Heuristic approach by Tabu search for resource allocation in communication networks

Peng, Yanjie

2005

Peng, Y. (2005). Heuristic approach by Tabu search for resource allocation in communication networks. Master's thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/3126>

<https://doi.org/10.32657/10356/3126>

Nanyang Technological University

Downloaded on 26 Apr 2023 17:41:18 SGT

A Heuristic Approach by Tabu Search for Resource Allocation in Communication Networks

Peng Yanjie

School of Electrical & Electronic Engineering

A thesis submitted to the Nanyang Technological University

in fulfilment of the requirement for the degree of

Master of Engineering

2005

Statement of Originality

I hereby certify that the work embodied in this thesis is the result of original research and has not been submitted for a higher degree to any other University or Institution.

20 APRIL 2005

Date

Peng Yanjie

Acknowledgments

I wish to express my deepest appreciation to my supervisor, Associate Professor Soong Boon Hee, for his professional guidance, invaluable instructions, and continuous encouragement.

I would like to express my sincere gratitude to Associate Professor Wang Lipo, for his helpful suggestions and comments.

I acknowledge the research scholarship provided by the School of Electrical & Electronic Engineering, Nanyang Technological University.

I am grateful to the technicians of Communication Laboratory IV, Mr. Joseph Lim Puay Chye and Ms. Hannah Lim Yoke Lan, for their useful supports.

I would like to thank my friends in Communication Laboratory IV for their generous sharing of technical knowledge and research experience.

Last but not least, I am indebted to my family for their understanding, support and encouragement.

Summary

With the huge increase in the demand for more capacity in communication networks, resources are limited and scarce. Consequently, resource allocation is a major challenge in the design and implementation of communication networks. The main objective is to improve resource utilization and users' satisfactions.

Three resource allocation problems are investigated in this thesis. They are routing and wavelength assignment (RWA) problem in wavelength-routed optical networks, channel assignment problem (CAP) in cellular telephone networks, and broadcast scheduling problem (BSP) in TDMA packet radio networks. The algorithms based on the tabu search (TS) paradigm are proposed for solving these combinatorial optimization problems.

The RWA problem is first solved by a simple greedy algorithm. Then a mixed tabu-greedy algorithm, which tries different sets of routes iteratively using tabu search and each set of routes is assigned wavelengths greedily, is proposed for RWA. The number of wavelengths is used as an estimate for the goodness of the choice.

The formulation for the static channel assignment (SCA) problem by Smith and Palaniswami is adopted. A standard tabu search algorithm is implemented to solve the fixed spectrum problem. The algorithm can also be applied to solve the minimum span problem with small modifications.

The BSP is divided into two sub-problems, i.e., minimization of the frame length and maximization of the throughput. The proposed mixed tabu-greedy algorithm has two steps accordingly. The first step, based on tabu search, finds a frame with the minimum length that transmits one and only one packet for each

station. The second step maximizes the throughput by adding greedily as many conflict-free assignments as possible to the schedule obtained in the first step.

The proposed algorithms have been evaluated on a number of benchmark problems by comparing their performance with some existing algorithms. The proposed algorithms have always produced better results in terms of resources used. Tabu search has proven to be an extremely promising heuristic for combinatorial optimization problems.

Table of Contents

Statement of Originality	i
Acknowledgments	ii
Summary	iii
Table of Contents	v
List of Figures	ix
List of Tables	x
CHAPTER 1	1
INTRODUCTION	1
1.1 Motivation	2
1.2 Objectives	3
1.3 Major Contribution of the Thesis	4
1.4 Organization of the Thesis	5
CHAPTER 2	7
PRINCIPLES OF TABU SEARCH	7
2.1 Ordinary Local Search	8
2.2 Use of Memory	9
2.3 General Operation	9

Table of Contents

2.4 Key Components.....	10
CHAPTER 3	15
ROUTING AND WAVELENGTH ASSIGNMENT	15
3.1 Introduction	16
3.1.1 Routing and Wavelength Assignment	17
3.2 Literature Review.....	18
3.2.1 Integer Linear Program	19
3.2.2 Decoupling	20
3.2.3 Edge-Disjoint Paths	21
3.3 Greedy Algorithm	22
3.3.1 Operation of GR-RWA	22
3.3.2 Difference between EDP-RWA and GR-RWA.....	23
3.4 Mixed Tabu-Greedy Algorithm	24
3.4.1 Routing Problem	25
3.4.2 Wavelength Assignment	26
3.5 Experimental Results and Discussion	28
CHAPTER 4	31
CHANNEL ASSIGNMENT PROBLEM	31
4.1 Background Information.....	32
4.1.1 Problem Description	33

Table of Contents

4.2 CAP Formulation	35
4.3 Tabu Search Algorithm.....	37
4.4 Simulation Results and Discussion	39
CHAPTER 5	48
BROADCAST SCHEDULING PROBLEM.....	48
5.1 Introduction and Literature Review	49
5.2 BSP Formulation	52
5.2.1 $P1$: Minimization of the Frame Length M	55
5.2.2 $P2$: Maximization of the Network Throughput τ	55
5.3 Two-step Algorithm	56
5.3.1 TS for $P1$	56
5.3.2 GR for $P2$	58
5.4 Computation Results and Discussion.....	59
5.4.1 Performance Metrics	59
5.4.2 Numerical Results and Discussion	60
CHAPTER 6	66
CONCLUSION AND RECOMMENDATIONS	66
6.1 Conclusion	67
6.2 Recommendations for Future Research	68
6.2.1 Wavelength Reassignment.....	68

Table of Contents

6.2.2 Base Station Location (BSL)	68
6.2.3 Joint Time Slot/Code Assignment (JTSCA).....	69
6.2.4 Evolutionary Multi-Objective Optimization (EMO)	69
Author's Publications	70
Bibliography	71

List of Figures

Figure 2.1: Flowchart of the standard tabu search algorithm	14
Figure 3.1: Four-node wavelength-routed optical network topology	17
Figure 3.2: Illustration of wavelength conversion	18
Figure 3.3: Pseudo code for the algorithm GR-RWA	23
Figure 4.1: Interference-free assignment for a four-cell, 11-channel network	34
Figure 4.2: Near interference-free assignment for a four-cell, 10-channel network	34
Figure 4.3: Cost tensor for the example four-cell network	37
Figure 5.1: Collision – (a) direct collision (b) hidden-station collision	51
Figure 5.2: The 15-station radio network used in the simulation	62
Figure 5.3: The 30-station radio network used in the simulation	62
Figure 5.4: The 40-station radio network used in the simulation	63
Figure 5.5: Broadcast schedule obtained for the 15-station network	63
Figure 5.6: Broadcast schedule obtained for the 30-station network	63
Figure 5.7: Broadcast schedule obtained for the 40-station network	64
Figure 5.8: Comparison of average delays by MFA, SVC, HNN-GA and TS-GR for the 15-station network	64
Figure 5.9: Comparison of average delays by MFA, SVC, HNN-GA, and TS-GR for the 30-station network	65
Figure 5.10: Comparison of average delays by MFA, SVC, HNN-GA, and TS-GR for the 40-station network	65

List of Tables

Table 3.1 : Comparison of TS-GR with EDP-RWA and GR-RWA, $N = 10$	29
Table 3.2: Comparison of TS-GR with EDP-RWA and GR-RWA, $N = 20$	29
Table 3.3: Comparison of GR-RWA with EDP-RWA, $N = 50$	30
Table 4.1: Problem descriptions for the HEX data set	41
Table 4.2: Problem descriptions for the KUNZ data set	41
Table 4.3: Comparison of TS with SD. SA. SONN ,HN, and HCHN	41
Table 4.4: Channel assignment for HEX1 with interference value 46	44
Table 4.5: Channel assignment for HEX2 with interference value 14	45
Table 4.6: Channel assignment for HEX3 with interference value 73	46
Table 4.7: Channel assignment for HEX4 with interference value 12	47
Table 5.1: Throughputs by MFA. SVC. HNN-GA, and TS-GR	61

CHAPTER 1

INTRODUCTION

The demand for more capacity in communication networks is growing and relentless. This demand is fueled by many factors. The fast growth of the Internet and World Wide Web, in terms of the number of users as well as the amount of time, is a major factor. The number of Internet users worldwide will exceed one billion by the end of 2005 [1]. Users connecting to the Internet via dialup lines typically stay on for an average of 20 minutes. In contrast, an average voice phone call lasts about three minutes. So an Internet call brings in about six times as much traffic into a network as a voice call [2]. The second factor is the fast growth of cellular telephone networks. The number of mobile phone users in China, the world's largest potential mobile phone market, reached 250 million at the end of 2003. And this number is expected to double to 500 million by 2007 [3]. Another factor is the growth of real-time high-quality, high-volume applications, resulting in an increasing demand for high-speed communications.

1.1 Motivation

The rapid and continuing growth of the capacity demand provides numerous challenges in the study, design and implementation of communication networks. *Resource allocation* is a major component of network utilization and quality-of-service provisioning. The resources of communication networks are primarily the bandwidths of their links and the buffer spaces at their switching nodes. With the increasing capacity demand and complexity of applications, resources are scarce and limited. These are to be properly provisioned and efficiently managed in order to achieve the highest resource utilization and maximal users' satisfactions. Optimal resource allocation plays an ever-increasing role in the design and analysis of communication networks. In general, such optimization problems are analytically and computationally intractable. So finding optimal feasible solutions for such problems by a heuristic approach motivated this research work.

Basically, networks consist of two portions, i.e., backbone portion and subscriber portion. The backbone portion requires huge bandwidths to support high-speed transmissions. Optical fiber transmission systems are widely deployed in this portion of networks. Wireline (twisted pair, coaxial cable, and optical fiber) and wireless (radio frequency and infrared) systems are widely deployed in the subscriber portion of networks. Fiber-to-the-home and wireless access will be the future trend. Fiber-to-the-home provides huge bandwidths to end-users, and wireless access provides great mobility and flexibility.

Therefore in this thesis, the resource allocation problems in optical and wireless networks are studied. Two types of wireless access networks are consider-

ed, i.e., cellular telephone networks and packet radio networks. And resources refer to the bandwidths of links.

1.2 Objectives

The objectives of this thesis are to find optimal solutions to three resource allocation problems in optical and wireless networks.

In wavelength-routed optical networks, the resource allocation problem is referred to as the *routing and wavelength assignment* (RWA) problem. To communicate with each other, an optical connection called a lightpath is set up between two entry nodes. Given a set of connection requests, RWA is to find a route and assign a wavelength to each connection request. The objective is to minimize the number of wavelengths used.

In cellular telephone networks, the resource allocation problem is referred to as the *channel assignment problem* (CAP). The term *channel* is general. It could be a fixed radio frequency (FDMA), a specific time slot within a frame (TDMA), or a particular code (CDMA), depending on the multiple access technique used. The CAP has the task of planning the reuse of the limited channels in a spectrum-efficient and interference-minimal way.

In packet radio networks (PRNs), the resource allocation problem is referred to as the *broadcast scheduling problem* (BSP). The BSP is to find a collision-free schedule of the transmissions of all the stations in a minimum number of time slots. The objective is to minimize the channel utilization and the average delay.

1.3 Major Contribution of the Thesis

In this thesis, novel approaches based on the *tabu search* (TS) paradigm are proposed for solving the resource allocation problems in optical and wireless networks.

In wavelength-routed optical networks, routing and wavelength assignment is divided into two sub-problems, i.e., routing problem and wavelength assignment problem. The routing scheme used is a *k*-shortest-path algorithm. The mixed tabu-greedy algorithm proposed tries different sets of routes iteratively. Each set of routes is assigned wavelengths, and the number of wavelengths is used as an estimate for the goodness of the choice. Tabu search can be used to make route selection and assign wavelengths. TS is used to select routes in this thesis. However, a greedy algorithm is employed to assign wavelengths since the running time of TS increases dramatically as a function of the number of nodes in the network.

In cellular telephone networks, the formulation for the static channel assignment (SCA) problem by Smith and Palaniswami is adopted. A standard tabu search algorithm is implemented to solve the fixed spectrum problem. The algorithm can also be applied to solve the minimum span problem with small modifications.

In packet radio networks, the broadcast scheduling problem is divided into two sub-problems. The first attempts to find a collision-free frame with the minimum frame length that allows every station to transmit once and only once. The second one is to maximize the throughput at the found frame length. The mixed tabu-greedy algorithm proposed has two steps accordingly. For the first step, a tabu

search algorithm is used and for the second step, a simple greedy algorithm is applied.

Numerical results have shown that the proposed algorithms produce better solutions to the resource allocation problems concerned in communication networks, compared with some existing algorithms.

1.4 Organization of the Thesis

The rest of the thesis is organized as follows.

In Chapter 2, the background and general operation of tabu search are presented. Key components of TS are also elaborated.

In Chapter 3, the background of wavelength-routed optical networks is introduced. The literature of routing and wavelength assignment is reviewed. A simple intuitive greedy algorithm for solving the routing and wavelength assignment problem is presented, followed by the mixed tabu-greedy algorithm. Experimental results are discussed and compared with those obtained by some existing algorithms.

In Chapter 4, the background and literature review of the channel assignment problem in cellular telephone networks are described. The nonlinear integer programming formulation of the static channel assignment problem by Smith and Palaniswami is stated. The tabu search algorithm for solving the CAP is presented. Computational results are discussed and compared with those obtained by some existing algorithms.

In Chapter 5, the background and literature review of the broadcast scheduling problem are introduced. The formulation and mixed tabu-greedy

algorithm for the BSP in packet radio networks are presented. Simulation results are discussed and compared with those obtained by some existing algorithms.

In Chapter 6, the conclusion of this research work is summarized and some ideas are recommended for futurework.

CHAPTER 2

PRINCIPLES OF TABU SEARCH

The term *tabu search* (TS) was first introduced in [4] by Fred Glover, on whose proposed ideas the basic form of TS is founded. Similar ideas were also sketched in [5] by Pierre Hansen. TS is comprehensively explained in [6]- [8].

Tabu search is a meta-heuristic method that guides a local heuristic search procedure to explore the solution space beyond local optimality. To accomplish this, TS allows even non-improving solutions and employs adaptive memory to avoid a cyclic behavior. The basic idea is to forbid certain moves that would return to recently visited solutions, by rendering them tabu.

2.1 Ordinary Local Search

Tabu search begins in the same way as ordinary local or neighborhood search, proceeding iteratively from one solution to another until a chosen termination criterion is satisfied. Each solution $s \in S$ has an associated neighborhood $N(s) \subset S$ and each solution $s' \in N(s)$ is reached from s by an operation called a move.

Given a feasible starting solution, ordinary local search, for example, a simple *descent* method, iterates the following steps:

- 1) Choose $s \in S$ as the initial solution.
- 2) Find $s' \in N(s)$ such that the objective function value $f(s')$ is the minimum within $N(s)$.
- 3) If $f(s') < f(s)$, designate s' to be the new s and go to 2).
- 4) Otherwise, s is the local optimum and the method stops.

Ordinary local search implementations are guaranteed to yield globally optimal solutions for the problems they are designed to handle. However, in most cases, ordinary local search terminates with a local optimum that is not a global optimum. Ordinary local search is easily trapped in a poor local optimum. To escape from the trap, non-improving solutions are allowed to be designated as the new solution to continue the search. This is called a *descent-ascent* method.

The major problem with a descent-ascent method is cycling due to the lack of the knowledge of the search history. To avoid cyclic behavior, search methods must incorporate memory, which is an integral component of any search that deserves to be called “intelligent”.

2.2 Use of Memory

Memory is used in tabu search to prevent the search from revisiting previously visited solutions (short-term memory) and explore the unvisited areas of the solution space (long-term memory). The memory used is both explicit and attributive. Explicit memory records complete solutions, typically consisting of elite solutions visited during the search. This memory is used less frequently, because clever data structures must be designed to avoid excessive memory requirements. Attributive memory records information about solution attributes that change in moving from one solution to another. The information recorded in the memory is classified tabu. Guided by the information collected during the search, TS makes responsive exploration, i.e., exploiting good solutions while exploring new promising regions. A bad strategic choice made by responsive exploration can yield more information than a good random choice.

Tabu search uses adaptive (flexible) memory. The information recorded in the memory changes with time and circumstance. Adaptive memory contrasts with rigid memory designs typical of branch and bound, in which branching strategies are fixed.

2.3 General Operation

In contrast with ordinary local search, tabu search goes beyond local optimality by performing even non-improving moves and incorporating adaptive memory to avoid cycling and explore less frequently visited solution space.

Given an initial solution, tabu search computes the neighborhood at each iteration, but selects the solution with the best objective function value within the neighborhood, even if it's worse than the current solution.

Tabu search forbids the selection of a solution in the neighborhood if it recognizes that solution as one selected at previous iterations. To identify a solution, TS records in a memory structure, called tabu list, some information, called attributes, of the selected solution. After a certain number of iterations, called tabu tenure, the attributes in the tabu list will be released. The tabu status of an attribute is not absolute, but can be overruled if some condition, called aspiration criterion, is verified. To stop the search progress, a termination criterion is predefined. The flowchart of the standard tabu search algorithm is shown in Figure 2.1.

2.4 Key Components

In this section, key parameters that appear in most tabu search implementations are explained.

Initial Solution: Tabu search starts with an initial feasible solution, which is in the same way as ordinary local or neighborhood search. The initial solution is selected at random or determined by a certain initialization procedure.

Move: A move is a simple modification to the current solution. A move varies from problem to problem, and even for the same problem instance, there is more than one possible case of moves.

Neighborhood The neighborhood $N(s)$ of the current solution s consists of all the new solutions that are one move from it. In general, it is computationally expensive to evaluate the entire neighborhood. Fortunately, memory, both short-

and long-term memory, is used to modify the neighborhood of the current solution. In short-term strategies, tabu classification serves to identify elements of $N(s)$ excluded from the modified neighborhood $N^*(s)$. In long-term considerations, $N^*(s)$ may also be expanded to include solutions not ordinarily found in $N(s)$.

The moves that produce $N^*(s)$ are not of equal importance. A subset of $N^*(s)$, often called *candidate list*, which contains the neighbors that are produced by some "influential" moves, is evaluated. And the best solution is selected as the new current solution.

Tabu search may be viewed as a dynamic neighborhood method. The candidate list of s is not a static set, but rather a set that can change according to the history of the search.

Short-Term Memory: The most commonly used short-term memory keeps track of solution attributes that have changed during the recent past, and is also called *recency-based* memory. Short-term memory is used to prevent the search from being trapped in local optimality.

To exploit this memory, selected attributes that occur in solutions recently visited are labeled tabu-active, and solutions that contain tabu-active attributes or particular combinations of these attributes, become tabu. This prevents certain solutions from the recent past from belonging to the neighborhood and hence from being revisited.

Long-Term Memory: In some applications, short-term memory components are sufficient to produce very high-quality solutions. However, in general, TS becomes significantly stronger by including long-term memory and its associated strategies. Long-term memory is normally employed to achieve a diversification

effect, encouraging the search to explore regions less frequently visited and discouraging the moves occurred too frequently in the search. The *frequency-based* approach is most commonly implemented. Frequency-based memory provides a type of information that complements the information provided by recency-based memory, broadening the foundation for selecting preferred moves. Transition frequencies keep track of how often attributes change, and residence frequencies take record of how often attributes are members of the solutions generated. Attributes that have greater frequency measures than a certain *frequency threshold* can trigger a tabu-activation rule.

Tabu List: Memory, both recency- and frequency-based, is managed by creating one or several tabu lists, which record the tabu-active attributes and implicitly or explicitly identify their current status. If memory space is sufficient to store one piece of information (e.g., a single integer) for each solution attribute used to define the tabu-activation rule, it is usually advantageous to record the iteration number that identifies when the tabu-active status of an attribute starts or ends.

Tabu Tenure: Tabu tenure is the number of iterations for which the tabu-active status of attributes in the tabu list will last. Tabu tenure can vary for different types or combinations of attributes, and can vary over different intervals of time or stages of search. Varying the tabu tenure during the search provides one way to induce a balance between closely examining one region and moving to different parts of the solution space.

A good range of tabu tenure values needs to be determined. Too small tenure values cannot guide the search to explore the solution space effectively. If the value is too large, on the other hand, it is possible that all moves are forbidden. This unnecessarily takes longer time to select a move.

Aspiration Criterion: Aspiration criteria are introduced in tabu search to determine when tabu-activation rules can be overridden, thus removing a tabu classification otherwise applied to a move. The appropriate use of such criteria can be very important for enabling tabu search to avoid missing good solutions and achieve its best performance levels.

One simplest instance of such criteria is aspiration by default. If all available moves are classified tabu, then the “least-tabu” move or a random move is selected and its tabu-active status is revoked.

A more widely used criterion is aspiration by objective. A tabu classification from a trial move is removed when the move yields a solution better than the best obtained so far.

Termination Criterion: There are several possible stopping criteria:

- 1) An optimal solution is found.
- 2) A preset time period is expired.
- 3) The neighborhood set is null.
- 4) The number of iterations is greater than the maximum number of iterations allowed.
- 5) The number of iterations performed since the current solution last changed is greater than a specified maximum number of iterations.

Figure 2.1 : Flowchart of the standard tabu search algorithm

CHAPTER 3

ROUTING AND WAVELENGTH ASSIGNMENT

Optical media constitute a promising alternative as a communication infrastructure. A single fiber offers a huge transmission bandwidth (a few tens of terabits per second) at a very low bit error rate ($\sim 10^{-15}$). However, the bandwidth offered by a single fiber is too large for a single application. Hence, it should be shared between users. This motivates the need for multiplexing. The idea of wavelength-division multiplexing (WDM) is to transmit data simultaneously at multiple carrier wavelengths over a fiber. These wavelengths do not interfere with each other provided that they are kept sufficiently far apart. Thus WDM makes a single fiber look like multiple virtual fibers, with each virtual fiber carrying a single data stream at a speed compatible with the electronic processing speed (~ 10 Gbps).

WDM networks are those where different signals using different wavelengths are placed on a single fiber. The fiber, which is called WDM link, connects nodes of WDM networks. Optical transmitters and receivers are located at

both ends. Most commonly used architectural approaches to implementing WDM systems are broadcast and select networks and wavelength-routed networks. Broadcast and select networks are suitable for local area networks, and not scalable to wide area. Wavelength-routed networks are the primary choice for wide area all-optical networks.

3.1 Introduction

Wavelength-routed WDM networks are made of optical cross-connects (OXC) connected to each other by fiber links. An OXC has a large number of ports (ranging from a few tens to thousands) and are able to switch wavelengths from one port to another. An optical connection is provided between two entry points, on which both transmitters and receivers are located. The wavelength used at each fiber from source to destination also determines the route. Consequently such systems are called *wavelength-routed* networks. In addition, the process of using wavelengths to route connections in optical networks is called wavelength routing. Once the connection is established, the network resources that are the wavelengths over each fiber are allocated during the hold time of the connection. This connection is called a *lightpath*. A simple network configuration is shown in Figure 3.1.

In the absence of wavelength converters, a lightpath is formed by selecting the same wavelength along the path. The constraint of selecting the same wavelength over each link is called the *wavelength-continuity* constraint. Wavelength-routed networks allocating different wavelengths on different links along a path are called wavelength-switched networks. If each node is armed with full wavelength-conversion capability, the network is equivalent to a circuit-

switched network. Only the routing problem needs to be addressed; wavelength assignment is not an issue. The idea of wavelength conversion is illustrated in Figure 3.2 [9]. In Figure 3.2(a), without wavelength conversion, the connection from node 1 to node 3 using wavelength λ_1 will be blocked since λ_1 is used by the connection from node 2 to node 3. In contrast, in Figure 3.2(b), the wavelength converter equipped at node 2 can convert wavelength λ_1 to the free wavelength λ_2 and thus it avoids wavelength conflict. Clearly, the presence of wavelength converters at routing nodes can reduce the blocking probability experienced by connection requests.

Figure 3.1: Four-node wavelength-routed optical network topology

3.1.1 Routing and Wavelength Assignment

In wavelength-routed networks, given a set of connection requests, the problem of providing a route and assigning a wavelength to each connection request is called the *routing and wavelength assignment* (RWA) problem. The wavelengths assigned must be such that no two lightpaths sharing a common physical fiber link use the same wavelength. In Figure 3.1, lightpaths 1-2-3 and 2-3 share a common fiber link 2-3, hence they are assigned two different wavelengths λ_1 and λ_2 . Lightpaths 1-2-4

and 2-3 do not have any common fiber links, so they can be assigned the same wavelength λ_1 .

Figure 3.2: Illustration of wavelength conversion

Practical limitations on transmission technology and optical devices restrict the number of available wavelengths on a fiber, so RWA is critically important in increasing the efficiency of wavelength-routed optical networks. With a good solution to this problem, more customers can be accommodated by the given system and fewer customers need to be rejected during periods of congestion.

3.2 Literature Review

Many algorithms have been proposed to solve the routing and wavelength assignment problem in the literature. These algorithms may be one of the three categories: RWA was formulated as an integer linear program; RWA was decoupled into routing problem and wavelength assignment problem; and RWA was solved by applying edge-disjoint paths algorithms.

3.2.1 Integer Linear Program

The static RWA problem with the wavelength-continuity constraint can be formulated as an integer linear program (ILP) [10],[11], in which the objective function is to minimize the flow in each link, which, in turn, corresponds to minimizing the number of lightpaths passing through a particular link. Let λ_{sdw} denote the traffic (the number of connection requests) from any source s to any destination d on any wavelength w . Two or more lightpaths may be set up between the same source-destination pair, if necessary, but each of them must employ a distinct wavelength, hence $\lambda_{sdw} \leq 1$. Let F_{ij}^{sdw} denote the traffic (the number of connection requests) from source s to destination d on link $i-j$ and wavelength w . $F_{ij}^{sdw} \leq 1$ since a wavelength on a link can be assigned to only one path. Given a network topology, a set of wavelengths, and the traffic matrix Λ in which Λ_{sd} denotes the number of connections needed between source s and destination d , the problem can be formulated as follows (which turns out to be an ILP):

$$\text{Minimize:} \quad F_{\max} \quad (3.1)$$

$$\text{such that} \quad F_{\max} \geq \sum_{s,d,w} F_{ij}^{sdw} \quad \forall ij \quad (3.2)$$

$$\sum_i F_{ij}^{sdw} - \sum_k F_{jk}^{sdw} = \begin{cases} -\lambda_{sdw}, & \text{if } s = j \\ \lambda_{sdw}, & \text{if } d = j \\ 0, & \text{otherwise} \end{cases} \quad (3.3)$$

$$\sum_w \lambda_{sdw} = \Lambda_{sd} \quad (3.4)$$

$$F_{ij}^{sdw} = 0, 1 \quad (3.5)$$

$$\sum_{s,d} F_{ij}^{sdw} \leq 1. \quad (3.6)$$

3.2.2 Decoupling

Most algorithms in the literature fall into this category. The RWA problem is decoupled into two sub-problems, i.e., routing sub-problem and wavelength assignment sub-problem, where the two sub-problems are solved separately.

3.2.2.1 Routing

For the routing sub-problem, there are two basic approaches that can be found in the literature: shortest-path (SP) [12]–[14] and k-shortest-path (K-SP) [15]–[17].

Shortest-Path: The shortest-path route for each connection request is calculated offline using the standard shortest-path algorithms such as Dijkstra's algorithm and any connection between the specified pair of nodes is established using the predetermined route. This approach to routing connections is very simple. However, the performance is relatively poor.

K-Shortest-Path: In this approach, a list of fixed candidate routes is maintained for each connection request. For example, these routes may include the shortest-path route, the second shortest-path route, the third shortest-path route, and so on. When a connection request arrives, the source node attempts to establish a connection on each of the candidate routes, until a route with a valid wavelength assignment is found. In this way, the routing problem is transformed into a selection problem.

3.2.2.2 Wavelength Assignment

After routes have been chosen for connections, assigning wavelengths to different routes in a manner that minimizes the number of wavelengths used under the wavelength-continuity constraint is equivalent to a graph-coloring problem. This

can be solved in a sequential, optimal or heuristic manner. The *sequential* approach sorts the routes to be assigned using the largest-number-of-neighbors-first [14], longest-path-first [18], [19], or random [18] scheme. Then, wavelengths are assigned to the sorted routes using the first-fit [18], or most-used [19] scheme. The *optimal* approach is actually exhaustive search, which always produces the best result but cannot handle large networks. *Heuristic* methods work well with graph-coloring problems. Several of these algorithms are simulated annealing (SA) [14], genetic algorithms (GAs [14], and tabu search (TS) [14], [20].

3.2.3 Edge-Disjoint Paths

The edge-disjoint paths algorithm was applied to RWA by Manohar *et al.* [21]. Given the graph $G(V, E)$ and the connection request set $C = \{(s_1, t_1), \dots, (s_k, t_k)\}$, C is said to be realizable in G if there exist mutually edge-disjoint paths $P_1, \dots, P_k$ so that P_i has endpoints s_i and t_i . The maximum edge-disjoint paths (MEDP) problem is to find a maximum-size realizable subset of C , given G and C . Let $C_1, \dots, C_i$ be a partition of C so that each element of the partition is a solution to the MEDP problem. The application to RWA is as follows. Since the paths in C_1 are edge disjoint, they can be assigned the same wavelength, say λ_1 . Now construct the set $C' = C - C_1$, i.e., C' is the set of connections not contained in C_1 . An iteration of MEDP on G and C' will give C_2 and the paths in C_2 can be assigned the same wavelength, say λ_2 . This process can be repeated until C' is empty for some i . Finding minimum such i is the problem of minimizing the number of wavelengths in wavelength-routed optical networks. The bounded greedy algorithm for MEDP that was first described in [22] was used by Manohar *et al.* The upper bound B for

the number of links in a path was chosen to be $\max(\text{diam}(G), \sqrt{|E|})$, where the diameter of G , $\text{diam}(G)$, is defined as the length of the longest path among shortest paths between all pairs of vertices, and E is the number of directed edges in G .

3.3 Greedy Algorithm

A simple greedy algorithm, called GR-RWA, is presented to solve the RWA problem in this section. This method is developed by modifying slightly the algorithm EDP-RWA in [21].

3.3.1 Operation of GR-RWA

Given the physical topology of a network and the connection request set $C = \{(s_1, t_1), \dots, (s_k, t_k)\}$. Let G_i denote the graph of the network in which all the routes will be assigned the same wavelength i . An upper bound B for the number of edges in a route is also defined. It has been argued by Kleinberg [22] that a reasonable value for B was $\max(\text{diam}(G), \sqrt{|E|})$, which was also used in [21].

The inputs to the algorithm are the graph G of the physical network, the connection request set C , and the bound B . The outputs of the algorithm are the number of wavelengths λ and P that is the set of pairs (c_j, P_j) ($j=1, \dots, k$), where c_j is a connection request and P_j is the lightpath assigned to it.

The algorithm operates as follows. First, select a request c_j from the set C , and try to find the shortest path P_j in the graph G_1 associated with the first wavelength. If such a path cannot be found on the first wavelength, the second wavelength is attempted. This process is repeated until P_j is realized on a certain

wavelength i . Then add (c_j, P_j) to P , delete c_j from C and remove all the edges on P_j from the graph G_i . After that, select a new request from C and find a lightpath for it. The search continues until all the requests have been routed. The pseudo code for this algorithm is shown in Figure 3.3. The shortest path P_j is selected by an offline Dijkstra's algorithm.

```

Begin
   $B = \max(\text{diam}(G), \sqrt{|E|})$ 
   $k$  = Number of connection requests in  $C$ 
   $\lambda = 0$ 
   $P = \emptyset$ 
  For  $j = 1$  to  $k$ 
    Select a request  $c_j$  from  $C$  at random
     $i = 1$ 
    flag = 1
    While (flag = 1) do
      If ( $\text{length}(P_j \text{ on } G_i) \leq B$ ) then
        Select  $P_j$  for  $c_j$ 
         $P = P \cup (c_j, P_j)$ 
         $G_i = G_i - P_j$ 
         $C = C - c_j$ 
        flag = 0
      Else
         $i = i + 1$ 
      EndIf
    EndWhile
    If ( $i > \lambda$ )
       $\lambda = i$ 
    EndIf
  EndFor
End

```

Figure 3.3: Pseudo code for the algorithm GR-RWA

3.3.2 Difference between EDP-RWA and GR-RWA

EDP-RWA works as follows.

- 1) A connection request c_j is selected from C at random.
- 2) If a route P_j can be calculated for c_j on G_i , c_j is deleted from C and the edges on P_j are removed from G_i .

- 3) Otherwise, go to 1). These are iterated k times, where k is the number of requests in C .
- 4) The first three steps are repeated with the new C and G_{i+1} .
- 5) The search continues until C is empty.

GR-RWA operates as follow.

- 1) A connection request c_j is selected from C at random.
- 2) A route P_j is calculated for c_j on the first G_i that can support c_j .
- 3) c_j is deleted from C and the edges on P_j are removed from G_i .
- 4) The search continues until C is empty.

The difference between the two algorithms is clear. EDP-RWA tries to route as many requests as possible on a wavelength before introducing a new one. GR-RWA introduces a new wavelength immediately after a request fails to be routed on the current one.

3.4 Mixed Tabu-Greedy Algorithm

The routing scheme used in this thesis is a k -shortest-path approach, and k is limited by the upper bound B . Candidate paths whose lengths are one, two, ..., B , in terms of the number of links, are calculated by a modified Dijkstra's algorithm, if such routes can be supported. Once the candidates are determined, tabu search is used to select one route for each connection request iteratively. Each set of routes is assigned wavelengths in a greedy manner. The objective function value is the number of wavelengths used to route all the connection requests.

3.4.1 Routing Problem

The tabu search implementation is shown as follows.

Initial Solution: Assigning one arbitrary route from the k candidates to each connection request produces the initial solution.

Move: A move is a basic flip-flop, which is generated by replacing the in-use route with an arbitrary one from the rest $k - 1$ unused candidates. In other words, one connection request is routed on a different path. A move can be characterized by a triplet $\langle r, j, i \rangle$, which means the pair (r, j) flip-flops to (r, i) . Connection request r routed on candidate route j currently will be routed on candidate route i .

Neighborhood: The neighborhood of the current solution S_c contains the new solutions that are one move from it. The size of the neighborhood is equal to the total number of connection requests here. The algorithm tries to route each connection request on a different path once an iteration.

Tabu List: Only short-term memory is implemented in this thesis. Every candidate route is reserved a space in the tabu list, which is a one-dimensional matrix L . The size of L equals the total number of candidates calculated for all the connection requests.

When connection request r routed currently on candidate j flip-flops to be routed on candidate i , the pair (r, i) is classified tabu. And the value of element l_i of the tabu list is set to the tabu tenure T . The value of each element of the tabu list decreases by one after an iteration until it becomes zero. Then the corresponding move is set free. At the end of an iteration, the best admissible solution is selected as the new current solution S_{CN} . Suppose S_{CN} was generated from S_c by a flip-flop of (r, j) to (r, i) . The pair (r, i) was labeled tabu immediately after the flip-flop.

However, its tabu-active status should be removed since, as an attribute of S_{CN} , the counterpart assignment of the new neighbor at the next iteration must be different from it. So it is unnecessary to tabu the pair (r, j) should be classified tabu simply because it occurred recently.

Tabu Tenure: The value is chosen to be

$$T = \alpha(k_r - 1) \quad (3.7)$$

where k_r is the number of candidate routes for connection request r and α typically takes values from 0.25 to 0.75. So T varies with α .

Aspiration Criterion: Aspiration by objective is used in this thesis. The move whose attribute is in the tabu list must be checked against the aspiration criterion to determine whether it is used or discarded. If it passes the check, i.e., the move leads to a solution with a cost smaller than the minimum cost obtained so far, the tabu-active status of the move is revoked for the current iteration. Otherwise, the move is discarded and a new random assignment is generated.

Termination Criterion: The search process stops when the number of iterations performed exceeds $R \times B$, where R is the total number of connection requests and B , the upper bound for the number of links in a path.

3.4.2 Wavelength Assignment

Wavelength assignment can be done in a sequential, heuristic or optimal manner. However, the heuristic (including tabu search) and optimal approaches are difficult to be applied to large networks. So a greedy algorithm is used to make wavelength assignment.

Two problems concerning wavelength assignment by a greedy method are the *selection order* in which the routes are to be assigned wavelengths, and the *selection rule* that which wavelength is assigned to a specific route. There are a few solutions to the two concerns. Two schemes that have been reported the best in the literature are adopted. The mechanism for the selection order is the largest-number-of-neighbors-first (LN) scheme, and the selection rule, first-fit (FF).

The LN scheme is based on the number of neighbors that are paths sharing one or more common fiber links. Intuitively, if a route has a large number of neighbors, it is hard to assign a wavelength since an unallocated identical wavelength must be found on more routes. The wavelength assignment of the route with the largest number of neighbors early avoids the need for using a large set of wavelengths. Therefore, by establishing LN lightpaths first, a better wavelength reuse should be achievable. This results in an overall smaller requirement of wavelengths for a given connection request set.

In the FF scheme, all the wavelengths are numbered. When searching for available wavelengths, a lower-numbered wavelength is considered before a higher-numbered one. The first available wavelength is then selected. This scheme requires no global information, so the complexity is low. The idea behind this scheme is to pack all of the in-use wavelengths towards the lower end of the wavelength space so that continuous longer routes towards the higher end of the wavelength space have a higher probability of being available.

3.5 Experimental Results and Discussion

Some numerical results obtained by GR-RWA and the mixed tabu-greedy algorithm are presented and compared with those obtained by EDP-RWA [21]. The performance is evaluated in terms of the number of wavelengths used to route a given set of connection requests in a given physical topology. The comparison is also made in terms of the execution time. The detail of the simulation is as follows. A random graph G for the physical topology is obtained by specifying the number of nodes in the graph N , and the probability of an edge, P_e between any node pair. The connection request set C is also generated randomly by populating the $N \times N$ matrix with 1's and 0's where the probability of 1's is specified by P_i and a 1 in position (i, j) indicates a connection request between node i and node j . For given G and C , ten runs are performed for every algorithm and the assignment that uses the least number of wavelengths is chosen.

Tables 3.1 – 3.3 compare the performance of GR-RWA with EDP-RWA. GR-RWA produces better results in terms of the number of wavelengths used, especially under heavy traffic. When the number of wavelengths needed becomes higher, the performance of GR-EDP becomes even better. EDP-RWA runs slightly faster than GR-RWA, from Table 3.3.

Tables 3.1 and 3.2 also compare the performance of the tabu-greedy algorithm (TS-GR) with EDP-RWA and GR-RWA. TS-GR has better results in terms of the number of wavelengths used than the two greedy algorithms. However, the computation time for TS-GR is much greater and increases dramatically with the demand of requests and the size of networks. If speed is the concern, GR-RWA may be the choice. If ultimate quality of solutions is the issue, TS-GR can be used.

Table 3.1 : Comparison of TS-GR with EDP-RWA and GR-RWA, $N = 10$

N	P_e	P_l	EDP-RWA		GR-RWA		TS-GR	
			Max/min λ	Time(s)	Max/min λ	Time(s)	Max/min λ	Time(s)
10	0.2	0.1	4 / 5	0.000	4 / 5	0.000	4 / 4	0.000
10	0.2	0.2	6 / 7	0.000	6 / 7	0.000	6 / 7	0.000
10	0.2	0.3	8 / 9	0.000	8 / 9	0.000	8 / 8	0.100
10	0.2	0.4	11 / 12	0.000	11 / 11	0.000	10 / 11	1.600
10	0.2	0.5	15 / 18	0.000	13 / 16	0.000	13 / 14	2.500
10	0.2	0.6	20 / 23	0.000	19 / 20	0.000	18 / 20	5.900
10	0.2	0.7	25 / 27	0.000	23 / 24	0.000	21 / 24	9.300
10	0.2	0.8	31 / 35	0.000	28 / 30	0.000	27 / 27	19.6
10	0.2	0.9	36 / 38	0.000	34 / 35	0.000	32 / 33	38.1

Table 3.2: Comparison of TS-GR with EDP-RWA and GR-RWA, $N = 20$

N	P_e	P_l	EDP-RWA		GR-RWA		TS-GR	
			Max/min λ	Time(s)	Max/min λ	Time(s)	Max/min λ	Time(s)
20	0.2	0.1	3 / 4	0.000	3 / 4	0.000	3 / 4	2.6
20	0.2	0.2	5 / 7	0.000	5 / 6	0.000	5 / 5	28
20	0.2	0.3	8 / 10	0.000	8 / 10	0.000	7 / 8	319
20	0.2	0.4	10 / 12	0.000	10 / 12	0.000	9 / 11	1364
20	0.2	0.5	15 / 18	0.000	14 / 16	0.000	12 / 15	3017
20	0.2	0.6	18 / 21	0.100	16 / 18	0.100	15 / 16	7254
20	0.2	0.7	23 / 25	0.100	20 / 26	0.100	18 / 25	10874
20	0.2	0.8	34 / 37	0.100	30 / 33	0.100	28 / 31	28269
20	0.2	0.9	41 / 46	0.200	36 / 40	0.200	35 / 38	57146

Table 3.3: Comparison of GR-RWA with EDP-RWA, $N = 50$

N	P_e	P_l	EDP-RWA		GR-RWA	
			Max/min λ	Time(s)	Max/min λ	Time(s)
50	0.2	0.1	2 / 3	0.500	2 / 3	0.500
50	0.2	0.2	5 / 5	2.000	5 / 5	2.000
50	0.2	0.3	7 / 8	4.400	6 / 7	4.600
50	0.2	0.4	9 / 10	6.300	7 / 9	6.500
50	0.2	0.5	13 / 16	7.400	12 / 14	7.800
50	0.2	0.6	16 / 16	9.600	14 / 16	9.900
50	0.2	0.7	20 / 25	12.1	18 / 21	12.6
50	0.2	0.8	24 / 27	16.7	20 / 24	17.2
50	0.2	0.9	31 / 36	21.4	26 / 30	21.7

CHAPTER 4

CHANNEL ASSIGNMENT PROBLEM

Since the early 1990s, the boom in mobile telecommunications has been seen in nearly all countries across the globe. The use of a radio interface in communications introduces a certain number of differences as compared with cable communication, one of which is the limited spectrum. The radio spectrum, and consequently the available capacity for radio access, is generally limited by national and international regulations. Unlike in fixed communications, where a growing population and demand can easily be satisfied by the installation of extra cables to connect subscribers to the network, the capacity of the radio spectrum cannot be extended arbitrarily. This problem is partly solved by the use of the cellular techniques, in which the service zone is divided into cells, each of which is served by a base station. The radio spectrum is used again and again as many times as possible in other geographical locations.

4.1 Background Information

With the ever-increasing popularity of mobile cellular communications, the limited radio spectrum places a restriction on this promising scenario. When two users are close geographically, interference occurs if they transmit on channels that are close in the radio spectrum. In general, interference can be classified into three categories: a *co-channel* interference – when the same channel is assigned in several cells from different clusters simultaneously; an *adjacent-channel* interference – when channels adjacent in the radio spectrum are used in cells adjacent geographically; and a *co-site* interference – when channels assigned to the same cell are not separated from each other by a minimal separation in frequency. There are at least two approaches coping with the lack of the radio spectrum. One is to improve the design and specifications of the hardware used to transmit and receive radio signals. If a very tight filter is available to remove undesired frequencies effectively, it is possible to increase the traffic capacity. This approach is important but not the only way. Optimal assignment of frequency channels is the other approach to solving the scarcity of the radio spectrum, which provides motivation for the *channel assignment problem* (CAP). The CAP is to assign the required number of channels to each base station in such a way that the radio spectrum is used most efficiently and the interference among users is minimized.

The channel assignment problem first appeared in the 1960s [23]. The development of new wireless services such as the first cellular networks led to scarcity of usable frequencies in the radio spectrum. In the late 1980s and early 1990s, the development of the GSM systems led to a rapidly increasing interest for channel assignment [24]. But projects on other applications such as military

wireless communication and radio/TV broadcasting contributed to the literature of channel assignment. So far, many methods have been proposed to tackle the channel assignment problem. Most early approaches used so-called greedy algorithms [25], [26]. Some heuristic methods included neural networks (NNs) [27] – [30], genetic algorithms (GAs) [31], [32], simulated annealing (SA) [33], [34], and tabu search (TS) [35], [36].

4.1.1 Problem Description

The channel assignment problem can be classified into two categories: static channel assignment (SCA) and dynamic channel assignment (DCA). A static channel assignment is made to satisfy the requirement for the immediate future based on the traffic forecast. The SCA is made only once and remains unchanged. As the traffic demand increases, it will become more and more difficult to maintain a satisfactory performance. A dynamic assignment strategy, which allows adaptive reassignment of channels, will need to be deployed. In this thesis, only the SCA is dealt with.

Suppose the whole network is divided into N cells and the total number of channels available is M . The channel demand is given in a one-dimensional matrix D . Noninterference constraints are stored in a symmetrical $N \times N$ compatibility matrix C . C_{ij} gives the minimal frequency separation between channels assigned to cells i and j , for interference-free assignments.

As an example, suppose the number of cells in the network is $N=4$, and the demand for channels in each of these cells is given by $D=[1,1,1,3]$. Consider the compatibility matrix suggested by Sivarajan *et al.* [26]

$$C = \begin{bmatrix} 5 & 4 & 0 & 0 \\ 4 & 5 & 0 & 1 \\ 0 & 0 & 5 & 2 \\ 0 & 1 & 2 & 5 \end{bmatrix}$$

The diagonal terms $C_{ii}=5$ indicate that any two channels assigned to the same cell must be at least five frequencies apart in order that no co-site interference exists. Channels assigned to cells 1 and 2 must be at least $C_{12}=4$ frequencies apart. Off-diagonal terms of $C_{ij}=1$ and $C_{ij}=2$ correspond to co-channel and adjacent-channel constraints, respectively. Thus, the solution represented in Figure 4.1 is one of the interference-free assignments that can be generated for $M=11$, and the solution in Figure 4.2 incurs a weak interference.

Figure 4.1 : Interference-free assignment for a four-cell, 11-channel network

Figure 4.2: Near interference-free assignment for a four-cell, 10-channel network

Given the above parameters, the SCA can be further divided into two types: minimum span problem and fixed spectrum problem. The minimum span problem is to minimize the span of channels (M) subject to the demand constraints (D) and the noninterference constraints (C), i.e., to determine the minimal number of channels (M) required to achieve an interference-free assignment, given N , C and D . The fixed spectrum problem is to minimize the severity of interference subject to the demand constraints (D), i.e., to minimize the interference level given all the four parameters, M , N , C and D . In practice, the total number of channels available is normally fixed, so the algorithm developed in this chapter is only applied to the fixed spectrum problem. However, with some small modifications, the algorithm can also be used to solve the minimum span problem.

4.2 CAP Formulation

The SCA problem can be formulated as various models. In this thesis, the formulation by Smith and Palaniswami [27] is used.

First, a set of binary variables are defined:

$$X_{j,k} = \begin{cases} 1, & \text{if channel } k \text{ is assigned to cell } j \\ 0, & \text{otherwise} \end{cases} \quad (4.1)$$

for $j = 1, \dots, N$ and $k = 1, \dots, M$. Suppose that $X_{j,k} = X_{i,l} = 1$, i.e., calls in cells j and i have been assigned channels k and l , respectively. One way to measure the degree of interference caused by such assignments is to weight each pair of assignments by a cost tensor $P_{j,i,m+1}$, where $m = |k - l|$ is the distance between channels k and l . If $k = l$, the interference cost should be at its maximum, with the cost decreasing until the two channels are far enough apart so that no interference exists.

The SCA then can be formulated as to minimize the total cost $F(X)$ of all assignments in the network, where $F(X)$ is given below.

$$F(X) = \sum_{j=1}^N \sum_{k=1}^M X_{j,k} \sum_{i=1}^N \sum_{l=1}^M P_{j,i,|k-l|+1} X_{i,l} \quad (4.2)$$

$$\text{subject to} \quad \sum_{k=1}^M X_{j,k} = D_j \quad \forall j = 1, \dots, N \quad (4.3)$$

$$X_{j,k} \in \{0, 1\} \quad \forall j = 1, \dots, N, \text{ and } k = 1, \dots, M. \quad (4.4)$$

The cost tensor P can be generated according to the recursive relation

$$P_{j,i,m+1} = \max(0, P_{j,i,m} - 1) \quad \forall m = 1, \dots, M-1 \quad (4.5)$$

$$P_{j,i,1} = C_{ji} \quad \text{for all } j, i \neq j \quad (4.6)$$

$$P_{j,i,1} = 0 \quad \text{for all } j = i. \quad (4.7)$$

The cost tensor is a cuboid, the front square of which is simply the compatibility matrix C with the diagonal terms overwritten to zero. This is because if $j = i$ and $k = l$, there should be no penalty incurred (there is effectively only a single call and no interference is possible). The third dimension of the tensor decreases the penalty linearly until the penalty becomes zero. Thus, the effective depth of the tensor is equivalent to the value of the maximum diagonal of C . This linear decrease in cost is a means of encouraging less severe violation of the noninterference constraints. If the network demands are such that violation is inevitable, the strength of the resulting signal is preferred to still be as strong as possible. The cost tensor for the example four-cell network is shown in Figure 4.3.

Figure 4.3: Cost tensor for the example four-cell network

4.3 Tabu Search Algorithm

A tabu search algorithm is presented for solving the fixed spectrum problem in this section. The algorithm can be easily modified to apply to the minimum span problem.

Initial Solution: Assigning at random the required number of channels to the corresponding cells produces the initial solution.

Move: A move is a basic flip-flop, which is generated by replacing an in-use channel with an unused one in a cell. The unused channel is generated randomly as long as it does not coincide with the in-use channels in the current cell. A move can be characterized by a triplet $\langle j, k, i \rangle$, which means the pair (j, k) flip-flops to (j, i) . Currently in-use channel k is replaced by unused channel i in cell j .

Candidate List: The conflict-based candidate list is used here. An assignment of channel k to cell j is said to be *conflicting* if such an assignment collides with at least one of other assignments in the current solution S_C . A new solution S_N is in the candidate list of S_C if S_N differs from S_C by exactly one conflicting assignment. And the attribute of the move that produced S_N from S_C must not be in the tabu list or the attribute is in the tabu list but it passes the aspiration check. Here, exactly one new solution is produced for each conflicting assignment. The size of the candidate list (the number of new solutions) is not greater than the total number of channels required in the entire network.

Tabu List: Only short-term memory is implemented in this thesis. The tabu list is a two-dimensional matrix L of dimension $N \times M$. When channel k assigned to cell j flip-flops to a new channel i , the pair (j, i) is classified tabu and the value of element l_{ji} of the tabu list is set to the tabu tenure T . The value of each element of the tabu list decreases by one after an iteration until it becomes zero, and then the corresponding move is set free. At the end of an iteration, the best admissible solution is selected as the new current solution S_{CN} . Suppose S_{CN} was generated from S_C by a flip-flop of (j, k) to (j, i) . The pair (j, i) was put in the tabu list immediately after the ‘flip-flop’. However, its tabu-active status should be removed since, as an attribute of S_{CN} , the counterpart assignment of the new neighbor at the next iteration must be different from it. So it is unnecessary to tabu the pair. Accordingly the pair (j, k) should be classified tabu simply because it occurred recently.

Tabu Tenure: The tenure value is selected to be

$$T = \alpha \frac{M - Dj}{Dj} \quad (4.8)$$

where M is the total number of channels available, D_j is the number of channels required in cell j , as defined previously and α typically takes values from 0.25 to 0.75.

Aspiration Criterion: Aspiration by objective is used in this thesis. The move whose attribute is in the tabu list must be checked against the aspiration criterion to determine whether it is used or discarded. If it passes the check, i.e., it leads to a solution with a cost smaller than the minimum cost obtained so far, the tabu-active status of the move is revoked for the current iteration. Otherwise, the move is discarded and a new random assignment is generated.

Termination Criterion: The search procedure stops when an interference-free solution is found or the preset six-hour time period expires, whichever comes earlier.

4.4 Simulation Results and Discussion

The implemented tabu search algorithm was tested on ten benchmark problems, which are divided into three groups. The first group contains two standard examples [26], the first of which is the small example used in previous sections, denoted by EX1. The second (denoted by EX2) is a slightly larger extension of EX1 with $N = 5$, $M = 17$, $D = [2, 2, 2, 4, 2]$. And the compatibility matrix is given by

$$C = \begin{bmatrix} 5 & 4 & 0 & 0 & 1 \\ 4 & 5 & 0 & 1 & 0 \\ 0 & 0 & 5 & 2 & 1 \\ 0 & 1 & 2 & 5 & 2 \\ 1 & 0 & 1 & 2 & 5 \end{bmatrix}.$$

The second group (denoted by HEX1 through to HEX4) is based on the 21-cell regular hexagonal network used by Sivarajan *et al.* [26]. The details are shown in Table 4.1. Two sets of demands are shown in D_1 and D_2 , and the compatibility matrices for the second group are stored in C_1 and C_2 .

The final set is derived from the topo-graphical data of an actual 24x21 km area around Helsinki, Finland, as used by Kunz [28]. This set is described in Table 4.2. The demand vector is D_3 and the compatibility matrix is C_3 .

$$D_1 = [2, 6, 2, 2, 2, 4, 4, 13, 19, 7, 4, 4, 7, 4, 9, 14, 7, 2, 2, 4, 2]$$

$$D_2 = [1, 1, 1, 2, 3, 6, 7, 6, 10, 10, 11, 5, 7, 6, 4, 4, 7, 5, 5, 5, 6]$$

$$D_3 = [10, 11, 9, 5, 9, 4, 5, 7, 4, 8, 8, 9, 10, 7, 7, 6, 4, 5, 5, 7, 6, 4, 5, 7, 5]$$

Ten runs were performed for each of the ten testing problems. Table 4.3 compares the results obtained in this thesis by using tabu search (TS) with the results given in [27] by using steepest descent (SD), simulated annealing (SA), self-organizing neural network (SONN), Hopfield network without hill climbing (HN) and hill-climbing Hopfield network (HCHN). “Min” represents the minimum interference found during the ten runs, while “Avg” is the average interference value. The smaller the interference values, the better the service of quality. TS obtains better average results for all data sets except EX1 and KUNZ3. In terms of minimum results, TS improves much for the HEX data set. Actual channel assignments for selected problems for which substantial improvements were obtained are shown in Tables 4.4 – 4.7.

Table 4.1 : Problem descriptions for the HEX data set

Problem	N	M	D	Co-channel	Adjacent	C_{ii}
HEX1	21	37	$[D_1]$	Yes	No	2
HEX2	21	91	$[D_1]$	Yes	Yes	3
HEX3	21	21	$[D_2]$	Yes	No	2
HEX4	21	56	$[D_2]$	Yes	Yes	3

Table 4.2: Problem descriptions for the KUNZ data set

Problem	N	M	C	D
KUNZ1	10	30	$[C_3]_{10}$	$[D_3]_{10}$
KUNZ2	15	44	$[C_3]_{15}$	$[D_3]_{15}$
KUNZ3	20	60	$[C_3]_{20}$	$[D_3]_{20}$
KUNZ4	25	73	$[C_3]$	$[D_3]$

Table 4.3: Comparison of TS with SD, SA, SONN, HN, and HCHN

Problem	SD		SA		SONN		HN		HCHN		TS	
	Avg	Min	Avg	Min	Avg	Min	Avg	Min	Avg	Min	Avg	Min
EX1	0.6	0	0.0	0	0.4	0	0.2	0	0.0	0	0	0
EX2	1.1	0	0.1	0	2.4	0	1.8	0	0.8	0	0	0
HEX1	56.8	55	50.7	49	53.0	52	49.0	48	48.7	48	46.6	46
HEX2	28.9	25	20.4	19	28.5	24	21.2	19	19.8	19	14.6	14
HEX3	88.6	84	82.9	79	87.2	84	81.6	79	80.3	78	73.8	73
HEX4	28.2	26	21.0	17	29.1	22	21.6	20	18.9	17	13.0	12
KUNZ1	24.4	22	21.6	21	22.0	21	22.1	21	21.1	20	20	20
KUNZ2	38.1	36	33.2	32	33.4	33	32.8	32	31.5	30	29	29
KUNZ3	17.9	15	13.9	13	14.4	14	13.2	13	13.0	13	13	13
KUNZ4	5.5	3	1.8	1	2.2	1	0.4	0	0.1	0	0	0

[illegible]

Chapter 4

Channel Assignment Problem

[illegible]

[illegible]

Table 4.4: Channel assignment for HEX1 with interference value 46

Cell No.	Demand of Channels	Assigned Channels
1	2	18,26
2	6	4,19,23,26,32,35
3	2	12,30
4	2	20,24
5	2	5,9
6	4	7,11,16,33
7	4	6,14,28,30
8	13	2,5,8,10,13,15,17,20,22,24,32,34,37
9	19	1,3,5,7,9,11,13,15,17,19,21,23,25,27,29,31,33,35,37
10	7	1,6,14,18,25,28,36
11	4	2,22,26,32
12	4	4,12,31,33
13	7	4,18,20,22,24,29,31
14	4	9,19,23,35
15	9	1,3,6,8,10,12,25,34,36
16	14	2,5,7,11,13,16,20,22,24,27,29,31,33,36
17	7	9,14,16,21,23,28,30
18	2	8,34
19	2	18,32
20	4	4,26,30,35
21	2	10,12

Table 4.5: Channel assignment for HEX2 with interference value 14

Cell No.	Demand of Channels	Assigned Channels
1	2	8,60
2	6	12,26,40,47,62,79
3	2	49,82
4	2	14,58
5	2	29,86
6	4	26,31,66,88
7	4	2,14,20,63
8	13	5,18,23,28,33,36,42,53,56,71,74,83,89
9	19	1,7,10,15,19,24,31,35,38,44,51,55,59,65,68,77,81,86,91
10	7	4,22,27,46,63,72,84
11	4	2,11,40,43
12	4	8,26,51,88
13	7	7,13,19,33,53,57,79
14	4	38,41,47,82
15	9	11,16,30,50,58,69,76,80,87
16	14	3,9,13,21,26,39,45,48,61,64,67,73,78,85
17	7	17,29,41,57,70,75,88
18	2	54,90
19	2	32,90
20	4	6,34,43,82
21	2	14,66

Table 4.6: Channel assignment for HEX3 with interference value 73

Cell No.	Demand of Channels	Assigned Channels
1	1	21
2	1	13
3	1	2
4	2	15,17
5	3	8,12,20
6	6	1,3,5,12,16,20
7	7	2,5,7,9,11,14,19
8	6	4,6,8,13,15,17
9	10	1,4,6,8,10,12,14,16,18,20
10	10	1,3,5,7,9,11,14,16,19,21
11	11	1,3,5,7,9,11,13,15,17,19,21
12	5	2,4,6,10,18
13	7	4,6,8,13,15,17,21
14	6	7,10,12,14,16,18
15	4	2,9,11,19
16	4	3,15,17,20
17	7	7,10,12,14,16,18,20
18	5	9,11,13,19,21
19	5	1,3,11,13,21
20	5	2,5,7,12,17
21	6	2,4,6,8,10,15

Table 4.7: Channel assignment for HEX4 with interference value 12

Cell No.	Demand of Channels	Assigned Channels
1	1	16
2	1	43
3	1	37
4	2	13,51
5	3	7,25,45
6	6	10,13,32,37,43,50
7	7	3,6,19,23,26,39,46
8	6	9,21,30,36,49,55
9	10	1,5,12,15,18,27,32,41,47,52
10	10	4,8,20,24,29,34,39,44,50,53
11	11	2,11,16,19,22,28,31,36,40,48,55
12	5	5,9,33,38,52
13	7	2,5,15,24,30,40,45
14	6	8,17,28,48,53,56
15	4	11,14,34,51
16	4	7,25,38,45
17	7	3,10,17,23,35,49,56
18	5	6,14,26,42,46
19	5	2,16,22,31,42
20	5	13,19,28,40,43
21	6	21,30,33,37,51,54

CHAPTER 5

BROADCAST SCHEDULING PROBLEM

Unlike cellular telephone networks that use circuit-switching techniques, packet radio networks (PRNs) combine traditional packet switching and computer networking techniques with algorithms to share broadcast radio channels.

Packet-switching capabilities can be found in many of the radio systems currently deployed, such as the third-generation (3G) mobile systems and wireless local area networks (WLANs). The main reason for using packet switching in these systems is that the typically bursty traffic is more efficiently transferred over packet-switched networks, than over traditional circuit-switched mobile telephony networks.

5.1 Introduction and Literature Review

A packet radio network [37], [38] provides high-speed wireless data communication services to a collection of stations located in a geographical region. Each station is equipped with a transmitter and receiver (transceiver) and a control unit. The transceiver may either transmit outgoing packets or listen to incoming packets. The control unit performs packet-switching operation. Stations communicate with each other through a shared broadcast radio channel. This gives rise to interference or collision where more than one station transmit packets on the common communication medium, which results in damage or loss of packets. There are two kinds of interference, i.e., primary and secondary interference [39]. *Primary interference* occurs when a transceiver is scheduled to perform more than one operation at the same time, for example, transmitting and receiving packets or receiving packets from more than one transmitter. *Secondary interference* occurs when a transmission from a neighboring transmitter unwittingly interferes at the receiving end of a communication between a transmitter and a receiver. Therefore packet transmissions must be scheduled to avoid interference. This leads to the packet transmission scheduling problem in PRNs [40] – [42]. There are two variations of this problem, namely broadcast scheduling problem (BSP) [49] – [56] and link scheduling problem (LSP) [43] – [47]. In a *broadcast schedule*, the entities to be scheduled are the stations. The transmission of a station is intended for, and must be received free of collision by, all of its neighbors. In this scheduling, primary interference is not tolerated and secondary interference does not arise. In a *link schedule*, the links between stations are scheduled, where a link exists if two stations can directly transmit packets to each other. The transmission of a station is

intended for a particular receiver, and there must not be collision at this receiver. Here, neither primary nor secondary interference is tolerated.

Scheduling protocols typically correspond to the medium access control layer in the IEEE 802 reference model. Such channel access protocols can be divided into two types: static and dynamic. In case of a *static* protocol, such as frequency-, time-, and code-division multiple access (FDMA, TDMA, and CDMA), an *a priori* assignment of the communication resources is made among the stations to avoid contention. *Dynamic* protocols such as ALOHA and CSMA do not have *a priori* resource allocation; instead resource requests are met on demand. Static assignment is well suited to a steady stream of traffic while dynamic allocation is well suited to a bursty traffic. In this chapter, the TDMA scheme is adopted and the BSP in TDMA packet radio networks is studied.

TDMA is a synchronous protocol that requires a global clock to which all stations are synchronized. In a TDMA network, time is divided into frames. Each frame consists of a sequence of fixed-length time slots that are assigned to stations to transmit or receive packets. Any stations that may result in collision must be scheduled to transmit at different time slots. There are two types of collision: direct collision and hidden-station collision. This is illustrated in Figure 5.1. In Figure 5.1(a), stations *a* and *b* are within the transmission range of each other. If *a* and *b* start transmissions at the same time, both the transceivers will be expected to transmit and receive simultaneously. Therefore collision occurs between the packets from the two stations, which is referred to as *direct collision*. In Figure 5.1(b), stations *a* and *b* are not within the transmission ranges of one another, but there is a third station *c* that is within the transmission ranges of both *a* and *b*. In this case, if *a*

and b start concurrent transmissions, c will be expected to receive from both a and b at the same time. This is referred to as *hidden-station collision*.

Figure 5.1: Collision – (a) direct collision (b) hidden-station collision

The BSP in TDMA networks can be defined as the scheduling of collision-free transmissions of all the stations so that each station transmits at least once in a frame. It will be repeated in the time axis if such a frame can be found. The frame structure directly influences two main network performance metrics, namely average delay and channel utilization. The low delay can be achieved by minimizing the frame length (the number of time slots in a frame). The channel utilization is directly proportional to the throughput at a fixed frame length. The throughput is defined as the number of simultaneous collision-free transmissions. The higher channel utilization contributes to lower the average delay in the case of the same frame length. This is because stations in the schedule with the higher channel utilization have more transmission chances than those in the schedule with the lower channel utilization. The BSP can be stated as to find the minimum frame length and maximum throughput for a PRN.

The BSP is a combinatorial problem shown to be NP-complete [48], [50], [53]. In [49], Cidon and Sidi proposed two distributed algorithms named round

robin algorithm and wait-for-neighbors algorithm, for maximizing the utilization of each time slot while taking into account the fairness issue. In [50] and [52], a distributed greedy algorithm and an efficient neural network approach, namely, mean field annealing, were applied to maximize the throughput at a given frame length. In [51], Funabiki and Takefuji proposed a centralized parallel algorithm based on neural network to find a collision-free frame whose length is given and each node broadcasts one packet in the frame. In [53], Wang and Ansari formulated the BSP as to find a schedule that had the minimum frame length and maximum throughput and they proposed an algorithm based on mean field annealing. In [54] - [56], the authors followed the formulation in [53] and the algorithms used were based on gradual neural network, sequential vertex coloring, and the mixed Hopfield neural network-genetic algorithm, respectively.

In this chapter, the BSP is divided into two sub-problems. The first attempts to find a collision-free frame with the minimum frame length that allows every station to transmit once and only once. The second is to maximize the throughput at the found frame length. The algorithm proposed has two steps accordingly. For the first step, a tabu search algorithm is used and for the second step, a simple greedy algorithm is applied. Numerical results have shown that the proposed algorithm can find better solutions in terms of throughputs and average delays than the algorithms in [53], [55], [56].

5.2 BSP Formulation

This thesis follows the formulation for the broadcast scheduling problem in [53]. Without loss of generality, some assumptions are first made. It is assumed that a

time slot is equal in length to the transmission time of a packet and all transmissions are synchronized to a single clock. It is also assumed that all transmitters have the same transmission range R . A packet radio network can be described by an undirected graph $G(V, E)$, where each vertex in Y represents a station in the network and each edge in E represents a link between two stations. If the distance between two stations i_1 and i_2 is less than R , there exists a link between them. Therefore, there is an undirected edge $e = (i_1, i_2) \in E$, and the two stations are said to be *one-hop* apart. If stations i_1 and i_2 are not within the transmission ranges of each other, but there is such an intermediate station i_3 that i_3 is within the transmission ranges of both i_1 and i_2 , i.e., $(i_1, i_3) \in E$ and $(i_2, i_3) \in E$, then stations i_1 and i_2 are *two-hop* apart. The topology of the network can be described by an $N \times N$ symmetric binary connectivity matrix C , where $N = |Y|$ is the number of stations in the network. $C = [c_{ji}]$ is defined by

$$c_{ji} = \begin{cases} 1, & \text{if } (j, i) \in E \text{ and } j \neq i \\ 0, & \text{otherwise.} \end{cases} \quad (5.1)$$

Transmissions of packets in the same time slot from stations that are one-hop apart incur direct collision. Transmissions of packets in the same time slot from stations that are two-hop apart induce hidden-station collision. Hence, in order to avoid collision, stations that are one-hop or two-hop apart must be scheduled to transmit in different time slots. Based on the connectivity matrix C and the above constraints, a new $N \times N$ symmetric matrix $F = [f_{ji}]$ called the compatibility matrix can be defined, where

$$f_{ji} = \begin{cases} 1, & \text{if stations } j \text{ and } i \text{ are one-hop or two-hop apart} \\ 0, & \text{otherwise.} \end{cases} \quad (5.2)$$

A feasible solution to the BSP must be free of collision and have at least one transmission for each station in the network. Such a feasible schedule can be expressed by an $N \times M$ binary matrix $S = [s_{jk}]$, where M is the number of time slots in a frame. The jk -th element of S is defined as

$$s_{jk} = \begin{cases} 1, & \text{if station } j \text{ transmits at time slot } k \\ 0, & \text{otherwise.} \end{cases} \quad (5.3)$$

The network throughput τ , then can be calculated by

$$\tau = \sum_{j=1}^N \sum_{k=1}^M s_{jk}. \quad (5.4)$$

The channel utilization ρ is related to the throughput by

$$\rho = \frac{\tau}{NM}. \quad (5.5)$$

Based on the above description, the BSP can be formulated as to find the optimal schedule S^{opt} that

- 1) satisfies the constraints

$$\sum_{k=1}^M s_{jk}^{opt} \geq 1 \quad \forall j = 1, \dots, N \quad (5.6)$$

$$\sum_{k=1}^M \sum_{j=1}^N \sum_{i=1}^N f_{ji} s_{jk}^{opt} s_{ik}^{opt} = 0; \quad (5.7)$$

- 2) achieves the shortest frame length, $M_{\min}$;
- 3) yields the maximum throughput, $\tau_{\max}$.

The first criterion guarantees the feasibility of the schedule. Equation (5.6) forces the schedule to allocate at least one time slot to each station. Equation (5.7) makes all transmissions free of collision.

The BSP formulated this way is a hard problem. The difficulty arises due to the constraint (5.6) that asks for at least one time slot for each station. To eliminate

this difficulty, the BSP is divided into two optimization problems that correspond to the second and third criteria in the BSP formulation.

5.2.1 *P1*: Minimization of the Frame Length M

First a new optimization problem *P1* is defined, whose objective is to find a collision-free schedule with one and only one transmission for each station. *P1* can be formulated as to find

$$S^{one} = [s_{jk}] \quad (5.8)$$

so that

$$\sum_{k=1}^M s_{jk}^{one} = 1 \quad \forall j = 1, \dots, N \quad (5.9)$$

$$\sum_{k=1}^M \sum_{j=1}^N \sum_{i=1}^N f_{ji} s_{jk}^{one} s_{ik}^{one} = 0. \quad (5.10)$$

If a feasible solution to *P1* can be found for a certain value of M , M is decreased by one and a new feasible solution to *P1* is searched for. This continues until either the search is not successful or the lower bound for M is reached. The initial value of M can be guessed from the lower bound for the frame length that is discussed in [53], [57]. In this chapter, the lower bound in [57] is adopted, which is tighter.

5.2.2 *P2*: Maximization of the Network Throughput τ

The problem of maximizing the throughput τ at the minimum frame length is referred to as *P2*. The formulation for *P2* is as follows:

$$\max \tau = \max \sum_{j=1}^N \sum_{k=1}^M s_{jk} \quad (5.11)$$

$$\text{subject to} \quad \sum_{k=1}^M s_{jk} \geq 1 \quad \forall j = 1, \dots, N \quad (5.12)$$

$$\sum_{k=1}^M \sum_{j=1}^N \sum_{i=1}^N f_{ji} s_{jk} s_{ik} = 0. \quad (5.13)$$

5.3 Two-step Algorithm

A two-step algorithm is proposed to solve the BSP in this section. The algorithm proceeds as follows. An initial value of the frame length is estimated from the lower bound. At the given frame length, a tabu search (TS) algorithm is used to solve *PI*. If a feasible solution to *PI* can be found, a shorter frame length is tried. This process continues until the lower bound is reached or the search for a feasible solution is not successful. After the schedule with the minimum frame length is obtained, a greedy (GR) algorithm for *P2* is used to maximize the throughput.

5.3.1 TS for *PI*

Suitable values or forms are selected for key components for the implementation of the tabu search algorithm. Initially, $S = [s_{jk}] = 0$ ($j = 1, \dots, N$ and $k = 1, \dots, M$). The cost function for a certain schedule S is defined as

$$\text{cost}(S) = \frac{\sum_{k=1}^M \sum_{j=1}^N \sum_{i=1}^N f_{ji} s_{jk} s_{ik}}{2}. \quad (5.14)$$

Division by 2 reflects the symmetry of the network topology and simplifies the computation.

Initial Solution: Assigning at random one time slot to each station produces the initial solution.

Move: A move is basic flip-flop that is generated by replacing an in-use time slot by an arbitrary unused one for a station. A move is characterized by a triplet $\langle j, k, i \rangle$, which means the pair (j, k) flip-flops to (j, i) . Station j will transmit at time slot i instead of k .

Candidate List: The conflict-based candidate list is used here. An assignment of time slot k to station j is said to be *conflicting* if such an assignment collides with at least one of other assignments in the current schedule S_C . A new solution S_N is in the candidate list of S_C if S_N differs from S_C by exactly one conflicting assignment. And the attribute of the move that produced S_N from S_C must not be in the tabu list or the attribute is in the tabu list but it passes the aspiration check. In this chapter, exactly one new solution is produced for each conflicting assignment. Since there is only one assignment to each station at this stage, the size of the candidate list (the number of new solutions) is not greater than the number of stations N .

Tabu List: Only short-term memory is implemented in this thesis. The tabu list is a two-dimensional matrix L of dimension $N \times M$. When time slot k assigned to station j flip-flops to a new time slot i , the pair (j, i) is classified tabu and the value of element l_{ji} of the tabu list is set to the tabu tenure T . The value of each element of the tabu list decreases by one after an iteration until it becomes zero and then the corresponding move is set free. At the end of an iteration, the best admissible solution is selected as the new current solution S_{CN} . Suppose S_{CN} was generated from S_C by a flip-flop of (j, k) to (j, i) . The pair (j, i) was put in the tabu list immediately after the flip-flop. However, its tabu-active status should be removed since, as an attribute of S_{CN} , the counterpart assignment of the new

neighbor at the next iteration must be different from it and it is unnecessary to tabu the pair. Accordingly the pair (j, k) should be classified tabu simply because it occurred recently.

Tabu Tenure: The tenure value is chosen to be

$$T = \alpha (M - 1) \quad (5.15)$$

where M is the number of time slots in a frame, as defined previously and α typically takes values from 0.25 to 0.75. So T varies with α .

Aspiration Criterion: Aspiration by objective is used in this thesis. The move whose attribute is in the tabu list must be checked against the aspiration criterion to determine whether it is used or discarded. If it passes the check, i.e., it leads to a solution with a smaller cost than the minimum cost obtained so far, the tabu-active status of the move is revoked for the current iteration. Otherwise, the move is discarded and a new random assignment is generated.

Termination Criterion: The search procedure stops when a collision-free schedule is found or the number of iterations performed exceeds $N \times M$, whichever comes earlier.

5.3.2 GR for P2

The throughput is maximized by adding as many conflict-free assignments as possible to the schedule obtained in the first step. An assignment of an available time slot to a station is added at one time. If it does not cause any collision, the added assignment is preserved; otherwise it is dropped and the next available time slot is tried. This process repeats until all available time slots are attempted to be assigned to all stations.

Here a proper order needs to be determined in which stations are assigned the available time slots. Assignment of a time slot to a station makes this slot unavailable to all of its one-hop and two-hop neighbors. For this reason, it is desirable to assign the available time slots first to the station that has less neighboring stations. Therefore stations are arranged in ascending order by the number of one-hop and two-hop neighbors [55].

5.4 Computation Results and Discussion

5.4.1 Performance Metrics

Resulting schedules can be evaluated by two performance metrics: channel utilization and average delay. Since the channel utilization is closely related to the throughput, the throughput is calculated and evaluated by Equation (5.4), for simplicity. To calculate the average delay, it is assumed that the network can be approximately modeled as $N \times M / D / I$ queues, where N is the number of stations. The detailed assumptions are as follows.

- 1) Packets have a fixed length, and the length of a time slot equals the time required to transmit a packet.
- 2) The inter-arrival time for each station j is statistically independent from other stations, and packets arrive according to a *Poisson* process with a rate of λ_j packets/slot. The total traffic in station j consists of the traffic incoming from other stations and the data from terminals attached to it. Packets are stored in buffers in each station and the buffer size is infinite.

- 3) The probability distribution of the service time of station j is deterministic and statically independent from other stations. The average service rate is μ_j packets/slot.
- 4) Packets can be transmitted only at the beginning of a time slot.

The average delay for each queue can be calculated by the *Pollaczek-Khinchin* formula [58] as follows:

$$D_j = \frac{1}{\mu_j} + \frac{\lambda_j}{2\mu_j(\mu_j - \lambda_j)}, \quad (5.16)$$

where λ_j is the average arrival rate and μ_j is the average service rate. An equal arrival rate for all queues is assumed. Hence,

$$\lambda_j = \frac{\lambda}{N} (\text{packets/slot}), \quad (5.17)$$

where λ is the total arrival rate. The average service rate for each queue is given by

$$\mu_j = \frac{\sum_{i=1}^M s_{ji}}{M} (\text{packets/slot}). \quad (5.18)$$

The total delay can be calculated by

$$D = \frac{\sum_{j=1}^N D_j}{N} (\text{slots}). \quad (5.19)$$

5.4.2 Numerical Results and Discussion

Three benchmark BSPs are scheduled, whose topologies are shown in Figures 5.2 – 5.4. The lower bounds for the frame lengths are 8, 10, and 8, respectively.

Table 5.1 compares the frame lengths and the throughputs obtained in this thesis by using the proposed algorithm (TS-GR) with those obtained by using mean field annealing (MFA) [53], sequential vertex coloring (SVC) [55], and the mixed

neural-genetic algorithm (HNN-GA) [56]. “NR” means that the corresponding frame length is not reachable. “NA” means the corresponding frame length can be obtained but the throughput at that frame length is not available. The proposed algorithm has produced higher throughputs for the 30-station network, which was claimed to be the most difficult among the three networks in [56]. Note that the minimum frame lengths for each case are underlined in Column M .

Table 5.1: Throughputs by MFA, SVC, HNN-GA, and TS-GR

N	M	MFA	SVC	HNN-GA	TS-GR
15	<u>8</u>	18	18	20	20
30	<u>10</u>	NR	NR	35	37
30	11	NR	37	40	43
30	12	39	NA	47	48
40	<u>8</u>	NR	60	67	68
40	9	71	NA	77	77

The optimal schedules obtained by the proposed algorithm are given in Figures 5.5 – 5.7 for the 15-, 30-, and 40-station networks, respectively. “ST” stands for the station number and “SL” stands for the time slot number. The frame lengths are optimal as can be verified from the lower bounds.

The comparisons of the average delays obtained from MFA, SVC, HNN-GA and the proposed TS-GR for the 15-, 30- and 40-station networks are shown in Figures 5.8 – 5.10. The frame length essentially determines the average delay. This can be verified from the wide gaps between the MFA and SVC curves, SVC and HNN-GA curves in Figure 5.9 and the MFA and SVC curves in Figure 5.10. The higher throughput contributes to lower the average delay slightly at the same frame length. This can be verified by the relatively narrow gaps in the three figures. In

Figure 5.8, the MFA and SVC curves almost coincide with one another since they have the same throughput and so do the HNN-GA and TS-GR curves.

Figure 5.2: The 15-station radio network used in the simulation

Figure 5.3: The 30-station radio network used in the simulation

neural-genetic algorithm (HNN-GA) [56]. “NR” means that the corresponding frame length is not reachable. “NA” means the corresponding frame length can be obtained but the throughput at that frame length is not available. The proposed algorithm has produced higher throughputs for the 30-station network, which was claimed to be the most difficult among the three networks in [56]. Note that the minimum frame lengths for each case are underlined in Column M .

Table 5.1: Throughputs by MFA, SVC, HNN-GA, and TS-GR

N	M	MFA	SVC	HNN-GA	TS-GR
15	<u>8</u>	18	18	20	20
30	<u>10</u>	NR	NR	35	37
30	11	NR	37	40	43
30	12	39	NA	47	48
40	<u>8</u>	NR	60	67	68
40	9	71	NA	77	77

The optimal schedules obtained by the proposed algorithm are given in Figures 5.5 – 5.7 for the 15-, 30-, and 40-station networks, respectively. “ST” stands for the station number and “SL” stands for the time slot number. The frame lengths are optimal as can be verified from the lower bounds.

The comparisons of the average delays obtained from MFA, SVC, HNN-GA and the proposed TS-GR for the 15-, 30- and 40-station networks are shown in Figures 5.8 – 5.10. The frame length essentially determines the average delay. This can be verified from the wide gaps between the MFA and SVC curves, SVC and HNN-GA curves in Figure 5.9 and the MFA and SVC curves in Figure 5.10. The higher throughput contributes to lower the average delay slightly at the same frame length. This can be verified by the relatively narrow gaps in the three figures. In

Figure 5.7: Broadcast schedule obtained for the 40-station network

Figure 5.8: Comparison of average delays by MFA, SVC, HNN-GA, and TS-GR for the 15-station network

Figure 5.9: Comparison of average delays by MFA, SVC, HNN-GA, and TS-GR for the 30-station network

Figure 5.10: Comparison of average delays by MFA, SVC, HNN-GA, and TS-GR for the 40-station network

CHAPTER 6

CONCLUSION AND RECOMMENDATIONS

In this final chapter, the thesis is concluded. The research work that has been carried out is summarized. In addition, recommendations for future research arising from the work that has been done are proposed and discussed.

6.1 Conclusion

This research work aimed to investigate and solve resource allocation problems in the design and optimization of communication networks. Three such problems have been studied: routing and wavelength assignment (RWA) problem in wavelength-routed optical networks, channel assignment problem (CAP) in cellular telephone networks, and broadcast scheduling problem (BSP) in TDMA packet radio networks. The algorithms based on the tabu search (TS) paradigm have been proposed for solving these combinatorial optimization problems.

The RWA problem was first solved by a simple greedy algorithm. Then, a mixed tabu-greedy algorithm, which tried different sets of routes iteratively using tabu search and each set of routes was assigned wavelengths greedily, was proposed for RWA. The number of wavelengths was used as a measure of how good the assignments were.

Next, the CAP was solved by a standard tabu search algorithm. Conflict-based candidate list and dynamic tabu tenure were used. The candidate list determined this way contained the new solutions that had been produced by “influential” moves. Dynamic tabu tenure provided one way to induce a balance between closely examining one region and moving to different parts of the solution space.

The BSP was divided into two sub-problems, i.e., minimization of the frame length and maximization of the throughput. The proposed mixed tabu-greedy algorithm had two steps accordingly. The first step, based on tabu search, found a frame with the minimum length that transmitted one and only one packet for each station. The second step maximized the throughput by adding greedily as many conflict-free assignments as possible to the schedule obtained in the first step.

Finally, the proposed algorithms have been evaluated by comparing their performance with some existing algorithms. The proposed algorithms have always produced better results in terms of resources used. Tabu search has proven to be an extremely promising heuristic for combinatorial optimization problems.

6.2 Recommendations for Future Research

Possible improvements and directions for future research work, related to the ideas presented and investigated in this thesis, are given in this section.

6.2.1 Wavelength Reassignment

Tabu search has proven to work well with wavelength assignment [14]. Best results will be achieved if TS is used to select routes and assign wavelengths. However, this is not practically feasible due to the extremely long execution time. In this thesis, a greedy algorithm employing the largest-number-of-neighbors-first and first-fit schemes was used for wavelength assignment. A simple greedy algorithm (denoted as GR-RWA), which introduced a new wavelength immediately after a connection request had failed to be routed on the current one, was also proposed for solving RWA. A future research topic could be to reassign wavelengths to the routes selected by GR-RWA, using tabu search.

6.2.2 Base Station Location (BSL)

Another challenging problem that arises when designing and setting up a new cellular network is the optimal location of base stations in the service area. The

objectives are to maximize the coverage area while minimizing the equipment cost. This problem can be formulated as a combinatorial optimization problem and can be solved by tabu search.

6.2.3 Joint Time Slot/Code Assignment (JTSCA)

As introduced in Chapter 5, the other packet transmission scheduling problem is the link scheduling problem. With the time/code division multiple access method adopted for collision-free communications, the link scheduling problem can be defined as to find a simultaneous assignment of a time slot and a code to each communication link, which is also referred to as the joint time slot/code assignment problem. The objective is to minimize the number of time slots subject to two constraints: 1) the number of codes must not exceed its upper limit and 2) any couple of links within the conflict distance must not be assigned to the same time slot/code pair.

6.2.4 Evolutionary Multi-Objective Optimization (EMO)

The three resource allocation problems dealt with in this thesis are all of single objective. And tabu search proved to be ideal for such problems. However, some resource allocation problems in communication networks are of multi-objective. One of the examples is the base station location problem in cellular telephone networks. Such problems can be converted into single-objective optimization problems and solved by tabu search. More naturally, they can be formulated as multi-objective optimization problems and solved by evolutionary multi-objective optimization (EMO) [59], which is a possible future alternative approach.

Author's Publications

1. YPeng, B.H. Soong, L. Wang, "Broadcast scheduling in packet radio networks using mixed tabu-greedy algorithm," *IEE Electron. Lett.*, vol. 40, no. 6, pp. 375 – 376, Mar. 2004.
2. YPeng, L. Wang, B.H. Soong, "Optimal channel assignment in cellular systems using tabu search," in *Proc. 14th IEEE PIMRC*, Sep. 2003, vol. 1, pp. 31 – 35.

Bibliography

- [1] <http://www.c-i-a.com/pr032102.htm>
- [2] R. Ramaswami and K. N. Sivarajan, *Optical Networks: A Practical Perspective*, 2nd ed. San Fransisco: Morgan Kaufmann Publishers, 2002.
- [3] http://english.peopledaily.com.cn/200311/10/eng20031110_127968.shtml
- [4] F. Glover, "Future paths for integer programming and links to artificial intelligence," *Comp. and Oper. Res.*, vol. 13, pp. 533 – 549, 1986.
- [5] P. Hansen, "The steepest ascent and mildest descent heuristic for combinational programming," *Congress on Numerical Methods in Combinational Optimization*, Capri, Italy, 1986.
- [6] F. Glover, "Tabu search – part I", *ORSA Journal on Computing*, vol. 1, no. 3, pp. 190 – 206, 1989.
- [7] F. Glover, "Tabu search – part II", *ORSA Journal on Computing*, vol. 2, no. 1, pp. 4 – 32, 1990.
- [8] F. Glover and M. Laguna, *Tabu Search*. Boston: Kluwer Academic Publishers, 1997.
- [9] X. Jia, X. Hu and D. Du, *Multiwavelength optical networks*. The Netherlands: Kluwer Academic Publishers, 2002.
- [10] B. Mukherjee, *Optical Communication Networks*. New York: McGraw-Hill, 1997.
- [11] R. Ramaswami and K. N. Sivarajan, "Routing and wavelength assignment in all-optical networks," *IEEE/ACM Trans. Networking*, vol. 3, no. 5, pp. 489 – 500, Oct. 1995.

-
- [12] C. Chen and S. Banerjee, "A new model for optimal routing and wavelength assignment in large wavelength-routed optical networks," in INFOCOM'96, 1996, vol. 1, pp. 164 – 171.
- [13] Z. Zhang and A. S. Acampora, "A heuristic wavelength assignment algorithm for multi-hop WDM networks with wavelength routing and wavelength reuse," *IEEE/ACM Trans. Networking*, vol. 3, no. 3, pp. 281 – 288, 1995.
- [14] E. Hyytia and J. Virtamo, "Wavelength assignment and routing in WDM networks," in *Nordic Teletraffic Seminar, (NTS) 14*, Copenhagen, Denmark, Aug. 1998.
- [15] M. Alanyali and E. Ayanoglu, "Provisioning algorithms for WDM optical networks," in INFOCOM98, 1998, vol. 2, pp. 910 – 918.
- [16] R. Sabella, E. Iannone, M. Listanti, M. Berdusco and S. Binetti, "Impact of transmission performance on path routing in all-optical transport networks," *IEEE Journal of Lightwave Technology*, vol. 16, no. 11, pp. 1965 – 1972, 1998.
- [17] S. Baroni and P. Bayvel, "Wavelength requirements in arbitrarily connected wavelength-routed optical networks," *IEEE Journal of Lightwave Technology*, vol. 15, no. 2, pp. 242 – 251, 1997.
- [18] I. Chlamtac, A. Ganz and G. Karmi, "Lightnet: lightpath based solutions for wide bandwidth WANs," in INFOCOM'90, 1990, vol. 3, pp. 1014– 1021.
- [19] G. Jeong and E. Ayanoglu, "Comparison of wavelength-interchanging and wavelength-selective cross-connects in multiwavelength all-optical networks," in INFOCOM'96, 1996, vol. 1, pp. 156 – 163.

- [20] A. Hertz and D. de Werra, "Using Tabu Search Techniques for Graph Coloring," *Computing*, vol. 39, pp. 345 – 351, 1987.
- [21] P. Manohar, D. Manjunath, and R. K. Shevgaonkar, "Routing and wavelength assignment in optical networks for edge-disjoint path algorithms," *IEEE Commun. Lett.*, vol. 6, no. 5, pp. 211 – 213, May 2002.
- [22] J. M. Kleinberg, "Approximation algorithms for disjoint paths problems," Ph.D. dissertation, MIT, Cambridge, May 1996.
- [23] B. H. Metzger, "Spectrum management technique," presented at the 38th National ORSA meeting, Detroit, Michigan, USA, Fall 1970.
- [24] A. Eisenblatter, "Frequency assignment in GSM networks: models, heuristics, and lower bound," Ph.D. thesis, Technische Universität Berlin, Berlin, Germany, 2001.
- [25] J. A. Zoellner and C. L. Beall, "A breakthrough in spectrum conserving frequency assignment technology," *IEEE Trans. Electromagn. Compat.*, vol. 19, pp. 313 – 319, 1977.
- [26] K. N. Sivarajan and R. J. McEliece, "Channel assignment in cellular radio," in *Proc. IEEE Veh. Technol. Conf.*, 1989, pp. 846 – 850.
- [27] R. Smith and M. Palaniswami, "Static and dynamic channel assignment using neural networks," *IEEE J. Select. Areas Commun.*, vol. 15, no. 2, pp. 238 – 249, Feb. 1997.
- [28] D. Kunz, "Channel assignment for cellular radio using neural networks," *IEEE Trans. Veh. Technol.*, vol. 40, no. 1, pp. 188 – 193, 1991.

- [29] 3. S. Kim, S. Park, P. Dowd and N. Nasrabadi, "Cellular radio channel assignment using a modified Hopfield network," *IEEE Trans. Veh. Technol.*, vol. 46, no. 4, pp. 957 – 967, 1996.
- [30] A. Hanamitsu and M. Ohta, "A maximum neural network with self-feedbacks for channel assignment in cellular mobile systems," In *Proc. Int. Joint Conf. Neural Networks*, 2002, vol. 3, pp. 2814 – 2818.
- [31] D. Beckmann and U. Killat, "A new strategy for the application of genetic algorithms to the channel-assignment problem," *IEEE Trans. Veh. Technol.*, vol. 48, no. 4, pp. 1261 – 1269, Jul. 1999.
- [32] C. Y. Ngo and V. O. K. Li, "Fixed channel assignment in cellular radio networks using a modified genetic algorithm," *IEEE Trans. Veh. Technol.*, vol. 47, no. 1, pp. 163 – 171, 1998.
- [33] M. Duque-Antón, D. Kunz and B. Rüber, "Channel assignment for cellular radio using simulated annealing," *IEEE Trans. Veh. Technol.*, vol. 42, pp. 14 – 21, 1993.
- [34] A. Dawood, M. M. OO, K. Ahmed and N. Rajatheva, "Efficient channel assignment algorithm under cochannel and adjacent channel interference constraint," in *Proc. IEEE Int. Conf. Pers. Wireless Commun.*, 2000, pp. 424 – 428.
- [35] D. H. Smith, R. K. Taplin and S. Hurley, "Frequency assignment with complex co-site constraints," *IEEE Trans. Electromagn. Compat.*, vol. 43, no. 2, pp. 210 – 218, May 2001.

- [36] A. Capone and M. Trubian, "Channel assignment problem in cellular systems: a new model and a tabu search algorithm," *IEEE Trans. Veh. Technol.*, vol. 48, no. 4, **pp.** 1252 – 1260, Jul. 1999.
- [37] R. E. Kahn, S. A. Gronemeyer, J. Burchfiel, and R. C. Kunzelman, "Advances in packet radio technology," in *Proc. IEEE*, Nov. 1978, vol. 66, no. 11, pp. 1468 – 1496.
- [38] "Special issue on packet radio networks," in *Proc. IEEE*, Jan. 1987, vol. 75, pp. 6 – 167.
- [39] A. Ephremides, J. E. Wieselthier, and D. J. Baker, "A design concept for reliable mobile radio networks with frequency hopping signaling," in *Proc. IEEE*, Jan. 1987, vol. 75, no. 1, **pp.** 56 – 73.
- [40] D. J. Baker and A. Ephremides, "The architectural organization of a mobile radio network via a distributed algorithm," *IEEE. Trans. Commun.*, vol. COM-29, **pp.** 1694– 1701, Nov. 1981.
- [41] D. J. Baker, J. Wieselthier, and A. Ephremides, "A distributed algorithm for scheduling the activation of links in a self-organizing mobile radio network," in *Proc. Int. Con. Commun.*, Philadelphia, PA, Jun. 1982, pp. 2F.6.1 – 2F.6.5.
- [42] D. J. Baker *et al.*, "The design and simulation of a mobile radio network with distributed control," *IEEE J. Select. Areas Commun.*, vol. SAC-2, **pp.** 226 – 237, Jan. 1984.
- [43] E. Arikan, "Some complexity results about packet radio networks," *IEEE Trans. Inf. Theory*, vol. IT-30, no. 4, pp. 681 – 685, Jul. 1984.

- [44] R. Nelson and L. Kleinrock, "Spatial TDMA: a collision-free multi-hop channel access protocol," *IEEE Trans. Commun.*, vol. COM-33, no. 9, pp. 934 – 944, Sep. 1985.
- [45] R. G. Ogier, "A decomposition method for optimal link scheduling," in *Proc. 24th Allerton Conf.*, Monticello, IL, Oct. 1986, pp. 822 – 823.
- [46] I. Chlamtac and A. Lerner, "Fair algorithms for maximal link activation in multi-hop radio networks," *IEEE Trans. Commun.*, vol. COM-35, no. 7, pp. 739 – 746, Jul. 1987.
- [47] B. Hajek and G. Sasaki, "Link scheduling in polynomial time," *IEEE Trans. Inf. Theory*, vol. 34, no. 5, pp. 910 – 917, Sep. 1988.
- [48] S. Even *et al.*, "On the NP-completeness of certain network testing problems," *Networks*, vol. 14, pp. 1 – 24, 1984.
- [49] I. Cidon and M. Sidi, "Distributed assignment algorithms for multi-hop packet radio networks," *IEEE Trans. Comput.*, vol. 38, no. 10, pp. 1353 – 1361, Oct. 1989.
- [50] A. Ephremides and T. V. Truong, "Scheduling broadcasts in multi-hop radio networks," *IEEE Trans. Commun.*, vol. 38, no. 4, pp. 456 – 460, Apr. 1990.
- [51] Funabiki and Y. Takefuji, "A parallel algorithm for broadcast scheduling problems in packet radio networks," *IEEE Trans. Commun.*, vol. 41, no. 6, pp. 828 – 831, Jun. 1993.
- [52] G. Wang and N. Ansari, "A neural network approach to broadcast scheduling in multi-hop radio networks," in *Proc. ICNN'94*, Jun. 1994, pp. 4699 – 4704.

- [53] G. Wang and N. Ansari, "Optimal broadcast scheduling in packet radio networks using mean field annealing," *IEEE J. Select. Areas Commun.*, vol. 15, pp. 250 – 259, Feb. 1997.
- [54] N. Funabiki and J. Kitamichi, "A gradual neural network algorithm for broadcast scheduling problems in packet radio networks," *IEICE Trans. Fundamentals*, vol. E82-A, no. 5, pp. 815 – 824, May 1999.
- [55] J. Yeo, H. Lee, and S. Kim, "An efficient broadcast scheduling algorithm for TDMA ad-hoc networks," *Computers & Operations Research* 29, pp. 1793 – 1806, 2002.
- [56] S. Salcedo-Sanz, C. Bousoño-Calzón, and A. R. Figueiras-Vidal, "A mixed neural-genetic algorithm for the broadcast scheduling problem," *IEEE Trans. Wireless Commun.*, vol. 2, no. 2, Mar. 2003.
- [57] D. Jungnickel, *Graphs, Networks and Algorithms*. Berlin, Germany: Springer-Verlag, 1999.
- [58] D. Bertsekas and R. Gallager, *Data Networks*, 2nd ed. Englewood Cliffs, NJ: Prentice-Hall, 1992.
- [59] K. Deb, *Multi-Objective Optimization using Evolutionary Algorithms*. Chichester, UK: John Wiley & Sons, 2001.