. 4, 2004, pp. 2420-2424

[28] E. Amaldi, A. Capone and F. Malucelli "Planning UMTS Base Station Location: Optimization Models with Power Control and Algorithms", IEEE Transactions on Wireless Communications, vol. 2, 2003, pp. 939-952

[29] K. Tutschku and P. Tran-Gia, "Spatial Traffic Estimation and Characterization for Mobile Communication Network Design", IEEE Journal on Selected Areas in Communications (JSAC), vol. 16, 1998, pp. 804-811

[30] R. Mathar and M. Schmeink, "Optimal Base Station Positioning and Channel Assignment for 3G Mobile Networks by Integer Programming", Annals of Operations Research, vol. 107, no. 1-42001, pp. 225-236

[31] J. Laiho, A. Wacker and T. Novosad, "Radio Network Planning and Optimisation for UMTS, 2nd Edition", John Wiley & Sons Ltd, 2006

[32] E. Amaldi, A. Capone, F. Malucelli and F. Signori, "UMTS Radio Planning: Optimizing Base Station Configuration", IEEE 56th Vehicular Technology Conference (VTC), vol. 2, 2002, pp. 768-772

[33] P. R. Gould, "Radio Planning of Third Generation Networks in Urban Areas", Conference on 3G mobile communication technologies, May 2002, pp. 64-68

[34] U. Okumura "Field Strength and Its Variablity in UHF and VHF Land-mobile Radio Service", Rev. Elec. Commun. Lab., vol. 16, 1968

[35] M. Hata, "Empirical Formula for Propagation Loss in Land Mobile Radio Services", IEEE Transactions on Vehicular Technology, vol. 29, 1980, pp. 317-325

[36] COST 231 Final Report, "Digital Mobile Radio towards Future Generation Systems", 1999. Available online: http:/www.lx.it.pt/cost231

[37] J. Yang, M. E. Aydin, J. Zhang and C. Maple, "UMTS Base Station Location Planning: A Mathematical Model and Heuristic Optimisation Algorithms", IET

communications, vol. 5, 2007, pp. 1007-1014

[38] H. Holma and A. Toskala, "WCDMA for UMTS - Radio Access for Third Generation Mobile Communications, 3rd Edition", John Wiley & Sons Ltd, 2004

[39] A. Molina, G. E. Athanasiadou and A. R. Nix, "The Automatic Location of Base-stations for Optimized Cellular Coverage: A New Combinatorial Approach", IEEE 49th Vehicular Technology Conference, vol.1, 1999, pp. 606-610

[40] E. Amaldi, A. Capone and F. Malucelli, "Discrete Models and Algorithms for the Capacitated Location Problems Arising in UMTS Network Planning", Proceedings of the 5th International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications, 2001, pp.827-831

[41] E. Amaldi, P. Belotti, A. Capone, F. Malucelli, "Optimizing Base Station Location and Configuration in UMTS Networks", Annals of Operations Research, vol. 146, no. 1, 2006, pp. 135-151

[42] M. E. Aydin, J. Yang and J. Zhang, "A Comparative Investigation on Heuristic Optimization of WCDMA Radio Networks", Applications of Evolutinary Computing, LNCS 4448, vol. 4448,2007, pp. 111– 120

[43] J. Harmatos, A. Szentesi and I. Godor, "Cost-based UMTS Transport Network Topology Optimisation", International Conference on Computer Communication (ICCC), 1999

[44] A. Quintero, S. Pierre, "A Mimetic Algorithm for Assigning Cells to Switches in Cellular Mobile Networks", IEEE Communications Letters, vol. 6, 2002, pp. 484-486

[45] S. Menon, R. Gupta, "Assigning Cells to Switches in Cellular Networks by Incorporating a Pricing Mechanism into Simulated Annealing", IEEE

Transactions on Systems, Man, and Cybernetics, Part B, vol. 34, 2004, pp. 558-565

[46] A. Merchant and B. Sengupta, “Assignment of Cells to Switches in PCS Networks,” IEEE/ACM Trans. Networking, vol. 3, Oct.1995, pp. 521-526

[47] S. Pierre and S. Houeto, “A Tabu Search Approach for Assigning Cells to Switches in Cellular Mobile Networks”, Computer Communications, vol. 25, 2002, pp. 464-477

[48] Y. Wu and S. Pierre, “Optimization of Access Network Design in 3G Networks”, IEEE Canadian Conference on Electrical and Computer Engineering (CCECE), vol. 2, 2003, pp. 781-784

[49] A. Szlovenscsak, I. Godor, J. Harmatos and T. Cinkler, “Planning Reliable UMTS Terrestrial Access Networks”, IEEE Communication Magazine, 40, January 2002, pp. 66-72

[50] C. Charnsripinyo and D. Tipper, “Topological Design of 3G Wireless Backhaul Networks for Service Assurance”, Proceedings.5th International Workshop on Design of Reliable Communication Networks (DRCN), 2005, pp. 9

[51] Y. Wu and S. Pierre, “A New Hybrid Constraint-Based Approach for 3G Network Planning”, IEEE Communications Letters, vol. 8, 2004, pp. 277-279

[52] J. Harmatos, A. Szentesi and I. Godor, “Planning of Tree-Topology UMTS Terrestrial Access Networks”, 11th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIRMC), vol. 1, 2000, pp. 353-357

[53] J. L. Bentley: "Multidimensional Binary Search Trees used for Associative Searching", Comm. of the ACM, 18.9.1972. pp. 509 - 517

[54] U. Lauther: "A Data Structure for Gridless Routing", Proc. 17th Design Automation Conf., 1980, pp. 603-609

[55] S. Kundu and J. Misra, “A Linear Tree Partitioning Algorithm”, SLAM Journal

on Computing, vol. 6, 1977, pp. 151-154

[56] R. Shalak, K.Sandrasegaran, J. Agbinya and S. Subenthiran, “UMTS Core Network Planning Model and Comparison of Vendor Product Performance”, 6th International Conference on Advanced Communication Technology, vol. 2, 2004, pp. 685-689

[57] D. Ricciato, R. Pilz and E. Hasenleithner, “Measurement-Based Optimization of a 3G Core Network: A Case Study”, 6th International Conference on Next Generation Teletraffic and Wired/Wireless Advanced Networking, vol. 4003, 2006, pp. 70-82

[58] P. Gajowniczek, M. Pjoro, A. Arvidsson, “VP Reconfiguration through Simulated Allocation”, Thirteenth Nordic Tele-traffic Seminar(NTS 13), 1996, pp.251-260

[59] S. Chamberland and S. Pierre, “On the Design Problem of Cellular Wireless Networks”, Wireless Networks, vol. 11, 2005, pp. 489–496

[60] R. Jonker and A. Volgenant, “A Shortest Augmenting Path Algorithm for Dense and Sparse Linear Assignment Problems”, Computing, vol. 38, no. 4, 1987, pp. 325- 340

[61] M. St-Hilaire, S. Chamberland and S. Pierre, “A Local Search Heuristic for the Global Planning of UMTS Networks”, IWCMC, 2006

[62] D. Beasley, D. R. Bull and R. R. Martin, “An Overview of Genetic Algorithm: Part 1, Fundamentals”, University Computing, vol. 15, 1993, pp. 58-69

[63] J. E. Beasley and P. C. Chu, “A Genetic Algorithm for the Set Covering Problem”, European Journal of Operational Research, vol. 94, 1996, pp. 392-404

[64] M. Ionita and H. Luchian, “Two Problem Independent Methods for Generating Initial Solutions”, IEEE Congress on Evolutionary Computation, vol. 2, 2005,

pp. 1442-1447

[65] O. Peyran and W. Zhuang, “Educating Initial Solutions for Genetic Algorithms: A chip Planning Oplanning Optimization Example”, Institute of High Performance Computing

[66] K. Sastry, D. Goldberg, “Genetic Algorithm”, Search Methodologies, chapter 4, Springer, 2005, pp. 97-125

[67] D. E. Goldberg, “Optimal Initial Population Size for Binary-coded Genetic Algorithms”, TCGA, Report No. 85001, University of Alabama, Tuscaloosa.

[68] J. J. Grefenstette, “Optimization of Control Parameters for Genetic Algorithms”. IEEE Transactions on Systems, Man and Cybernetics (SMC), vol. 16, 1986, pp. 122-128

[69] J. D. Schaffer, R. A. Caruana, L. J. Eshelman and R. Das, “A Study of Control Parameters Affecting Online Performance of Genetic Algorithms for Function Optimization”, Proceedings of 3rd International Conference on Genetic Algorithms, 1989, pp. 51-60

[70] D. E. Goldberg, K. Deb and J. H. Clark, “Genetic Algorithms, Noise, and the Sizing of Populations”, Complex Systems, vol. 6, 1992, pp. 333-362.

[71] C. R. Reeves, “Modern Heuristic Techniques for Combinatorial Problems”, John Wiley & Sons, 1993

[72] B.T. Zhang and J.J. Kim, “Comparison of Selection Methods for Evolutionary Optimization, Evolutionary Optimization”, An International Journal on the Internet, vol. 2, no. 1, 2000, pp. 55-70

[73] B. L. Miller and D. E. Goldberg, “Genetic Algorithms, Tournament Selection, and the Effects of Noise”, Complex Systems, vol. 9, 1995, pp. 193-212

[74] M. Chakraborty and U. K. Chakraborty, “An Analysis of Linear Ranking and Binary Tournament Selection in Genetic Algorithms”, Proceedings of 1997

International Conference on Information, Communications and Signal Processing (ICICS), vol. 1, September 1997, pp. 407-411

[75] L. Davis, “Handbook of Genetic Algorithms”, Van Nostrand Reinhold, 1991, pp. 46-50

[76] M.W. van der. Ham, “Simulated Annealing Applied in Coding Theory”, Eindhoven University of Technology, 1988

[77] K.A. Dowsland, “Simulated Annealing” in “Modern Heuristic Techniques for Combinatorial Problems”, (ed. C.R. Reeves), McGraw-Hill, 1995

[78] P. J. M. van Laarhoven and E. H. L. Aarts, “Simulated Annealing: Theory and Applications”, D. Reidel Publishing Company, 1987

[79] K. Binder, “Monte Carlo Methods in Statistical Physics”, Springer, New York, 1979

[80] E. Rodriguez-Tello, J. Hao, J. Torres-Jimenez, “An Effective Two-stage Simulated Annealing Algorithm for the Minimum Linear Arrangement Problem”, Computers and Operations Research, vol. 35, 2008, pp. 3331-3346

[81] H. Das, P. T. Cummings and M. D. LeVan, “Scheduling of Serial Multiproduct Batch Processes Via Simulated Annealing”, Computers & Chemical Engineering, vol.14 no.12, 1990, pp.1351-1362

[82] T. G. Crainic, M. Toulouse and M. Gendreau, “Parallel Asynchronous Tabu Search for Multicommodity Location-allocation with Balancing Requirements”, Annals of Operations Research, vol. 63, no, 2, 1996, pp. 277-299

[83] S. Nahar, S. Sahni and E. Shragowitz, “Experiments with Simulated Annealing”, Proceedings of the 22nd ACM/IEEE Conference on Design Automation, 1985, pp.748-752

[84] B. L. Golden and C. C. Skiscim, “Using Simulated Annealing to Solve Routing

and Location Problems", Naval Research Logistics Quarterly, vol. 33, 1986, pp. 261-279

[85] M. Lundy, A. Mees, "Convergence of an Annealing Algorithm", Mathematical Programming, vol. 34, 1986, pp. 111-124

[86] W. P. Swartz, Jr. "Automatic Layout of Analog and Digital Mixed Macro/Standard Cell Integrated Circuits", Ph.D. Dissertation, Yale University, 1993

[87] P. K. Lai, "Synthesis of High-Performance Digital Cells in ASTRX/OBLX", M.Sc. Dissertation, Carnegie Mellon University, 1994

[88] J. A. Boyan, "Learning Evaluation Functions for Global Optimization", Ph.D. Dissertation, Carnegie Mellon University, 1998

[89] C. Sechen and A. Sangiovanni-Vincentelli, "The TimberWolf Placement and Routing Package", IEEE Journal of Solid-State Circuits, vol. 20, 1985, pp. 510-522

Appendix A1: Simulation results (problem set 2)

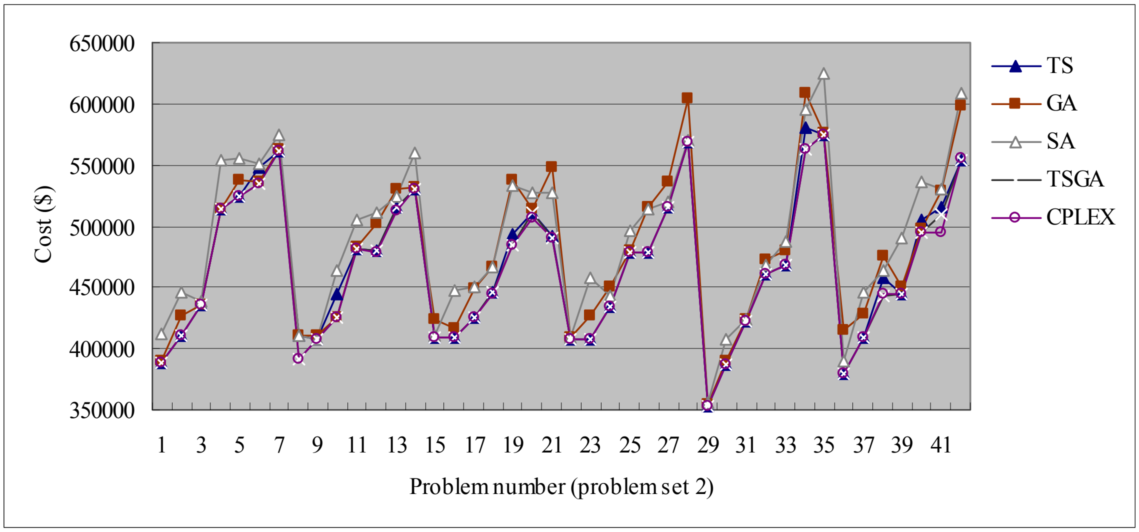
#	CPLEX		TS			GA			SA			TSGA		
	cost ($)	CPU (sec)	cost ($)	CP (sec)	Gap %	cost ($)	CPU (sec)	Gap %	cost ($)	CPU (sec)	Gap %	cost ($)	CPU (sec)	Gap %
1	388864	9	388864	17	0.00	389256	647	0.10	411991	44	5.95	388864	81	0.00
2	410622	9	410622	26	0.00	426212	75	3.80	446474	400	8.73	410622	119	0.00
3	435407	8	435407	31	0.00	436207	92	0.18	438397	475	0.69	435407	169	0.00
4	513864	77	513864	49	0.00	513864	137	0.00	553208	1037	7.66	513864	276	0.00
5	525085	2047	525085	58	0.00	537153	238	2.30	555548	986	5.80	525085	327	0.00
6	534430	243	548298	73	2.59	535616	299	0.22	550546	931	3.02	534430	392	0.00
7	561163	128	561256	75	0.02	562574	330	0.25	574283	1075	2.34	561163	437	0.00
8	390984	27	409988	86	4.86	411029	103	5.13	409988	923	4.86	390984	235	0.00
9	407811	424	407811	123	0.00	410418	146	0.64	407811	1242	0.00	407811	346	0.00
10	425379	62	444598	160	4.52	425385	221	0.00	463350	1443	8.93	425379	441	0.00
11	481962	7417	481962	219	0.00	482906	244	0.20	504909	1741	4.76	481962	640	0.00
12	478975	4229	479372	283	0.08	502389	387	4.89	511055	2292	6.70	479315	1027	0.07
13	512540	108000	515735	426	-	530653	552	-	523899	2747	-	515735	1119	-
14	529916	108000	529645	461	-	531097	519	-	559560	3325	-	529645	1320	-
15	408903	56	409305	240	0.10	424037	163	3.70	410873	1997	0.48	408903	461	0.00
16	408523	410	408523	379	0.00	416062	259	1.85	447523	3130	9.55	408523	969	0.00
17	424782	456	425763	557	0.23	448489	300	5.58	451078	3326	6.19	424782	1076	0.00
18	445063	4127	446020	669	0.22	466411	492	4.80	466531	4260	4.82	445912	1349	0.19
19	483921	108000	493879	1121	-	537678	825	-	533342	5295	-	483921	1924	-
20	506824	108000	511040	1255	-	514380	826	-	527676	7050	-	511040	2320	-
21	491093	108000	491300	1382	-	548607	1376	-	527677	8246	-	491093	2654	-
22	408254	343	408254	33	0.00	409330	54	0.26	408572	175	0.08	408254	101	0.00
23	408077	726	408077	40	0.00	426516	72	4.52	458318	1086	12.31	408077	133	0.00
24	433538	1825	433538	50	0.00	450526	87	3.92	443446	1180	2.29	433538	174	0.00
25	478810	74891	478810	63	0.00	480747	114	0.40	496119	1488	3.62	478810	254	0.00
26	479195	39972	479195	73	0.00	515676	153	7.61	513886	1797	7.24	479195	368	0.00
27	515532	10547	515532	81	0.00	535996	166	3.97	519701	1765	0.81	515532	347	0.00
28	568262	42228	568262	102	0.00	604861	272	6.44	570628	2742	0.42	568262	578	0.00
29	353291	1865	353291	87	0.00	353923	82	0.18	355224	3552	0.55	353332	255	0.01
30	387632	3008	387632	132	0.00	389455	116	0.47	407767	3445	5.19	387632	390	0.00
31	422820	3415	423099	159	0.07	424566	182	0.41	424086	3079	0.30	423044	461	0.05
32	460447	108000	460500	191	-	472990	210	-	467976	3587	-	460589	659	-
33	468437	108000	468479	239	-	480189	241	-	487788	4165	-	468479	891	-
34	563519	108000	580021	380	-	608866	406	-	594645	5112	-	563519	1641	-
35	574211	58073	574211	397	0.00	576845	387	0.46	624149	6414	8.70	574211	1273	0.00
36	379051	634	379051	197	0.00	415448	150	9.60	390306	5664	2.97	379051	528	0.00
37	408567	5515	409003	341	0.11	428205	182	4.81	446583	6464	9.30	408567	859	0.00
38	444434	108000	458324	440	-	475813	326	-	463633	7342	-	443620	1559	-
39	443937	28515	443937	526	0.00	449964	371	1.36	490472	8176	10.48	443937	1508	0.00
40	494696	108000	504442	819	-	498109	483	-	536660	9122	-	494626	2062	-
41	494675	102700	514862	994	4.08	528592	622	6.86	530862	12468	7.32	510337	2485	3.17
42	554977	OM-71170	554320	1394	-	598730	889	-	608134	14756	-	553686	4202	-
Average				340	0.92		321	2.73		3640	4.77		884	0.11

Appendix A2: Simulation results (problem set 3)

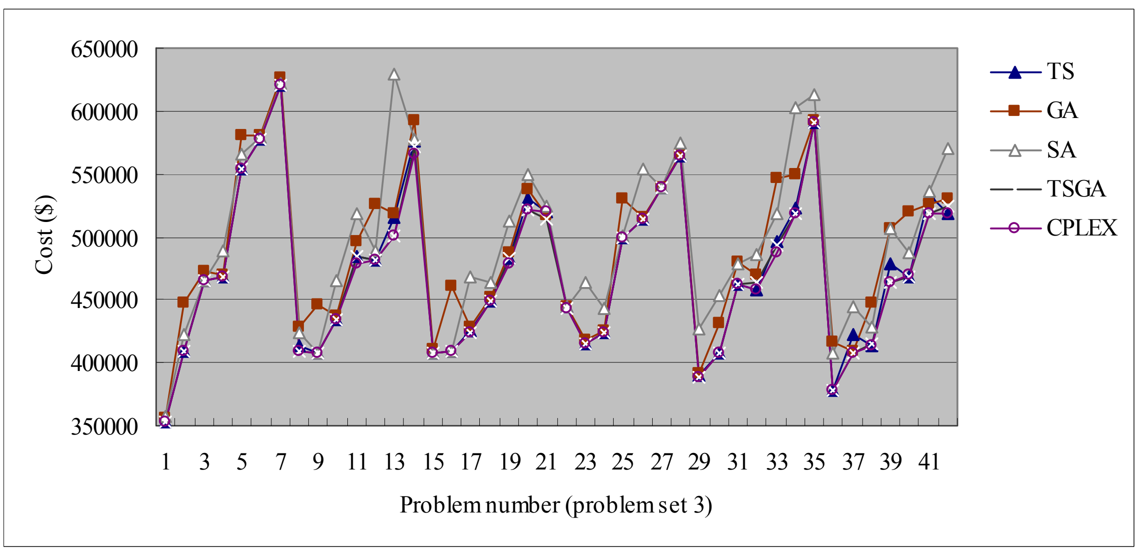
#	CPLEX		TS			GA			SA			TSGA		
	cost ($)	CPU (sec)	cost ($)	CPU (sec)	Gap %	cost ($)	CPU (sec)	Gap %	cost ($)	CPU (sec)	Gap %	cost ($)	CPU (sec)	Gap %
1	352888	11	352888	12	0.00	355463	56	0.73	357520	98	1.31	352888	73	0.00
2	408427	7	408427	18	0.00	447746	757	9.63	422758	443	3.51	408427	135	0.00
3	464833	78	464833	24	0.00	473207	105	1.80	464833	488	0.00	464833	161	0.00
4	468583	86	468583	27	0.00	469103	128	0.11	488748	656	4.30	468583	224	0.00
5	554082	392	554082	37	0.00	580298	204	4.73	565767	846	2.11	554082	332	0.00
6	577165	142	577165	47	0.00	580551	307	0.59	578534	1152	0.24	577165	482	0.00
7	621145	61	621145	50	0.00	626405	490	0.85	621466	1487	0.05	621145	772	0.00
8	408722	29	413497	55	1.17	429010	102	4.96	423655	879	3.65	408722	299	0.00
9	407550	142	407550	84	0.00	446195	150	9.48	407723	1222	0.04	407550	356	0.00
10	434416	245	434544	129	0.03	437117	225	0.62	465162	1480	7.08	434544	548	0.03
11	479210	10	484318	152	1.07	496178	241	3.54	519005	1871	8.30	484318	715	1.07
12	480889	10157	480889	200	0.00	525739	257	9.33	488622	2022	1.61	480889	859	0.00
13	500633	52796	515083	251	2.89	518587	354	3.59	628846	3187	25.61	500633	1001	0.00
14	565295	108000	575387	283	-	592105	456	-	576983	3540	-	570994	1338	-
15	407336	53	407336	135	0.00	410469	157	0.77	407336	1851	0.00	407336	459	0.00
16	408732	502	408732	260	0.00	460555	239	12.68	409568	2929	0.20	408732	778	0.00
17	424489	1094	424830	341	0.08	427623	314	0.74	468899	3399	10.46	424489	1062	0.00
18	449092	10068	449266	443	0.04	452570	410	0.77	463656	4425	3.24	449266	1507	0.04
19	479215	108000	483573	777	-	487059	507	-	512377	5885	-	483641	2021	-
20	522121	108000	529749	941	-	538159	636	-	549212	7420	-	522121	2593	-
21	520151	108000	519880	981	-	516778	772	-	525089	6772	-	514304	3029	-
22	443549	724	444007	33	0.10	444457	58	0.20	444363	1103	0.18	443549	98	0.00
23	414879	207	414879	41	0.00	417971	70	0.75	463694	1033	11.77	414879	122	0.00
24	424113	6902	424113	51	0.00	424717	86	0.14	443805	1306	4.64	424113	177	0.00
25	498800	4815	498945	55	0.03	530252	157	6.31	500175	1279	0.28	498800	245	0.00
26	513931	54468	513931	76	0.00	515754	125	0.35	553276	2342	7.66	513931	338	0.00
27	538627	1234	538627	77	0.00	538804	165	0.03	539027	1681	0.07	538627	315	0.00
28	564573	108000	564573	92	-	565330	198	-	574063	2134	-	564573	454	-
29	388744	459	389613	87	0.22	390654	81	0.49	426166	3014	9.63	388776	236	0.01
30	408350	2787	408350	121	0.00	431146	119	5.58	453590	3164	11.08	408350	353	0.00
31	462081	108000	462081	180	-	480245	187	-	478302	2965	-	462081	542	-
32	458618	108000	458618	207	-	469666	210	-	486684	3664	-	464093	913	-
33	487590	108000	497005	312	-	546447	286	-	517775	4848	-	492875	1006	-
34	518465	61096	523591	355	0.99	549545	316	5.99	602744	4830	16.26	518465	1086	0.00
35	590650	61917	590650	425	0.00	592046	453	0.24	612919	5693	3.77	590650	1509	0.00
36	378741	731	378741	193	0.00	416252	149	9.90	408063	5809	7.74	378741	488	0.00
37	407638	5734	422349	328	3.61	408978	229	0.33	444273	6786	8.99	407638	876	0.00
38	413777	19957	413777	435	0.00	448188	290	8.32	427899	7097	3.41	414576	1144	0.19
39	463245	108000	478255	584	-	506795	382	-	506908	7473	-	463245	1609	-
40	470419	108000	467943	769	-	519727	479	-	487495	9359	-	467943	1920	-
41	517945	23427	532959	1104	2.90	526198	621	1.59	536917	10068	3.66	518150	2487	0.04
42	519121	72327	519121	1214	0.00	529604	624	2.02	570362	12962	9.87	523935	3353	0.93
Average				285	0.41		289	3.35		3587	5.34		905	0.07

Appendix A3: Simulation results (problem set 4)

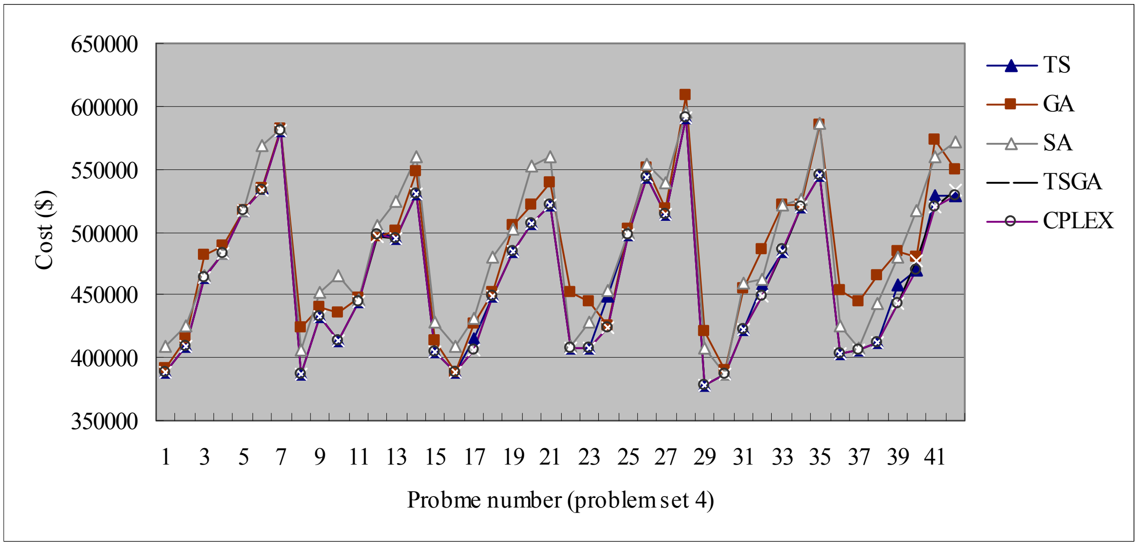
#	CPLEX		TS			GA			SA			TSGA		
	cost ($)	CPU (sec)	cost ($)	CPU (sec)	Gap %	cost ($)	CPU (sec)	Gap %	cost ($)	CPU (sec)	Gap %	cost ($)	CPU (sec)	Gap %
1	388735	14	388735	12	0.00	391551	75	0.72	409007	9	5.21	388735	91	0.00
2	409394	235	409394	20	0.00	415034	92	1.38	425503	400	3.93	409394	135	0.00
3	464072	435	464072	26	0.00	481074	124	3.66	464608	522	0.12	464072	222	0.00
4	482456	67	482456	32	0.00	489159	144	1.39	482477	587	0.00	482456	248	0.00
5	516559	54	516559	38	0.00	516559	169	0.00	516559	619	0.00	516559	297	0.00
6	532663	644	534030	43	0.26	534030	195	0.26	569026	980	6.83	532663	387	0.00
7	580382	30	580382	51	0.00	582513	326	0.37	582583	1240	0.38	580382	558	0.00
8	387395	76	387395	55	0.00	423960	111	9.44	406828	847	5.02	387395	241	0.00
9	433132	587	433132	90	0.00	439649	169	1.50	451538	1128	4.25	433132	382	0.00
10	413642	49563	413642	104	0.00	435643	194	5.32	465554	1433	12.55	413642	498	0.00
11	444635	471	444635	133	0.00	447803	270	0.71	446010	1669	0.31	444635	647	0.00
12	497182	108000	501322	227	-	496976	360	-	505181	2207	-	496892	1013	-
13	495200	31323	495270	223	0.01	500576	403	1.09	523793	2467	5.77	495503	1049	0.06
14	530249	32618	530249	335	0.00	548316	430	3.41	559982	2866	5.61	530249	1274	0.00
15	404007	224	404007	152	0.00	414144	191	2.51	427889	1951	5.91	404007	520	0.00
16	388683	123	388683	214	0.00	388804	236	0.03	408854	2600	5.19	388683	758	0.00
17	405438	1443	414408	273	2.21	427052	329	5.33	431956	3474	6.54	405438	1146	0.00
18	449262	108000	449262	533	-	452519	462	-	480046	4475	-	449262	1716	-
19	483918	26662	483918	567	0.00	505154	535	4.39	502644	5407	3.87	483918	2026	0.00
20	507022	108000	507022	856	-	521452	668	-	552147	7174	-	507022	2755	-
21	521103	108000	520731	1028	-	539363	877	-	559807	8974	-	520731	2907	-
22	408354	514	408354	33	0.00	451567	60	10.58	408471	814	0.03	408354	95	0.00
23	407155	8098	407155	44	0.00	444853	66	9.26	428948	1054	5.35	407155	153	0.00
24	423372	39086	448337	49	5.90	424673	78	0.31	453886	1254	7.21	423372	169	0.00
25	497960	47354	497960	59	0.00	502586	109	0.93	499500	1326	0.31	497960	229	0.00
26	543675	105816	543675	80	0.00	550962	123	1.34	553481	1814	1.80	543675	399	0.00
27	514479	6063	514479	80	0.00	517902	135	0.67	539792	1542	4.92	514479	301	0.00
28	590474	29769	590474	88	0.00	608292	400	3.02	595719	2306		590474	638	0.00
29	378408	1901	378408	79	0.00	420558	90	11.14	407434	3402	7.67	378408	238	0.00
30	386680	4704	386680	109	0.00	390301	113	0.94	387395	3662	0.18	386680	345	0.00
31	422824	36192	422824	174	0.00	454531	161	7.50	459504	3201	8.68	422824	547	0.00
32	449627	108000	458323	243	-	485262	198	-	462021	3589	-	449627	711	-
33	485292	108000	484886	265	-	520833	269	-	521077	3842	-	484886	793	-
34	520165	108000	520165	299	-	521606	280	-	525162	4531	-	520165	1008	-
35	545761	108000	545098	417	-	585642	412	-	585835	5541	-	545098	1278	-
36	403471	9128	403471	199	0.00	453644	165	12.44	425529	5577	5.47	403471	542	0.00
37	406529	OM-78219	406529	315	-	444254	219	-	408022	6065	-	406529	889	-
38	411943	OM-106483	411943	430	-	465020	291	-	443111	6971	-	411943	1150	-
39	443828	108000	458374	530	-	484984	388	-	480674	8537	-	443104	2008	-
40	469314	OM-51194	469314	853	-	479455	566	-	516676	9995	-	477678	1991	-
41	519686	OM-58711	528871	1193	-	572659	612	-	559812	10828	-	519686	2602	-
42	528388	OM-62300	528388	1025	-	549031	668	-	571785	10858	-	533971	2749	-
Average				276	0.30		280	3.56		3309	4.07		898	0.00



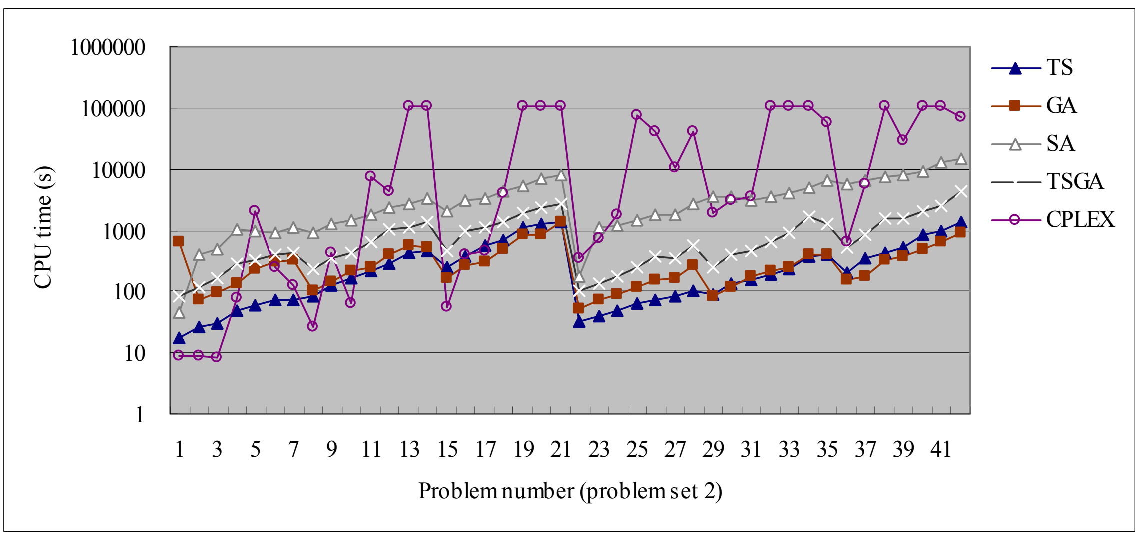
Appendix B1: Solution comparison (problem set 2)



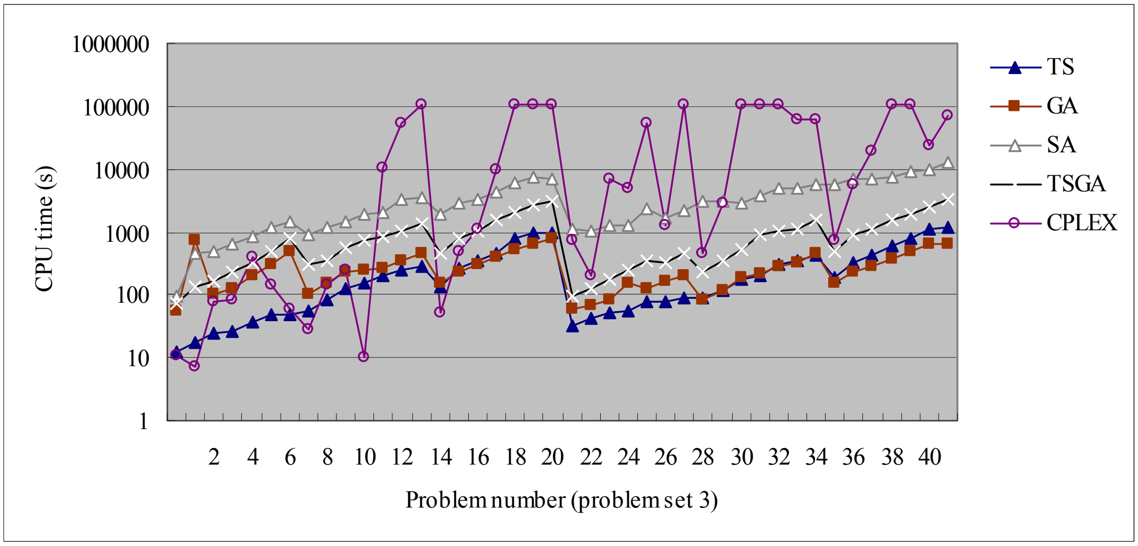
Appendix B2: Solution comparison (problem set 3)



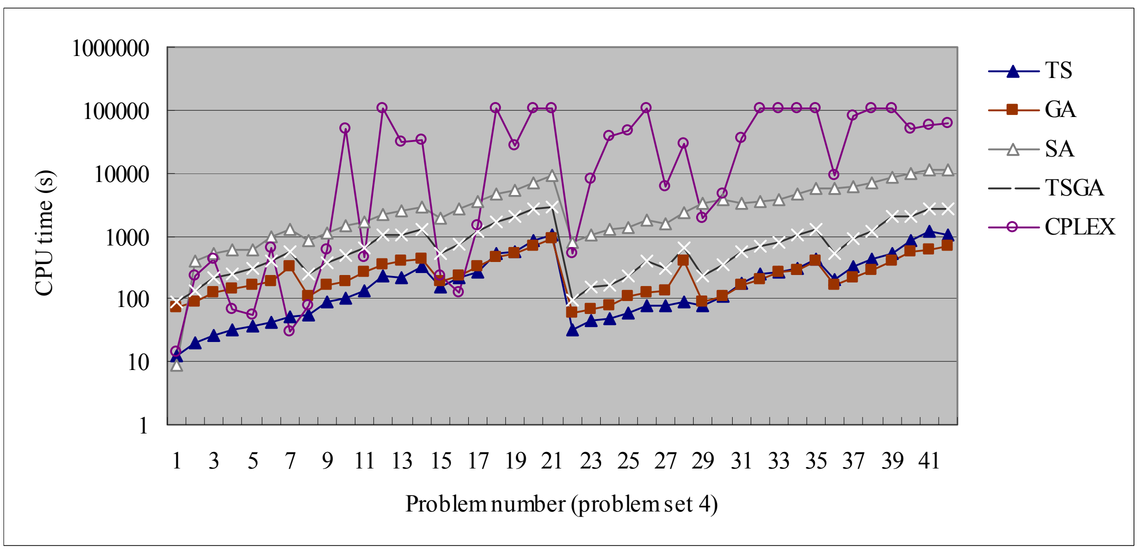
Appendix B3: Solution comparison (problem set 4)



Appendix C1: CPU time comparison (problem set 2)



Appendix C2: CPU time comparison (problem set 3)



Appendix C3: CPU time comparison (problem set 4)