

國立交通大學

資訊工程學系

博士論文

WDM 網路中多點傳輸與波長指派問題研究



**Multicast Routing and Wavelength Assignment in
WDM Networks**

研究生：陳明崇

指導教授：曾憲雄 博士

中華民國九十五年十一月


WDM 網路中多點傳輸與波長指派問題研究

**Multicast Routing and Wavelength Assignment in
WDM Networks**

研究生: 陳明崇
指導教授: 曾憲雄

Student: Ming-Tsung Chen
Advisor: Dr. Shian-Shyong Tseng

國立交通大學
資訊工程學系
博士論文



A Dissertation
Submitted to Department of Computer Science
College of Computer Science
National Chiao Tung University
in partial Fulfillment of the Requirements
for the Degree of
Doctor of Philosophy
in

Computer Science

November 2006

Hsinchu, Taiwan, Republic of China

中華民國九十五年十一月

WDM 網路中多點傳輸與波長指派問題研究

學生：陳明崇

指導教授：曾憲雄

國立交通大學 資訊工程學系

摘要

在可見的未來，WDM 網路將被用來建置主幹網路，因此，建置多點傳輸功能，以提供多變的網路應用需求是必要的。在本篇論文中，我們擴展路由與波長指派問題（routing and wavelength assignment problem, RWA）的研究，重新定義在 WDM 網路具傳輸延遲限制的多點傳輸與波長指派的新問題，簡稱 MRWAP-DC。在此問題中，多點傳輸需求具傳輸延遲限制且網路節點具不同的分光能力或波長轉換能力，多點傳輸代價定義為傳輸代價與所需光波長數的線性加權，其目的為對每一個需求尋找一個光樹狀傳輸路由集合（light-forest），在最小的多點傳輸代價下，使得這些多點傳輸需求可以在傳輸延遲限制內，成功的將資料傳輸至所有的目的節點。在本篇論文中，MRWAP-DC 將被完整定義與描述，提出一個整數線性規劃（ILP）方法，使得 MRWAP-DC 問題可被轉換成條件定義最佳化問題，利用 CPLEX 線性規劃工具以找出問題的最佳解。雖然 ILP 方法可用來找出滿足條件的最佳解，但只適合解決小規模的網路路由與波長指派問題，因此，我們提出兩個啟發式演算法（meta-heuristic）：螞蟻演算法（Ant Colony Optimization）與基因遺傳演算法（Genetic Algorithm）以解決兩個簡化的問題—URWAP-DC-SR 和 MRP-DC-WWC-SR。此外，針對動態網路路由與波長指派問題，執行時間為重要的考慮因素，因此，我們提出兩個直覺演算法： k -最短路徑近似演算法（Near- k -Shortest-Path-based Heuristic, NKSPH）與反覆尋解模型（Iterative Solution Model, ISM），以處理大規模網路的動態路由與波長指派問題。由實驗的數據結果，這兩個直覺演算法可以找到接近最佳解的近似解。

在本篇論文中，不僅利用 ILP 方法來定義 MRWAP-DC 問題，同時提出四種不同

解決這類簡化問題的方法，透過利用 ILP 方法所得到的最佳解比較，這些方法可以找到接近最佳解的近似解，但花費的執行時間卻遠比 ILP 方法少。

關鍵詞: RWA、ACO、GA、WDM、多重傳輸路由、波長指派、多點傳輸需求、分頻多工光纖網路、傳輸延遲限制、光樹狀傳輸路由、分光能力、整數線規劃、遺傳演算法、螞蟻演算法。



Multicast Routing and Wavelength Assignment in WDM Networks

Student: Ming-Tsung Chen

Advisor: Dr. Shian-Shyong Tseng

Department of Computer Science
National Chiao Tung University

Abstract

Because optical wavelength division multiplexing (WDM) networks are expected to be realized for building up backbone in the near future, multicasting in WDM networks needs to be addressed for various network applications. This dissertation studies an extended routing and wavelength assignment (RWA) problem called multicast routing and wavelength assignment problem with delay constraint (MRWAP-DC) that incorporates delay constraints in WDM networks having heterogeneous light splitting capabilities. The objective is to find a light-forest for a multicast whose multicast cost, defined as a weighted combination of communication cost and wavelength consumption, is minimal. An integer linear programming (ILP) formulation is proposed to formulate and solve the special problem of MRWAP-DC, MRWAP-DC-WWC. Experimental results show that using CPLEX to solve the ILP formulation can optimally deal with small-scale networks. Therefore, we develop two heuristics, Near- k -Shortest-Path-based Heuristic (NKSPH) and Iterative Solution Model (ISM), to solve the problem in large-scale networks. Numerical results indicate that the proposed heuristic algorithms can produce approximate solutions of good quality in an acceptable time. This dissertation also investigates two special problems, URWAP-DC-SR and MRP-DC-WWC-SR by two meta-heuristics ant colony optimization (ACO) and genetic algorithm (GA). We compare the performance of the proposed algorithms with the ILP

formulations. Solutions found by these meta-heuristics are approximately equal to those found by the ILP formulations, and the elapsed execution times are far less than that demanded by the ILP formulations.

Keywords: Multicast routing, wavelength assignment, multicast request, WDM network, delay bound, light-tree, light splitting capacity, integer linear program, genetic algorithm, ant colony optimization.



誌 謝

首先感謝恩師曾憲雄教授多年來的指導與鼓勵，沒有曾教授的指導，本論文將無法完成；在曾教授不厭其煩的指導與鼓勵，引導與訓練獨立研究的方法及技巧，不僅完成本論文，同時也讓我瞭解到研究的本質與程序。此外，要感謝許多人的協助、體諒與鼓勵，本校財務金融研究所林妙聰教授、資工系陳榮傑教授與簡榮宏教授的建議，使得本篇論文的理論與實務內容，更加趨於完整與堅實。其中，特別感謝林妙聰教授給予的鼓勵和指導，啟發我能從其他觀點去檢閱自己研究上的盲點和不足，不僅對論文結構提供建議，更進而協助完成 ACO 相關論文。同時也要感謝論文口試委員清華大學資工系黃能富教授、台灣大學電機系王勝德教授、高雄大學電機系洪宗貝教授、中央研究院資訊所陳孟彰教授等提供的精闢見解與建議，才能讓本篇論文更加完整。成功大學統計系趙昌泰副教授提供 CPLEX 線性規劃工具，使得本論文中相關實驗得以順利完成。同實驗室的學長、同學、學弟妹、與助理的幫忙，使得本篇論文實際的研究探索過程，能夠順利地進行與論文完成。而中華電信研究所的長官與同事長期的支持與包容，亦是不可或缺的精神支柱。

進入交通大學資科所攻讀博士學位時，正是開發公司內重要資訊系統的關鍵時期，每天焚膏繼晷的不停工作與念書，一切仍歷歷在目、彷彿昨日，當時小朋友才就讀幼稚園，如今已是國中一年級的新生了，非常感謝家人的鼓勵與支持，由其是我的太太淑英，在我攻讀博士學位與工作的雙重壓力之下，若沒有她的任勞任怨、無怨無悔的奉獻付出，我將無法心無旁騖的專心於博士論文的研究，將眼前遭遇的困難轉化為克服困境的動力，衝破難關，繼續堅持到底而完成學業。

僅將這一篇論文，獻給老師、實驗室的學長與學弟妹、口試委員、以及最親愛的父母、家人淑英、宣劭、芃穎，與每一位給予我幫助及支持我的人。

Contents

List of Figures	iv
List of Tables	vi
Chapter 1 Introduction	1
1.1 Motivation	1
1.2 Contribution	3
1.3 Reader's Guide	4
Chapter 2 Overview of Optical Networks.....	8
2.1 Characteristics of Optical Networks	8
2.2 Optical Multiplexing Transmission System	11
2.3 Optical Equipment.....	13
2.4 Evolution of Optical Networks	16
2.5 Multicast.....	19
Chapter 3 Preliminaries of Routing and Wavelength Assignment Problem.....	23
3.1 Routing and Wavelength Assignment Problem (RWAP).....	23
3.2 Unicast RWAP (URWAP).....	25
3.3 Multicast RWAP (MRWAP)	27
3.4 RWAP Problem Model.....	29
Chapter 4 Multicast Routing and Wavelength Assignment Problem with Delay	
Constraints	31
4.1 Formulation of MRWAP-DC	32
4.2 Studied Problems and Methods	42
4.3 Simulation Scheme	44
Chapter 5 ILP Formation for MRWAP-DC-WWC	46
5.1 Formulation	47
5.2 Proof of the ILP Formulation	51
5.3 Experiments.....	55
5.3.1 Comparisons of Different Wavelength Consumption Ratios	56
5.3.2 Performance Assessment of ILP	57
5.4 Conclusion.....	58
Chapter 6 Ant Colony Optimization (ACO) for URWAP-DC-SR.....	61

6.1 Concept of the ACO	64
6.2 Initialization in the ACO	66
6.3 State Transition Rule	67
6.4 Pheromone Updating Rule	69
6.5 Stopping Criterion	71
6.6 Computational Experiments	71
6.6.1 Introduction of Transmission Delays to the ILP Formulation	72
6.6.2 Comparisons between the ACO and the ILP Formulation.....	72
6.6.3 Comparisons of Iterations	75
6.7 Conclusion.....	76
Chapter 7 Genetic Algorithm (GA) for MRP-DC-WWC-SR.....	89
7.1 Problem Formulation.....	90
7.2 Concept of GA	98
7.2.1 Selection / Reproduction.....	99
7.2.2 Crossover	100
7.2.3 Mutation.....	101
7.3 GA for MRP-DC-WWC-SR.....	101
7.3.1 Chromosomal Encoding Scheme.....	101
7.3.2 Crossover Operator.....	104
7.3.3 Mutation.....	105
7.3.4 Fitness Function Definition	108
7.3.5 Chromosome Repair	109
7.3.6 Replacement Strategy	110
7.3.7 Termination Rules.....	110
7.4 Experiments	111
7.4.1 Performance Assessment of the GA model.....	112
7.4.2 Comparisons between GA, 3PM, and ILP.....	115
7.4.3 Comparisons among Four Mutation Heuristics.....	115
7.5 Conclusion.....	125
Chapter 8 Heuristics for MRWAP-DC-WWC-SR	126
8.1 NKSPH for MRWAP-DC-WWC-SR	127
8.2 ISM for MRWAP-DC.....	131
8.1.1 Selecting Wavelength Procedure (SWP).....	131
8.1.2 Finding Assigned Light-Tree Procedure (FALP).....	133
8.3. Experiments.....	138
8.3.1 Comparisons for Different Values of k in NKSPH	138
8.3.2 Comparisons among ILP, NKSPH, and ISM.....	139
8.3.3 Comparisons between MaxDepth and MaxDest.....	140

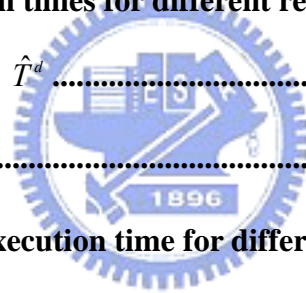
8.3.4 Comparisons between MaxE and MinR	142
8.4. Conclusion.....	143
Chapter 9 Conclusion and Future Work.....	147
References	151



List of Figures

Figure 2-1: Essential optical network.....	9
Figure 2-2: WDM for three wavelengths	11
Figure 2-3: Wavelength add/drop multiplexer (WADM).....	14
Figure 2-5: Three multicast schemes	21
Figure 4-1: WDM network and multicast trees for a request r , $(v_9, \{v_0, v_5, v_8, v_{10}\}, 3.3)$	40
Figure 4-2: Problem Lattice	43
Figure 5.1: WDM network G	49
Figure 6-1: ACO framework	65
Figure 7-1: Example of $MSpT^c(T, D = \{v_0, v_5, v_8, v_{10}\})$	91
Figure 7-2: Sample of multicast tree T	95
Figure 7-3: Two light-trees converted from Figure 7-2.....	97
Figure 7-4: GA procedure	98
Figure 7-5: Demanded generations for different PSs and different requests in net_1 ..	116
Figure 7-6: Demanded generations for different PSs and requests in net_2	116
Figure 7-7: Multicast costs for different generations in net_1	117
Figure 7-8: Multicast costs for different generations in net_2	117
Figure 7-9: Promotion Percentages of multicast costs for different requests	118
Figure 7-10: Elapsed execution times for different mutation probabilities (MPs) in net_1	118
Figure 7-11: Average generations for different mutation probabilities (MPs) in net_1	119
Figure 7-12: Elapsed execution times for different crossover probabilities (CPs) in net_1	119

.....	119
Figure 7-13: Average generations for different crossover probabilities (<i>CPs</i>) in <i>net</i> ₁	120
Figure 7-14: Multicast costs for different requests in <i>net</i> ₁	120
Figure 7-15: Multicast costs for different requests in <i>net</i> ₁	121
Figure 7-16: Multicast costs for different generations in <i>net</i> ₂	121
Figure 7-17: Multicast costs for different requests in <i>net</i> ₂	122
Figure 7-18: Average promotion percentages of multicast cost for different requests	122
Figure 7-19: Elapsed execution times for different requests	123
Figure 7-20: Multicast costs for different generations in <i>net</i> ₁	123
Figure 7-21: Multicast costs for different generations in <i>net</i> ₁	124
Figure 7-22: Elapsed execution times for different requests in <i>net</i> ₁	124
Figure 8-1: A weak candidate \hat{T}^d	133
Figure 8-2: $\hat{T}^d \cup P^c(v, u)$	134
Figure 8-3: Comparisons of execution time for different requests	141
Figure 8-4: Comparisons of multicast cost distance for different requests.....	142



List of Tables

Table 4-1: The relations of applying methods and studied problems	44
Table 5-1: Experimental results for different β/α	60
Table 5-2: Experimental results for different networks by using ILP.....	60
Table 6-1: Average execution time (sec.) for different χ values and different networks	77
Table 6-2: Experimental requests of 200 requests routed in 40 nodes	78
Table 6-3: Experimental requests of 200 requests routed in 50 nodes	79
Table 6-4: Experimental requests of 200 requests routed in 60 nodes	79
Table 6-5: Results of 200 requests routed in 40 nodes	81
Table 6-6: Results of 200 requests routed in 50 nodes	83
Table 6-7: Results of 200 requests routed in 60 nodes	85
Table 6-8: Average results for different networks ($n=40, 50, 60$)	87
Table 8-1: Experimental results for different values of k in NKSPH	144
Table 8-2: Experimental results among ILP, NKSPH, and ISM.....	145
Table 8-3: Worst cases of NKSPH and ISM among test groups	146
Table 8-4: Comparisons of <i>MaxE</i> and <i>MinR</i>	146
Table 9-1: The further works in extending proposed methods or proposing new methods	150

Chapter 1 Introduction

1.1 Motivation

Due to the explosive growth of the Internet and bandwidth-intensive applications, such as HDTV, videoconferencing, and video-on-demand system, a new technology to provide high-bandwidth transport is required. By the help of optical technologies and optical fibers to provide an excellent medium for transmitting tremendous amount of data in the speed of fifty terabits per second, the requirement can be realized by optical networks which are a type of high-capacity telecommunication networks. Among the proposed network architectures, WDM (wavelength division multiplexing) network has undoubtedly become the solution adopted in order to increase the capacity of long-haul wide area networks; furthermore, the deployment of WDM networks has also recently been considered in the metro networks. However, despite the fact that many commercially available systems are ready to be installed, there are still many challenges that have to be resolved before WDM can be widely deployed in the metro network environment.

Based on the attractive communication bandwidth in WDM networks, many new network applications are inspiring new communication models in which multicast is an important communication of a point to multipoint to distribute multimedia content (or data). Several commercial protocols including synchronous digital hierarchy (SDH), synchronous optical network (SONET), asynchronous transfer mode (ATM), and internet protocol (IP), are

investigated to implement them in WDM network. These protocols form a WDM protocol stack. The preliminary requirement in these protocols is how to find a route and how to assign one or more wavelengths to all links in the route such that the data will be sent to all destinations. The route is a set of transmission connections between two nodes and the union of the connections can be viewed as a multicast tree. The problem of finding a multicast tree and assigning wavelengths to transmit data is an important and interesting research topic in transporting data. The multicast tree and assigned wavelengths may be determined in different layer in the protocol stack; for example, in the protocol IP over WDM, they will be determined in IP layer.

The problem of determining a multicast tree and assigning wavelengths to links in the tree is called the multicast routing and wavelength assignment problem (MRWAP). In previous research, the well known problem, routing and wavelength assignment problem (RWAP) has been shown to be NP-hard, where the RWAP is to find a light-path from a source to a destination for each request and assign a wavelength to each link in the light-path for a set of connection requests. Therefore, the MRWAP is also NP-hard and expected to be computationally challenging. Although the MRWAP has been studied since the 1990s in traditional copper networks and since the 1999s in WDM networks, using different features including network topology (ring, mesh, star, tree, or graph), the routers with or without the capacity of optical wavelength conversion and with or without electric wavelength conversion, the routers with or without light splitting capability, the multicast request being static or dynamic, and different objective functions, will introduce different MRWAP instances in WDM networks.

In WDM networks, deploying the light splitting capability and the wavelength conversion capability to optical routers implies to demonstrate a WDM network in different construction costs. Due to the sophisticated architecture of the routers with light splitting

capability, using the routers with superior light splitting capacities to build a network is usually more expensive than using those with inferior light splitting capacities or without ones. Therefore, a WDM network with heterogeneous light splitting capabilities (WDM-He network), in which the light splitting capacities of all routers could be different, can better reflect the real-world requirement.

For the new network applications, such as videoconferencing and video-on-demand system, to guarantee video and audio signals to be efficiently transmitted in interactive multimedia applications is very important. Therefore, transmission delays from a source to each destination need to be limited under a given delay bound (value), where the delay bound may be determined according to the degree of emergence, data priority, or application type of the data. Delay bound constraints are thus realistic to reflect the demand about data transmission in the future. A request with delay bound dictates that it needs to be successfully transmitted before its given delay constraint is violated. To combine the real-world requirements in constructing WDM networks and transmitting requests, we will extend these previous researches by adding delay bound constraints in WDM-He networks. The MRWAP to route requests subject to delay constraints is denoted by MRWAP-DC. In this dissertation, not only the MWRAP-DC is well formulated but also integer linear programming (ILP) and two meta-heuristic, ant colony optimization (ACO), genetic algorithm (GA), and two heuristics are proposed to explore different variants of the MRWAP-DC.

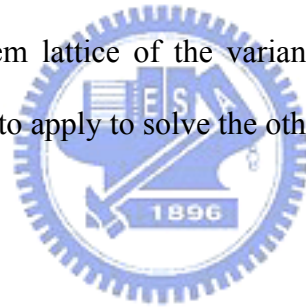
1.2 Contribution

The major contributions conveyed in this dissertation are outlined as follows:

- (1) A new problem, MRWAP-DC, is defined and formulated.
- (2) Variants of the MRWAP-DC including MRWAP-DC-WWC, URWAP-DC-SR,

MRP-DC-WWC-SR, and MRWAP-DC-WWC-SR are studied.

- (3) Four methods, ILP, GA, ACO, two heuristics (Near- k -Shortest-Path-based heuristic (NKSPH) and Iterative Solution Model (ISM)), are proposed to examine the MRWAP-DC-WWC, URWAP-DC-SR, MRP-DC-WWC-SR, and MRWAP-DC-WWC-SR, respectively.
- (4) In the ILP formulation, the formulation has been not only shown to find optimal solution but also demonstrated by the CPLEX.
- (5) A well-known simulation model is used to simulate these proposed methods. For the different problems, the experimental results are conducted to compare GA, ACO, and two heuristics with the ILP formulations. The statistics of the experiments reveal that the proposed algorithms are effective as well as efficient.
- (6) According to the problem lattice of the variants of MRWAP-DC, these proposed methods may be refined to apply to solve the other variants.



1.3 Reader's Guide

The remainder of this dissertation is organized as follows. In Chapter 2, the characteristics of optical networks, multiplexing transmission systems, optical equipment, evolution of optical networks, and multicast will be introduced. The introduction not only gives an overview of what is the optical network but also helps us to understand that the studied problems in this dissertation are reasonable. The preliminaries of routing and wavelength assignment problem (RWAP) are given in Chapter 3. According to the number of destinations required to be routed, the RWAP is divided into two types, unicast RWAP (URWAP) and multicast RWAP (MRWAP). Previous research on the URWAP and MRWAP will be surveyed. In the end of this chapter, a model of the RWAP will be proposed such that the complexity and relationship among previous studies can be compared. The problems

discussed in this dissertation can be represented by this model.

In Chapter 4, the MRWAP-DC will be formulated to define as a general problem. Nevertheless, due to this generalization, the MRWAP-DC is hard to solve in an affordable execution time. The variants, including the MRWAP-DC-WWC, URWAP-DC-SR, MRP-DC-WWC-SR, and MRWAP-DC-WWC-SR, are special cases of the MRWAP-DC by setting the request set to only contain single request, setting the multicast request to be a unicast request, and setting the network without wavelength conversion. These variants will be explored by different methods introduced in following chapters.

In Chapter 5, an ILP (Integer Linear Programming) formulation will be proposed to solve the MRWAP-DC which is more difficult than the RWAP-DC due to the multicasting feature. The tool CPLEX will be used to implement the ILP formulation. The simulation results obtained by the ILP method will be viewed as a baseline for the comparison with meta-heuristics. Although the ILP model can be deployed to find optimal solutions, the execution time is not affordable for large-scale networks. Moreover, the MRWAP-DC exhibits much more complicated structures; it is unlikely to follow the ILP approach to produce optimal solutions in an acceptable time.

Chapter 6 will address a design of ant colony optimization (ACO), which is a meta-heuristic developed in early 1990s. The ACO uses natural metaphor inspired by the behavior of ant colonies to solve complex combinatorial optimization problems for finding near-optimal solutions. It has demonstrated significant strengths in many application areas, such as the traveling salesman problem, graph coloring problem, quadratic assignment problem, generalized minimum spanning tree problem, scheduling problems, and minimum weight vertex cover problem. An ACO algorithm will be designed for the URWAP-DC-SR and comparisons between ACO and ILP will be made. Therefore, our study will not only extend the application areas of the ACO approach but also suggest a new viable method for coping

with the complex optimization problems arising from the WDM domain

In Chapter 7, a genetic algorithm (GA) will be introduced for the MRP-DC-WWC-SR. The set of possible solutions of the problem is the search space in GA. A solution in the search space is called an individual whose genotype is composed of a set of chromosomes represented by sequences of 0's and 1's. These chromosomes of individuals could dominate phenotypes of individuals. Each individual is associated with an objective function value called fitness. A good individual is the one that has a high or low fitness value depending upon the problem's goal as maximization or minimization. The strength of a chromosome in the individual is represented by its fitness value and the chromosomes of the individuals are to be carried to the next generation. A set of individuals with associated fitness values is called the population. This population at a given stage of GA is called a generation. The best individual was found in each generation at which the individual with that best fitness value was discovered. The general GA proceeds to include five basic operations, individual coding, selection/reproduction, crossover, mutation, and replacement. A GA algorithm will be developed to solve the MRP-DC-WWC-SR. We will compare its performance with the ILP model.

For routing a set of requests in a large-scale network, where the network provides more wavelengths, more requests are issued and the requests have enormous destinations, the ILP, ACO and GA are all time-demanding in solving the MRWAP-DC or the UEWAP-DC. In Chapter 8, two efficient heuristics, Near- k -Shortest-Path-based Heuristic (*NKSPH*) and Iterative Solution Model (*ISM*), will be proposed to find feasible approximate solutions. Based on the k -shortest light-paths between the source and each destination, NKSPH can find near optimal solutions and reduce the failure opportunity by using the adjustment in the value of k according to the execution time, where increasing the value of k will enlarge the searching space but provide a better opportunity of reaching the optimal solution. Conclusion

and future work will be given in Chapter 9 to make a summary and review about these proposed methods. In this chapter, we will outline several works worthy to research further including extending the proposed methods to solve the other variants, relaxing these integral variables in the ILP formulation to be real variables to become a relaxed-ILP formulation, and introducing simulated annealing (SA) algorithm.



Chapter 2 Overview of Optical Networks

Apart from providing high transport capacity at low costs (approximately \$0.3 per yard), optical fibers has low bit error rate (received fraction in error is approximated to 10^{-12} , compared to 10^{-6} in copper cable), low signal attenuation (0.2 decibels per kilometer, 0.2 dB/km), low signal distortion, low power requirement, low material, small spatial requirement, and high immune to interference and crosstalk in the view of material characteristic. Based on optical technologies, different optical equipments are developed to provide the functions of signal generating, signal regenerating, signal shaping, adding/dropping signal, wavelength splitting, multiplexing, and wavelength conversion. By the exploration in the transmission bandwidth and coordination in optical equipments, different types of optical networks are demonstrated; for examples, point-to-point optical network, Gigabit Ethernet, broadcast-and-select network, linear lightwave networks, and wavelength routing network. A good survey on growth of optical networks can be found in [10]. In this Chapter, the characteristics of optical networks, multiplexing transmission system, optical equipments, evolution of optical networks, and multicast are introduced. The introduction not only gives an overview of optical networks but also justifies the problems studied in this dissertation.

2.1 Characteristics of Optical Networks

Compared with traditional networks in which data is converted to electrons to travel

through copper cable, data in optical networks is converted to bits of light called *photons* and then transmitted over optical fibers. Optical networks provide faster transmission than traditional networks because photons moving in a fiber do not affect the others and are not affected by stray photons outside the fiber. An essential optical network is shown in Figure 2-1. Four elements including optical transmitter, optical fiber, signal regenerator, and optical receiver are required. The optical transmitter consists of a modulation circuit for coding input electric signal and a driver circuit to drive light source (LED, Light Emitting Diode, or laser) to produce a beam of light. That is, electrical binary information will be modulated into a sequence of on/off light pluses, and then they will be transmitted into the optical fiber. The light beam will become attenuant in a long transmission; thus, the signal regeneration will amplify the power of the beam. When the signal pluses are propagated to the receiver, they will be demodulated back into electrical signal.

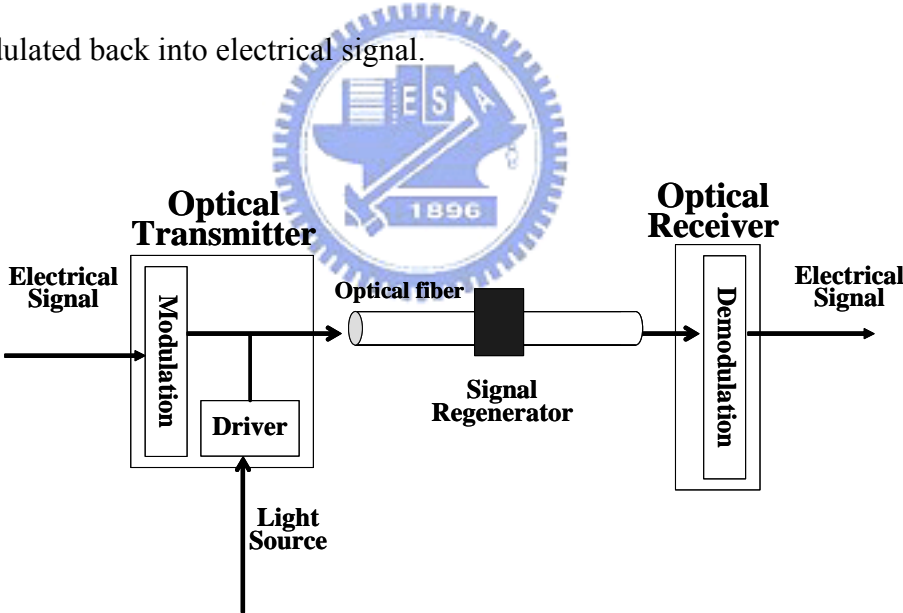


Figure 2-1: Essential optical network

Optical fiber consists of three parts, core glass, cladding glass, and plastic jacket. Due to the characteristic, the core glass with a higher index of refraction than the cladding glass, a ray of light from the core glass approaches the surface of the cladding glass will result in reflection which partial ray is reflected back into the core glass and refraction which partial

ray go through the surface of the cladding glass. When the incident angle is less than a specified value, refraction will not occur such that the ray of light is completely reflected back into the core glass. It is called *total internal reflection*. Using total internal reflection in optical fibers, a ray can be propagated at the core glass over a long distance in low signal attenuation. Because indexes of refraction for different wavelengths are different, the incident angle to produce total internal reflection will be different for rays with different wavelength. A *mode* is defined as a ray of light that enters an optical fiber at a particular angle. According to the number of modes provided in an optical fiber, optical fibers are classified into two types, multi-mode and single-mode. Multi-mode fiber whose core glass will be 50 microns or so allows multiple modes of light to propagate through the fiber. Single-mode fiber whose core glass will be 8 microns or so allows only one. Multi-mode fiber uses LED as the light-generating device, while single-mode fiber generally uses laser.

The photons transmitted in optical fibers will be affected by attenuation and dispersion. Attenuation, which is the loss of signal power propagated over some distance and computed as $10\log_{10}(\text{transmitted power}/\text{received power})$, limits the maximum transmitted data rate or bandwidth capacity, and maximum distance. Attenuation is primarily caused by three factors; (1) scattering of light from molecular level irregularities in the glass structure; (2) light absorbed by residual materials, such as metals or water ions [8], within the core glass and cladding glass; and (3) light leakage due to bending, splices, connectors, or other outside forces. Dispersion, which is time distortion of an optical signal that results in pulse broadening, causes a waveform to become significantly distorted and can result in unacceptable levels of composite second-order distortion. There are two dispersions, *modal dispersion* [70] and *chromatic dispersion* [18], in transmitting signal in fibers. The two types can be balanced to produce a wavelength of zero dispersion anywhere within the 1,310 nm to 1,650 nm. In order to overcome attenuation and dispersion, electrical or optical signal regenerators (repeaters) are

used to amplify the power of rays. Apart from amplifying signal power at full wavelength simultaneously, the function of operation in amplification can be classified into signal reshaping and signal reclocking. The former reproduces the attenuated signal by reshaping the plus shape of each bit and eliminating noise. The latter reproduces the attenuated signal by synchronizing the signal to its original bit pattern and bit rate. Three types of signal regenerators are implemented to provide the corresponding operations: 3R (including signal reshaping and signal reclocking operations), 2R (only including signal reshaping operation), and 1R (simple generator without signal reshaping and signal reclocking operations).

2.2 Optical Multiplexing Transmission System

The way to exploit the fiber’s huge bandwidth is to introduce concurrency among multiple user transmissions. The concurrency may be provided according to wavelength, time slots, or wave shape, and thus the technologies of wavelength-division multiplexing (WDM), optical time-division multiplexing (O-TDM), and optical code-division multiplexing (O-CDM) are developed to provide flexibility.

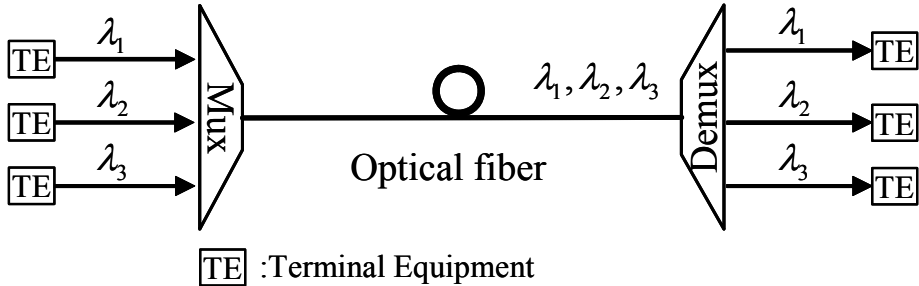


Figure 2-2: WDM for three wavelengths

For a signal transmitted in a wavelength which corresponds to an end user operating at

electric speed, the WDM provides the function in which each signal is modulated and several signals will be combined and transmitted simultaneously in a same optical fiber. As shown in Figure 2-2, three user's terminal equipment can be multiplexed on the same fiber. WDM systems are popular with telecommunications companies because they allow them to expand the capacity of the network without laying more fiber links. According to market segments, WDM systems are divided into dense and coarse WDM. Systems with more than eight active wavelengths per fiber are generally considered Dense WDM (DWDM) systems; otherwise, they are classified as coarse WDM (CWDM). Since DWDM is more expensive than CWDM, DWDM tends to be used at a higher level in the communications hierarchy.

O-TDM is a technique used to increase the bandwidth of a single wavelength channel. In O-TDM, communication channels are assigned based on time slots in a frame. Incoming electronic data is imprinted upon the pulse stream via an electro-optic modulator. Thus, the time-multiplexed data can be encoded inside a sub-nanosecond (ns) time slot. Many such timeslots are time-interleaved into a frame format, sent through the optical fiber, and demultiplexed at the receiver. Under O-TDM, each end-user should be able to synchronize to within one time slot.

The basic concept of O-CDM is found from spread-spectrum communication technique [58], which is a means of transmission, where the signal occupies a bandwidth in excess of the minimum necessary to send the information; the band spread is accomplished by means of a code that is independent of the data, and a synchronized reception with the code at the receiver is used for despreading and subsequent data recovery. O-CDM provides a class of new multiplexing techniques extending the technique of CDMA (code-division multiple-access) [73]. In O-CDM, each data in channel is encoded with the specific code such that only intended user with the corrected code can recover the encoded information. The selection of the desired signal from among all of the other signals on the channel is based on matched

filtering. The output of the optical decoder is the correlation between the input signal and the matched filter. Thus, a proper choice of optical codes allows signals from all connected network nodes to be carried without interference between signals. Therefore, simultaneous multiple access can be achieved without complex network protocols. In sum, O-TDM and O-CDM are relatively less attractive than WDM, and then WDM is the current favorite multiplexing technology for long-haul communications in optical communication networks.

2.3 Optical Equipment

For a single-mode fiber whose potential bandwidth is nearly 50 Tb/s, the data rate is nearly ten thousand times of electronic data rates of a few gigabits per second (Gb/s) in traditional network using copper fiber. Because end-user's equipment operates at electronic rate and the optical signals are transmitted in media (optical fibers) at optical data rate, the effort should be elapsed to tap into this huge mismatch of data rate between optic and electron. In the past years, there are several optical equipments are developed such that the optical networks can provide more bandwidth and complexity. Erbium-doped fiber amplifier (EDFA), wavelength add/drop multiplexer (WADM), and optical wavelength crossconnect (OXC) are significant equipment.

A conventional repeater puts a modulated optical signal through three stages: (1) optical-to-electronic conversion, (2) electronic signal amplification, and (3) electronic-to-optical conversion. The three stages usually are abbreviated to OEO conversions. The repeaters limit the bandwidth of the signals that can be transmitted in long spans of optical fiber. Eliminating complex and inefficient OEO conversion, an Erbium-doped fiber amplifier (EDFA) is an optical repeater that amplifies a modulated beam directly without OEO conversions. The device uses a short length of optical fiber doped with the rare-earth element erbium. When the

signal-carrying beams pass through this fiber, external energy (called optical pump laser) is applied to energize the erbium ions. Because the light beams carried by a fiber are attenuated as they travel through the material, this necessitates the use of repeaters in spans of optical fiber longer than about 100 kilometers. Using EDFA, the WDM networks will provide the way of amplifying all wavelengths at the same time to increase the realizable bandwidth and transmitting distance.

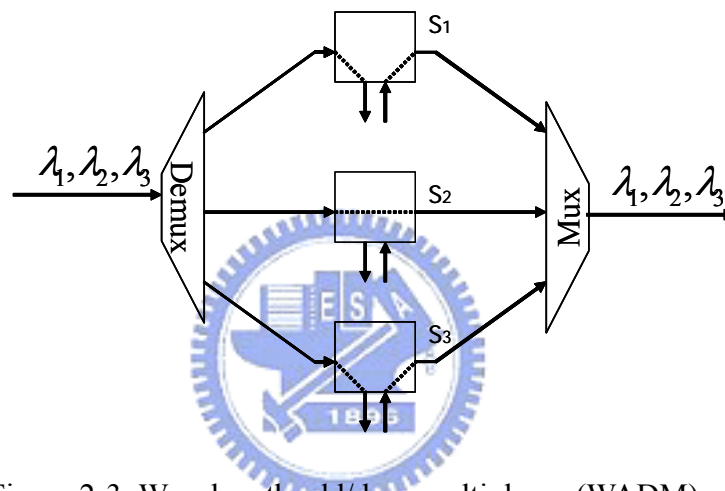


Figure 2-3: Wavelength add/drop multiplexer (WADM)

While EDFA and WDM technology is used to provide a large scale network in which two nodes may be connected in more than 300 kilometers distance, it is necessary to drop or to add some traffic at internal nodes. Based on the requirement, the wavelength add/drop multiplexer (WADM) can be realized by using demultiplexers, switches, and multiplexers. For example, the WADM with three wavelengths shown in Figure 2-3 requires a demultiplexer, three 2x2 switches (one switch per wavelength), and a multiplexer. Two statuses, ‘bar’ state and ‘cross’ state, are respectively provided the function of the signal on corresponding wavelength passing through the WADM, and the function of the signal on the corresponding wavelength being dropped locally and another signal being added to the same

wavelength at the WADM. In Figure 2-3, because switches S_1 and S_3 are in "cross" status, the signals in λ_1 and λ_3 will be added/dropped in the WADM. Using the WADM, the unusable signals will be dropped such that the user's signals can be added to the data beam to transmit.

When a wavelength is used to carry user's signals, the data will be switched from a specified input port to a specified output port in the same wavelength. It is not realistic to require that the same wavelength is occupied from the source node to the destination node. This type of switch is incapable of converting the data from one wavelength to another one; otherwise, the switch must have the capability of wavelength conversion [63] and it is called a OXC with wavelength conversion shown in Figure 2-4 which includes two input ports and two output ports to connect two input fiber links and two output fiber links with three wavelengths. Using appropriate fiber interconnection devices can provide the flexibility of routing signals between different wavelengths. For providing the capability, several pulses generated to convert the wavelength will give rise to transmission delays. In earlier research [75], the delayed-interference signal-wavelength converter (DISC) was proposed to provide 3.8-THz-shifted (from 1530 to 1560-nm) by generating more than 14-ps-long pulses. Recently, 40Gb/s all-optical wavelength converters comprising an SOA (semiconductor optical amplifier) and a plus reformatting optical filter were demonstrated in [44]. The pulses are generated or reformatted and thus cause delayed transmission. Although eliminating wavelength conversion capability can significantly reduce the cost of constructing a switch, it may result in reduction of network efficiency because the same wavelength must be available on each link of the constructed route. Therefore, it may be more realistic to construct switches with wavelength conversion. Deploying a part of switches with wavelength conversion in networks can be a viable alternative to balance the installation cost and efficiency. Networks of this type are called a *WDM network with sparse wavelength conversion*.

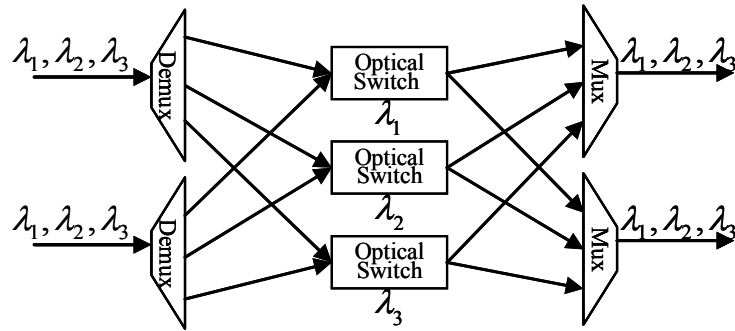


Figure 2-4: OXC with 2 input ports and 2 output ports

2.4 Evolution of Optical Networks

In multi-mode fibers, different wavelengths can be propagated in a certain distance. Different optical networks implementing different technologies and architectures are demonstrated in the evolution of optical networks. Since optical fibers had been manufactured in 1970s, a variety of optical networks have come into existence to replace traditional networks using copper cables. Optical networks can be classified according to different criteria. According to network topology, they can be ring, mesh, or star. Ring topology is superior to mesh topology in many ways: (1) the number of rings increases linearly with the number of nodes, (2) fault tolerance, (3) load sharing, and (4) reduced load at the router and no need for buffering. According the number of hops in optical networks, they are classified into single-hop networks and multi-hop networks. Nevertheless, according to evolution of optical networks, there are three generations in the trend of optical network development. The first generation started when fibers were chosen to replace copper media. The second generation emerged by providing network functionality by electronics. The third generation demonstrated in 1999 was intelligent optical networks to provide the capacity of routing and signaling for optical paths.

Optical transmission was first introduced in the first generation. Data signals must be

converted between optical and electronic equipment and the protocols used in copper-based network are still deployed. Due to the burden of OEO conversions, only a small fraction of bandwidth, less than 0.1 percent, is utilized. FDDI (Fiber Distributed Data Interface) and Gigabit Ethernet are two major products in this generation to provide 100~200 Mb/s and 1~10Gb/s bandwidths.

Breakthroughs in technologies of WDM and EDFA, the second generation networks exploit the bandwidth of optical fibers by traditional electric network equipment, such as switches, amplifiers, and so on. The broadcast-and-select network (BASN) [28] is a representative product which consists of a passive star coupler (PSC) and connected nodes to form a star-like network. Each node equipped with one or more fixed-tuned or tunable optical transmitters and one or more fixed-tuned or tunable optical receivers. Each transmitter in each source node will be tuned to use a different wavelength such that all the signals are transmitted simultaneously to PSC. All transmitted signals will be combined in the PSC, and then broadcasted to all nodes. All destination nodes will tune their receivers to the corrected wavelength such that the signals propagated in the wavelength will be received by the receiver. The requirement of fast tunability is required in transmitters and receivers. Because each transmission in BASN needs to be broadcasted to all other nodes, not only most of the transmitted power is wasted on receivers but also the number of transmitted messages is limited by the number of wavelengths in the network. Therefore, although BASN is suitable for local or metropolitan area networks, it is not suitable for wide area networks.

The third generation is wavelength-based routing networks that are presented as a scalable alternative by the help of optical WADM and optical crossconnector (OXC) [84]. To avoid wastage of transmit power, channeling a signal from the transmitter of a source node to the receiver of destination node along a restricted route is needed instead of letting it spread out over the entire network as in BSAN. Therefore, at each intermediate node on the route,

light coming in at one incoming port in a given wavelength is routed out of one and only one outbound port by a wavelength router. Not only the need for traffic is groomed but also the capacity gain provided by wavelength conversion is justified. In order to transmit signals more efficiently, the problem of the virtual topology design for offline traffic environment and the problem of finding a route and assigning a wavelength are imperative problems. By the different technologies adopted to build wavelength-based routing networks, there can be linear lightwave networks (LLN) and wavelength routing networks (WRN).

In LLN discussed in [3], nodes are classified into two types *end nodes* and *routing nodes*. *Routing nodes* provide the function to multiplex and to demultiplex optical signals in wavebands but not in wavelengths, where wavebands partitioned from lower attenuation band (for examples, 1550 nm band) consist of a number or eight wavebands transmitted in fibers. Each waveband can be partitioned into a number of wavelengths. *End nodes* provide the function to multiplex and to demultiplex optical signals in wavelengths but not wavebands. The objective of the architecture is to provide purely optical connections on demand, supporting a high degree of flexibility, including user-chosen modulation formats (digital or analog) and user-chosen bitrates (or bandwidths).

To build a more flexible multipoint optical network such that the signal can be routed based on wavelength level, the wavelength routing networks (WRN) [70] is developed by deploying optical wavelength crossconnect (OXC) in networks. Three problems in BASN, lacking of wavelength reuse, power splitting loss, and scalability to WAN (wide area network) can be resolved. Using point-to-point optical fiber links to connect input ports and output ports in OXCs, a WRN with an arbitrary topology can be established and data can be rerouted to other optical switches based on wavelengths. Data will be sent from one node to another according to the wavelength-level connections that exist between every two consecutive switches. A WRN which carries data without any intermediate OEO conversion is called an

all-optical transparent WRN [8][70]. The type of optical networks deploying the WDM technology is called WDM networks, including BASN, LLN, and WRN.

2.5 Multicast

To provide high bandwidth network communication, several commercial protocols including synchronous digital hierarchy (SDH), synchronous optical network (SONET), asynchronous transfer mode (ATM), and internet protocol (IP), are investigated to implement them in WDM networks [74]. For example, in the protocol IP over WDM, the transmitted data will be packeted based upon IP protocols. To provide more flexibility of services, various existing protocols over WDM could be directly supported in wavelength channels.

Due to the attractive communication bandwidth in WDM networks, many new network applications (distributed databases [9], replicated file systems [51], resource allocation in distributed systems [26], distributed process management [13], distributed games [6], replicated procedure calls [15], and teleconferencing [55], and so on) are inspiring new communication models, among which *multicast* is an important communication of a point to multipoint to distribute multimedia content or data. In multicast, data will be sent from a single source (transmitter) to multiple destinations (receivers) and the route of transmitting the type of requests is tree-like structure called a multicast tree.

Multicast in traditional copper networks has been well studied since the 1990s. Multicast Backbone (MBONE) [24], which can be seen as an overlay of the internet exploring applications of multicast over IP layer by using the reliable multicast transport protocol (RMTP) [57] for IP, is the first demonstrator. RMTP is used to reliability guarantee in application development. For example, distributed databases which need to be certain that all members of a multicast group agree on which packets have been received. The only service demanded by RMTP from the underlying network is the establishment of a multicast tree

from the sender to the receivers, where the multicast tree can be set up by multicast routing protocols (such as, DVMRP [61], PIM [66], or CBT[5]). The function of RMTP is to deliver packets from the sender to the receivers in sequence along the multicast tree, independent of how the tree is created and resources are allocated. Roca, *et al.* [77], gave an overview of most of the directions taken by research in multicast research. Mir [52] gave a survey of techniques, architectures, and algorithms for multicasting data in communication switch networks.

Nevertheless, a realistic demonstrator of multicast in WDM networks seems in development. For the protocol stack, multicast can be implemented at different layers; for example, WDM layer, SDH/SONET layer, ATM layer, or IP layer. Three schemes, IP multicast, Multiple-unicast (or IP multicast via WDM multicast), and WDM multicast for multicasting data in IP over WDM networks, were introduced by Qiao *et al.* [62]. In IP multicast, the multicast tree is constructed in the IP layer, and each node will make copies of data and transmit each copy to each successor. As shown in Figure 2-5 (a), v_2 is a branch node to pass data to v_3 and v_4 . Therefore, it is necessary to make a copy in v_3 , to send the copy to passing through v_2 to v_4 . Because it requires OEO conversions of packing data at each router, this scheme is not only inefficient but also unaffordable. To avoid these OEO conversions, Multiple-unicast is proposed to construct a virtual topology consisting of a set of light-paths from the source to all destinations, where the number of light-paths may equal the number of destinations. Because data will only be copied in the source node, the transmission delay of OEO conversion is still required. Besides, if some link is shared by more than one light-path, each light-path would need a different wavelength for routing the data. As shown in Figure 2-5(b), two light-paths, v_1 - v_2 - v_3 and v_1 - v_2 - v_4 , would need two different wavelengths λ_1 and λ_2 because the link between v_1 and v_2 is shared. If each light-path requires one specific wavelength, the wavelength consumption may become unaffordable. The WDM multicast

scheme is thus proposed to reduce wavelength consumption.

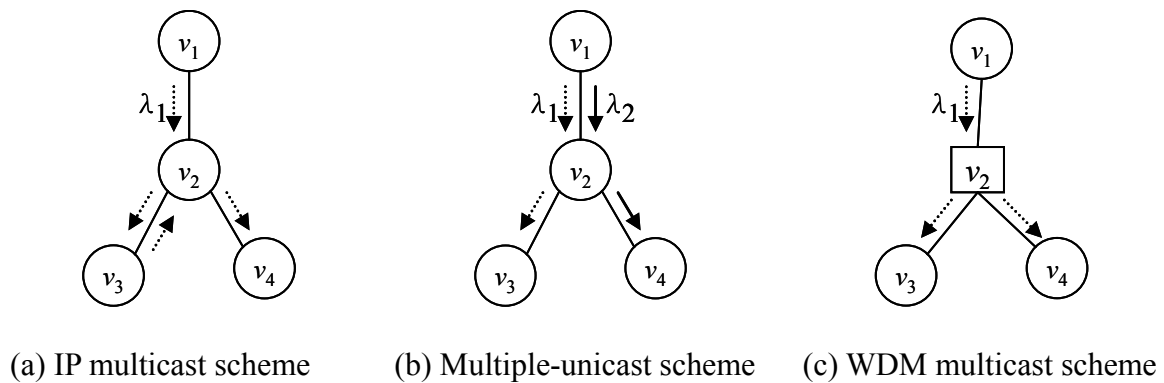


Figure 2-5: Three multicast schemes

To avoid making copies in the source node and sending a separate copy to each receiver using different wavelength, light signals need to be duplicated using optical splitters [53] or tap [60] for providing multicasting in the WDM layer; that is, WDM multicast is implemented by using a multicast tree in the WDM layer, in which the root represents the multicast source. Following the links in the multicast tree, the same data is transmitted only once on each link. Nevertheless, the optical splitters make the switch architecture complex and also cause power loss that requires optical amplifiers, but no OEO conversion is required and wavelength consumption is thus saved.

In an optical network, a tap [60] is an optical device which taps a small amount of the power of the signal from an optical fiber, and allows the signal to continue with negligible power degradation. An n -way splitter [53] is an optical device which splits an input signal among n outputs, but the power of each output will be reduced to $\frac{1}{n}$ th of the original signal. The ability to split the signal without the knowledge of the signal's characteristics allows an all-optical network to realize multicasting without the need for buffering. Therefore, all-optical networks are much more powerful than electronic networks in which store-and-

forward is necessary to achieve multicasting. In general, a node implementing optical splitter is called an *MC (multicast capable)* node as introduced in [62]; otherwise, it is called an *MI (multicast incapable)* node. For a multicast tree, if each branch nodes with more than two outbound links to connect the other nodes is an MC node, the multicast tree will be a light-tree [67]. Because the split signals can be transmitted by links to other nodes concurrently, locating an MC node for routing data to several destinations would have significant wavelength saving over multiple-unicast. To describe the splitting capacity, the *light splitting capacity* of a node is used to indicate the maximum number of split signals at an output port. The light splitting capacity of an MC node (respectively, MI node) is greater than (respectively, equal to) 1. As shown in Figure 2-5(c), since v_2 is an MC node, only wavelength λ_1 is required for routing data to v_3 and v_4 and wavelength λ_2 can be saved.



Chapter 3 Preliminaries of Routing and Wavelength Assignment Problem

To demonstrate the applications in WDM networks, several issues consisting of topology design (employing OXC, optical fiber, generator, and so on) and reconfigure, routing and wavelength assignment, multicast routing and wavelength assignment, traffic grooming, and IP-over-WDM need to be investigated. In this chapter, preliminaries of routing and wavelength assignment problem (RWAP) are introduced. According to the number of destinations required to be routed to, the RWAP is divided into two types, unicast RWAP (URWAP) and multicast RWAP (MRWAP). Besides, relevant research into the URWAP and MRWAP are surveyed. In the end of this chapter, an RWAP model is proposed such that the relationship to previous studied problems can be compared. The problems discussed in this dissertation can be demonstrated using this model.

3.1 Routing and Wavelength Assignment Problem (RWAP)

In optical networks, the traffic is usually grouped into sessions, and a session is a workstation engaged in specified activity which requires data transmission, which can be classified into two types, one-to-one (unicast) and one-to-many (multicast). A session can be described by an ordered pair (s, D) , where s is the source (a 'send' workstation from which data is transported) and D is a set of destinations ('receive' workstations to which data is

transported). According to the number of destinations in D , $m = |D|$, the activity is unicast when $m = 1$; otherwise, it is multicast. In order to complete each session successfully, how to construct a route and assign wavelengths to each link in the route is an important and interesting research topic. The problem is called the Routing and Wavelength Assignment problem (RWAP) that has been shown to be NP-hard [14]. Preliminarily, RWAP can be partitioned into two sub-problems: (1) routing problem (RP) – discussing the way to find the route of transmitting one or more sessions and (2) wavelength assignment problem (WAP) – discussing how to assign wavelengths to the links in the fixed or given routes.

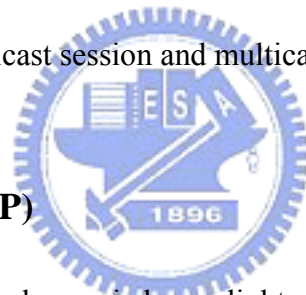
In general, the features in WDM networks include network topology (ring, mesh, star, tree, or arbitrary graph), the routers with or without the capacities of optical wavelength conversion or electric wavelength conversion, the routes with or without light splitting capability. Different variants of RWAP will be issued from discussing the WDM networks with different features. In RWAP, the requirement which establishes a session for transmitting data is called a (connection) request. Typically, requests can be static or dynamic, depending on whether they are known in advance or not. The RWAPs for discussing static requests and dynamical requests are called a static RWAP and a dynamical RWAP, respectively. Besides, different objectives will result in different methods or algorithms proposed to examine these RWAPs.

For the entire set of given static requests, the static RWAP is to set up routes for these requests while minimizing network resources (such as the numbers of links and wavelengths). Alternatively, the objective can also be to set up as many of these connection sessions as possible for a given network topology. It is worthy to note that the goal is usually to predict a long-term traffic requirement among routers in networks. Nevertheless, dynamic multicast requests arrive at the network one by one in a random way. It is necessary to tear down some connected route and establish new routes in response to the traffic change in the network.

Unlike static RWAP, dynamic RWAP must be processed online and responsively when a new request arrives. The algorithms for dynamic RWAP usually perform more poorly than those for static RWAP due to no information about network status and the requirement of short response time. The objective is usually to minimize the amount of connection blocking or the number of connected links in the route, or to maximize the number of connection sessions that are established in the network at any times.

Nevertheless, according to the variation between routes in unicast sessions which can be carried on a light-path and routes in multicast sessions which can be carried on one or a set of multicast trees, the studies on RWAP can be classified into unicast RWAP (URWAP) and multicast RWAP (MRWAP). Because the variation of routes is more significant than the variation of connection requests (static requests vs. dynamical requests), the preliminary of RWAP will be introduced with unicast session and multicast session.

3.2 Unicast RWAP (URWAP)



Because a unicast session can be carried on a light-path, the URWAPs are known as the Static Light-path Establishment (SLE) problem or the Dynamical Light-path Establishment (DLE) problem for the static and the dynamical connection requests. RWAP needs to follow the wavelength-continuity constraint that the wavelength used in the input port in a router is the same as that used in the output port. Based on the constraint, two integer linear program (ILP) formulations were proposed in [40][64] to maximize the number of established connections for a fixed number of wavelengths and a given set of unicast requests. A new ILP formulation [54] can be used for networks without wavelength conversion and easily extended for networks with sparse wavelength conversion. In [54], a dynamic and stochastically varying demand model which takes into account the effect of the uncertain future demands and availability of resources. The scenario of the demand model is considered

as follows: (1) a set of unicast requests is first required to be established (static traffic); (2) additional unicast requests arrive randomly one a time and are assigned routes and wavelengths without rerouting the existing light-paths; (3) unicast requests are terminated randomly as well. However, since the number of wavelengths is limited, some of the unicast requests will be blocked. Assuming that there is a penalty associated with blocking a light-path, the goal is to minimize the expected value of the sum of the blocking penalty. A survey of URWAP can be found in [43].

As the wavelength-continuity constraint is removed by the use of wavelength converters [84], network blocking performance is reduced, wavelength reuse is increased, higher loads are supported, and network throughput is enhanced. The benefits of wavelength conversion provided in WDM networks were introduced in [38]. Due to the complexity of wavelength converters, wavelength converters remain expensive such that it may not be economically viable to equip all the routers in a WDM network with these devices. Two architectures, share-per-node and share-per-link, were proposed for switches sharing converters in [42]. In the share-per-node structure, all converters at the switching node are collected in a converter bank, where the converter bank is a collection of a few wavelength converters which can convert any input wavelength to any output wavelength. In this architecture, only the wavelengths which require conversion are directed to the converter bank and converted wavelengths are then switched to the appropriate outbound link by the second (small) optical switch. In the share-per-link structure, each outbound link is provided with a dedicated converter bank which can be accessed only by those light-paths traveling on that particular outbound fiber link. Yoshima, *et al.* [83] proposed and demonstrated a packet-based optical multicasting. In this paper, an optical code (OC) label in the packet header combined with multicast-capable packet switch enables packet-based optical multicasting.

Therefore, it may be possible that a subset of routers have wavelength conversion,

wavelength converters are shared by more than one links, or the wavelength converter with limited wavelength conversion that the parts of wavelengths can be converted, where the former is usually called sparse wavelength conversion. The problem of designing a WDM network with limited wavelength conversion was introduced by Iness *et al.* [31]. The effect of sparse wavelength conversion on connection blocking was examined in Subramaniam *et al.* [72]. Optimal placement of limited wavelength converters in mesh networks and arbitrary networks was discussed in Arora [2] and Iness [32][60], respectively. For the placement of full wavelength converters and the placement of limited-range converters introduced in Yates [82], Houmaidi and Bassiouni [30] who developed the HYBRID algorithm by extending the k-MDS algorithm, which is used to select the best set of nodes that will be equipped with full-conversion capability. Ho and Lee [29] proposed a dynamic algorithm to minimize the number of wavelength translations in WDM networks with full-range converters and in WDM networks with limited-range converters. A hybrid algorithm based on the combination of mobile agents technique and genetic algorithm was proposed by Lei and Jiang [43] to explore the dynamic RWA problem in WDM networks with sparse wavelength conversion. Using mobile agents to cooperatively explore the network states and to continuously update the routing tables, the hybrid algorithm determined the first population of routes for a new request based on the routing table of its source node.

3.3 Multicast RWAP (MRWAP)

For the multicast session, the routing problem which explores the way to find a multicast tree is usually called a multicast routing problem (*MRP*) [3][38][81]. For the case that the route is fixed or given, how to assign wavelengths to links in the route is called a multicast wavelength assignment problem (*WAP*) [17][25]. The problem of combining *MRP* and *WAP* is called a multicast routing and wavelength assignment problem (*MRWAP*) [1][12][33][45]

[46][47][50][67][68] [71][84][85]. Although the MRP in IP networks has been well studied [7][34][36][37] and efficient multicast routing algorithms or protocols [28][79] have been developed in use for many years, a commercial algorithm or protocol used in WDM layer seem in development. However, several algorithms or heuristics have been proposed to explore the MRWAP.

In previous research, Ali and Deogun [1] have investigated the MRP in networks employing tap-and-continue switches which have limited multicast capabilities. Liang and Shen [46] have investigated the problem of finding minimum-cost for traversing a link on some wavelength and for wavelength conversion when the path has to switch to a different wavelength at some nodes. Sahin and Azizoglu [68] have investigated the problem under various splitting policies, and Malli *et al.* [50] have investigated the problem under a sparse splitting model. Sahasrabuddhe and Mukherjee [67] have formulated the RWAP for multi-hop multicast routing in packet-switched networks as a mixed-integer linear programming problem. In [85], the MRWAP with sparse light splitting is studied. Four heuristic algorithms Reroute-to-Source, Reroute-to-Any, Member-First, and Member-Only, were proposed to construct routing tree. However, the focus in is on building a multicast tree meeting all the predetermined constraints rather than on the optimal use of wavelength converters. How to minimize the number of wavelengths used in a multicast session was discussed. The argument is that the fewer the number of wavelengths in a multicast tree will lead to the fewer the number of wavelength conversions.

In the study [85], the nodes in networks were assumed to lack for the capability of wavelength conversion. Furthermore, the MRWAP deploying the nodes with wavelength conversion were studied by Sreenath *et al.* [71]. The MC nodes with wavelength conversion were called virtual sources in [71]. The virtual source approach consisting of two phases, networking partitioning phase and tree generation phase, was proposed to construct a

multicast tree. Minimizing communication cost, wavelength conversion cost and wavelength consumption of light-forest subject to a transmission delay bound was not discussed in [85] and [71]. The multicast routing problem involving in wavelength assignment is called the multicast routing and wavelength assignment problem. Jia *et al.* [33] and Chen [12] solved the problem by decomposing the problem into two sub-problems, multicast routing and wavelength assignment, so as to reduce the complexity. In [33], the problem for routing a request with a delay bound was solved under the assumption that every node in network has light splitting capability. Two integrated algorithms corresponding to the two sub-problems were proposed to minimize the sum of wavelength cost and communication cost. Considering wavelength cost, conversion cost, and initial transmission cost, Chen [12] proposed an integrated approximation algorithm to deal with the problem without delay bounds and to subject to minimize the total cost of a multicast session. For routing on a network with power splitters having full range wavelength conversion and with wavelength converters having an unlimited splitting capacity, a mixed integer programming model was proposed by Yang *et al.* [84] to solve the multicast routing and wavelength assignment for light-trees with delay bounds. In their dissertation, the objective was not only to minimize the number of used fibers and to obtain the optimal placement of power splitters but also to design the logical topology based on light-trees for multiple connection demands. The study of [84] is based on the assumption that a multicast request is routed only by a light-tree. It is possible that no light-tree can be found to satisfy the delay bound constraint and to cover all destinations in the network without enough power splitters or enough wavelength converters.

3.4 RWAP Problem Model

In order to distinguish these problems, a six-field notation $RWAP(S, \mathcal{R}, \mathcal{L}, \mathcal{D}, \mathcal{W}, \mathcal{T})$ is used to represent the RWAPs. The fields are described as follows.

\mathcal{S} : session type, $\mathcal{S} = \{unicast, multicast\}$.

\mathcal{R} : request type, $\mathcal{R} = \{static, dynamical\}$.

\mathcal{L} : light splitting capacity, $\mathcal{L} = \{no, sparse, limited\}$.

\mathcal{D} : requests with delay bound, $\mathcal{D} = \{no, delay\}$.

\mathcal{W} : wavelength conversion capability, $\mathcal{W} = \{no, sparse, share, limited-range\}$.

\mathcal{T} : network topology, $\mathcal{T} = \{ring, mesh, star, tree, graph\}$.



Chapter 4 Multicast Routing and Wavelength Assignment Problem with Delay Constraints

In [12][33][71][84] and [85], all MC nodes were assumed to have the capability of splitting an input signal to multiple output signals. The number of output signals in each MC node is no smaller than the outbound edges of the MC node so that all destinations connecting to the MC node can be routed successfully. In this dissertation, MC nodes of this type are called unrestricted MC (UMC) nodes. Due to the sophisticated architecture [10] of MC nodes, using the MC nodes with superior light splitting capacities to build network is usually more expensive than using those with inferior light splitting capacities or MI nodes. Therefore, a WDM network with heterogeneous light splitting capabilities (WDM-He network), in which the light splitting capacities of all nodes could be different, addressed in this dissertation can better reflect the real-world requirement.

To the best of our knowledge, only a limited number of papers have been reported on the MRWAP for routing multicast requests with a delay bound in a WDM-He network with or without wavelength conversion incorporating the objective of minimizing the total cost incorporating communication cost and wavelength consumption. This type of problem is called MRWAP-DC. To better provide a realistic objective function to reflect the cost for routing a request, a linear combination of communication cost and wavelength consumption, α (communication cost) + β (wavelength consumption) is considered. This objective is called

the *multicast cost* hereafter in this dissertation. Notice that communication cost ratio α and wavelength consumption ratio β can be appropriately chosen according to the topology and the load of the network.

Due to the constraints of delay bounds and limited light splitting capacities, a light-tree may be insufficient for routing a request to all destinations. For each request, a set of light-trees, called *light-forest*, will be required. Therefore, the MRWAP-DC is to find an optimal light-forest with a minimum multicast cost to route a set of requests known in advance or a requests dynamically arriving to a WDM-He network. Since the minimum Steiner tree problem (MSTP) [35] can be reduced to the studied problem by considering the minimum communication cost of connecting the source and all destinations in the request with unlimited delay bound by setting $\alpha = 1$ and $\beta = 0$, the MRMR-DC is therefore also NP-hard. The MRWAP-DC is difficult to solve because many issues are simultaneously taken into account: the request is associated with a delay bound, the network deploys MI nodes and heterogeneous MC nodes, the node with or without wavelength conversion, and a light-forest is evaluated by multicast cost. To simplify our discussion, the network used in the rest of this dissertation stands for the WDM-He network. Because the MRWAP-DC is NP-hard, it is very unlikely to optimally solve it in polynomial time.

In the following chapters, the variants of the MRWAP-DC will be introduced and formulated by integer linear programming (ILP). We shall also apply different solution methods, including genetic algorithm (GA), ant colony optimization (ACO) and simple heuristics, to derive approximate solutions in an acceptable time.

4.1 Formulation of MRWAP-DC

The MRWAP-DC is based on the following assumptions.

- (1) The WDM network is an arbitrary connective graph.

- (2) All links in the network are directed and provide the same wavelengths.
- (3) Some of the nodes are MC nodes whose light splitting capacities could be different.
- (4) Nodes may provide the capability of wavelength conversion.

Before proceeding to the problem statements and formulation, we introduce the notation that will be used in this chapter throughout this dissertation.

Notation:

- V : set of nodes in the given WDM network;
- E : set of directed links in the given WDM network;
- M : set of wavelengths available for data transmission;
- R : set of multicast requests to be routed;
- n : number of nodes;
- m : number of directed links;
- γ : number of different wavelengths in M ;
- N_R : number of multicast requests in R ;
- r^x : $r^x = (s^x, D^x, \Delta^x) \in R$, multicast request r^x from source s^x to destination set D^x subject to delay bound Δ^x ;
- q^x : number of destinations in D^x ;
- $\zeta_{i^x}^x$: specified destination in D^x for $1 \leq i^x \leq q^x$;
- $e(v_i, v_j)$: directed edge from node v_i to node v_j , e_{ij} for short;
- $c(e_{ij})$: communication cost on e_{ij} , c_{ij} for short;
- $d(e_{ij})$: transmission delay on e_{ij} , d_{ij} for short;
- $w(e_{ij}, \lambda_l)$: wavelength usage on e_{ij} , w_{ijl} for short;
- $\hat{c}(v_i)$: wavelength conversion cost at node v_i , \hat{c}_i for short; $\hat{c}_i = \infty$ or $\hat{c}_i = 0$ when node v_i does not provide the wavelength conversion capability or the wavelength

conversion cost is ignored;

$\hat{d}(v_i)$: wavelength conversion delay at node v_i , \hat{d}_i for short; $\hat{d}_i = \infty$ or $\hat{d}_i = 0$ when node v_i does not provide the wavelength conversion capability or the wavelength conversion delay is ignored;

T_k^x : specified light-tree used to route the specified request r^x ;

$V(T_k^x)$: set of nodes in T_k^x ;

$E(T^x)$: set of edges in T^x ;

$in(T_k^x, v_i)$: the number of inbound edges of node v_i in T_k^x ;

$out(T_k^x, v_i)$: the number of outbound edges of node v_i in T_k^x ;

$h^{T_k^x}$: $h^{T_k^x} : E(T_k^x) \rightarrow M$, wavelength assignment function used to describe what wavelength in edge e_{ij} is assigned for T_k^x ;

$h^{T_k^x}(e_{ij})$: assigned wavelength in e_{ij} for T_k^x , $h_{ij}^{T_k^x}$ for short;

Γ^x : assigned light-forest for request r^x , $\Gamma^x = \{(T_k^x, h^{T_k^x}) \mid 1 \leq k \leq N_{\Gamma^x}\}$, where T_k^x is a light-tree and N_{Γ^x} is the number of light-tree in Γ^x ;

Γ : set of light-forest for R , $\Gamma = \{\Gamma^x \mid 1 \leq x \leq N_R\}$;

i, j : node index, $1 \leq i, j \leq n$;

l : wavelength index, $1 \leq l \leq \gamma$;

x : request index, $1 \leq x \leq N_R$;

i^x : destination index in D^x , $1 \leq i^x \leq q^x$;

A weighted graph $G = (V, W, E, \theta, c, d, \hat{c}, \hat{d}, w)$ is used to present a WDM-He network with node set $V = \{v_1, v_2, \dots, v_n\}$, directed optical link set $E = \{e_1, e_2, \dots, e_m\}$, and wavelength

set $M = \{\lambda_1, \lambda_2, \dots, \lambda_\gamma\}$. Function $\theta : V \rightarrow \mathcal{N}$ defines the light splitting capacity of switches, function $c : (V, V) \rightarrow \mathcal{R}^+$ defines the communication cost of links, function $d : (V, V) \rightarrow \mathcal{R}^+$ specifies the transmission delay over links, function $\hat{c} : V \rightarrow \mathcal{R}^+$ defines wavelength conversion cost of nodes, function $\hat{d} : V \rightarrow \mathcal{R}^+$ defines wavelength conversion delay of nodes. Binary function $w : (E, M) \rightarrow \{0, 1\}$ is used to dictate whether a wavelength is used over a link, and binary function $e : (V, V) \rightarrow \{0, 1\}$ represents whether one link exists to connect two nodes or not.

In this dissertation, to reduce the representation of the functional value, the notation of functional values will ignore parentheses and the input variables; for example, for two nodes v_i and v_j , $v_i, v_j \in V$, $1 \leq i, j \leq n$, e_{ij} represents the functional value $e(v_i, v_j)$ for short and $e_{ij} = 1$ represents that there is one link from v_i to v_j . For some λ_l over link e_{ij} , $1 \leq l \leq \gamma$, $1 \leq i, j \leq m$, $w(e_{ij}, \lambda_l) = 1$ indicates that wavelength λ_l can be used to route data; otherwise, $w(e_{ij}, \lambda_l) = 0$. w_{ijl} will be used to represent the value $w(e_{ij}, \lambda_l)$ for short. Extending the notations, c_{ij} (i.e., $c(v_i, v_j)$) and d_{ij} (i.e., $d(v_i, v_j)$) represent the communication cost and transmission delay from node v_i to node v_j . v_i will be an MC node when $\theta_i > 1$; otherwise, $\theta_i = 1$. Moreover, the light splitting capacities of MC nodes may be different. At node v_i , wavelength communication cost and wavelength conversion delay are denoted by \hat{c}_i and \hat{d}_i , respectively. The values of \hat{c}_i and \hat{d}_i are set to be an infinity when node v_i does not provide the wavelength conversion capability or are set to be an zero when the wavelength conversion cost and the wavelength conversion delay are ignored.

For a set of multicast requests $R = \{r^1, r^2, \dots, r^{N_R}\}$, $r^x \in R$, $1 \leq x \leq N_R$, a multicast request r^x with a delay bound Δ^x is represented as (v_{s^x}, D^x, Δ^x) with $D^x = \{\zeta_1^x, \zeta_2^x, \dots, \zeta_{q^x}^x\}$ and indicates that the data needs to be routed from a certain source v_{s^x} to all destinations $\zeta_{i^x}^x$, $1 \leq i^x \leq q^x$, where $v_{s^x} \in V$, $D^x \subseteq V - \{v_{s^x}\}$ is a set of destinations, $|D^x| = q^x$, and the transmission

delay must be bounded by the bound Δ^x , where s^x represents v_{s^x} for short (i.e., $(v_{s^x}, D^x, \Delta^x) = (s^x, D^x, \Delta^x)$). For different sources, destinations and emergence levels, the delay bounds may be different. A tighter delay bound will result in fewer routes to be chosen and make the request likely to be suspended. For most of the cases, the delay bound of a request may be determined through previous experiences concerning the specified source, destinations, and application domain

Due to the effects of two constraints, the nodes with slight splitting capacities and requests with delay bounds, a light-tree may route data to a subset of the destinations for a request; therefore, several light-trees are required to cooperate for finishing the complete transmission. Let T_k^x denote some light-tree used to route request r^x , $V(T_k^x)$ and $E(T_k^x)$ denote the node set and the edge set in T_k^x , and $in(T_k^x, v_i)$ and $out(T_k^x, v_i)$ denote the number of inbound edges and the number of outbound edges of node v_i in T_k^x . The following conditions will be satisfied when T_k^x is a light-tree for r^x :

- (1) T_k^x is a tree;
- (2) $in(T_k^x, s^x) = 0$;
- (3) $\forall v_i \in V(T_k^x) - \{s^x\}, in(T_k^x, v_i) = 1$;
- (4) $\forall v_i \in V(T_k^x), out(T_k^x, v_i) \leq \theta_i$.

The first condition ensures that T_k^x is a connected tree rooted at s^x . A light-tree can be viewed as a routing topology from the root; therefore, the last condition ensures that each internal node must have a light splitting capacity sufficient for splitting the input signal to transmit to all associated nodes. If node v_i satisfies the last condition, it is called *feasible*. One may note that the transmission delay of T_k^x has not been discussed till now. The transmission

delay of T_k^x may exceed the given delay bound such that T_k^x cannot be used to route r^x .

Wavelength assignment for light-tree is to assign a specified wavelength for each link in a light-tree. A wavelength assignment function $h^{T_k^x} : E(T_k^x) \rightarrow M$ is used to describe which wavelength over e_{ij} is assigned to T_k^x . For example, $h^{T_k^x}(e_{ij}) = \lambda_l$ ($h_{ij}^{T_k^x} = l$ for short) means that λ_l over e_{ij} is assigned to T_k^x . It is worthy to note that $w(e_{ij}, \lambda_l) = 1$ must be satisfied. The pair $(T_k^x, h^{T_k^x})$ represents a light-tree designated for r^x ; that is, T_k^x is assigned specified wavelengths according to the value of wavelength assignment function $h^{T_k^x}$. Because a light-tree can be decomposed into a set of light-paths from the root to each destination in the light-tree, the transmission delay of a light-tree will be equivalent to the maximum of the transmission delays of these light-paths. Therefore, the multicast cost and the transmission delay of T_k^x using the wavelength assign function $h^{T_k^x}$ can be defined as:

$$\text{communication cost : } c(T_k^x, h^{T_k^x}) = \sum_{e_{ij} \in E(T_k^x)} c_{ij} + \sum_{e_{ij}, e_{jl} \in E(T_k^x)} \left\lceil \frac{|h_{ij}^{T_k^x} - h_{jl}^{T_k^x}|}{\gamma} \right\rceil \cdot \hat{c}_j ; \quad (4-1)$$

$$\text{transmission delay : } d(T_k^x, h^{T_k^x}) = \max_{\zeta^x \in D^x \cap V(T_k^x)} d(P_{T_k^x}(s^x, \zeta^x)), \quad (4-2)$$

where $c_{ij} = c(e_{ij})$, $h_{ij}^{T_k^x} = h^{T_k^x}(e_{ij})$, $\hat{c}_j = \hat{c}(e_{ij})$, and $P_{T_k^x}(s^x, \zeta^x)$ is a light-path in T_k^x from s^x to destination ζ^x , $\zeta^x \in D^x$, and $d(P_{T_k^x}(s^x, \zeta^x)) = \sum_{e_{ij} \in E(P_{T_k^x}(s^x, \zeta^x))} d_{ij} +$

$\sum_{e_{ij}, e_{jl} \in E(P_{T_k^x}(s^x, \zeta^x))} \left\lceil \frac{|h_{ij}^{T_k^x} - h_{jl}^{T_k^x}|}{\gamma} \right\rceil \cdot \hat{d}_j$. Assigning different wavelengths to the two connected edges,

e_{ij} and e_{jl} , implies $h_{ij}^{T_k^x} \neq h_{jl}^{T_k^x}$ and the wavelength conversion cost \hat{c}_j and wavelength

conversion delay \hat{d}_j in v_j will be required. Due to $|h_{ij}^{T_k^x} - h_{jl}^{T_k^x}| < \gamma$, $\left\lceil \frac{|h_{ij}^{T_k^x} - h_{jl}^{T_k^x}|}{\gamma} \right\rceil$ will be either 0 or 1 depending on whether $h_{ij}^{T_k^x} = h_{jl}^{T_k^x}$ or $h_{ij}^{T_k^x} \neq h_{jl}^{T_k^x}$. Therefore, $\left\lceil \frac{|h_{ij}^{T_k^x} - h_{jl}^{T_k^x}|}{\gamma} \right\rceil \cdot \hat{c}_j$ and $\left\lceil \frac{|h_{ij}^{T_k^x} - h_{jl}^{T_k^x}|}{\gamma} \right\rceil \cdot \hat{d}_j$ represent the required wavelength conversion cost and wavelength conversion delay in node v_j determined by the two values $h_{ij}^{T_k^x}$ and $h_{jl}^{T_k^x}$.

The MRWAP-DC can be represented by (G, R) . For each request r^x in R , it is possible that one or more than one light-trees required such that r^x is routed to all destinations successfully. The set of these assigned light-trees called an *assigned light-forest* are represented as $\Gamma^x = \{(T_k^x, h^{T_k^x}) \mid 1 \leq k \leq N_{\Gamma^x}\}$, where N_{Γ^x} is the number of assigned light-trees in Γ^x . The set $\Gamma = \{\Gamma^x \mid 1 \leq x \leq N_R\}$ will represent a feasible solution to (G, R) when the following conditions are satisfied:

$$\text{destination constraint : } \forall r^x, r^x \in R, \bigcup_{(T_k^x, h^{T_k^x}) \in \Gamma^x} (V(T_k^x) \cap D^x) = D^x ; \quad (4-3)$$

$$\text{delay constraint : } \forall r^x, r^x \in R, \forall k, 1 \leq k \leq N_{\Gamma^x}, d(T_k^x, h^{T_k^x}) \leq \Delta^x . \quad (4-4)$$

The destination constraint and delay constraint ensure that all destinations will be routed and the transmission delays of all assigned light-trees are bounded by the delay bound, respectively. An assigned light-forest satisfying Eq. (4-3) and (4-4) is called a *feasible light-forest* which implies that r^x can be rerouted successfully. The communication cost of Γ , $c(\Gamma)$, can be defined as:

$$c(\Gamma) = \sum_{\Gamma^x \in \Gamma} c(\Gamma^x) = \sum_{\Gamma^x \in \Gamma} \sum_{(T_k^x, h^{T_k^x}) \in \Gamma^x} c(T_k^x, h^{T_k^x}) . \quad (4-5)$$

For a set of requests, the wavelength consumption of Γ , $\omega(\Gamma)$, is equal to the total number of different wavelengths used in each $h^{T_k^x}$. In this dissertation, $\omega(\Gamma)$ is not defined as the number of different wavelengths used to route all requests but the sum of the numbers of wavelengths used in each assigned light-forest. The definition is given so as to reduce the amount of wavelengths required for routing all requests, Therefore, $\omega(\Gamma^x)$ and $\omega(T_k^x, h^{T_k^x})$ represent the wavelength consumption in Γ^x and in $(T_k^x, h^{T_k^x})$. Because one or more than one wavelengths are shared among wavelength assignment functions, it is worthy to note that $\omega(\Gamma) = \sum_{1 \leq x \leq N_R} \omega(\Gamma^x)$ but $\sum_{(T_k^x, h^{T_k^x}) \in \Gamma^x} \omega(T_k^x, h^{T_k^x}) \geq \omega(\Gamma^x)$.

The objective function, *multicast cost function* f , in the MRWAP-DC is defined as:

$$f(\Gamma) = \alpha \cdot c(\Gamma) + \beta \cdot \omega(\Gamma), \quad (4-6)$$

where α and β are *communication cost ratio* and *wavelength consumption ratio*, respectively.

Therefore, the MRWAP-DC is to find a solution with minimum multicast cost to reroute a set of given requests with a delay bound and it will be formally defined as follows.

DEFINITION 4.1: MRWAP-DC : For a weighted graph $G = (V, M, E, \theta, c, d, \hat{c}, \hat{d}, w)$ and a set of requests $R = \{r^1, r^2, \dots, r^{N_R}\}$, the problem is to find $\Gamma = \{\Gamma^x \mid 1 \leq x \leq N_R\}$ and $\Gamma^x = \{(T_k^x, h^{T_k^x}) \mid 1 \leq k \leq N_{\Gamma^x}\}$ for each $r^x = (s^x, D^x, \Delta^x)$, $1 \leq x \leq N_R$, so as to minimize

$$f(\Gamma) = \alpha \cdot c(\Gamma) + \beta \cdot \omega(\Gamma) \quad \text{for } \Gamma = \{\Gamma^x \mid 1 \leq x \leq N_R\},$$

$$\text{where } c(\Gamma) = \sum_{\Gamma^x \in \Gamma} \sum_{(T_k^x, h^{T_k^x}) \in \Gamma^x} c(T_k^x, h^{T_k^x}) \quad \text{and } \omega(\Gamma) = \sum_{\Gamma^x \in \Gamma} \omega(\Gamma^x)$$

such that $\forall \Gamma^x \in \Gamma$ and $\forall (T_k^x, h^{T_k^x}) \in \Gamma^x$ satisfy the following conditions:

$$(1) \quad \forall (T_k^x, h^{T_k^x}) \in \Gamma^x, \quad in(T_k^x, s^x) = 0 \quad (\text{source constraint});$$

- (2) $\forall (T_k^x, h^{T_k^x}) \in \Gamma^x, \forall v_i \in V(T_k^x) - \{s^x\}, \text{in}(T_k^x, v_i) = 1$ (input constraint);
- (3) $\forall (T_k^x, h^{T_k^x}) \in \Gamma^x, \forall v_i \in V(T_k^x), \text{out}(T_k^x, v_i) \leq \theta_i$ (capacity constraint);
- (4) $\bigcup_{(T_k^x, h^{T_k^x}) \in \Gamma^x} (V(T_k^x) \cap D^x) = D^x$ (destination constraint);
- (5) $\forall k, 1 \leq k \leq N_{\Gamma^x}, d(T_k^x, h^{T_k^x}) \leq \Delta^x$ (delay constraint);
- (6) $\forall e_{ij}, e_{ij} \in T_k^x, w(e_{ij}, h_{ij}^{T_k^x}) = 1$ (wavelength constraint).

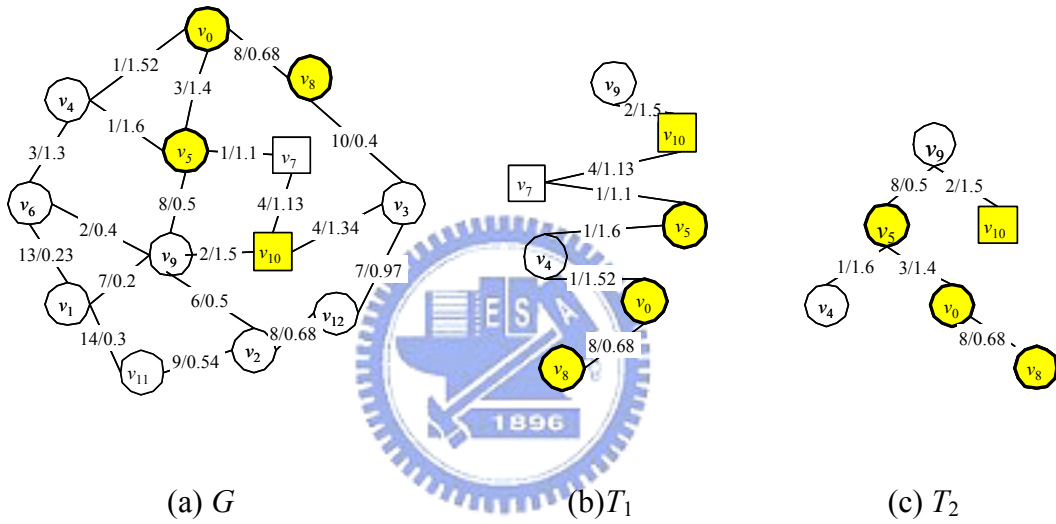


Figure 4-1: WDM network and multicast trees for a request $r, (v_9, \{v_0, v_5, v_8, v_{10}\}, 3.3)$

EXAMPLE 4.1 : As shown in Figure 4-1(a), graph G represents a WDM network with 13 nodes and 19 links. Nodes v_7 and v_{10} are MC nodes. Each link in the graph is associated with a value-pair “ a/b ”, where a and b are the communication cost and the transmission delay of the link, respectively. The wavelength conversion cost and wavelength conversion delay of node are ignored. For a given request $r, (v_9, \{v_0, v_5, v_8, v_{10}\}, 3.3)$, two multicast trees T_1 and T_2 are shown in Figure 4-2. (b) and (c). T_1 is a light-tree because it does not include any infeasible node. Nevertheless, T_2 includes an infeasible node v_5 (for $\text{out}(T_2, v_5) = 2$ to represent the outbound edges of v_5 in T_2 and $\theta(v_5) = 1, \text{out}(T_2, v_5) > \theta(v_5)$) such that it is not a light-tree. ■

By the definition of the MRWAP-DC, a feasible solution to (G, R) is a set of assigned light-forest (i.e., $\Gamma = \{\Gamma^x \mid 1 \leq x \leq N_R\}$, $\Gamma^x = \{(T_k^x, h^{T_k^x}) \mid 1 \leq k \leq N_{\Gamma^x}\}$) for routing all requests in R . The procedure of solving the MRWAP-DC includes two objectives: finding a light-forest for each request which is called a multicast routing problem (MRP) and assigning a wavelength to each link in each light-tree which is called a wavelength assignment problem (WAP). Two approaches, integrated approach which two objectives is involved in one-phase procedure and separated approach which two objectives will be investigated in two independent phases, seem very reasonable to explore the problem. The procedures implementing the integrated approach and the separated approach are called one-phase procedure and two-phase procedure, respectively. Because the MRWAP-DC involves delay constraints and light splitting capacity constraints, the problem will become more complicated than previous research. The complex conditions will cause the MRWAP-DC hard to be examined by using one-phase procedure such that the exhaustive search and the ILP formulation are hard to find an optimal or efficient solution in an affordable execution time. Therefore, two one-phase procedures which are a type of meta-heuristic including ant colony optimization (ACO) and genetic algorithm (GA) are proposed to examine the variants of the MRWAP-DC. Moreover, a critical challenge of the two-phase approach is how to avoid the situations that the light-trees found in the multicast routing phase cannot be assigned appropriate wavelengths in the wavelength assignment phase. In this dissertation, two two-phase procedures, Near- k -Shortest-Path-based Heuristic (NKSPH) and Iterative Solution Model (ISM), will be proposed to examine the studied problems. The details will be discussed in the following chapter.

4.2 Studied Problems and Methods

To simplify the notation of the studied problems, several abbreviations used in following problems include:

- URWAP : unicast routing and wavelength assignment problem;
- MRWAP : multicast routing and wavelength assignment problem;
- DC : delay constraint;
- WWC : without wavelength conversion;
- SR : single request.

Before the discussion about the MRWAP-DC, exploring the URWAP-DC will benefit to investigate the MRWAP-DC. Similar to the MRWAP-DC on DEFINITION 4.1, the dissimilarity is that the number of destinations in a request is equal to 1. That is, the URWAP-DC is a special case of the MRWAP-DC. In this dissertation, the variants of the URWAP-DC, the URWAP-DC-SR whose number of requests in R is equal to 1, the URWAP-DC-WWC in which the nodes do not implement wavelength conversion capability, and the URWAP-DC-WWC-SR combining the two additional conditions in the URWAP-DC-SR and the URWAP-DC-WWC, are introduced. Among these problems, the URWAP-DC is the most general problem. Extending the study of the URWAP-DC, the special cases of the MRWAP-DC, MRWAP-DC-WWC-SR and MRWAP-DC-SR, are also studied. The MRP-DC-WWC-SR which discusses the multicast routing is a sub-problem of the MRWAP-DC-WWC-SR. By the five-field notation for problem model in Chapter 3, the following eight problems will be introduced and the problem lattice is shown in Figure 4-2.

URWAP-DC-WWC-SR(*unicast, dynamic, no, delay, no, graph*)

URWAP-DC-WWC(*unicast, static, no, delay, no, graph*)

URWAP-DC-SR(*unicast, dynamic, no, delay, sparse, graph*)

URWAP-DC(*unicast, static, no, delay, sparse, graph*)

MRWAP-DC-WWC-SR(*multicast, dynamical, sparse & limited, delay, no, graph*)

MRWAP-DC-SR(*multicast, dynamical, sparse & limited, delay, sparse, graph*)

MRWAP-DC-WWC(*multicast, static, sparse & limited, delay, sparse, graph*)

MRWAP-DC(*multicast, static, sparse & limited, delay, sparse, graph*)

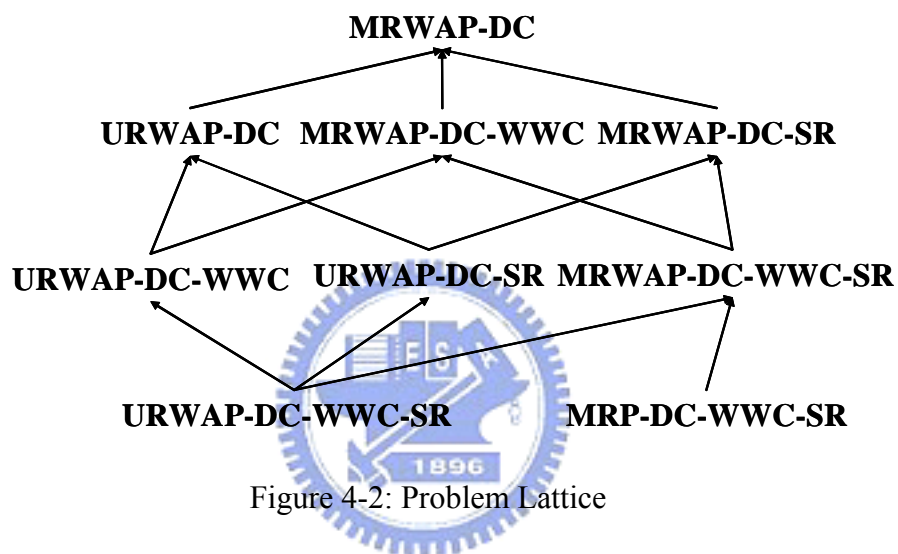


Figure 4-2: Problem Lattice

As seen in the problem lattice shown in Figure 4-2, the direction of each arrow represents the generalization of problem; for example, the MRWAP-DC-WWC is a generalized problem of the URWAP-DC-WWC and the MRWAP-DC-WWC-SR. In other words, both the URWAP-DC-WWC and the MRWAP-DC-WWC-SR are two specialized problems of the MRWAP-DC-WWC. The generalized problem considers more adjustable conditions than the specialized problem such that all specialized problems can be reduced from their generalization problems.

Following the study, the complexity of the MRWAP-DC makes the ILP formulation or meta-heuristics ACO and GA hard to find a feasible solution in an affordable execution time. For practical applications, finding a feasible solution in a timely manner is more important

than finding an optimal solution using an exceedingly long time. Therefore, two heuristics, Near-k-Shortest-Path-based heuristic (NKSPH) and Iterative Solution model (ISM), will be proposed to find approximate solutions efficiently. Different method could be applied to these problems, but the methods may be inefficient to solve these problems or the same formulation of the generalized problem could not be nicely applied to solve its specialized problems. Therefore, different methods are applied to solve different problems.

In this dissertation, five different solution methods, including ILP formulation, ACO, GA, NKSPH, and ISM are proposed and introduced. These methods will not be applied to all problems, but we choose suitable methods for each specific problem and use the same formulation to the corresponding specialized problems as far as possible. The relationship between the solution methods and the studied problems are shown in Table 4-1.

Table 4-1: The relations of applying methods and studied problems

	ILP	ACO	GA	NKSPH	ISM
MRP-DC-WWC-SR	Yes		Yes	Yes	Yes
URWAP-DC-SR	Yes	Yes		Yes	Yes
MRWAP-DC-WWC-SR	Yes			Yes	Yes
MRWAP-DC-WWC	Yes				

Note : “Yes” means that the problem will be examined by the method proposed in this dissertation

4.3 Simulation Scheme

To study the performance of the proposed methods in this dissertation, we designed and conducted a series of computational simulations. The scheme used in following simulations can be referred to Waxman [78]. In the scheme, there are n nodes randomly distributed over a

rectangular grid with integer coordinates. In a network topology generated for experiments, each directed link from node u to node v is associated with the probability function $P(u, v) = \lambda \exp(-p(u, v) / \gamma\delta)$, where $p(u, v)$ is the coordinate distance between u and v , δ is the maximum distance between each two nodes, and λ and γ are control variables selected from interval $(0, 1]$. In the probability function, a larger value of λ produces a network with higher link densities, and small value of γ increases the densities of short links relative to longer ones.

The communication cost from node u to node v is defined by taking the value of the distance between them on the grid. The transmission delay is randomly generated from interval $[1, 5]$. For each request $r = (s, d, \Delta)$, s and d are generated in a random manner. The delay bound Δ must be reasonable, for otherwise it is unlikely to find a feasible light-path or feasible light-tree. To generate a request with a reasonable delay bound, we use the value according to the minimum of transmission delays from the source to destinations found by applying Dijkstra's shortest path algorithm [19], and set Δ to be equal to χ times of the derived minimum transmission delay, where χ is a control parameter dictating the tightness between delay bound and minimum transmission delay. For example, $\chi = 3$ means that all delay bounds of requests were set to be $3 \times$ their minimum transmission delay. In our experiments, we set $\lambda = 0.7$, $\gamma = 0.7$, and the size of rectangular grid = 100 to simulate network with different numbers of nodes. Moreover, 15% of the nodes are equipped with wavelength conversion and 15% of nodes are equipped with light splitting capability for the case that the networks with a wavelength conversion capability and the case that the networks with a light splitting capability.

Chapter 5 ILP Formation for MRWAP-DC-WWC

Because the wavelength conversion delay and the wavelength conversion cost are ignored, a simplified ILP formulation is introduced for the special MRWAP-DC-WWC. Seven properties are proposed to show that the ILP formulation can solve the studied problem.

The notation used in our ILP formulation is given as follows.

e : some link in E ;

v : some node in V ;

λ : some wavelength in M ;

ζ : some destination belonging to some request;

$I(v)$: set of inbound edges to node v in V ;

$O(v)$: set of outbound edges from node v in V ;

$y_e^{x,\lambda,\zeta}$: binary variable indicating whether wavelength λ over link e is used for the light-path from s^x to ζ , where $\zeta \in D^x$; i.e., $y_e^{x,\lambda,\zeta} = 1$, if yes; 0, otherwise;

$\chi_e^{x,\lambda}$: binary variable indicating whether wavelength λ over link e is used for the request r^x , i.e., $\chi_e^{x,\lambda} = 1$, if yes; 0, otherwise;

$z^{x,\lambda}$: binary variable indicating whether wavelength λ is used for the request r^x , i.e., $z^{x,\lambda} = 1$, if yes; 0, otherwise.



5.1 Formulation

According to the formulation of the MRWAP-DC, the transmission delay is computed by Eq. (4-4). Because a light-tree may be viewed as a combination of a set of light-paths from the root to each destination contained in the light-tree, the transmission delay of a light-tree can be viewed as the maximum of the transmission delays on the light-paths. Therefore, variables $y_e^{x,\lambda,\zeta}$, $\chi_e^{x,\lambda}$, and $z^{x,\lambda}$ are used for computing the transmission delays on the light-paths from the source to destinations, the communication costs of light-tree for each request r^x , and the wavelength consumptions of light-forests. In this section, an ILP formulation will be developed.

The MRWAP-DC-WWC can be formulated as follows.

Objective function

$$\text{Minimize } \alpha \sum_{r^x \in R} \sum_{\lambda \in M} \sum_{e \in E} \chi_e^{x,\lambda} c(e) + \beta \sum_{r^x \in R} \sum_{\lambda \in M} z^{x,\lambda}$$

Subject to

$$(c1) \quad \forall r^x \in R, \forall \zeta \in D^x, \quad \sum_{\lambda \in M} \sum_{e \in I(s^x)} y_e^{x,\lambda,\zeta} - \sum_{\lambda \in M} \sum_{e \in O(s^x)} y_e^{x,\lambda,\zeta} = -1 \quad (\text{source constraints})$$

$$(c2) \quad \forall r^x \in R, \forall \zeta \in D^x, \quad \sum_{\lambda \in M} \sum_{e \in I(\zeta)} y_e^{x,\lambda,\zeta} - \sum_{\lambda \in M} \sum_{e \in O(\zeta)} y_e^{x,\lambda,\zeta} = 1 \quad (\text{target constraints})$$

$$(c3) \quad \forall r^x \in R, \forall \zeta \in D^x, \forall v \in V - \{s^x, \zeta\}, \forall \lambda \in M, \quad \sum_{e \in I(v)} y_e^{x,\lambda,\zeta} - \sum_{e \in O(v)} y_e^{x,\lambda,\zeta} = 0$$

(wavelength continuity constraints)

$$(c4) \quad \forall v \in V, \forall \lambda \in M, \quad \sum_{r^x \in R} \sum_{e \in I(v)} \chi_e^{x,\lambda} \leq 1 \quad (\text{input constraints})$$

$$(c5) \quad \forall r^x \in R, \forall v \in V, \forall \lambda \in M, \quad \sum_{e \in O(v)} \chi_e^{x,\lambda} \leq \theta(v) \quad (\text{capacity constraints})$$

$$(c6) \quad \forall r^x \in R, \forall \zeta \in D^x, \forall e \in E, \forall \lambda \in M, \quad \chi_e^{x,\lambda} \geq y_e^{x,\lambda,\zeta} \quad (\text{link usage constraints})$$

$$(c7) \quad \forall r^x \in R, \forall e \in E, \forall \lambda \in M \quad \chi_e^{x,\lambda} \leq z^{x,\lambda} \cdot w(e, \lambda) \quad (\text{wavelength usage constraints})$$

$$(c8) \quad \forall r^x \in R, \forall \zeta \in D^x, \forall \lambda \in M, \quad \sum_{e \in E} y_e^{x,\lambda,\zeta} d(e) \leq \Delta (\text{delay constraints})$$

$$(c9) \quad \forall r^x \in R, \forall \zeta \in D^x, \forall e \in E, \forall \lambda \in M, \quad y_e^{x,\lambda,\zeta} \in \{0, 1\} \text{ (0-1 constraints)}$$

$$(c10) \quad \forall r^x \in R, \forall e \in E, \forall \lambda \in M, \quad \chi_e^{x,\lambda} \in \{0, 1\} \text{ (0-1 constraints)}$$

$$(c11) \quad \forall r^x \in R, \forall \lambda \in M, \quad z^{x,\lambda} \in \{0, 1\} \text{ (0-1 constraints)}$$

In the above formulation, source constraints (c1), target constraints (c2), and wavelength continuity constraints (c3) ensure that a light-path using some specified wavelength from the source r^x to each destination $\zeta \in D^x$ can be found, and that each node except the source and destinations should pass a signal (information) from its input port to its output port. Since at most one signal can enter the input port using the same wavelength, we need input constraints (c4). Moreover, the number of split output signals must be bounded by the light splitting capacity. The capacity constraints (c5) ensure $\sum_{e \in O(v)} \chi_e^{x,\lambda}$, the number of outbound edges from v for routing the signals of the output port to other nodes using λ , should be smaller than or equal to $\theta(v)$. For some ζ , $y_e^{x,\lambda,\zeta} = 1$ implies that the wavelength λ in e will be assigned to reroute r^x (i.e., $\chi_e^{x,\lambda} = 1$) and that λ will be used to rerouted r^x (i.e., $z^{x,\lambda} = 1$). Therefore, the constraint $\chi_e^{x,\lambda} \geq y_e^{x,\lambda,\zeta}$ must be held by constraint set (c6). Constraint set (c7) ensures that only the link with unused wavelengths ($w(e, \lambda) = 1$) can be used. Therefore, the value of $\chi_e^{x,\lambda}$ may be equal to 1 only when $z^{x,\lambda} = 1$ and $w(e, \lambda) = 1$. According to the above discussion, $\bigcup_{y_e^{x,\lambda,\zeta}=1} e$ (i.e., $\bigcup_{\forall e \in E, y_e^{x,\lambda,\zeta}=1} e$) will be proved to be a light-path from s^x to ζ using wavelength λ or a null set, and its transmission delay can be represented by $\sum_{e \in E} y_e^{x,\lambda,\zeta} d(e)$, where $\bigcup_{y_e^{x,\lambda,\zeta}=1} e$ is an empty set represents λ will not be used to reroute r^x . Constraint set (c8) is used to specify the delay bound constraint for each request. Constraint set (c9) ~ (c11) limits all variables to be 1

or 0. For the first term in the objective function, $\chi_e^{x,\lambda} = 1$ means that wavelength λ over link e needs to reroute r^x , so $\sum_{e \in E} \chi_e^{x,\lambda} c(e)$ is the communication cost of using wavelength λ . Therefore, the first term represents the total communication cost of all requests. We know that the number of used wavelengths for a request can be represented by $\sum_{\lambda \in M} z^{x,\lambda}$. So, the second term in the objective function represents the total wavelength consumption for all requests.

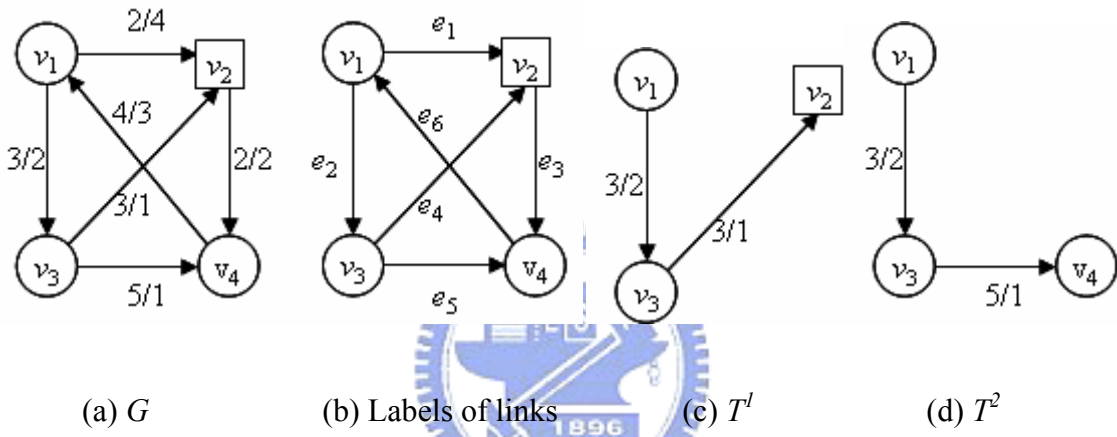


Figure 5.1: WDM network G

EXAMPLE 5.1: As shown in Figure 5-1(a), the graph G represents a network with 4 nodes ($n = 4$), 6 directed links, and v_2 is an *MC* node. Each link is associated with a value-pair “ a/b ”, where a and b are communication cost and transmission delay, and wavelengths λ_1 and λ_1 are deployed over each link. The labels of directed links in E are described in Figure 5-1 (b). Suppose there is only one request $r^1(v_1, \{v_2, v_4\}, 3)$. Both $y_i^{j,k}$, χ_i^j , and z^j are used to replace $y_{e_i}^{1,\lambda_j,v_k}$, $\chi_{e_i}^{1,\lambda_j}$, and z^{1,λ_j} for $1 \leq i \leq 6$, $1 \leq j \leq 2$, and $k = 2$ or 4 ($v_k \in \{v_2, v_4\}$) for short. The samples in the *ILP* formulation of the problem having 24 variables of type $y_{e_i}^{1,\lambda_j,v_k}$, 12 variables of type $\chi_{e_i}^{1,\lambda_j}$, and 2 variables of type z^{1,λ_j} could be described as follows.

$$\text{Minimize } \alpha(2\chi_1^1 + 3\chi_2^1 + 2\chi_3^1 + 3\chi_4^1 + 5\chi_5^1 + 4\chi_6^1 + 2\chi_1^2 + 3\chi_2^2 + 2\chi_3^2 + 3\chi_4^2 + 5\chi_5^2 + 4\chi_6^2) + \beta(z^1 + z^2)$$

The following samples in the ILP formulation describe each set of constraints.

$$(c1) : y_6^{1,2} + y_6^{2,2} - y_1^{1,2} - y_1^{2,2} - y_2^{1,2} - y_2^{2,2} = -1 \text{ (for } d_i = v_2)$$

$$(c2) : y_1^{1,2} + y_1^{2,2} + y_4^{1,2} + y_4^{2,2} - y_3^{1,2} - y_3^{2,2} = 1 \text{ (for } d_i = v_2)$$

$$(c3) : y_2^{1,2} - y_4^{1,2} - y_5^{1,2} = 0 \text{ (for } v = v_3, d_i = v_2, \lambda = \lambda_1)$$

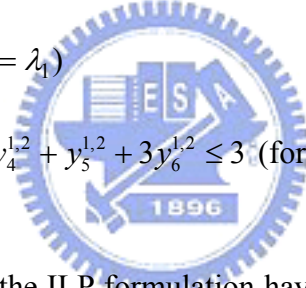
$$(c4) : \chi_1^1 + \chi_4^1 \leq 1 \text{ (for } v = v_2, \lambda = \lambda_1)$$

$$(c5) : \chi_4^1 + \chi_5^1 \leq 1 \text{ (for } v = v_3, \lambda = \lambda_1, \theta(v_3) = 1)$$

$$(c6) : \chi_2^1 \geq y_2^{1,2} \text{ (for } e = e_2, d_i = v_2, \lambda = \lambda_1)$$

$$(c7) : \chi_2^1 \leq z^1 \text{ (for } e = e_2, \lambda = \lambda_1)$$

$$(c8) : 4y_1^{1,2} + 2y_2^{1,2} + 2y_3^{1,2} + y_4^{1,2} + y_5^{1,2} + 3y_6^{1,2} \leq 3 \text{ (for } d_i = v_2, \lambda = \lambda_1)$$



Therefore, in the solution of the ILP formulation having 68 constraints (there are 2, 2, 8, 8, 8, 24, 12, and 4 constraints for sets from (c1) to (c8), respectively), variables except for $y_2^{1,2}, y_4^{1,2}, y_2^{2,4}, y_5^{2,4}, \chi_2^1, \chi_4^1, \chi_2^2$, and χ_5^2 are 0. The multicast cost of the solution is 16 (i.e., $1 \times (0 + 3 + 0 + 3 + 0 + 0 + 0 + 3 + 0 + 0 + 5 + 0) + 1 \times (1 + 1) = 16$) for $\alpha = 1$ and $\beta = 1$. ■

In the ILP formulation, since the numbers of variables, $y_e^{x,\lambda,\zeta}$, $\chi_e^{x,\lambda}$, and $z^{x,\lambda}$, are $m\gamma$, $m\gamma$, and γ , respectively. The total number of variables is $(q+1)m\gamma + \gamma$. Since the numbers of constraints (1) ~ (8) are q , q , $(n-2)q\gamma$, $n\gamma$, $n\gamma$, $m\gamma$, $m\gamma$, and $q\gamma$, respectively. The total number of constraints will be equal to $(nq+mq+2n+m-q)\gamma+2q$. Due to $m > n > q$, the complexities of variables and constraints are $O(m\gamma)$ and $O(nq\gamma)$, respectively.

Using the ILP formulation, the solution would be proved in following properties to be an optimal light-forest to solve the MRWAP-DC-WWC.

5.2 Proof of the ILP Formulation

In the previous formulations, “for each request” (i.e., “ $\forall r^x \in R$ ”) appears in each constraints set from (c1) to (c11). In the following description of properties, the representation of “for each request” is ignored for short.

PROPERTY 5.1: For each $\zeta \in D^x$, there exists one edge $e \in O(s^x)$ and one wavelength $\lambda \in M$, such that $y_e^{x,\lambda,d} = 1$ in each solution.

Proof: According to constraints (c1), we have $\sum_{\lambda \in M} \sum_{e \in O(s^x)} y_e^{x,\lambda,\zeta} \geq 1$ for each $\zeta \in D^x$ in the solution satisfying all constraints in the formulation. Therefore, there exists at latest one specific $e \in O(s^x)$ and one $\lambda \in M$, such that $y_e^{x,\lambda,d} = 1$ for each ζ . ■

PROPERTY 5.2: There exists only one light-path from s^x to ζ using some wavelength λ .

Proof: According to PROPERTY 5.1, $\forall \zeta \in D^x$, $\exists e_l \in O(s^x)$ and $\lambda \in M$, such that $y_{e_l}^{x,\lambda,\zeta} = 1$.

We assume that there is no light-path from s^x to ζ ; that is, the terminal node in the light-path is not ζ . Without loss of generality, suppose the light-path is represented by $\hat{p}^{x,\lambda} = \langle e_1, e_2, \dots,$

$e_{a-1} \rangle$, where e_j is the edge from v_j to v_{j+1} and $v_a \neq \zeta$. For all e_j , $1 \leq j \leq a-1$, we have $y_{e_j}^{x,\lambda,\zeta} = 1$.

Nevertheless, for $v_a \neq \zeta$ such that $\hat{p}^{x,\lambda}$ ends at v_a , we have $\sum_{e \in O(v_a)} y_e^{x,\lambda,\zeta} = 0$. It implies

$\sum_{e \in I(v_a)} y_e^{x,\lambda,\zeta} - \sum_{e \in O(v_a)} y_e^{x,\lambda,\zeta} \geq 1$ and (c3) is violated. A contradiction arises. Therefore, we may

conclude that $v_a = \zeta$ and $\hat{p}^{x,\lambda}$ is a light-path from s^x to ζ using wavelength λ . If more than one light-path exists for routing the data to ζ , then the objective function value is not minimum. The proof is complete. ■

According to PROPERTY 5.2, $\hat{p}^{x,\lambda}(s^x, \zeta)$ is a light-path when $y_{e_1}^{x,\lambda,\zeta} = 1$ for some e_1 , $e_1 \in O(s^x)$; otherwise, $\hat{p}^{x,\lambda}(s^x, \zeta)$ is a empty set. Thus, the union of the $e \in E$ with $y_e^{x,\lambda,\zeta} = 1$ will be $\hat{p}^{x,\lambda}(s^x, \zeta)$; that is, $\hat{p}^{x,\lambda}(s^x, \zeta) = \bigcup_{y_e^{x,\lambda,\zeta}=1} e$. Therefore, $\bigcup_{y_e^{x,\lambda,\zeta}=1} e$ may be a light-path or an empty set when λ is used or not used from s^x to ζ . There are exactly q^x non-empty light-paths among $\hat{p}^{x,\lambda}(s^x, \zeta)$ for all $\lambda \in M$ and for all $\zeta \in D^x$. Because the objective is to minimize the multicast cost, it can be seen that a loop never exists in $\hat{p}^{x,\lambda}(s^x, \zeta)$. To simplify our discussion, the directions of a light-path and a light-tree are ignored in the rest of this dissertation. Besides, $\hat{p}^{x,\lambda}(s^x, \zeta)$ represents a non-empty light-path.

PROPERTY 5.3: A graph obtained by merging all light-paths using the same wavelength is a light-tree.

Proof: Assume that $\bigcup_{\zeta \in D^x} \hat{p}^{x,\lambda}(s^x, \zeta)$, the union of all light-paths using λ , is not a tree; that is, there exists at least one cycle in $\bigcup_{\zeta \in D^x} \hat{p}^{x,\lambda}(s^x, \zeta)$. Suppose the cycle is formed by two different sub-light-paths between two specific nodes u and v in two light-paths. Therefore, there are two input signals using λ entering v which will cause input constraints (c4) to be violated. Moreover, the set of capacity constraints (c5) ensures that the number of split signals of each internal node in $\bigcup_{\zeta \in D^x} \hat{p}^{x,\lambda}(s^x, \zeta)$ is not greater than its light splitting capacity. Therefore,

$\bigcup_{\zeta \in D^x} \hat{p}^{x,\lambda}(s^x, \zeta)$ is a light-tree. ■

EXAMPLE 5.2: For the solution discussed in EXAMPLE 5.1, because $\chi_2^1, \chi_4^1, \chi_2^2$, and χ_5^2 are equal to 1, the light-forest consisting of two light-trees, $T^1 = \bigcup_{x_e^1=1} e = \{e_2, e_4\}$ and $T^2 = \bigcup_{x_e^2=1} e = \{e_2, e_5\}$ shown in Figure 5-1 (c) and (d) can be obtained. ■

For each $e \in E$, because $\chi_e^{x,\lambda} = 1$ will let $z^{x,\lambda} = 1$ satisfied and may increase the objective function value, $\chi_e^{x,\lambda}$ is set to 1 for satisfying (c6) and $z^{x,\lambda}$ is set to 1 for satisfying (c7) only; otherwise, the link usage constraints (c6) and wavelength usage constraints (c7) are violated or the objective function value is not minimum. Therefore, $\chi_e^{x,\lambda} = 1$ and $z^{x,\lambda} = 1$ when $y_e^{x,\lambda,\zeta} = 1$. Suppose that $T_\lambda^x = \bigcup_{\zeta \in D^x} \hat{p}^{x,\lambda}(s^x, \zeta)$. Since $\hat{p}^{x,\lambda}(s^x, \zeta) = \bigcup_{y_e^{x,\lambda,\zeta}=1} e$, $T_\lambda^x = \bigcup_{\zeta \in D^x} \bigcup_{y_e^{x,\lambda,\zeta}=1} e = \bigcup_{\chi_e^{x,\lambda}=1} e$ represents a light-tree and each link in T_λ^x will assign λ to route the request to each destination $\zeta \in V(T^{x,\lambda}) \cap D^x$.

PROPERTY 5.4: A feasible light-forest for each request will be found in each solution of the ILP formulation.

Proof:

According to PROPERTY 5.2 and 5.3, for each destination we can find a light-path using some wavelength and these light-paths using λ can form a light-tree $T_\lambda^x = \bigcup_{\chi_e^{x,\lambda}=1} e$. For the delay constraints (c8) ensuring $\forall \zeta \in D^x, \sum_{e \in E} y_e^{x,\lambda,\zeta} d(e) \leq \Delta$ and $\hat{p}^{x,\lambda}(s^x, \zeta) = \bigcup_{y_e^{x,\lambda,\zeta}=1} e$ being a light-path in T_λ^x , $d(\hat{p}^{x,\lambda}(s^x, \zeta)) \leq \Delta^x$ is obtained. $d(T^{x,\lambda}) = \max_{\zeta \in D^x \cap V(T^{x,\lambda})} d(\hat{p}^{x,\lambda}(s^x, \zeta)) \leq \Delta^x$ is

satisfied. Therefore, it can be seen that $\Gamma^x = \{(T_\lambda^x, h^{T_\lambda^x}) \mid T_\lambda^x = \bigcup_{\chi_e^{x,\lambda}=1} e, \forall \lambda \in M, \text{ for } T_\lambda^x, T_\lambda^x \text{ is not a empty set}\}$ is a feasible assigned light-forest for rerouting r^x , where $h^{T_\lambda^x} : E(T_\lambda^x) \rightarrow \{\lambda\}$ is a wavelength assignment function to assign all links in T_λ^x to use λ . ■

PROPERTY 5.5: The communication cost of the light-forest Γ^x is equal to $\sum_{\lambda \in M} \sum_{e \in E} \chi_e^{x,\lambda} \cdot c(e)$.

Proof: As defined above, $c(T_\lambda^x) = \sum_{e \in T_\lambda^x} c(e)$ and $T_\lambda^x = \bigcup_{\chi_e^{x,\lambda}=1} e$, we have $c(T_\lambda^x) = \sum_{\chi_e^{x,\lambda}=1} c(e) = \sum_{e \in E} \chi_e^{x,\lambda} \cdot c(e)$. According to PROPERTY 5.4, we may conclude the communication cost of Γ^x to be $\sum_{\lambda \in M} c(T_\lambda^x) = \sum_{\lambda \in M} \sum_{e \in E} \chi_e^{x,\lambda} \cdot c(e)$. ■

PROPERTY 5.6: The wavelength consumption is $\sum_{\lambda \in M} z^{x,\lambda}$.

Proof: By PROPERTY 5.3, if $\chi_e^{x,\lambda} = 1$, then link e will be contained in some light-tree using λ ($T_\lambda^x = \bigcup_{\chi_e^{x,\lambda}=1} e$). That is, the wavelength λ needs to be used to route the request (which implies $z^{x,\lambda} = 1$) for $\chi_e^{x,\lambda} = 1$ in (c7). That is, $\chi_e^{x,\lambda} = 1$ implies that T_λ^x is not a empty set and $z^{x,\lambda} = 1$. We know that $z^{x,\lambda}$ is set to 1 for satisfying (c7) only; otherwise, (c7) is violated or the value of the objective function is not a minimum. Therefore, the wavelength consumption is $\sum_{\lambda \in M} z^{x,\lambda}$. ■

PROPERTY 5.7: The objective function defined in the ILP formulation is equivalent to the multicast cost function.

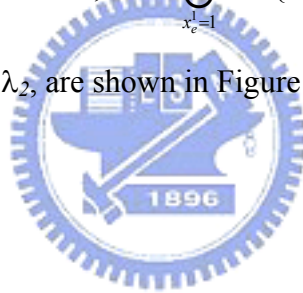
Proof: Suppose $\Gamma = \{\Gamma^x \mid 1 \leq x \leq N_R\}$ is a set of assigned light-forests. As defined above and PROPERTY 5.5 and 5.6, $f(\Gamma^x) = \alpha \cdot \sum_{\lambda \in M} \sum_{e \in E} \chi_e^{x,\lambda} c(e) + \beta \cdot \sum_{\lambda \in M} z^{x,\lambda}$. The property readily follows.

■

According to PROPERTY 5.4 and 5.7, each solution of the ILP formulation must be the optimal light-forest with minimum multicast cost and the solution can be mapped to be light-forests.

EXAMPLE 5.3: For the ILP formulation discussed in EXAMPLE 5.1, the solution will be found.

In the solution, $y_2^{1,2}, y_4^{1,2}, y_2^{2,4}, y_5^{2,4}, \chi_2^1, \chi_4^1, \chi_2^2, \chi_5^2, z^1$, and z^2 all have value 1 and the other variables have value 0. The multicast cost of the solution is $1 \times (0 + 3 + 0 + 3 + 0 + 0 + 0 + 3 + 0 + 0 + 5 + 0) + 1 \times (1 + 1) = 16$ when $\alpha = 1$ and $\beta = 1$. Because z^1 and z^2 are equal to 1, the light-forest consisting of two light-trees, $T_1 = \bigcup_{x_e^1=1} e = \{e_2, e_4\}$ using wavelength λ_1 and $T_2 = \bigcup_{x_e^2=1} e = \{e_2, e_5\}$ using wavelength λ_2 , are shown in Figure 5-1(c) and (d). ■



5.3 Experiments

In this dissertation, our work focuses on how to find the optimal light-forests such that nodes in the networks can be set up to route all requests. The approach used in this simulation to evaluate the performance of the ILP formulation is defined in Section 4.3. In order to reduce the elapsed execution time, we assume that there is one request in the request set R . In the experiments, eight types of networks were tested: 30 nodes ($n = 30$), 40 nodes ($n = 40$), 50 nodes ($n = 50$), 60 nodes ($n = 60$), 70 nodes ($n = 70$), 80 nodes ($n = 80$), 90 nodes ($n = 90$), and 100 nodes ($n = 100$), for each of which 60 different requests were randomly generated. Each 60 requests are categorized into 3 groups corresponding to 2 destinations ($q = 2$), 3 destinations ($q = 3$), and 4 destinations ($q = 4$). Because there is exact one request in R , $r = (s,$

D, Δ) used to represent the request, where s and the destinations in D were generated randomly.

The experiments consist of two parts: comparisons of different wavelength consumption ratios and performance assessment of the ILP. Our codes were implemented in C++ with the linear programming tool CPLEX on a computer with an Intel P4 2.4GHz CPU and 2GB RAM.

5.3.1 Comparisons of Different Wavelength Consumption Ratios

The multicast cost depends on different values of communication cost ratio (α) and wavelength consumption ratio (β). We shall study the effects of different values of β/α . The values of β/α are set to be 0.1 ($\alpha = 1$ and $\beta = 0.1$), 1 ($\alpha = 1$ and $\beta = 1$), 10 ($\alpha = 1$ and $\beta = 10$), 50 ($\alpha = 1$ and $\beta = 50$), and 100 ($\alpha = 1$ and $\beta = 100$). For each combination of β/α and χ , we route 5 requests in the network with 50 nodes ($n = 50$). We keep track of the average communication cost (CC), average wavelength consumption (ω), and average elapsed execution time (ET) (in seconds) for the 5 requests. The results are summarized in Table 5-1. Several observations can be made as follows:

- (1) For larger values of χ , the impact of different values of β/α on execution times is not significant. However, the elapsed execution times increase dramatically as the value of β/α grows and the delay bound becomes tighter (i.e., smaller values of χ). For example, for $\beta/\alpha = 0.1, 1, 10, 50,$ and 100 , ET is 53.87, 47.69, 67.79, 46.53, and 67.04 seconds when $\chi = 3.0$ and ET is 886.31, 1,652.01, 2,953.35, 2,644.63, and 3,533.81 seconds when $\chi = 1.2$. From the results, we know that elapsed execution time is unacceptably long when the delay bound becomes tighter. For example, when $\chi = 1.2$, the average execution time is more than 3,533.81 seconds for $\beta/\alpha = 100$. Therefore, the ILP formulation cannot solve the MRWAP-DC-WWC well when

the specified delay bound of a request is close to the minimum transmission delay.

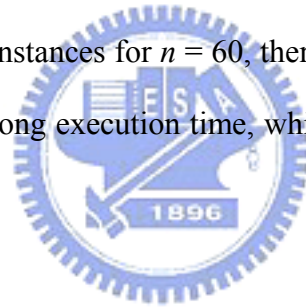
- (2) The ratio of β to α can be set properly such that a request can be routed by less communication cost or wavelength consumption. For example, the average wavelength consumption slightly decreases from 1.6 to 1.0 as the ratio increases rapidly from 0.1 to 100. For $\beta/\alpha = 0.1$ or 1 in $\chi = 3.0$, the communication cost $CC = 79.4$ is only determined by the wavelength consumption cost. For $\beta/\alpha = 100$ and $\chi = 3.0$, the wavelength consumption cost, the product of wavelength consumption and β ($\beta \times \omega = 100 \times 1.0$), is dominated by the communication cost. It is therefore reasonable to adjust the ratio to balance the load of each wavelength and each link. Moreover, dynamic adjustment may be a reasonable policy.

5.3.2 Performance Assessment of ILP

The test instances in Part (1) are again used here. We set $\alpha = 1$, $\beta = 1$, and $\chi = 1.2$. Table 5-2 shows the results from routing 60 requests in five networks ($n = 30, 40, 50, 60, 70$). For each combination of network types (n/m : the number of nodes/the number of edges) and the number of destinations (q), the first block shows the number of ILP variables ($\#NV$), the number of constraints ($\#NR$), the number of nonzero constraint entries ($\#NZ$) in these constraints. The first performance index we are interested in is the number of requests that are successfully solved ($\#Succ$). Performance indices resulted from the successfully solved instances include the minimum elapsed execution time (ET_{\min}), the maximum elapsed execution time (ET_{\max}), the average minimum elapsed execution time of the first 3 requests ($ET_{\min3}$), the average maximum elapsed execution time of the first 3 requests ($ET_{\max3}$), the average elapsed execution time of other requests (ET_{other}), and the average overall elapsed execution time (ET). From the numerical results, we have the following observations:

- (1) The elapsed execution time is more than 34 hours (for $ET_{\max} = 123,882.13$ seconds)

for some request ($q = 3$) in the network with 60 nodes ($n = 60$), and the ILP formulation cannot be used to solve the network with more than 70 nodes or requests with more than 3 destinations in a reasonable time. For example, $ET_{\min 3} = 73,268.66$ seconds in $n = 60$ and $q = 4$. In general, the execution time is proportional to the number of destinations and the number of nodes. For networks with more nodes, to fulfill the request becomes hard. That is, the number of solved requests decreases sharply when the number of nodes increases. For example, $\#Succ = 18$ for $n = 60$ and $q = 3$, but $\#Succ = 8$ for $n = 70$ and $q = 3$; furthermore, $\#Succ = 6$ for $n = 60$ and $q = 4$, but almost no request for $n = 70$ and $q = 4$ can be solved. This observation helps explain the phenomenon that $ET_{\max} = 123,882.13$ seconds for $n = 60$ and $q = 3$ is much greater than $ET_{\max} = 73,856.50$ seconds for $n = 70$ and $q = 3$. Amongst the 18 solved instances for $n = 60$, there could be some instances that have used an extraordinarily long execution time, while the 6 solved instances for $n = 70$ might be easier to solve.



5.4 Conclusion

In this chapter, the MRWAP-DC-WWC problem is explored first. To solve the problem to optimality, we formulate the problem in an ILP formulation to obtain the optimal solutions. Nevertheless, the optimal solutions are not obtained in an affordable execution time when the delay bound becomes tight, when the networks are equipped with more nodes, or when the numbers of destinations in the requests increase. Because the ILP formulation cannot solve instances of the MRWA-DC-WWC in large-scale networks, it is necessary to propose a heuristic to derive approximate solutions in a reasonable time, and two greedy heuristics will be introduced in the following chapters. Results from our computational study show that the ILP formulation can be used to solve the MRWA-DC-WWC in the networks of a limited

number of nodes. In our study, we find that how to determine adaptive communication cost ratio and wavelength consumption ratio could be an interesting topic.



Table 5-1: Experimental results for different β/α

χ	$\beta/\alpha = 0.1$			$\beta/\alpha = 1$			$\beta/\alpha = 10$			$\beta/\alpha = 50$			$\beta/\alpha = 100$		
	<i>CC</i>	<i>w</i>	<i>ET</i>	<i>CC</i>	<i>w</i>	<i>ET</i>	<i>CC</i>	<i>w</i>	<i>ET</i>	<i>CC</i>	<i>w</i>	<i>ET</i>	<i>CC</i>	<i>w</i>	<i>ET</i>
3.0	79.4	1.6	53.87	79.4	1.6	47.69	81.2	1.0	67.79	81.2	1.0	46.53	81.2	1.0	67.04
2.0	79.6	1.6	81.99	79.6	1.6	74.50	82.2	1.0	127.27	82.2	1.0	127.46	82.2	1.0	127.18
1.5	84.2	1.8	340.33	84.2	1.8	409.39	84.6	1.2	369.14	86.8	1.0	269.49	86.8	1.0	405.02
1.4	85.0	1.8	539.96	85.0	1.8	505.13	85.4	1.2	586.04	87.6	1.0	767.96	87.6	1.0	1,149.81
1.3	88.0	2.0	510.91	88.0	2.0	737.73	88.4	1.4	664.81	88.6	1.0	847.96	88.6	1.0	1,088.45
1.2	89.4	2.0	886.31	89.4	2.0	1,652.01	89.8	1.4	2,953.35	92.0	1.0	2,644.63	92.0	1.0	3,533.81

Table 5-2: Experimental results for different networks by using ILP

<i>n/m</i>	<i>q</i>	<i>#NV</i>	<i>#NR</i>	<i>#NZ</i>	<i>#Succ</i>	<i>ET_{min}</i>	<i>ET_{max}</i>	<i>ET_{min3}</i>	<i>ET_{max3}</i>	<i>ET_{other}</i>	<i>ET</i>
30/236	2	3,540	2,954	14,160	20	0.6	13,256.31	0.79	4,842.18	14.50	816.83
	3	4,720	4,281	20,060	19	0.2	2,355.52	0.85	1,560.76	24.28	263.18
	4	5,900	5,608	25,960	19	3.6	82,981.89	9.36	42,145.73	640.36	7,094.21
40/456	2	6,840	5,354	27,360	20	0.1	43,923.91	0.49	17,562.67	69.10	2,682.85
	3	9,120	7,831	38,760	16	8.5	27,638.92	18.39	16,583.96	469.58	3,406.43
	4	11,400	10,308	50,160	14	17.1	66,999.05	60.04	29,153.12	1,226.44	6,960.79
50/588	2	8,820	6,874	35,280	20	0.8	41,063.38	1.99	14,460.90	125.21	2,257.08
	3	11,760	10,061	49,980	19	2.3	11,014.69	18.68	10,014.08	819.63	2,144.92
	4	14,700	13,248	64,680	18	25.6	74,096.81	65.25	46,930.01	2,590.33	7,096.11
60/964	2	14,460	10,834	57,840	20	0.2	1,253.83	4.57	1,120.02	147.71	2,272.09
	3	19,280	15,951	81,940	18	10.7	123,882.13	53.64	89,361.25	4,710.74	18,042.98
	4	24,100	21,068	106,040	6	202.2	73,268.66	73,268.66	-	-	22,266.95
70/1124	2	16,860	12,634	67,440	18	2.1	84,512.84	10.81	84,512.84	1,018.16	5,488.86
	3	22,480	18,601	95,540	8	9.8	73,856.50	343.21	-	30,475.86	17,561.86
	4	-	-	-	-	-	-	-	-	-	-

Chapter 6 Ant Colony Optimization (ACO) for URWAP-DC-SR

For the MRWAP-DC-WWC, the ILP formulation could not find a feasible solution in an affordable execution time for routing more requests or the network with more nodes and links. In this chapter, we address a design of ant colony optimization (ACO), which is a meta-heuristic developed in the early 1990s [11][22]. The ACO uses natural metaphor inspired by the behavior of ant colonies to solve complex combinatorial optimization problems for finding near-optimal solutions. It has demonstrated significant strengths in many application areas, such as the traveling salesman problem, graph coloring problem, quadratic assignment problem, generalized minimum spanning tree problem, scheduling problems, and minimum weight vertex cover problem, just to name a few. The details of ACO design will be described later.

In [27][41][49][76], the ACO has been used to solve the URWAP-SR or MRWAP-DC-SR, but communication cost, wavelength conversion cost and delay bound were not incorporated in their studies. To the best of knowledge, Varela and Sinclair [76] is the first paper applying the ACO to cope with the URWAP. In their design, each ant keeps a tabu list of previously visited nodes to avoid dead-ends and cycles and to allow backtracking, where backtracking means that an ant will reversely pop out its previous location to alter the visited nodes when the already found partial tour is blocked. Garlick [27] extended the ACO applications to the dynamic URWAP-SR by using length and congestion information in

making routing decisions to reduce the possibility of network blocking in the tour-constructing phase. In [41], a survey and comparison on ACO applications to routing and load-balancing issues were presented. The authors also compare different ACO mitigating stagnation features, including evaporation, aging, pheromone smoothing and limiting, privileged pheromone laying and pheromone-heuristic control. Both [27] and [76] used the shortest path algorithm or minimum edges of paths to find a light-path as the heuristic ingredient of the ACO. However, delay bounds make the heuristic inappropriate for solving the URWAP-DC or MRWAP-DC. Moreover, the realistic concerns about wavelength conversion in switches lead to a more complicated problem. While the ACO has been applied to solve some specific case of the URWAP or MRWAP, no results have been reported on the complex, but realistic, problem involving delay bound, communication cost, and wavelength conversion simultaneously. In this dissertation, we shall design new ACO features to produce solutions to the studied problem, URWAP-DC-SR.

For a given (G, R) , the number of R is equal to 1 and the number of destination set is equal to 1. Therefore, (G, r) is used to distinguish the URWAP-DC-SR from the MRWAP-DC, where $r = (s, \zeta, \Delta)$. Unlike the MRWAP-DC which is to find a set of assigned light-forests, the URWAP-DC-SR is to find an assigned light-path for each unicast request. The assigned light-path is a wavelength-based light-path connection. For the network with N_w wavelengths, there are N_w wavelength links on each link. Every $e_{ij} \in E$ is a direct link from v_i to v_j , the wavelength link in e_{ij} represents as e_{ijl} , $1 \leq l \leq N_w$, and $c(e_{ijl}) = c(e_{ij})$ and $d(e_{ijl}) = d(e_{ij})$. Therefore, when a light-path includes two wavelength links e_{ijl} and $e_{jkl'}$ ($l \neq l'$), the switch j must provide the wavelength conversion capability such that the signal passing from e_{ij} to enter the input port of j using wavelength l can be transmitted to switch k from the output port of j using wavelength l' .

The URWAP-DC-SR seeks to find an assigned light-path P that consists of a sequence of connected wavelength links. Because e_{ijl} is contained in P means that the wavelength l in e_{ij}

will be used to route r , the wavelength assignment function h^P will equal to the function which set wavelength links in P to be 1 and the others to be 0. That is, when e_{ijl} is chosen in P ($e_{ijl} \in P$), $h^P(e_{ij}, l) = 1$ (i.e., $h_{ij}^P = l$). Therefore, P is a solution to (G, r) . Unlink the representation of light-forest in the MRWAP-DC, the solution in the URWAP-DC-SR is presented by P .

The communication cost $c(P)$ and the transmission delay $d(P)$ of P could be calculated as follows:

$$\text{communication cost : } c(P) = \sum_{e_{ijl} \in P} T_c(e_{ijl}), \quad (6-1)$$

$$\text{transmission delay : } d(P) = \sum_{e_{ijl} \in P} T_d(e_{ijl}), \quad (6-2)$$

where $T_c(e_{ijl}) = c_{ijl} + \sum_{e_{jkl'} \in P} \left\lceil \frac{|l-l'|}{N_W} \right\rceil \cdot \hat{c}_j$ and $T_d(e_{ijl}) = d_{ijl} + \sum_{e_{jkl'} \in P} \left\lceil \frac{|l-l'|}{N_W} \right\rceil \cdot \hat{d}_j$ represents the

communication cost and the transmission delay of passing signal from v_i through v_j using e_{ijl} .

For $e_{ijl} \in P$ and v_j is not a destination, there exists some node v_k and wavelength $\lambda_{l'}$ such that $e_{jkl'} \in P$; otherwise, P is not a light-path from the source to the destination. Although there is a

summary operator sigma in the notations $\sum_{e_{jkl'} \in P} \left\lceil \frac{|l-l'|}{N_W} \right\rceil \cdot \hat{c}_j$ and $\sum_{e_{jkl'} \in P} \left\lceil \frac{|l-l'|}{N_W} \right\rceil \cdot \hat{d}_j$, the sigma

operator is only contributed to represent the wavelength conversion cost and delay.

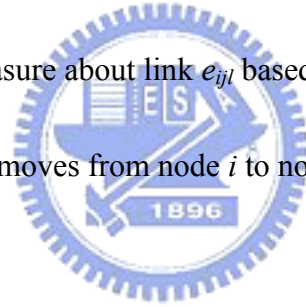
In the formula, they are computed based on the used wavelength links to reduce the complexity. An assigned light-path P is a feasible solution when the following three conditions are all satisfied:

- (1) the origin of P is s ;
- (2) the destination node of P is ζ ; and
- (3) the transmission delay of P is no greater than the delay bound (i.e., $d(P) \leq \Delta$).

In the rest of dissertation, for notational convenience assigned light-path and feasible solution will be replaced with light-path and solution if no confusion would arise.

In this section, we develop several features for the deployment of the ACO. The notation used in our ACO design is given in the following:

- b : number of ants;
- ξ : percentage of the b ants to distribute at the source and destination for each request;
- A^k : set of nodes accessible to ant k for r ;
- $\bar{\tau}_{ijl}$: initial pheromone on wavelength link e_{ijl} ;
- τ_{ijl} : dynamic desirability measure (pheromone intensity) on wavelength link e_{ijl} ;
- η_{ijl}^k : static desirability measure about link e_{ijl} based on a heuristic value for ant k ;
- p_{ijl}^k : probability that ant k moves from node i to node j using wavelength l (i.e., using e_{ijl});
- P^k : light-path traversed by ant k ;
- e : some wavelength link in a light-path.



6.1 Concept of the ACO

The ACO is a family of meta-heuristics that are inspired by the natural optimization mechanism conducted by real ants. The general framework of the ACO algorithm is shown Figure 6-1. In the ACO framework, the underlying environment for the ant colony to explore through is a directed graph, possibly with weights assigned to the edges. Therefore, a studied problem is usually represented by a weighted graph. The ACO system starts by distributing a set of artificial ants onto the graph. Each ant will construct a tour that corresponds to

a solution to the original problem. When all ants attain solutions, they share their information via pheromone and then another iteration commences. The process is repeated until some pre-specified criterion is satisfied. The optimization mechanism of the above-mentioned process is carried out by two important features: state transition rules and pheromone updating rules. A state transition rule is used for an ant to determine which node it will visit next (Step 2.2.1). The pheromone updating rules dynamically updates the pheromone intensities (or, in simple words, the degree of preference) on the edges (Steps 2.2.2 and 2.3). For general discussion on the philosophy and design detail, the reader is referred to [11][21].

Although the ACO has been applied to deal with the static UWRAP [76] and the dynamic URWAP [27], the proposed approach does not work for the URWAP-DC-SR. For example, the backtracking method for avoiding dead-ants in [27] and [76] cannot be used in the URWAP-DC-SR because transmission delays need to be taken into account. The existence of delay bounds of requests stipulates the global pheromone updating rule to test whether some ants arrive at the destinations successfully abiding by the delay bound. In this section, we propose and design an ACO algorithm that can produce approximate solutions with all realistic constraints incorporated.

ACO Framework

Step 1: Initialization

Step 2: Repeat

2.1. Each ant is positioned at some node

2.2. Repeat

2.2.1. Each ant moves to a next node according to the state transition rule

2.2.2. Apply the local pheromone updating rule

Until all ants have constructed a complete tour or encountered a dead-end

2.3. Apply the global pheromone updating rule

Until the stopping criterion is met

Figure 6-1: ACO framework

6.2 Initialization in the ACO

In our ACO design for the URWAP-DC-SR, the initialization phase includes two parts, dispatching ants to nodes and initializing pheromone on edges. For the first part, the trail of an ant can be viewed as a light-path. Therefore, it is reasonable to expect that an ant will start from the source and stop at the destination of the request. The optimal trail with the minimum communication cost indicates that it is an optimal routing light-path for the request. The strategy that finds trails from the source towards destination is called forward searching. On the other hand, backward searching refers to the strategy starting from the destination. Combining both strategies, we can let the ants begin their searching sessions randomly at either the source s or the destination ζ . In this dissertation, parameter ξ is given to adjust the percentage of b ants to be initially dispatched to s ; that is, the numbers of ants initially positioned at the source and the destination are $\xi \times b$ and $(1-\xi) \times b$, respectively.

For the second initialization task, applying some heuristics will provide informative guidance to determine the initial pheromone, and possibly shorten the time required by finding a near optimal or even optimal solution. Taking into account the objective of minimizing the total communication cost of routing a given request, the initial pheromone $\bar{\tau}_{ijl}$ on each wavelength link e_{ijl} is defined as:

$$\bar{\tau}_{ijl} = \begin{cases} 0, & \text{if } w_{ijl} = 0; \\ 1 + \frac{1}{\sum_{x \in V} \frac{1}{w_{ixl} c(e_{ixl})}}, & \text{if } w_{ijl} = 1. \end{cases} \quad (6-3)$$

Recall the definition of $w_{ijl} = 0$, which indicates that wavelength link e_{ijl} is infeasible. In order to prevent an ant from traversing an infeasible wavelength link, the initial pheromone of that wavelength link is set to 0 (Eq. (6-3)). If the wavelength link is viable, the initial

pheromone will be set as in Eq. (6-4) to let a wavelength link with less communication cost have a higher intensity of initial pheromone than the others.

6.3 State Transition Rule

The state transition rule presented in this dissertation features the following two aspects. First, unlike the ACO research [69] proposed solutions through the exploration of the power set of the vertex set, the solution in this paper will obtain a path of wavelength links from the source to the destination. Depending on whether the switches provide wavelength conversion or not, each ant can choose wavelength links using the same or different wavelengths to route data to next switch. That is, the solution to the URWAP-DC-SR can be viewed as a sequence of wavelength links, and the preference information (including pheromone intensity and local heuristic value) is deposited on the wavelength links. Secondly, the local heuristic used in most of previous research is static (that is, the value will not change during the optimization process). Shyu *et al.* [69] deployed a dynamic heuristic to reflect the situation that the access preference for a wavelength link changes over time depending on which wavelength links have been already selected. Based on the above concerns, we modify the state transition rule, which defines the probability that ant k at node i uses wavelength link e_{ijl} to route data to node j as follows:

$$p_{ijl}^k = \begin{cases} 1, & \text{if } q < q_0 \text{ and } \langle j, l \rangle = \arg \max_{x \in A^k, t \in W} \{ w_{ixt} \tau_{ixt} (\eta_{ixt}^k)^{\hat{\beta}} \}; & (6-5) \\ 0, & \text{if } q < q_0 \text{ and } \langle j, l \rangle \neq \arg \max_{x \in A^k, t \in W} \{ w_{ixt} \tau_{ixt} (\eta_{ixt}^k)^{\hat{\beta}} \} & (6-6) \\ \frac{w_{ijl} \tau_{ijl} (\eta_{ijl}^k)^{\hat{\beta}}}{\sum_{x \in A^k} w_{ixl} \tau_{ixl} (\eta_{ixl}^k)^{\hat{\beta}}}, & \text{if } q \geq q_0, & (6-7) \end{cases}$$

where A^k denotes the set of accessible nodes for ant k to visit such that no node can be traversed for more than once, τ_{ijl} is the dynamic desirability measure about the access to the

wavelength link e_{ijl} , η_{ijl}^k is the static desirability measure about the same wavelength link based on a problem-specific local heuristic, and $\hat{\beta}$ is the parameter controlling the relative significance between the two measures. Following the same line of reasoning in Eq. (6-3) and (6-4), w_{ijl} is added to each equation to guarantee that any infeasible wavelength link, i.e. $w_{ijl} = 0$, cannot be chosen.

The value of p_{ijl}^k can be determined according to a random number q drawn from the open interval (0, 1). If q is less than a specified threshold q_0 , the wavelength link e_{ijl} with the maximum product $w_{ixt} \tau_{ixt} (\eta_{ixt}^k)^{\hat{\beta}}$ is always selected (see Eq. (6-5) and Eq. (6-6)); otherwise, the wavelength link is selected according to the probability given in Eq. (6-7). That is, the state transition rule is a controlled trade-off scheme between the exploitation search and the exploration search of the problem space. Note that the probability value p_{ijl}^k depends on which wavelength link the ant uses to construct the light-path (trail) and that the transmission delay of the light-path is constrained by the delay bound. When the trail exceeds the delay bound, the value of η_{ijl}^k will be set to be 0 in Eq. (6-8), which will be defined in the next paragraph. Therefore, such a wavelength link will not be selected in Eq. (6-5) or Eq. (6-7). It thus highly suggests that the quality and feasibility of a solution depend on the wavelength links selected; that is, the communication cost and the transmission delay of light-path reflect the quality of the solution found by some ant and whether the solution is feasible or not, respectively. The value of variable τ_{ijl} , which gradually reflects the global preference for link e_{ijl} , is updated according to the quality of the final solution constructed at the end of each cycle. The details will be described in the next subsection. Local preference is incorporated to reflect the objective of communication cost minimization subject to transmission delay bounds. When there is no feasible solution constructed by the ant colony, determining a feasible solution, if exists, becomes more crucial. Therefore, the local preference needs to

reflect the status of whether a feasible solution has been found thus far. During the solution-finding session, if no feasible solution has been encountered, the local preference will center on how to find a feasible solution subject to the transmission delays of links; otherwise, the communication cost is taken into account. Therefore, the value of variable η_{ijl}^k , which evaluates the local preference of ant k for wavelength link e_{ijl} , changes dynamically and is given by

$$\eta_{ijl}^k = \begin{cases} 0, & \text{if } d(P^k) + T_d(e_{ijl}) > \Delta; & (6-8) \\ \frac{w_{ijl}}{T_d(e_{ijl})}, & \text{if no feasible solution has been found;} & (6-9) \\ \frac{w_{ijl}}{T_c(e_{ijl})}, & \text{otherwise,} & (6-10) \end{cases}$$

where η_{ijl}^k can be seen as the inverse of transmission delay (Eq. (6-9)) or the inverse of communication cost (Eq. (6-10)), depending on whether the ACO system has explored some feasible solution or not. In the sequel, the proposed dynamic local heuristic favors the feasible wavelength link that has either minimum transmission delays or minimum communication costs.

6.4 Pheromone Updating Rule

In the proposed system, we apply global and local pheromone updating rules as follows. First, at the end of each cycle we keep track of the best feasible solution P^{best} and the worst infeasible solution P^{worst} encountered by the colony. Our idea is to encourage the ants to follow links in P^{best} and avoid links in P^{worst} in the following cycles. This idea is realized by reinforcing (respectively, lessening) the intensities of the pheromone currently left on the

wavelength links in P^{best} (respectively, P^{worst}). The pheromone τ_{ijl} on wavelength link e_{ijl} is updated according to the following global updating rule:

$$\tau_{ijl} = (1 - \rho)\tau_{ijl} + \rho \sum_k \tau'_{ijl} - \rho \sum_k \tau''_{ijl}, \quad (6-11)$$

$$\text{where } \tau'_{ijl} = \begin{cases} \frac{1}{T_c(e_{ijl})} & \text{if } e_{ijl} \in P^{best}; \\ \frac{1}{\sum_{e \in P^{best}} T_c(e)} & \\ 0, & \text{otherwise,} \end{cases} \quad (6-12)$$

$$\text{and } \tau''_{ijl} = \begin{cases} \frac{T_d(e_{ijl})}{\sum_{e \in P^{worst}} T_d(e)} & \text{if } e_{ijl} \in P^{worst}; \\ 0, & \text{otherwise,} \end{cases} \quad (6-13)$$

Parameter $\rho \in (0, 1)$ simulates the evaporation rate of the pheromone intensity and enables the algorithm to reduce the significance of inferior links or forget the bad decisions previously made.

Secondly, we activate the local pheromone updating rule to shuffle the solutions and prevent from early convergence, i.e. all the ants make the same decisions. The local updating rule is performed at the end of each step when each ant selects a new wavelength link e_{ijl} . The pheromone intensity on link e_{ijl} is updated by

$$\tau_{ijl} = (1 - \varphi)\tau_{ijl} + \varphi\bar{\tau}_{ijl}, \quad (6-14)$$

where $\varphi \in (0, 1)$ is a parameter adjusting the current pheromone previously laid on e_{ijl} and $\bar{\tau}_{ijl}$ is the initial value of pheromone laid on e_{ijl} . Note that the local updating rule decreases the pheromone intensity on the link just visited by an ant and makes the selected links less attractive to other ants. The effect of the process will direct the exploration session of an ant

toward the links that have not yet been visited by other ants.

6.5 Stopping Criterion

Like other meta-heuristics, different types of stopping criteria can be used for the ACO. We cite the following four for the reader's interest.

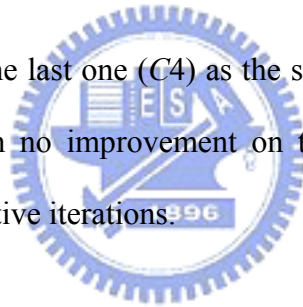
C1: The number of iterations is greater than a specified iteration limit;

C2: The execution time is longer than a specified CPU time limit;

C3: The averages of communication costs of ants in some consecutive iterations become the same; or

C4: The number of consecutive iterations in which no improvement attained on the incumbent solution is greater than a specified limit.

In this dissertation, we use the last one (C4) as the stopping criteria of our ACO design, i.e. the algorithm will stop when no improvement on the incumbent solution is achieved within a given number of consecutive iterations.



6.6 Computational Experiments

This research focuses on determining an assigned light-path of low communication cost such that switches in the network can be set up to reroute a request. To study the performance of the proposed approach, we designed and conducted a series of computational simulations. The scheme used in our simulation is defined in Section 4.3 or referred to Waxman [78]. As for the parameters setting of the ACO algorithm, preliminary experiments suggest $\xi = 0.5$, $\hat{\beta} = 1$, $\rho = 0.7$, $\varphi = 0.9$, and $q_0 = 0.5$.

The experiments include three parts: (1) introduction of transmission delays to the ILP formulation; (2) comparisons between the ACO algorithm and the ILP formulation; (3) investigation on the number of iterations exerted in the ACO algorithm. The codes were

written in C++. The platform is a personal computer with an Intel P4 2.4GHz CPU and 1GB RAM.

6.6.1 Introduction of Transmission Delays to the ILP Formulation

The ILP formulation used in the simulation to solve the MRWA-DC-WWC is adapted from that proposed in Chapter 5. It was implemented using the linear programming tool CPLEX. Three types of networks were tested: 40 switches ($n = 40$), 50 switches ($n = 50$), and 60 switches ($n = 60$), for each of which 200 different requests were randomly generated. Five wavelengths were provided for the networks. The delay bound was set to be χ times of the minimum transmission delay between the source and the destination in each request; for example, $\chi = 3.0$ means that all delay bounds of the 200 requests were set to be $3.0 \times$ the minimum transmission delays. For each combination of values of χ and network types, the elapsed run times, each of which are averaged over 200 requests, are summarized in Table 6-1. The experimental results suggest that the elapsed execution times increase sharply as the number of switches grows or the delay bound becomes tighter (i.e., smaller values of χ). For example, when $\chi = 1.1$, the average execution time is more than 1,380 seconds. Therefore, the ILP formulation cannot solve the MRWAP-DC-WWC well when the number of switches is more 70 or the specified delay bound of a request is close to the minimum transmission delay. They are similar to the results shown on Table 5-1 and will be used to compare with the experimental results attained from ACO method.

6.6.2 Comparisons between the ACO and the ILP Formulation

In this part, we define the stopping criterion for the ACO algorithm by terminating the execution when 2,000 iterations are reached or the incumbent value is equal to the optimal one. The same experiment settings were also applied to observe the solutions found by the

ACO approach. Experimental results are shown in Tables 6-2, 6-3 and 6-4 for the networks with 40 switches, 50 switches, and 60 switches. The solutions found by the ILP formulation are used as the baseline for comparisons. In these tables, the first two columns show the value of χ and the number of ants. We kept track of the scenarios of the ACO algorithm at iteration 1,000 and iteration 2,000. Recall that the algorithm will stop before entering later iterations if it encounters an optimal solution. Consider the major column entitled “1,000 Iterations”. Four sub-columns summarize the computational statistics at the end of the 1,000-th iteration.

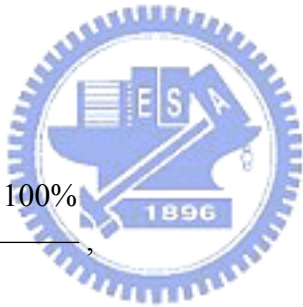
#Fea: number of requests for which feasible solutions are found;

#Opt: number of requests that are optimally solved;

Dev: average communication cost deviation of the found solutions from the optimal ones;

ET: average execution time.

Dev is defined as:

$$Dev = \frac{\sum_r \frac{c(P_r^{feas}) - c(P_r^{opt})}{c(P_r^{opt})} \times 100\%}{\#Fea}, \quad (6-15)$$


where $c(P_r^{feas})$ and $c(P_r^{opt})$ are the communication cost of the feasible solution P_r^{feas} found at the end of some iteration in the ACO algorithm and the communication cost of the optimal solution P_r^{opt} found by the ILP formulation, respectively. The second part reports the results at the end of 2,000 iterations. When the ACO algorithm finished processing 200 requests, we also keep track the number of requests that have been optimally solved (column *#Opt*). The sub-columns *Iter* and *ET* contain the average number of iterations and the average execution time required to produce these optimal solutions. The last major column *Non-Opt* records information on those test cases for which no optimal solutions were found. Sub-column *Iter* records the iteration at which the best feasible solution was encountered.

From the numerical results, we have the following observations:

- (1) For the case that χ has a tight value, it is guaranteed to find a feasible solution with fewer iterations or fewer ants. According the following three sets of experimental results in $\chi = 1.1$ of Table 6-3, (i) $\#Fea = 190$ in $b = 20$ and at the end of 1,000 iterations, (ii) $\#Fea = 200$ in $b = 100$ and at the end of 1,000 iterations, and (iii) $\#Fea = 198$ in $b = 20$ and at the end of 2,000 iterations, the first and the second sets of results demonstrate that more ants can benefit to find feasible solutions. Moreover, the first and the third sets of results demonstrate that execution with more cycles will have a higher probability of finding feasible solutions.
- (2) When a request with tight delay bound which is close to the minimum transmission delay, it seems to take less execution time because the ants were soon trapped and because a tight delay bound diminishes the number of viable wavelength links. Nevertheless, it may have a lower possibility of finding feasible solutions. For $\chi = 1.1$ and $\chi = 1.5$ in Table 6-3, we have $\#Fea = 198$ and $\#Fea = 200$, and $ET = 0.545$ seconds and 1.160 seconds at the end of 1,000 iterations for $b = 60$.
- (3) According to the comparisons from Tables 6-1 to 6-4, the execution time of the ACO algorithm is not sensitive to the change of the number of switches and the number of delay bound; but the time required by the ILP formulation highly depends on the change of the two numbers. For example, the ET values of the ACO for $\chi = 1.1$, $n = 40$, 50, and 60 are less than 1 seconds, but the corresponding ET values of the ILP are more than 32 seconds, 395 seconds, 1380 seconds. This demonstrates the robustness and superiority of the ACO algorithm for the URWAP-DC-SR.
- (4) Although a larger number of iterations and ants deployed in the ACO can reduce the communication cost of feasible solutions, the long execution time may be inefficient. The maximum average numbers of iterations to optimally solve optimally and non-optimally requests are 122 and 321 for $n = 40$, 123 and 440 for $n = 50$, and 169 and 420 for $n = 60$. Therefore, the Stopping criterion adopts the combination of that a

given number of consecutive iterations within which no improvement on solutions is attained and a given limited number of iterations, which may be more reasonable. The comparisons about the number of iterations will be discussed in the following section.

6.6.3 Comparisons of Iterations

This part is dedicated to investigating the number of consecutive iterations within which no improvement is attained on solution values. Experimental results are shown in Tables 6-5, 6-6 and 6-7 for different networks. Five groups with 200, 400, 600, 800, and 1000 consecutive iterations are studied, and statistics $\#Fea$, $\#Opt$, ET , and Dev are discussed. According to the experimental results, the stopping criterion can provide the performance with less execution time and approximated deviation in average. Summarizing the numerical results of Tables 6-5 to 6-7, the average experimental results of different χ values are shown in Table 6-8. The number of consecutive iterations could be determined by the response time, number of ants, and tightness degree (χ). Nevertheless, the deviation and execution time seem to be the reasonable factors. From different criteria, we make several observations:

- (1) The value of Dev decreases steadily for the increase of the number of ants and the increase of the number of iterations. For example, in Table 6-5, the values of Dev are 3.90%, 2.59%, 1.37%, 1.10%, and 0.74% for 200, 400, 600, 800, 1,000 iterations, and are 2.32%, 1.86%, 1.03%, 0.86%, and 0.93% for 40, 60, 80, 100, and 110 ants in $\chi = 3.0$. Nevertheless, it is not clear which factor's increase has impacts on the decrease the Dev values.
- (2) In Table 6-8, the ACO algorithm needed a larger number of consecutive iterations and fewer ants seem to provide lower deviation and to take longer run time than the ACO algorithms that used a smaller number of consecutive iterations and more ants in the networks with more switches. For example, in the case of $n = 60$, the values of

ET and Dev are 3.622 and 3.84% in $b = 40$ and 1,000 iterations, and 2.165 and 4.85% in $b = 130$ and 200 iterations. Nevertheless, in the case of $n = 40$, the values of ET and Dev are 1.751 and 3.57% in $b = 20$ and 1,000 iterations, and 1.502 and 2.93% in $b = 110$ and 200 iterations.

- (3) Although it is hard to determine the most appropriate ant population and the number of iterations, the colony with a greater number of ants seem to let the ACO algorithm to find feasible solution more efficiently. To route the requests with less deviation and high success probability, we suggest that the ant population may be set as the number of nodes in the network plus 20, and that the value of consecutive iterations is set as large as possible.

6.7 Conclusion

In this study, we have designed and implemented an ACO approach for solving the URWAP-DC-SR. To adjust the ACO approach to meet the specific characteristics of the studied problem, a wavelength-link-based graph has been constructed for the ants to traverse on. The effectiveness and robustness of the ACO approach have been examined through extensive experiments. The experimental results have clearly evinced that our proposed ACO algorithm can find approximate solutions with average deviations of less than 4% from the optimal ones with an average elapsed execution time only about 0.1% of that required by an ILP formulation. Moreover, the ACO algorithm still works well in solving the URWAP-DC for large-scale networks, for which the IP formulation fails to provide optimal solutions in a reasonable time.

The purpose of this study is not to address superiority over other meta-heuristic in solving the URWAP-DC-SR. Our focus is set on addressing the applicability as well as the capability of the ACO algorithm in dealing with the URWAP-DC. Our study will not only extend the application areas of the ACO approach but also suggest a new viable

method for coping with the complex optimization problems arising from the WDM domain.

Table 6-1: Average execution time (sec.) for different χ values and different networks

χ	$n = 40$	$n = 50$	$n = 60$
3.0	0.216	0.369	1.987
2.0	0.583	0.682	3.496
1.5	1.463	1.691	9.889
1.4	2.245	4.192	19.382
1.3	3.664	9.758	55.751
1.2	4.679	10.746	79.421
1.1	32.789	395.394	1380.914



Table 6-2: Experimental requests of 200 requests routed in 40 nodes

χ	b	1,000 Iterations				2,000 Iterations				Opt		Non-Opt	
		#Fea	#Opt	Dev	ET	#Fea	#Opt	Dev	ET	Iter	ET	Iter	ET
3.0	20	200	156	1.49%	0.654	200	169	0.70%	1.045	205	0.463	619	4.218
	40	200	169	0.71%	0.992	200	179	0.35%	1.503	177	0.763	516	7.807
	60	200	178	0.37%	1.132	200	184	0.26%	1.604	128	0.767	390	11.236
	80	200	178	0.33%	1.467	200	182	0.26%	2.146	121	0.980	434	13.944
	100	200	185	0.26%	1.190	200	190	0.15%	1.655	92	0.854	344	16.870
2.0	20	200	153	4.39%	0.558	200	163	3.74%	0.906	229	0.403	429	3.124
	40	200	163	2.37%	0.826	200	168	1.46%	1.312	149	0.488	564	5.641
	60	200	164	1.45%	1.046	200	174	1.00%	1.668	166	0.714	264	8.052
	80	200	168	1.65%	1.341	200	173	1.29%	2.082	138	0.820	390	10.171
	100	200	172	0.98%	1.346	200	180	0.62%	2.149	133	0.941	405	13.021
1.5	20	199	152	3.73%	0.438	200	161	2.31%	0.748	174	0.235	513	2.868
	40	199	161	2.34%	0.699	200	170	1.69%	1.090	160	0.406	357	4.964
	60	200	160	2.69%	0.943	200	169	1.94%	1.534	149	0.521	242	7.055
	80	200	172	1.81%	0.985	200	178	1.11%	1.558	123	0.589	389	9.393
	100	200	172	2.21%	1.104	200	177	1.97%	1.817	96	0.561	253	11.483
1.4	20	200	152	5.43%	0.420	200	155	4.28%	0.732	130	0.170	342	2.667
	40	200	168	2.54%	0.569	200	171	2.40%	0.943	112	0.269	392	4.917
	60	200	159	3.20%	0.952	200	167	2.23%	1.581	156	0.559	364	6.753
	80	200	171	1.95%	0.882	200	179	1.25%	1.420	117	0.556	335	8.779
	100	200	172	2.31%	1.088	200	177	1.53%	1.804	110	0.605	337	11.031
1.3	20	200	159	4.79%	0.352	200	168	3.01%	0.593	163	0.210	270	2.604
	40	200	173	2.71%	0.432	200	180	1.58%	0.682	119	0.286	201	4.247
	60	200	171	2.93%	0.669	200	177	2.62%	1.062	115	0.399	142	6.167
	80	200	170	2.57%	0.818	200	175	2.10%	1.349	79	0.353	577	8.316
	100	200	184	0.76%	0.757	200	186	0.60%	1.138	86	0.472	184	9.987
1.2	20	200	164	4.19%	0.349	200	169	3.59%	0.549	132	0.182	189	2.549
	40	200	167	3.39%	0.465	200	177	2.14%	0.764	135	0.301	366	4.331
	60	200	170	2.93%	0.645	200	175	2.34%	1.077	102	0.334	301	6.276
	80	200	173	2.04%	0.783	200	176	1.87%	1.303	81	0.346	173	8.325
	100	200	178	2.10%	0.845	200	181	1.03%	1.346	83	0.435	267	10.026
1.1	20	199	180	1.96%	0.204	199	182	1.90%	0.318	69	0.102	69	2.502
	40	199	178	2.07%	0.315	200	182	1.75%	0.517	72	0.157	328	4.161
	60	200	184	1.85%	0.382	200	188	1.21%	0.596	72	0.243	52	6.127
	80	200	185	1.56%	0.430	200	188	1.34%	0.694	62	0.259	43	7.515
	100	200	186	1.57%	0.464	200	191	0.77%	0.737	68	0.328	181	9.421

Table 6-3: Experimental requests of 200 requests routed in 50 nodes

χ	b	1,000 Iterations				2,000 Iterations				Opt		Non-Opt	
		#Fea	#Opt	Dev	ET	#Fea	#Opt	Dev	ET	Iter	ET	Iter	ET
3.0	20	200	169	0.98%	0.614	200	174	0.61%	1.010	139	0.367	832	5.313
	40	200	172	0.73%	0.922	200	181	0.46%	1.455	122	0.585	572	9.743
	60	200	178	0.49%	1.204	200	185	0.31%	1.868	123	0.883	739	14.014
	80	200	181	0.46%	1.408	200	185	0.30%	2.223	99	0.979	694	17.562
	100	200	181	0.44%	1.618	200	185	0.26%	2.543	80	0.907	835	22.724
2.0	20	200	152	3.78%	0.621	200	156	2.21%	1.089	129	0.256	706	4.045
	40	200	156	2.21%	1.017	200	165	1.28%	1.697	166	0.609	747	6.825
	60	200	164	1.50%	1.342	200	174	0.79%	2.009	168	0.880	515	9.561
	80	200	167	1.89%	1.516	200	176	0.87%	2.397	145	0.958	625	12.951
	100	200	170	0.94%	1.762	200	175	0.80%	2.825	109	0.907	378	16.248
1.5	20	200	143	8.61%	0.618	200	149	6.71%	1.084	142	0.247	506	3.527
	40	200	151	5.50%	0.948	200	158	4.67%	1.655	143	0.418	367	6.308
	60	200	154	3.88%	1.160	200	161	2.95%	2.066	120	0.492	535	8.566
	80	200	152	3.82%	1.665	200	159	2.69%	2.875	123	0.740	461	11.154
	100	200	155	2.98%	1.865	200	161	2.48%	3.271	108	0.785	489	13.532
1.4	20	200	142	8.71%	0.587	200	153	7.02%	1.013	171	0.286	274	3.379
	40	200	156	6.11%	0.843	200	161	4.11%	1.457	123	0.369	521	5.950
	60	200	162	4.98%	1.024	200	170	3.56%	1.732	144	0.541	427	8.479
	80	200	165	4.27%	1.362	200	172	3.03%	2.183	145	0.788	422	10.752
	100	200	165	3.51%	1.379	200	170	2.44%	2.452	85	0.574	411	13.091
1.3	20	198	155	6.75%	0.478	199	164	5.80%	0.826	161	0.285	231	3.287
	40	200	158	7.11%	0.751	200	166	5.24%	1.291	140	0.393	519	5.675
	60	199	162	5.16%	0.960	199	172	3.87%	1.649	157	0.648	322	7.800
	80	200	170	3.74%	1.004	200	175	2.67%	1.688	112	0.545	299	9.692
	100	200	164	4.00%	1.274	200	169	2.75%	2.265	90	0.510	435	11.830
1.2	20	197	154	6.80%	0.432	199	165	5.63%	0.739	163	0.259	269	3.003
	40	200	161	5.82%	0.622	200	171	4.72%	1.018	137	0.328	304	5.084
	60	199	166	5.74%	0.772	200	173	4.34%	1.333	106	0.369	380	7.514
	80	200	169	5.11%	1.042	200	173	4.66%	1.743	111	0.512	294	9.625
	100	200	171	4.23%	1.109	200	174	3.76%	1.927	77	0.450	188	11.813
1.1	20	190	169	4.37%	0.347	198	178	4.61%	0.547	150	0.247	370	2.972
	40	198	170	4.56%	0.488	199	176	3.68%	0.830	106	0.279	272	4.878
	60	198	182	2.43%	0.545	198	187	1.52%	0.847	102	0.426	199	6.908
	80	199	181	2.03%	0.660	199	183	1.86%	1.074	70	0.378	61	8.571
	100	200	185	2.73%	0.620	200	186	2.39%	0.978	53	0.299	188	10.003

Table 6-4: Experimental requests of 200 requests routed in 60 nodes

χ	b	1,000 Iterations				2,000 Iterations				Opt		Non-Opt	
		#Fea	#Opt	Dev	ET	#Fea	#Opt	Dev	ET	Iter	ET	Iter	ET
3.0	20	200	152	1.40%	1.147	200	159	1.04%	1.847	220	0.681	731	6.371
	40	200	156	0.93%	1.793	200	168	0.62%	2.870	236	1.262	597	11.313
	60	200	158	0.92%	2.288	200	168	0.64%	3.751	168	1.373	592	16.237
	80	200	168	1.11%	2.550	200	177	0.37%	4.020	161	1.704	621	21.847
	100	200	174	0.47%	2.643	200	181	0.32%	4.098	133	1.768	603	26.297
2.0	20	200	133	4.83%	1.015	200	144	3.12%	1.768	237	0.597	668	4.778
	40	200	141	3.65%	1.473	200	150	1.94%	2.579	187	0.802	500	7.910
	60	200	146	2.51%	2.018	200	157	1.92%	3.370	194	1.190	411	11.331
	80	200	149	2.61%	2.445	200	159	1.90%	4.172	185	1.479	475	14.618
	100	200	150	1.90%	2.961	200	164	1.52%	4.781	213	2.058	382	17.188
1.5	20	200	140	7.15%	0.869	200	151	5.66%	1.416	223	0.491	465	4.268
	40	200	149	6.18%	1.246	200	159	4.77%	2.020	210	0.730	468	7.025
	60	200	158	4.21%	1.483	200	164	3.35%	2.464	165	0.831	498	9.905
	80	200	158	3.52%	1.803	200	169	2.16%	2.872	177	1.140	441	12.311
	100	200	163	2.78%	1.958	200	172	1.72%	3.085	179	1.273	380	14.214
1.4	20	200	139	7.61%	0.784	200	152	5.95%	1.298	252	0.500	315	3.828
	40	200	150	6.33%	1.081	200	160	3.37%	1.801	199	0.639	522	6.445
	60	200	153	4.66%	1.506	200	161	4.21%	2.460	188	0.841	302	9.144
	80	200	165	3.51%	1.473	200	174	2.27%	2.323	173	0.981	377	11.307
	100	200	156	3.84%	2.065	200	171	1.98%	3.367	210	1.498	789	14.388
1.3	20	199	143	6.40%	0.550	200	155	4.31%	0.927	187	0.297	330	3.098
	40	200	148	6.12%	0.718	200	158	4.18%	1.221	154	0.352	252	4.488
	60	200	159	4.59%	0.848	200	164	3.86%	1.453	130	0.418	376	6.167
	80	200	165	2.74%	1.073	200	172	2.05%	1.657	156	0.640	237	7.907
	100	200	172	3.03%	1.083	200	178	2.21%	1.682	143	0.704	277	9.594
1.2	20	199	146	6.61%	0.458	200	156	5.01%	0.773	163	0.227	336	2.711
	40	199	158	5.40%	0.636	200	167	2.81%	1.020	151	0.352	570	4.402
	60	200	160	4.18%	0.822	200	167	2.74%	1.381	142	0.447	367	6.111
	80	200	162	3.69%	0.975	200	165	3.11%	1.703	98	0.381	340	7.936
	100	200	165	3.79%	1.135	200	169	2.59%	1.951	112	0.568	425	9.493
1.1	20	193	167	2.77%	0.366	196	170	2.74%	0.595	127	0.251	225	2.548
	40	196	170	2.77%	0.506	198	174	2.39%	0.827	113	0.317	258	4.239
	60	196	170	2.74%	0.657	197	177	1.96%	1.076	114	0.469	196	5.748
	80	198	178	2.31%	0.650	199	181	2.08%	1.030	77	0.360	262	7.415
	100	198	179	1.65%	0.763	199	189	1.02%	1.146	132	0.693	106	8.936

Table 6-5: Results of 200 requests routed in 40 nodes

χ	b	200 Iterations				400 Iterations				600 Iterations				800 Iterations				1,000 Iterations			
		#Fea	#Opt	ET	Dev	#Fea	#Opt	ET	Dev	#Fea	#Opt	ET	Dev	#Fea	#Opt	ET	Dev	#Fea	#Opt	ET	Dev
3.0	20	200	137	0.543	3.90%	200	149	1.118	2.59%	200	151	1.595	1.37%	200	166	2.175	1.10%	200	163	2.689	0.74%
	40	200	149	0.963	2.32%	200	165	1.978	1.12%	200	165	2.897	0.77%	200	174	3.871	0.48%	200	173	4.543	0.49%
	60	200	155	1.381	1.86%	200	168	2.695	0.87%	200	174	4.036	0.55%	200	173	5.191	0.56%	200	184	6.561	0.26%
	80	200	165	1.803	1.03%	200	171	3.546	0.68%	200	171	5.197	0.54%	200	180	6.719	0.38%	200	179	8.181	0.34%
	100	200	165	2.280	0.86%	200	177	4.101	0.46%	200	174	6.237	0.45%	200	187	8.174	0.22%	200	185	10.242	0.31%
	110	200	165	2.444	0.93%	200	172	4.688	0.63%	200	182	6.907	0.32%	200	183	8.859	0.31%	200	184	11.360	0.26%
2.0	20	200	126	0.441	10.21%	200	134	0.846	5.03%	200	148	1.232	3.90%	200	157	1.661	4.64%	200	146	2.001	3.46%
	40	200	142	0.736	3.78%	200	147	1.461	3.78%	200	153	2.148	2.30%	200	157	2.805	2.89%	200	163	3.565	2.01%
	60	200	143	1.049	4.10%	200	155	2.005	2.14%	200	163	3.060	2.72%	200	167	4.043	1.35%	200	172	4.815	1.24%
	80	200	146	1.333	5.81%	200	168	2.662	1.53%	200	165	3.874	1.29%	200	169	5.246	1.47%	200	175	6.442	0.77%
	100	200	154	1.594	2.11%	200	164	3.133	2.12%	200	167	4.704	1.23%	200	169	6.220	1.89%	200	176	7.652	1.05%
	110	200	156	1.763	2.62%	200	164	3.616	1.58%	200	170	5.091	1.38%	200	172	6.824	0.94%	200	174	8.560	0.85%
1.5	20	200	136	0.331	9.47%	200	137	0.674	9.09%	200	145	0.979	7.83%	200	147	1.381	5.71%	200	158	1.682	4.61%
	40	200	143	0.611	6.40%	200	145	1.135	4.74%	200	157	1.720	4.23%	200	160	2.259	3.40%	200	164	2.897	3.06%
	60	200	149	0.850	6.97%	200	158	1.618	3.57%	200	165	2.393	3.09%	200	167	3.256	1.93%	200	167	4.086	2.61%
	80	200	154	1.051	4.40%	200	169	2.156	1.95%	200	164	3.103	4.17%	200	170	4.231	2.70%	200	172	5.013	1.86%
	100	200	155	1.345	3.85%	200	163	2.534	2.99%	200	171	3.889	2.15%	200	176	5.105	1.22%	200	175	6.397	1.70%
	110	200	159	1.458	3.70%	200	167	2.877	2.93%	200	168	4.258	1.57%	200	172	5.563	1.80%	200	174	6.913	1.36%
1.4	20	200	132	0.339	9.80%	200	141	0.662	5.91%	200	150	0.936	6.08%	200	157	1.281	4.17%	200	154	1.592	4.48%
	40	200	142	0.587	6.54%	200	148	1.081	5.31%	200	156	1.656	3.84%	200	160	2.259	2.59%	200	164	2.592	2.63%
	60	200	150	0.783	5.15%	200	158	1.561	4.70%	200	162	2.372	2.92%	200	162	3.008	2.94%	200	172	3.797	2.29%
	80	200	157	0.972	5.01%	200	166	1.999	1.91%	200	165	2.960	1.50%	200	165	3.884	2.27%	200	172	4.824	1.93%

Table 6-5: (Continued.)

	100	200	156	1.262	3.51%	200	160	2.471	3.33%	200	171	3.703	1.56%	200	172	4.783	1.77%	200	173	6.138	1.71%
	110	200	157	1.334	4.21%	200	162	2.715	2.49%	200	174	3.945	1.50%	200	168	5.119	2.64%	200	173	6.402	1.95%
1.3	20	197	140	0.313	7.44%	197	154	0.607	5.49%	199	158	0.915	5.16%	199	162	1.210	4.43%	200	163	1.513	4.94%
	40	199	151	0.537	5.99%	200	162	1.120	3.81%	200	167	1.549	3.56%	200	165	2.092	3.39%	200	168	2.465	3.53%
	60	198	156	0.742	4.89%	200	164	1.533	3.83%	200	169	2.167	2.85%	200	178	2.863	1.72%	200	175	3.554	2.83%
	80	200	157	0.919	4.16%	200	174	1.901	2.08%	200	173	2.826	2.79%	200	176	3.667	1.49%	200	175	4.563	1.40%
	100	200	169	1.188	3.25%	200	167	2.295	2.71%	200	172	3.333	2.79%	200	177	4.410	1.99%	200	177	5.701	1.82%
	110	200	168	1.236	3.41%	200	175	2.570	1.79%	200	174	3.769	3.25%	200	172	4.828	2.22%	200	183	6.239	1.10%
1.2	20	194	145	0.292	7.07%	200	156	0.625	5.02%	200	162	0.877	4.43%	200	158	1.108	4.91%	200	162	1.456	4.53%
	40	197	153	0.524	6.00%	200	163	0.960	4.10%	200	173	1.452	2.84%	200	165	1.969	4.41%	200	174	2.491	4.12%
	60	198	158	0.760	4.12%	200	170	1.400	3.26%	200	167	2.050	3.45%	200	170	2.727	3.35%	200	175	3.375	1.96%
	80	199	166	0.858	4.29%	200	170	1.845	3.51%	200	173	2.598	3.19%	200	171	3.502	3.10%	200	179	4.255	1.55%
	100	200	165	1.090	3.78%	200	170	2.139	3.41%	200	173	3.261	2.12%	200	179	4.266	1.99%	200	175	5.342	1.89%
	110	200	170	1.217	3.28%	200	172	2.361	3.35%	200	176	3.529	2.39%	200	173	4.694	1.62%	200	180	5.623	1.52%
1.1	20	192	156	0.265	3.78%	193	169	0.515	2.88%	194	166	0.779	2.88%	197	168	1.067	2.53%	198	175	1.322	2.23%
	40	198	174	0.495	2.69%	194	168	0.875	2.67%	199	178	1.389	2.64%	200	181	1.822	2.22%	197	179	2.139	1.98%
	60	196	170	0.655	2.27%	199	180	1.229	1.91%	197	172	1.866	2.61%	200	180	2.493	2.17%	199	182	3.082	1.85%
	80	197	170	0.815	2.19%	197	175	1.658	2.10%	198	180	2.427	1.72%	199	180	3.210	1.77%	200	181	3.967	1.98%
	100	199	178	1.009	2.01%	200	178	1.934	2.30%	199	182	2.996	1.90%	200	184	3.885	1.90%	200	187	4.835	1.58%
	110	197	173	1.065	2.36%	200	181	2.182	1.86%	200	187	3.204	1.42%	200	182	4.177	1.55%	200	185	5.234	1.52%

Table 6-6: Results of 200 requests routed in 50 nodes

χ	b	200 Iterations				400 Iterations				600 Iterations				800 Iterations				1,000 Iterations			
		#Fea	#Opt	ET	Dev	#Fea	#Opt	ET	Dev	#Fea	#Opt	ET	Dev	#Fea	#Opt	ET	Dev	#Fea	#Opt	ET	Dev
3.0	30	200	157	0.886	1.59%	200	168	1.683	1.15%	200	167	2.542	0.91%	200	175	3.398	0.60%	200	178	4.174	0.46%
	50	200	167	1.461	1.24%	200	172	2.551	0.84%	200	182	4.022	0.41%	200	180	5.141	0.49%	200	184	6.338	0.38%
	70	200	165	1.709	0.94%	200	175	3.410	0.87%	200	181	5.104	0.40%	200	186	6.960	0.33%	200	183	8.132	0.31%
	90	200	167	2.169	0.90%	200	179	4.271	0.54%	200	183	6.267	0.47%	200	184	8.364	0.37%	200	184	10.322	0.37%
	110	200	176	2.663	0.94%	200	182	5.048	0.42%	200	184	7.589	0.28%	200	188	9.862	0.28%	200	188	12.159	0.31%
	120	200	179	2.751	0.59%	200	178	5.430	0.42%	200	189	8.310	0.21%	200	185	10.731	0.38%	200	187	13.177	0.30%
2.0	30	200	144	0.686	5.90%	200	162	1.308	4.57%	200	165	1.977	2.45%	200	166	2.652	1.60%	200	171	3.264	1.82%
	50	200	158	1.023	2.61%	200	165	2.058	2.75%	200	178	2.988	0.87%	200	176	3.808	1.43%	200	178	4.849	0.75%
	70	200	163	1.335	3.13%	200	174	2.703	1.61%	200	171	3.842	1.26%	200	180	5.187	0.56%	200	177	6.431	1.24%
	90	200	168	1.700	2.59%	200	172	3.410	1.48%	200	180	4.784	0.59%	200	178	6.189	0.70%	200	181	7.706	0.62%
	110	200	168	2.036	2.15%	200	174	3.869	1.55%	200	176	5.893	0.81%	200	186	7.854	0.62%	200	184	9.359	0.72%
	120	200	172	2.195	1.56%	200	173	4.293	0.97%	200	184	6.275	0.59%	200	181	8.284	0.63%	200	181	10.304	0.59%
1.5	30	200	134	0.553	12.27%	200	150	1.104	9.89%	200	158	1.674	6.79%	200	150	2.142	6.67%	200	160	2.734	4.95%
	50	200	143	0.823	8.20%	200	159	1.632	6.25%	200	158	2.399	7.70%	200	161	3.088	5.48%	200	164	4.006	4.88%
	70	200	151	1.095	7.27%	200	154	2.088	7.62%	200	171	3.149	4.14%	200	165	4.201	4.24%	200	171	5.309	2.41%
	90	200	153	1.342	7.13%	200	162	2.664	5.29%	200	167	4.092	3.86%	200	168	5.262	4.70%	200	172	6.625	2.92%
	110	200	154	1.624	6.78%	200	163	3.216	4.59%	200	168	4.756	3.75%	200	171	6.464	2.45%	200	173	7.699	2.91%
	120	200	161	1.749	6.13%	200	163	3.288	5.13%	200	174	5.244	2.94%	200	171	6.876	1.84%	200	179	8.578	2.19%
1.4	30	200	138	0.528	11.40%	200	152	1.018	9.33%	200	154	1.562	8.38%	200	156	2.128	5.63%	200	156	2.500	6.31%
	50	200	146	0.766	9.00%	200	148	1.517	7.53%	200	158	2.306	5.56%	200	160	2.936	5.51%	200	159	3.865	4.34%
	70	200	148	1.029	10.31%	200	156	2.027	7.71%	200	161	3.038	5.06%	200	164	4.100	4.10%	200	169	4.923	4.67%
	90	200	151	1.344	6.65%	200	162	2.434	3.86%	200	161	3.784	3.63%	200	166	4.909	4.48%	200	170	6.289	3.83%

Table 6-6: (Continued.)

	110	200	158	1.556	7.27%	200	161	2.963	5.12%	200	169	4.437	3.28%	200	165	5.950	3.55%	200	172	7.480	2.30%
	120	200	148	1.629	8.03%	200	158	3.259	5.11%	200	165	4.734	3.89%	200	167	6.402	3.66%	200	169	8.060	3.66%
1.3	30	198	143	0.503	8.82%	198	143	0.971	10.45%	200	155	1.458	6.58%	199	157	1.964	6.79%	200	155	2.374	6.18%
	50	198	150	0.771	7.54%	200	151	1.380	7.20%	200	153	2.131	6.64%	200	162	2.777	5.61%	200	158	3.595	5.35%
	70	199	151	0.962	8.64%	199	158	1.907	5.22%	199	159	2.813	6.05%	200	172	3.663	4.18%	199	168	4.594	4.83%
	90	197	153	1.157	7.24%	200	157	2.404	6.60%	199	165	3.494	4.63%	200	166	4.528	3.95%	200	167	5.783	3.99%
	110	199	148	1.415	8.27%	200	162	2.827	6.16%	200	169	4.062	4.29%	200	168	5.475	4.80%	200	172	7.040	4.02%
	120	199	153	1.540	6.01%	200	160	2.983	6.30%	200	167	4.551	5.26%	200	171	5.834	3.63%	200	169	7.423	3.71%
1.2	30	192	147	0.488	8.22%	195	154	0.714	7.21%	198	150	1.086	8.80%	200	156	1.357	6.38%	199	161	1.785	6.51%
	50	195	149	0.673	8.61%	198	156	1.096	6.89%	198	163	1.562	5.55%	199	167	2.096	5.42%	200	167	2.696	4.67%
	70	199	152	0.728	6.73%	198	163	1.428	6.41%	200	167	2.018	5.58%	200	163	2.764	5.34%	200	173	3.411	4.63%
	90	196	157	0.872	6.43%	199	165	1.727	5.14%	200	166	2.485	5.78%	200	172	3.405	4.51%	200	171	4.321	4.09%
	110	200	158	1.046	7.44%	200	164	2.061	4.31%	200	168	3.019	4.59%	200	171	4.147	3.67%	200	170	5.171	4.77%
	120	199	161	1.095	6.58%	200	167	2.165	4.06%	200	171	3.140	5.59%	200	175	4.343	4.21%	200	172	5.512	3.95%
1.1	30	180	152	0.364	5.51%	189	166	0.694	4.40%	190	161	1.032	5.05%	192	171	1.287	3.62%	196	173	1.667	3.20%
	50	187	157	0.521	4.39%	191	169	0.983	4.50%	198	177	1.511	3.56%	198	174	1.913	3.62%	198	176	2.396	3.67%
	70	188	158	0.660	4.81%	195	171	1.320	4.00%	197	176	1.940	3.02%	198	175	2.460	3.30%	199	180	2.951	3.30%
	90	190	164	0.806	3.83%	196	171	1.584	4.32%	199	174	2.339	4.00%	198	180	3.066	3.20%	198	181	3.740	2.25%
	110	192	171	0.956	4.35%	197	172	1.931	4.34%	200	180	2.698	3.51%	199	178	3.689	3.60%	199	180	4.536	2.63%
	120	195	167	1.077	4.64%	198	170	2.088	4.13%	199	170	2.951	3.89%	198	175	3.960	3.22%	198	173	4.850	3.33%

Table 6-7: Results of 200 requests routed in 60 nodes

χ	b	200 Iterations				400 Iterations				600 Iterations				800 Iterations				1,000 Iterations			
		#Fea	#Opt	ET	Dev	#Fea	#Opt	ET	Dev	#Fea	#Opt	ET	Dev	#Fea	#Opt	ET	Dev	#Fea	#Opt	ET	Dev
3.0	40	200	164	1.150	1.44%	200	169	2.102	0.78%	200	173	3.136	0.81%	200	182	4.127	0.41%	200	180	5.452	0.46%
	60	200	168	1.577	0.76%	200	176	3.075	0.63%	200	180	4.444	0.44%	200	181	5.842	0.36%	200	182	7.310	0.46%
	80	200	176	2.041	0.68%	200	175	3.854	0.71%	200	182	5.646	0.38%	200	185	7.360	0.35%	200	183	9.354	0.36%
	100	200	175	2.473	0.73%	200	182	4.698	0.44%	200	181	6.809	0.40%	200	186	8.992	0.31%	200	189	11.441	0.28%
	120	200	176	2.663	0.94%	200	182	5.048	0.42%	200	184	7.589	0.28%	200	188	9.862	0.28%	200	188	12.159	0.31%
	130	200	179	2.751	0.59%	200	178	5.430	0.42%	200	189	8.310	0.21%	200	185	10.731	0.38%	200	187	13.177	0.30%
2.0	40	200	159	0.837	6.79%	200	166	1.709	2.53%	200	172	2.487	1.40%	200	169	3.328	1.55%	200	174	4.139	0.90%
	60	200	156	1.165	4.35%	200	169	2.318	1.75%	200	180	3.465	2.16%	200	175	4.442	0.89%	200	182	5.509	1.46%
	80	200	167	1.466	3.23%	200	169	2.887	2.17%	200	176	4.325	1.52%	200	187	5.955	0.40%	200	184	7.108	0.52%
	100	200	173	1.836	2.27%	200	176	3.481	1.50%	200	181	5.417	0.78%	200	183	6.973	0.65%	200	184	8.513	0.53%
	120	200	168	2.036	2.15%	200	174	3.869	1.55%	200	176	5.893	0.81%	200	186	7.854	0.62%	200	184	9.359	0.72%
	130	200	172	2.195	1.56%	200	173	4.293	0.97%	200	184	6.275	0.59%	200	181	8.284	0.63%	200	181	10.304	0.59%
1.5	40	199	137	0.735	9.36%	200	153	1.335	7.80%	200	160	2.033	6.15%	200	161	2.691	5.51%	200	161	3.483	4.88%
	60	200	150	0.982	7.12%	200	151	1.867	7.86%	200	155	2.851	6.49%	200	164	3.631	4.76%	200	170	4.761	3.91%
	80	200	153	1.226	6.99%	200	163	2.369	5.58%	200	164	3.738	3.17%	200	166	4.754	4.18%	200	170	5.892	3.17%
	100	200	156	1.497	6.88%	200	165	2.910	5.57%	200	172	4.392	2.93%	200	173	5.593	3.28%	200	171	7.045	3.14%
	120	200	154	1.624	6.78%	200	163	3.216	4.59%	200	168	4.756	3.75%	200	171	6.464	2.45%	200	173	7.699	2.91%
	130	200	161	1.749	6.13%	200	163	3.288	5.13%	200	174	5.244	2.94%	200	171	6.876	1.84%	200	179	8.578	2.19%
1.4	40	200	135	0.700	10.32%	200	144	1.310	9.78%	200	156	2.013	6.42%	200	159	2.658	5.27%	200	157	3.253	4.78%
	60	199	144	0.898	9.94%	200	151	1.741	7.22%	200	157	2.652	7.71%	200	158	3.655	5.38%	200	170	4.395	3.09%
	80	200	149	1.179	8.71%	200	155	2.316	4.67%	200	162	3.350	5.59%	200	168	4.557	3.95%	200	169	5.686	3.25%
	100	200	151	1.424	9.14%	200	161	2.763	5.03%	200	162	4.042	5.11%	200	168	5.550	3.04%	200	170	6.861	3.84%

Table 6-7: (Continued.)

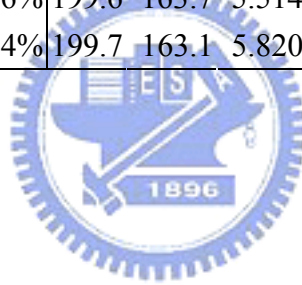
	120	200	158	1.556	7.27%	200	161	2.963	5.12%	200	169	4.437	3.28%	200	165	5.950	3.55%	200	172	7.480	2.30%
	130	200	148	1.629	8.03%	200	158	3.259	5.11%	200	165	4.734	3.89%	200	167	6.402	3.66%	200	169	8.060	3.66%
1.3	40	196	142	0.654	10.83%	196	155	1.168	7.99%	199	155	1.766	7.38%	200	156	2.398	5.85%	199	157	3.005	4.81%
	60	200	151	0.866	9.24%	198	155	1.703	6.64%	199	160	2.389	6.22%	199	160	3.247	7.09%	200	168	4.097	3.63%
	80	199	156	1.085	7.84%	200	155	2.072	5.81%	200	161	3.140	5.55%	199	167	4.164	4.30%	200	164	5.238	4.01%
	100	199	153	1.280	8.12%	200	155	2.589	6.11%	199	163	3.987	4.64%	200	169	5.172	4.07%	200	172	6.218	3.81%
	120	199	148	1.415	8.27%	200	162	2.827	6.16%	200	169	4.062	4.29%	200	168	5.475	4.80%	200	172	7.040	4.02%
	130	199	153	1.540	6.01%	200	160	2.983	6.30%	200	167	4.551	5.26%	200	171	5.834	3.63%	200	169	7.423	3.71%
1.2	40	193	145	0.615	8.70%	198	162	0.891	6.42%	199	159	1.326	6.84%	199	167	1.773	5.01%	199	164	2.157	5.37%
	60	195	150	0.640	8.91%	198	159	1.231	6.82%	199	161	1.810	5.51%	199	167	2.351	5.12%	200	166	3.043	4.57%
	80	196	153	0.779	7.33%	199	163	1.575	6.08%	199	163	2.217	5.53%	199	165	2.949	5.57%	200	169	3.928	4.54%
	100	199	152	1.005	7.34%	200	161	1.887	5.56%	200	170	2.703	4.03%	200	171	3.772	3.90%	200	169	4.819	3.52%
	120	200	158	1.046	7.44%	200	164	2.061	4.31%	200	168	3.019	4.59%	200	171	4.147	3.67%	200	170	5.171	4.77%
	130	199	161	1.095	6.58%	200	167	2.165	4.06%	200	171	3.140	5.59%	200	175	4.343	4.21%	200	172	5.512	3.95%
1.1	40	186	157	0.438	4.70%	192	170	0.887	3.94%	194	168	1.268	4.12%	197	170	1.591	4.27%	196	172	2.021	4.23%
	60	189	164	0.586	4.46%	195	160	1.161	5.35%	198	173	1.721	3.31%	197	175	2.194	3.49%	198	172	2.661	4.50%
	80	191	164	0.780	4.86%	197	176	1.440	3.93%	196	174	2.136	2.93%	197	174	2.722	3.82%	199	178	3.454	2.57%
	100	195	167	0.912	4.78%	196	170	1.783	4.07%	196	179	2.497	3.06%	198	178	3.346	3.06%	198	182	4.099	2.47%
	120	192	171	0.956	4.35%	197	172	1.931	4.34%	200	180	2.698	3.51%	199	178	3.689	3.60%	199	180	4.536	2.63%
	130	195	167	1.077	4.64%	198	170	2.088	4.13%	199	170	2.951	3.89%	198	175	3.960	3.22%	198	173	4.850	3.33%

Table 6-8: Average results for different networks ($n=40, 50, 60$)

n	b	200 Iterations				400 Iterations				600 Iterations				800 Iterations				1,000 Iterations			
		#Fea	#Opt	ET	Dev	#Fea	#Opt	ET	Dev	#Fea	#Opt	ET	Dev	#Fea	#Opt	ET	Dev	#Fea	#Opt	ET	Dev
40	20	197.6	138.9	0.361	7.38%	198.6	148.6	0.721	5.14%	199.0	154.3	1.045	4.52%	199.4	159.3	1.412	3.93%	199.7	160.1	1.751	3.57%
	30	198.0	145.0	0.499	5.58%	199.3	154.3	0.984	4.24%	199.7	160.0	1.438	3.63%	199.6	163.0	1.915	3.02%	200.0	168.0	2.369	2.51%
	40	199.1	150.6	0.636	4.82%	199.1	156.9	1.230	3.65%	199.9	164.1	1.830	2.88%	200.0	166.0	2.439	2.77%	199.6	169.3	2.956	2.55%
	50	199.6	152.7	0.750	4.50%	200.0	160.9	1.489	3.43%	200.0	163.1	2.192	3.11%	199.9	170.0	2.953	2.27%	200.0	172.1	3.594	2.18%
	60	198.9	154.4	0.889	4.19%	199.9	164.7	1.720	2.90%	199.6	167.4	2.563	2.60%	200.0	171.0	3.369	2.00%	199.9	175.3	4.181	1.86%
	70	199.1	155.9	1.009	4.06%	199.7	164.4	1.987	2.72%	199.9	171.4	2.960	2.08%	199.9	173.0	3.898	1.99%	200.0	174.1	4.741	1.78%
	80	199.4	159.3	1.107	3.84%	199.6	170.4	2.252	1.97%	199.7	170.1	3.284	2.17%	199.9	173.0	4.351	1.88%	200.0	176.1	5.321	1.40%
	90	199.9	163.1	1.261	3.35%	199.9	168.9	2.491	2.39%	199.9	173.3	3.626	1.89%	199.9	174.4	4.837	1.66%	200.0	178.3	6.003	1.20%
	100	199.9	163.1	1.395	2.77%	200.0	168.4	2.658	2.47%	199.9	172.9	4.017	1.74%	200.0	177.7	5.263	1.57%	200.0	178.3	6.615	1.44%
	110	199.6	164.0	1.502	2.93%	200.0	170.4	3.001	2.09%	200.0	175.9	4.386	1.69%	200.0	174.6	5.724	1.58%	200.0	179.0	7.190	1.22%
50	30	195.7	145.0	0.572	7.67%	197.4	156.4	1.070	6.71%	198.3	158.6	1.619	5.57%	198.7	161.6	2.133	4.47%	199.3	164.9	2.642	4.20%
	40	196.3	148.4	0.733	7.45%	198.0	159.9	1.343	5.61%	198.9	163.3	2.004	4.73%	199.4	166.3	2.652	3.98%	199.1	166.4	3.358	3.63%
	50	197.1	152.9	0.863	5.94%	198.4	160.0	1.602	5.14%	199.4	167.0	2.417	4.33%	199.6	168.6	3.108	3.94%	199.7	169.4	3.964	3.43%
	60	197.6	154.7	0.959	6.40%	198.7	160.1	1.871	5.18%	199.4	166.6	2.762	4.55%	199.3	168.6	3.623	3.87%	199.7	172.9	4.539	3.09%
	70	198.0	155.4	1.074	5.98%	198.9	164.4	2.126	4.78%	199.4	169.4	3.129	3.64%	199.7	172.1	4.191	3.15%	199.7	174.4	5.107	3.06%
	80	198.0	159.7	1.222	5.66%	199.4	165.1	2.359	4.14%	199.3	168.9	3.507	3.52%	199.3	173.1	4.637	3.22%	199.9	173.9	5.809	2.63%
	90	197.6	159.0	1.341	4.97%	199.3	166.9	2.642	3.89%	199.7	170.9	3.892	3.28%	199.7	173.4	5.103	3.13%	199.7	175.1	6.398	2.58%
	100	199.0	161.0	1.490	5.61%	199.4	167.1	2.873	4.04%	199.3	172.6	4.264	2.99%	199.7	175.4	5.628	2.62%	199.7	176.7	6.999	2.51%
	110	198.7	161.9	1.614	5.31%	199.6	168.3	3.131	3.78%	200.0	173.4	4.636	2.93%	199.9	175.3	6.206	2.71%	199.9	177.0	7.635	2.52%
	120	199.0	163.0	1.719	4.79%	199.7	167.0	3.358	3.73%	199.9	174.3	5.029	3.20%	199.7	175.0	6.633	2.51%	199.7	175.7	8.272	2.53%

Table 6-8: (Continued.)

40	194.3	133.0	0.838	7.43%	197.6	144.6	1.587	5.42%	199.0	149.7	2.315	4.97%	198.7	152.6	2.953	4.37%	199.4	157.9	3.622	3.84%
50	197.3	137.3	0.974	7.00%	197.9	145.9	1.887	5.56%	199.1	152.1	2.745	4.72%	198.9	155.3	3.480	4.03%	199.0	157.7	4.325	3.27%
60	197.1	139.4	1.139	6.00%	199.0	148.9	2.171	4.90%	198.7	155.1	3.183	4.19%	199.1	159.6	4.056	3.37%	199.4	162.0	4.973	3.09%
70	197.7	142.4	1.280	5.98%	198.9	150.3	2.505	4.36%	199.7	157.9	3.609	3.65%	199.4	160.0	4.595	3.25%	199.9	162.9	5.709	2.85%
60 80	198.3	141.3	1.432	5.98%	198.7	154.6	2.733	4.24%	199.1	157.1	3.978	3.87%	199.7	162.1	5.090	2.67%	200.0	164.7	6.391	2.40%
90	198.1	143.9	1.554	5.49%	198.9	153.3	3.063	3.98%	199.4	161.4	4.446	3.21%	199.4	162.9	5.679	2.90%	199.9	165.4	6.992	2.52%
100	198.4	144.6	1.713	5.62%	199.0	158.3	3.371	3.40%	199.1	162.1	4.721	3.15%	199.7	164.7	6.230	2.63%	199.6	166.7	7.662	2.53%
110	199.0	149.9	1.876	5.25%	198.7	158.3	3.652	3.42%	199.4	163.6	5.090	3.18%	199.4	166.0	6.705	2.54%	199.9	169.0	8.347	2.25%
120	198.9	151.6	2.028	4.41%	199.6	159.7	3.908	3.26%	199.6	163.7	5.514	2.83%	199.6	167.6	7.197	2.42%	199.6	171.1	9.071	2.34%
130	198.7	150.6	2.165	4.85%	199.4	160.1	4.077	3.04%	199.7	163.1	5.820	2.77%	199.6	168.3	7.701	2.35%	199.9	171.1	9.574	1.94%



Chapter 7 Genetic Algorithm (GA) for MRP-DC-WWC-SR

In this chapter, genetic algorithm (GA) method is applied to examine the MRP-DC-WWC-SR, a sub-problem of the MRWAP-DC-WWC-SR, which does not require wavelength assignment. In the proposed GA model, a destination-oriented encoding scheme used to encode individual's ploidy chromosomes, repaired-gene used to concatenate two disconnected genes, four types of operators (*Chromosome Crossover*, *Individual Crossover*, *Chromosome Mutation*, and *Individual Mutation*) and four mutation heuristics (*Random Mutation*, *Minimum Cost First Mutation*, *Minimum Delay First Mutation*, and *Hybrid Mutation*) used in evolution processing, will be incorporated to find solutions whose multicast costs are as small as possible. In the encoding scheme, the number of chromosomes is equal to the number of destinations in a request, and each chromosome indicates a light-path from the source to a destination. The objective of the encoding scheme is to attain the presentation of the individual's phenotype to be a multicast tree covering all destinations. The formulations for computing communication cost and transmission delay of a light-tree in Chapter 4 are not efficient to compute to a multicast tree. Therefore, new formulations for solving the studied problem will be proposed in the chapter.

In the rest of the chapter, first, the problem formulation for the MRP-DC-WWC-SR is introduced. Unlike the previous proposed formulations, the method of merging light-paths to

become a multicast tree and new formulations to compute communication cost, number of demanded wavelengths, and transmission delay of a multicast tree will be redefined. Secondly, the concept of GA method is introduced that is conducive to comprehend the major components of the framework of GA. Thirdly, according to the properties of studied problem, each component is well formulated such that the solutions for the problem are explored. Finally, three parts of experiments are discussed such the performance assessment of the GA model will be demonstrated.

7.1 Problem Formulation

Let $P_i(u, v) = \langle w_1^i, w_2^i, \dots, w_{l_i-1}^i, w_{l_i}^i \rangle$ represent an arbitrary light-path from u to v passing through $l_i - 2$ intermediate nodes, where $w_1^i = u$, $w_{l_i}^i = v$, and l_i is the number of nodes in $P_i(u, v)$. Let $e_{w_j^i, w_{j+1}^i}$ denote the link connecting nodes w_j^i and w_{j+1}^i , $1 \leq j \leq l_i - 1$; that is, l_i nodes are connected by $l_i - 1$ links. Set $\mathbf{P}(u, v) = \{P_i(u, v) \mid P_i(u, v) = \langle w_1^i, w_2^i, \dots, w_{l_i-1}^i, w_{l_i}^i \rangle, \forall w_j^i, w_j^i \in V, 1 \leq j \leq l_i\}$ contains all light-paths from u to v . The communication cost and the transmission delay of $P_i(u, v)$, denoted by $c(P_i(u, v))$ and $d(P_i(u, v))$, are given as:

$$\text{Communication cost: } c(P_i(u, v)) = \sum_{e \in P_i(u, v)} c(e) = c(e_{w_1^i, w_2^i}) + c(e_{w_2^i, w_3^i}) + \dots + c(e_{w_{l_i-1}^i, w_{l_i}^i}) \quad (7-1)$$

$$\text{Transmission delay: } d(P_i(u, v)) = \sum_{e \in P_i(u, v)} d(e) = d(e_{w_1^i, w_2^i}) + d(e_{w_2^i, w_3^i}) + \dots + d(e_{w_{l_i-1}^i, w_{l_i}^i}) \quad (7-2)$$

Among these potential light-paths in $\mathbf{P}(u, v)$, the one with the minimum communication cost and the one with minimum transmission delay are called the *minimum cost light-path*

(MCLP) denoted by $P^c(u, v)$ and *minimum delay light-path* (MDLP) denoted by $P^d(u, v)$; that is,

$$\text{MCLP: } P^c(u, v), \text{ where } c(P^c(u, v)) = \min_{P_i(u, v) \in P(u, v)} c(P_i(u, v)), \quad (7-3)$$

$$\text{MDLP: } P^d(u, v), \text{ where } c(P^d(u, v)) = \min_{P_i(u, v) \in P(u, v)} d(P_i(u, v)). \quad (7-4)$$

Given q light-paths, $P(u_1, v_1)$, $P(u_2, v_2)$, ..., and $P(u_q, v_q)$, they can be merged into a graph, represented as $\bigcup_{i=1}^q P(u_i, v_i)$. A multicast tree $MSP T^c(\bigcup_{i=1}^q P(u_i, v_i), D)$ is a tree obtained

by applying Prim's minimum spanning tree ($MSP T$) algorithm [59] to find the $MSP T$ with minimum sum of communication cost from $\bigcup_{i=1}^q P(u_i, v_i)$ first, and by eliminating all leaf nodes which are not contained in D .

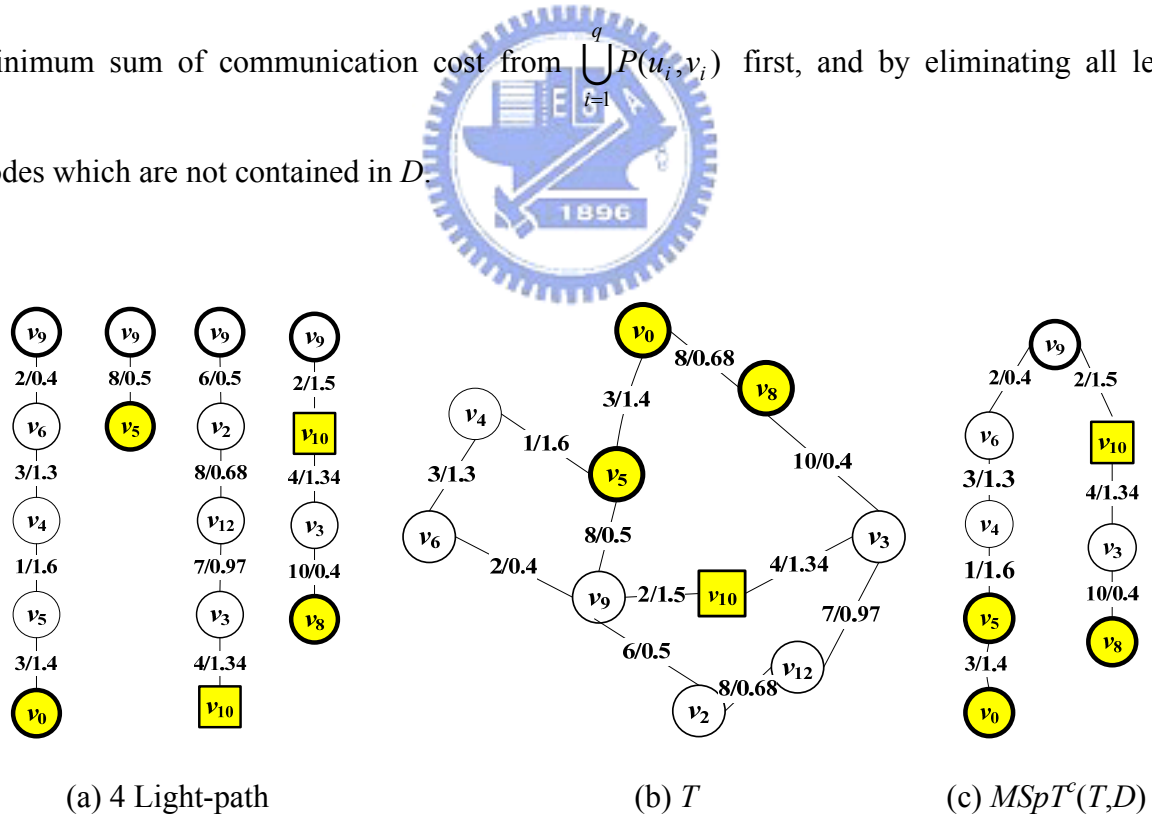


Figure 7-1: Example of $MSP T^c(T, D = \{v_0, v_5, v_8, v_{10}\})$

EXAMPLE 7.1: For the network and the request shown in Figure 4-1 (a) and EXAMPLE 4.1, four light-paths, $P_1(v_9, v_0) = \langle v_9, v_6, v_4, v_5, v_0 \rangle$, $P_2(v_9, v_5) = \langle v_9, v_5 \rangle$, $P_3(v_9, v_{10}) = \langle v_9, v_2, v_{12}, v_3, v_{10} \rangle$, $P_4(v_9, v_8) = \langle v_9, v_{10}, v_3, v_8 \rangle$, and $D = \{v_0, v_5, v_{10}, v_8\}$ are shown in Figure 7-1(a). The graphs, $T = P_1(v_9, v_0) \cup P_2(v_9, v_5) \cup P_3(v_9, v_{10}) \cup P_4(v_9, v_8)$ and $MSpT^c(T, D = \{v_0, v_5, v_8, v_{10}\})$, are shown in Figures 7-1(b) and (c), respectively. ■

According to the properties of light-trees, all internal nodes are feasible such that one wavelength is required to route the request to all destinations in the light-tree. The communication cost and the wavelength consumption of the light-tree can be computed using Eqs. (4-1) ~ (4-6). Nevertheless, a multicast tree may include infeasible nodes (whose light splitting capacities are smaller than their outbound degrees) such that one wavelength is not sufficient to route the request to all destinations in the multicast tree. The number of required wavelengths is determined by the number of light-trees separated from the multicast tree. Although Eqs. (4-1) ~ (4-6) are good for computing the communication cost and the transmission delay of a light-tree, they can not be used to compute those of a multicast tree. A new formulation will be proposed to compute the communication cost and the transmission delay. Besides, the number of latent required wavelengths (wavelength consumption) also needs to be computed to reflect the efficient of the multicast tree. To address the formulations for multicast trees, we assume the multicast tree T rooted at s include τ sub-trees ST_t rooted at s_t for $t, 1 \leq t \leq \tau$. Wavelength consumption $\omega(T)$, communication cost $c(T)$, and transmission delay $d(T)$ of T are redefined as follows.

(1) *Wavelength consumption of T*

The wavelength consumption of T , $\omega(T)$, used to compute the required wavelength for a multicast tree is defined recursively as

$$\omega(T) = \begin{cases} 1 & T \text{ having a root node only} \\ \max\left(\left\lceil \frac{\sum_{1 \leq i \leq \tau} \omega(ST_i)}{\theta(s)} \right\rceil, \varpi(T)\right) & \text{otherwise} \end{cases}, \quad (7-5)$$

where $\varpi(T) = \max_{1 \leq i \leq \tau} \omega(ST_i)$.

For the special case that the depth of T equals 2, each ST_i degenerates into a single node. According to the definition of light splitting capacity, an arbitrary node v can split one signal at the input port to $\theta(v)$ signals at the output port. The tree connecting s with these $\theta(s)$ degenerated sub-trees will form a light-tree and one wavelength is required. Therefore, each combination of $\theta(s)$ light-trees is a light-tree such that the minimum of the required wavelengths is equal to the least integer greater than the number of leaves divided by $\theta(s)$;

that is, $\omega(T) = \left\lceil \frac{\tau}{\theta(s)} \right\rceil$. Due to $\varpi(T) = \max_{1 \leq i \leq \tau} \omega(ST_i) = 1$ and $\omega(T) = \max\left(\left\lceil \frac{\sum_{1 \leq i \leq \tau} \omega(ST_i)}{\theta(s)} \right\rceil, \varpi(T)\right) = \max\left(\left\lceil \frac{\tau}{\theta(s)} \right\rceil, \varpi(T)\right) = \left\lceil \frac{\tau}{\theta(s)} \right\rceil$, Eq. (7-5) is satisfied. More

noteworthy is that T can be divided into $\omega(T)$ light-trees.

Suppose Eq. (7-5) is satisfied when the depth of T is equal to k . When the depth of T is equal to $k + 1$, the depths of τ sub-trees ST_i are smaller than or equal to k . Therefore, $\omega(ST_i)$ can be computed by Eq. (7-5) and each ST_i can be divided into $\omega(ST_i)$ light-trees. The set of the divided light-trees from ST_i is denoted by Γ_i . When these τ sets of light-trees are sorted in decreasing order of the wavelength consumption value, one light-tree is picked out for each set of the first $\theta(s)$ sets and the picked light-tree is removed from the set; that is, $\theta(s)$ light-trees are picked. Connecting to s with these light-trees will form a new light-tree. When the number of remaining set is smaller than $\theta(s)$, the number of selected light-trees will be also smaller than $\theta(s)$. Therefore, the number of newly formed light-trees will be the maximum of

$$\left\lceil \frac{\sum_{1 \leq t \leq \tau} \omega(ST_t)}{\theta(s)} \right\rceil \text{ and } \varpi(T); \text{ that is } \omega(T) = \max \left(\left\lceil \frac{\sum_{1 \leq t \leq \tau} \omega(ST_t)}{\theta(s)} \right\rceil, \varpi(T) \right). \text{ Eq. (7-5) is satisfied}$$

when the depth of T is equal to $k + 1$. By induction, we can proof that Eq. (7-5) can be used to compute the wavelength consumptions of multicast trees.

(2) *The communication cost and the transmission delay of T*

The communication cost and the transmission delay of T are computed recursively as

$$c(T) = \sum_{1 \leq t \leq \tau} (\omega(ST_t)c(e_{s,s_t}) + c(ST_t)) \quad (7-6)$$

$$d(T) = \max_{1 \leq t \leq \tau} (d(ST_t) + d(e_{s,s_t})) \quad (7-7)$$

The total communication cost of a link for routing a request to all destinations is equal to the product of its wavelength consumption (i.e. required wavelengths) and the communication cost of the link. The number of required wavelengths of sub-trees is derived by Eq. (7-5), so the communication cost of passing e_{s,s_t} to ST_t is equal to $\omega(ST_t)c(e_{s,s_t})$; that is, the communication cost for routing to each destination in ST_t is $\omega(ST_t)c(e_{s,s_t}) + c(ST_t)$. The communication cost of T can be thus defined as in Eq. (7-6). Unlike the computation of communication cost, the sub-tree with maximum transmission delay will dominate the transmission delay of T . Due to the transmission delay of passing e_{s,s_t} to ST_t being $d(ST_t) + d(e_{s,s_t})$, the transmission delay for routing a request from s to all destinations in T can be formulated as Eq. (7-7).

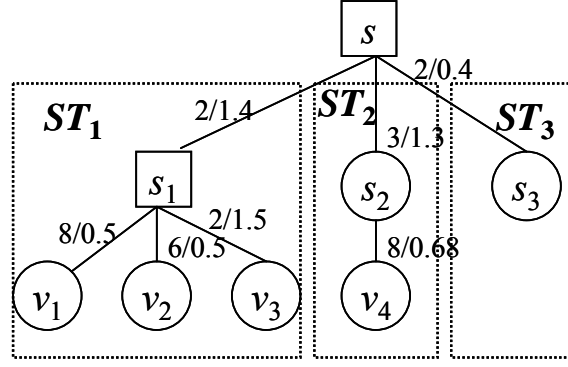


Figure 7-2: Sample of multicast tree T

EXAMPLE 7.2: Three sub-trees ($\tau = 3$) ST_1 , ST_2 and ST_3 in multicast tree T rooted at s are shown in Figure 7-2. In tree T , s and s_1 are MC nodes ($\theta(s) = 3$ and $\theta(s_1) = 2$), and the others are MI nodes. Due to $\omega(ST_1) = 1$, $\omega(ST_2) = 1$, $\omega(ST_3) = 1$, and $\omega(s) = 3$, we have

$\omega(s) = \max\left(\left\lceil \frac{1+1+1}{\theta(s)} \right\rceil, 1\right) = 2$, and $\omega(s_1) = 1$, $\omega(v_1) = 1$, $\omega(v_2) = 1$, $\omega(v_3) = 1$, $\omega(v_4) = 1$, $\omega(s_2) = 1$, $\omega(s_3) = 1$, and $\omega(v_4) = 1$.


$\omega(T) = \max_{1 \leq i \leq 3} \omega(ST_i) = 2$ and $\omega(T) = \max\left(\left\lceil \frac{\sum_{1 \leq i \leq 3} \omega(ST_i)}{\theta(s)} \right\rceil, \omega(T)\right) = \max\left(\left\lceil \frac{2+1+1}{3} \right\rceil, 2\right) = 2$ are found.

The communication costs and the transmission delays of ST_1 , ST_2 , and ST_3 can be computed as $c(ST_1) = 1 \times 8 + 1 \times 6 + 1 \times 2 = 16$, $d(ST_1) = \max(0.5, 0.5, 1.5) = 1.5$, $c(ST_2) = 8$, $d(ST_2) = 0.68$, $c(ST_3) = 0$, and $d(ST_3) = 0$, respectively. Therefore, $c(T) = (\omega(ST_1)c(e_{s,s_1}) + c(ST_1)) + (\omega(ST_2)c(e_{s,s_2}) + c(ST_2)) + (\omega(ST_3)c(e_{s,s_3}) + c(ST_3)) = (2 \times 2 + 16) + (1 \times 3 + 8) + (1 \times 2 + 0) = 33$ and $d(T) = \max(d(ST_1) + d(e_{s,s_1}), d(ST_2) + d(e_{s,s_2}), d(ST_3) + d(e_{s,s_3})) = \max(1.5 + 1.4, 0.68 + 1.3, 0 + 0.4) = 2.9$ can be obtained. ■

A multicast tree T is called a *candidate* if its transmission delay does not exceed the delay bound Δ . In the studied problem, the objective is to find a set of light-trees (i.e., a light-forest) with the minimum multicast cost. A candidate may be converted into a set of light-trees for satisfying the capacity constraint. The set of the derived light-trees, called an *equivalent light-forest*, constitutes a solution. Therefore, for a candidate \hat{T} with $\omega(\hat{T}) > 1$

(i.e., \hat{T} is not a light-tree), an equivalent light-forest $\Gamma = \{\hat{T}_1, \hat{T}_2, \dots, \hat{T}_{\omega(\hat{T})}\}$ is obtained from converting \hat{T} by the following procedure *Converting* such that $\hat{T} = \bigcup_{k=1}^{\omega(\hat{T})} (\hat{T}_k)$, $c(\hat{T}) = \sum_{k=1}^{\omega(\hat{T})} c(\hat{T}_k)$, $d(\hat{T}) = \max_{1 \leq k \leq \omega(\hat{T})} d(\hat{T}_k)$, $f(\hat{T}) = \sum_{k=1}^{\omega(\hat{T})} f(\hat{T}_k)$, and $\omega(\hat{T}_k) = 1$ for $1 \leq k \leq \omega(\hat{T})$, where $\omega(\hat{T}_k) = 1$ indicates that \hat{T}_k is a light-tree. Therefore, once a candidate is found, the corresponding equivalent light-forest will be a solution of the MRP-DC-WWC-SR. It is worthy to note that an optimal candidate implies an optimal light-forest that is an optimal solution.

Converting (\hat{T}, s)
// \hat{T} is a multicast tree and s is the source of request r .
{
1. if s is a leaf node
2. $T = \emptyset$
3. insert T into $S(s)$ // kept the converted sub-trees in s
4. return
5. else
6. $\tau = |Child(s)|$
7. for $i = 1$ to τ
8. *Converting* ($\hat{T}, Child(s, i)$)
9. end for-loop
10. while $Child(s) \neq \emptyset$ // $Child(s)$ keep all successors of s
11. choose the first $\theta(s)$ successors form $Child(s)$ in the order of
 the numbers of light-trees
12. insert $\bigcup_{1 \leq i \leq \theta(s)} (e_{s, Child(s, i)} \cup S(Child(s, i), 1))$ into $S(s)$
13. delete $S(Child(s, i), 1)$ for $1 \leq i \leq \theta(s)$
14. remove $Child(s, i)$ when $S(Child(s, i))$ is empty for $1 \leq i \leq \theta(s)$
15. end loop
16. return
}



The primary concept in *Conversing* procedure is to convert infeasible nodes in the candidate to construct a set of light-trees (a light-forest). Suppose that each node u has two buffers $S(u)$ and $Child(u)$ to kept light-trees with root u and all successors of u in \hat{T} , where $S(u, i)$ and $Child(u, i)$ are the i -th light-tree and the i -th successor of u , respectively. For the case that u is a leaf node, it must be noted that there is no light-tree stored in $S(u)$ and $|S(u)| = 0$. For the case that there are τ successors in u , because u can split an input signal into $\theta(u)$ output signals to other nodes, the request can be routed to $\theta(u)$ successors by a specified wavelength. Therefore, routing $\theta(u)$ light-trees chosen from $S(u)$ will need one wavelength, and then connecting u and the $\theta(u)$ light-trees will form a light-tree rooted at u . Repeating the procedure, all light-trees rooted at u will be constructed. For $|S(u)|$ light-trees rooted at u , $|S(u)|$ wavelengths are required to route the request passing through u . For a candidate \hat{T} , $S(s)$, the light-forest keeping $|S(s)|$ light-trees is the corresponding equivalent light-forest of \hat{T} .

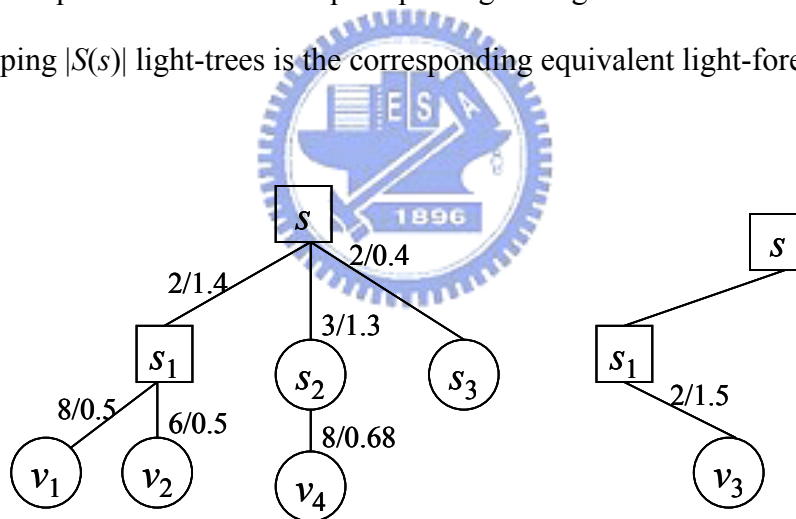


Figure 7-3: Two light-trees converted from Figure 7-2

EXAMPLE 7.3: Suppose that the multicast tree shown in Figure 7-2 is a candidate. The candidate requiring 2 wavelengths as computed in EXAMPLE 7.2 can be converted into two light-trees as shown in Figure 7-3. In other words, the two light-trees can be merged into a graph which is equivalent to the candidate, and the sum of the communication costs of the two light-trees equals the communication cost of the candidate. ■

For a given request, there may be several different candidates eligible for routing the request. The multicast cost function f defined in Eq. (4-6) is redefined for candidate T as

$$f(T) = \alpha \times c(T) + \beta \times \omega(T). \quad (7-8)$$

7.2 Concept of GA

The search space in GA method consists of all possible solutions to the problem [20]. A solution in the search space is called an *individual* whose genotype is composed of a set of *chromosomes* represented by sequences of 0s and 1s. These chromosomes of individuals could dominate phenotypes of individuals. Each individual has an associated objective function called *fitness*. A good individual is the one that has a high/low fitness value depending upon the problem (maximization/minimization). The strength of a chromosome in the individual is determined by its *fitness value* and the chromosomes of these individuals are carried to the next generation. A set of individuals with associated fitness values is called a *population*. This population at a given stage of GA is called a *generation*. The best individual was found in a generation at which the individual with best fitness value was discovered. The general GA proceeds as follows:

Genetic Algorithm

Step 1: Initialize the population

Step 2: Repeat

2.1 Choose parents from population; /* Selection/Reproduction */

2.2 Construct offspring by combining parents; /* Crossover/mutation */

2.3 If offspring is suitable then

 Replace worst individuals with better offspring; /* replacement */

Until the terminal condition is met

Figure 7-4: GA procedure

There are three major components in the while loop for GA:

(1) The process of selecting good individuals from the current generation to be carried to the next generation is called *selection/reproduction* (Step 2.1).

(2) The process of shuffling two randomly selected strings (chromosomes) in the two individuals to generate new offspring is called *crossover*. Sometimes one or more bits of a chromosome are complemented to generate a new offspring. This process of complementation is called *mutation* (Step 2.2).

(3) The replacement of worst performing individuals based on the fitness value is called *replacement* (Step 2.3).

The population size is finite in each generation of GA, which implies that only relatively fit individuals in generation j will be carried to the next generation $j+1$. The power of GA method comes from the fact that the algorithm terminates rapidly to obtain an optimal or near optimal solution. The iterative process is terminated when the situation in GA satisfy the terminal condition [65].

According to different characteristics of the studied problems, different individual representations, different fitness functions, different crossovers, and different mutations will be proposed such that each studied problem is well examined. The three main components are introduced as follows.

7.2.1 Selection / Reproduction

Since the population size in each generation is limited, only the finite number of good individuals (with high fitness value) will be copied into the mating pool depending on their fitness values. The individuals with higher fitness values contribute more copies to the mating pool than those with lower fitness values. This can be achieved by assigning proportionately a

higher probability of copying an individual that has a higher fitness value. The selection/reproduction uses the fitness values of the individuals obtained after evaluating the objective function. It uses a biased roulette wheel [16][65] for the selection of individuals to be taken into the mating pool. It ensures that highly fit individuals with high fitness values have a more number of offspring in the mating pool. Each individual I_i in the current generation is allotted a roulette wheel slot sized in proportion Pr_i to its fitness value. This proportion Pr_i can be defined as follows. Let $Of(I_i)$ be the actual fitness value of the individual I_i in a generation, $Sum = \sum_{I_i \text{ in population}} Of(I_i)$ be the sum of the fitness values of all individuals in the generation, and let $Pr_i = Of(I_i)/Sum$. When the roulette wheel is spun, there is a greater chance that a better individual will be copied into the mating pool because a good individual occupies a larger area on the roulette wheel.



7.2.2 Crossover

This operator involves two steps: first, the two individuals are selected from the mating pool at random for mating, and secondly, the crossover point c is selected uniformly at random in the interval $[1, l)$ (i.e., $1 \leq c < l$) for each pair of chromosomes in the two chosen individuals, where l is the length of the chromosome. Two new chromosomes called *offspring-chromosome* are then obtained by swapping all characters between positions $c + 1$ and n , and the two new individuals called *offspring* owning the *offspring-chromosomes* with different phenotypes are also obtained. Consider the following example that is shown using two chromosomes, P and Q , each of length $n = 6$ bit vectors.

Chromosome P: $\langle u_1, u_2, u_3, u_4, u_5, u_6 \rangle$

Chromosome Q: $\langle v_1, v_2, v_3, v_4, v_5, v_6 \rangle$,

where u_i and v_j are fixed length bit vectors for $1 \leq i, j \leq 6$.

Let crossover point c be 3rd bit vector from left. The bit vectors between 4 and 6 are swapped and bit vectors between 1 and 3 remain unchanged, then the two offspring-chromosomes R and S can be obtained as follows:

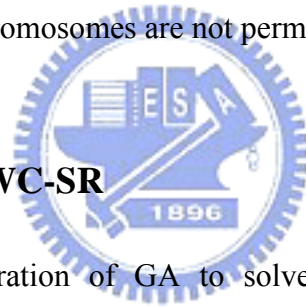
Offspring-chromosome R: $\langle u_1, u_2, u_3, v_4, v_5, v_6 \rangle$

Offspring-chromosome S: $\langle v_1, v_2, v_3, u_4, u_5, u_6 \rangle$

7.2.3 Mutation

The combined operation of reproduction and crossover may sometimes lose potentially useful information from the chromosome. To overcome this problem, the mutation implemented by complementing a bit (0 to 1 and vice versa) in some bit vector at random is introduced to ensure that good chromosomes are not permanently lost.

7.3 GA for MRP-DC-WWC-SR



In this section, a demonstration of GA to solve the MRP-DC-WWC-SR will be introduced. In the MRP-DC-WWC-SR, one multicast request will be rerouted at a time; therefore, (G, r) is used to describe the MRP-DC-WWC-SR, where $r = (s, D, \Delta)$ and there are q destinations in D . A destination-oriented encoding scheme used to encode individual's genotype is proposed first, and then the detail components in the GA procedure will be examined.

7.3.1 Chromosomal Encoding Scheme

In our model, the *polyploid chromosome* [16] represents individual's genotype, where the number of chromosomes is equal to the number of destinations in the routing request.

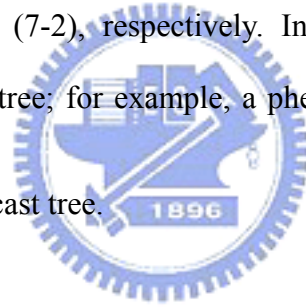
Since the light-paths between a source and all destinations could be merged into a multicast tree evaluated by Eq. (7-5) ~ Eq. (7-7), we employ a *destination-oriented* encoding scheme that each individual consists of q chromosomes using positive integer numbers to represent q light-paths from the source to q destinations. For an arbitrary k , the chromosome $C_i^k = \langle w_1^k, w_2^k, \dots, w_{l_i^k}^k \rangle$ in the individual $I_i = \langle C_i^1, C_i^2, \dots, C_i^q \rangle = \prod_{1 \leq k \leq q} C_i^k$ represents a

light-path from $w_1^k = s$ to $w_{l_i^k}^k = \zeta_k$, where l_i^k is the number of nodes, $\zeta_k \in D$, and $w_j^k \in V$, for $1 \leq j \leq l_i^k$. The part of a chromosome is called *gene*. $c(C_i^k) = \sum_{j=1}^{l_i^k-1} c(e_{w_j^k, w_{j+1}^k})$ and

$d(C_i^k) = \sum_{j=1}^{l_i^k-1} d(e_{w_j^k, w_{j+1}^k})$ represents the communication cost and the transmission delay of C_i^k

according to Eq. (7-1) and Eq. (7-2), respectively. In our model, the phenotype of an individual represents a multicast tree; for example, a phenotype of I_i will be represented by

$MSpT^c(\bigcup_{k=1}^q C_i^k, D)$ that is a multicast tree.



EXAMPLE 7.4: The four light-paths from v_9 to v_0, v_5, v_{10} , and v_8 for $(v_9, \{v_0, v_5, v_{10}, v_8\}, 3.3)$

shown in Figure 7-1 (a) represent the chromosomes of individual I_j as follows:

$$C_j^1 = \langle v_9, v_6, v_4, v_5, v_0 \rangle, l_j^1 = 5,$$

$$C_j^2 = \langle v_9, v_5 \rangle, l_j^2 = 2,$$

$$C_j^3 = \langle v_9, v_2, v_{12}, v_3, v_{10} \rangle, l_j^3 = 5,$$

$$C_j^4 = \langle v_9, v_{10}, v_3, v_8 \rangle, l_j^4 = 4,$$

$$I_j = \langle C_j^1, C_j^2, C_j^3, C_j^4 \rangle = \prod_{1 \leq k \leq 4} C_j^k. \quad \blacksquare$$

Each chromosome represents a light-path from the source to a destination, and each edge

of light-path needs to correspond directly to a physical optical fiber in WDM networks. Nevertheless, when the evolution is in progress, the two new offspring-chromosomes will be obtained by concatenating two genes of parent's chromosomes in crossover or in mutation. The catenation causes the offspring to own *disturbed chromosome* that contain a nonexistent fiber-link at latest. Therefore, an operation, *cat*, is used to concatenate and repair the disturbed chromosome for preventing the new offspring-chromosome from using an invalid optical link. Formally, the catenation of two genes, $g_i^{x_1, x_2} = \langle w_{x_1}^k, w_{x_1+1}^k, \dots, w_{x_2}^k \rangle \in C_i^k$ for $1 \leq x_1, x_2 \leq l_i^k$ and $g_j^{y_1, y_2} = \langle \bar{w}_{y_1}^k, \bar{w}_{y_1+1}^k, \dots, \bar{w}_{y_2}^k \rangle \in C_j^k$ for $1 \leq y_1, y_2 \leq l_j^k$, denoted with $cat(g_i^{x_1, x_2}, g_j^{y_1, y_2})$ can be defined as :

$$cat(g_i^{x_1, x_2}, g_j^{y_1, y_2}) = \begin{cases} \langle w_{x_1}^k, w_{x_1+1}^k, \dots, w_{x_2}^k, \bar{w}_{y_1+1}^k, \bar{w}_{y_1+2}^k, \dots, \bar{w}_{y_2}^k \rangle & \text{if } w_{x_1}^k = \bar{w}_{y_1}^k \\ \langle w_{x_1}^k, w_{x_1+1}^k, \dots, w_{x_2}^k, \bar{w}_{y_1}^k, \bar{w}_{y_1+1}^k, \dots, \bar{w}_{y_2}^k \rangle & \text{if } w_{x_2}^k \neq \bar{w}_{y_1}^k \text{ and } e_{w_{x_2}^k, \bar{w}_{y_1}^k} \in E \\ cat(cat(g_i^{x_1, x_2}, P_x(w_{x_2}^k, \bar{w}_{y_1}^k)), g_j^{y_1, y_2}) & \text{if } w_{x_2}^k \neq \bar{w}_{y_1}^k \text{ and } e_{w_{x_2}^k, \bar{w}_{y_1}^k} \notin E \end{cases} \quad (7-9)$$

where $P_x(w_{x_2}^k, \bar{w}_{y_1}^k) \in \mathbf{P}(w_{x_2}^k, \bar{w}_{y_1}^k)$ called as a *repaired-gene* is a randomly-chosen light-path between $w_{x_2}^k$ and $\bar{w}_{y_1}^k$. Because each chromosome describes a light-path, the catenation of two genes makes sense when they belong to two chromosomes which route to same destination. Therefore, the above description $g_i^{x_1, x_2}$ and $g_j^{y_1, y_2}$ will belong to two chromosomes C_i^k and C_j^k of individual I_i and I_j , respectively.

EXAMPLE 7.5: Suppose that two genes $\langle v_9, v_2 \rangle$ and $\langle v_3, v_8 \rangle$ are given. In Figure 7-1(a), $\langle v_9, v_2 \rangle$ and $\langle v_3, v_8 \rangle$ cannot be concatenated into $\langle v_9, v_2, v_3, v_8 \rangle$ because the link e_{v_2, v_3} between v_2 and v_3 does not exist in E . Therefore, some repaired-gene $\langle v_2, v_{12}, v_3 \rangle$ should be chosen arbitrarily from $\mathbf{P}(v_2, v_3)$ and we have

$$\begin{aligned}
cat(\langle v_9, v_2 \rangle, \langle v_3, v_8 \rangle) &= cat(cat(\langle v_9, v_2 \rangle, \langle v_2, v_{12}, v_3 \rangle), \langle v_3, v_8 \rangle) \\
&= cat(\langle v_9, v_2, v_{12}, v_3 \rangle, \langle v_3, v_8 \rangle) \\
&= \langle v_9, v_2, v_{12}, v_3, v_8 \rangle. \quad \blacksquare
\end{aligned}$$

7.3.2 Crossover Operator

There are two types of crossover operators used randomly in the development of this GA model: (1) *Chromosome Crossover (CC)*, (2) *Individual Crossover (IC)*. Suppose that the two individuals $I_i = \prod_{1 \leq k \leq q} C_i^k$ and $I_j = \prod_{1 \leq k \leq q} C_j^k$, and chromosomes $C_i^k = \langle w_1^k, w_2^k, \dots, w_{l_i^k}^k \rangle$ in I_i and $C_j^k = \langle \bar{w}_1^k, \bar{w}_2^k, \dots, \bar{w}_{l_j^k}^k \rangle$ in I_j are given for some k , where $w_1^k = \bar{w}_1^k = s$ and $w_{l_i^k}^k = w_{l_j^k}^k = \zeta_k \in D$. The two operators are defined as follows.

- *Chromosome Crossover (CC)*: Because the first nodes of C_i^k and C_j^k have the same source, two crossover points x and y ($x \leq y$) will be selected randomly from 2 to $\min(l_i^k, l_j^k)$. These two offspring-chromosomes \hat{C}_i^k and \hat{C}_j^k are described as follows.

case 1 : $x = y$

$$\hat{C}_i^k = cat(\langle w_1^k, w_2^k, \dots, w_{x-1}^k \rangle, \langle \bar{w}_x^k, \bar{w}_{x+1}^k, \dots, \bar{w}_{l_j^k}^k \rangle)$$

$$\hat{C}_j^k = cat(\langle \bar{w}_1^k, \bar{w}_2^k, \dots, \bar{w}_{x-1}^k \rangle, \langle w_x^k, w_{x+1}^k, \dots, w_{l_i^k}^k \rangle)$$

case 2 : $x < y$

$$\hat{C}_i^k = cat(cat(\langle w_1^k, w_2^k, \dots, w_{x-1}^k \rangle, \langle \bar{w}_x^k, \bar{w}_{x+1}^k, \dots, \bar{w}_y^k \rangle), \langle w_{y+1}^k, w_{y+2}^k, \dots, w_{l_i^k}^k \rangle)$$

$$\hat{C}_j^k = cat(cat(\langle \bar{w}_1^k, \bar{w}_2^k, \dots, \bar{w}_{x-1}^k \rangle, \langle w_x^k, w_{x+1}^k, \dots, w_y^k \rangle), \langle \bar{w}_{y+1}^k, \bar{w}_{y+2}^k, \dots, \bar{w}_{l_j^k}^k \rangle)$$

- *Individual Crossover (IC)*: Randomly select two crossover points x and y ($x \leq y$) from 1 to q . These two offspring \hat{I}_i and \hat{I}_j are described as follows.

case 1 : $x = y$

$$\hat{I}_i = \langle C_i^1, C_i^2, \dots, C_i^{x-1}, C_i^x, \dots, C_j^q \rangle$$

$$\hat{I}_j = \langle C_j^1, C_j^2, \dots, C_j^{x-1}, C_i^x, \dots, C_i^q \rangle$$

case 2 : $x < y$

$$\hat{I}_i = \langle C_i^1, C_i^2, \dots, C_i^{x-1}, C_j^x, C_j^{x+1}, \dots, C_j^y, C_i^{y+1}, \dots, C_i^q \rangle$$

$$\hat{I}_j = \langle C_j^1, C_j^2, \dots, C_j^{x-1}, C_i^x, C_i^{x+1}, \dots, C_i^y, C_j^{y+1}, \dots, C_j^q \rangle$$

In the crossover operations, the repaired-gene is chosen from the MCLP and the MDLP according to whether a feasible solution has been found till current generation or not. That is, the MDLP or the MCLP is chosen to reflect that finding a feasible solution or finding a better feasible solution is the major target.



7.3.3 Mutation

There are two types of mutation operators used in the development of this GA model: (1) *Chromosome Mutation (CM)*, (2) *Individual Mutation (IM)*. The two operators are defined as:

- *Chromosome Mutation (CM)*: Randomly select two mutation points x and y ($x \leq y$) from 2 to l_i^k . The chromosome \hat{C}_i^k mutated from C_i^k is presented as follows.

case 1 : $x = y$

$$\hat{C}_i^k = \text{cat}(\langle w_1^k, w_2^k, \dots, w_{x-1}^k \rangle, P(w_x^k, w_{l_i^k}^k)), \text{ where } P(w_x^k, w_{l_i^k}^k) \in \mathbf{P}(w_x^k, w_{l_i^k}^k).$$

case 2 : $x < y$

$$\hat{C}_i^k = \text{cat}(\text{cat}(\langle w_1^k, w_2^k, \dots, w_{x-1}^k \rangle, P(w_x^k, w_y^k)), \langle w_{y+1}^k, w_2^k, \dots, w_{l_i^k}^k \rangle),$$

where $P(w_x^k, w_y^k) \in \mathbf{P}(w_x^k, w_y^k)$.

- *Individual Mutation (IM)*: Randomly select two mutation points x and y ($x \leq y$) from 1

to q . The chromosome \hat{I}_i mutated from \hat{I}_j is presented as follows.

case 1 : $x = y$

$$\hat{I}_i = \langle C_i^1, \dots, C_i^{x-1}, P(s, \zeta_x), C_i^{x+1}, \dots, C_i^q \rangle, \text{ where } P(s, \zeta_x) \in \mathbf{P}(s, \zeta_x)$$

case 2 : $x < y$

$$\hat{I}_i = \langle C_i^1, \dots, C_i^{x-1}, P(s, \zeta_x), P(s, \zeta_{x+1}), \dots, P(s, \zeta_y), C_i^{y+1}, \dots, C_i^q \rangle,$$

where $P(s, \zeta_z) \in \mathbf{P}(s, \zeta_z)$ for $x \leq z \leq y$.

The mutation not only insures the population against permanent fixation at any particularity locus but also spoils the better chromosome. Since a chromosome is used to represent a light-path, the mutation operator implies that another new light-path (chromosome) would be established by choosing randomly a node called *mutation node* in the chromosome (light-path), and rerouting the mutation node to another node called *rerouting node*, where the rerouting node can be determined by using different rerouting approaches. In our GA model, the four different mutation heuristics, *Random Mutation (RM)*, *Minimum Cost First Mutation (MCFM)*, *Minimum Delay First Mutation (MDFM)*, and *Hybrid Mutation (HM)* providing different rerouting approaches in Chromosome Mutation to reflect the problem characteristics are described as follows.

- (1) *Random Mutation (RM)*: The rerouting node is chosen irregularly from the neighboring nodes.
- (2) *Minimum Cost First Mutation (MCFM)*: The heuristic of *MCFM* gives the population more evolution pressure that the rerouting node is chosen according to the mutation probability of link to decrease communication cost. All mutation probabilities of light-paths between the mutation node u and all neighborhood nodes would be computed according to their communication costs such that the link with high communication

cost will give lower mutation probability. Therefore, a neighborhood node with a high mutation probability will have a greater chance to be chosen than those with a lower mutation probability. In *MCFM*, the mutation probability of the link from u to v ($e(u, v) \in E$), $Pr_{MCFM}(u, v)$, is defined as :

$$Pr_{MCFM}(u, v) = \frac{\frac{1}{c(P^c(u, v))}}{\sum_{e(u, x) \in E} \frac{1}{c(P^c(u, x))}}. \quad (7-10)$$

(3) *Minimum Delay First Mutation (MDFM)*: In *MDFM*, the link with a high transmission delay will give a lower mutation probability. Therefore, the mutation probability of the link from u to v ($e(u, v) \in E$), $Pr_{MDFM}(u, v)$, is determined by their transmission delay and defined as :



$$Pr_{MDFM}(u, v) = \frac{\frac{1}{d(P^d(u, v))}}{\sum_{e(u, x) \in E} \frac{1}{d(P^d(u, x))}}. \quad (7-11)$$

(4) *Hybrid Mutation (HM)*: The *HM* hybridizes the *RM*, *MCFM*, and *MDFM* to construct an intelligent rerouting approach according to the features of chromosome, the features include whether the transmission delay is greater than Δ or not and whether the communication cost is minimal or not. For a given chromosome C_i^k , the *HM* adopts the appropriate mutation heuristic from *RM*, *MCFM*, or *MDFM* according to the following rules:

Adopt MCFM : if $c(C_i^k) > c(P^c(s, \zeta_k))$ and $d(C_i^k) \leq \Delta$;

Adopt MDFM : if $d(C_i^k) > \Delta$;

Adopt RM : if $c(C_i^k) = c(P^c(s, \zeta_k))$ and $d(C_i^k) \leq \Delta$.

7.3.4 Fitness Function Definition

Generally, GA method uses a fitness function to evaluate all individuals in a population. According to the phenotype of individual I_j , the *corresponding multicast tree* T_{I_j} , $T_{I_j} = MSpT^c(\bigcup_{k=1}^q C_j^k, D)$, could be determined by the chromosomes in I_j . It is not necessarily true that the corresponding multicast tree is a light-tree or a candidate. Since the goal is to find a feasible light-forest with a minimum multicast cost, the multicast cost function $f(T_{I_j})$ can be viewed as a fitness function associated with each individual. The objective is to minimize $f(T_{I_j})$ defined in Eq. (7-8).

In our encoding schema, because each chromosome represents a light-path from the source to each destination, the destination constraint is always satisfied. Nevertheless, the delay constraint will complicate the GA formulation so it is not involved in encoding chromosomes or in evolution proceeding. To avoid the chromosome whose phenotype violate the delay constraint leaving in the population, we need to attach a penalty by using a *penalty function* to the fitness function. Therefore, the individual with a non-zero penalty implies that there is at least one chromosome violating the delay constraint, and the penalty will increase sharply when the transmission delay of a light-path is far from the delay bound. Considering the delay constraint, the formulation in Eq. (7-8) above can be rewritten in another form:

$$\text{Minimize } fitness(T_{I_j}) = f(T_{I_j}) + \hat{\beta} \times Penalty(T_{I_j}), \quad (7-12)$$

$$\text{where } Penalty(T_{I_j}) = \begin{cases} f(T_{I_j}) \cdot \exp\left(\frac{d(T_{I_j}) - \Delta}{\Delta}\right) & \text{if } d(T_{I_j}) > \Delta \\ 0 & \text{otherwise} \end{cases} \quad \text{and } \hat{\beta} \text{ is the penalty}$$

weight which should be greater than the number of destinations.

In the Selection/Reproduction stage, each individual I_j is given a probability $Pr(I_j)$ for being selected as parent that is disproportional to their fitness because our objective is

minimizing the fitness value. Therefore, the $Pr(I_j)$ can be redefined as :

$$Pr(I_j) = \frac{\frac{1}{fitness(T_{I_j})}}{\sum_{I_x \text{ in population}} \frac{1}{fitness(T_{I_x})}} \quad (7-13)$$

7.3.5 Chromosome Repair

The catenation operator in Eq. (7-9) used in crossover and mutation operations causes that the new chromosome concatenated two genes may contain a useless gene which is cyclic sub-path. The cyclic sub-path in chromosome will be duplicated in generation proceeding more and more seriously. For example, suppose that the repaired-gene in EXAMPLE 7.4 is $\langle v_2, v_9, v_{10}, v_3 \rangle$. The $cat(\langle v_9, v_2 \rangle, \langle v_3, v_8 \rangle) = \langle v_9, v_2, v_9, v_{10}, v_3, v_8 \rangle$ contain a cyclic sub-path $\langle v_9, v_2, v_9 \rangle$. Therefore, the following procedure *Repairing_Chromosome* is introduced so as to repair these offspring chromosomes. Using the procedure, $\langle v_9, v_2, v_9, v_{10}, v_3, v_8 \rangle$ could be reduced into $\langle v_9, v_{10}, v_3, v_8 \rangle$.

```

Repairing_Chromosome (C =  $\langle u_1, u_2, \dots, u_{l_i^k} \rangle$ )
// C is a chromosome, and  $l_i^k$  is the number of nodes in C.
// starting(e) and end(e) are the starting node and the end node of edge e in the order of
light-path.
{
  1. x = 1 ;
  2. while (x  $\neq$   $l_i^k$ )
  3.   y = x + 1;
  4.   while (y  $\neq$   $l_i^k$ )
  4.     if ( $u_x = u_y$ ) then
  5.       remove  $u_z$  from C for x < z  $\leq$  y;
           x = y;
           break;
  7.     else
  8.       y = y + 1;
  9.   end while-loop;
}

```

```
9.end while-loop;  
}
```

7.3.6 Replacement Strategy

For creating a new generation after crossover and mutation are carried out on the individuals of the previous generation, there are many replacement strategies proposed in the literature and a good discussion can be found in [16]. The most common strategies are to probabilistically replace the poorest performing individuals from the previous generation, but the *elitist strategy* appends the best performing individual of a previous generation to the current population and thereby ensures that the individual with the best fitness value always survives to the next generation.

The algorithm developed here combines both the concepts mentioned above. Each offspring generated after crossover and mutation is added to the new generation if it has a better fitness value than both of its parents. If the fitness value of an offspring is better than only one of the parents, we select an individual randomly from the better parents and the offspring. If the offspring is worse than both parents, any one of the parents is selected at random for the next generation. This ensures that the best individual is carried to the next generation, while the worst is not carried to the succeeding generations.

7.3.7 Termination Rules

The execution of GA can be terminated using any one of the following rules:

- R1: when the average and maximum fitness values are got above a predetermined threshold;
- R2: when the average and maximum fitness values in a generation become the same; or
- R3: when the number of generations exceeds an upper bound specified by the user.

The best value for a given problem can be obtained from the GA when the algorithm is terminated using R2 [16]. But, R3 is chosen in this paper to reduce the elapsed execution time.

7.4 Experiments

Our work focuses on how to find a near optimal light-forest such that destination, delay, and degree constraints are all satisfied. The approach used in this simulation to evaluate the performance assessment of our GA model have been introduced in Section 4.3 or referred to Waxman [78]. We assume that Δ equals to 1.2 times of the maximum in the following experiments (i.e., $\chi = 1.2$ discussed in Chapter 5.3). Two different WDM networks, net_1 with 30 nodes ($n = 30$) and net_2 with 100 nodes ($n = 100$), are used to test the GA model. 20 requests are generated randomly to simulate different requests with different numbers of destinations for net_1 and net_2 , i.e., the notation of “ $q = 4$ ” represents a request with 4 destinations. In the following experiments, the multicast cost and the elapsed execution stand for the average elapsed execution time and average multicast cost.

The experiments consist in three parts: (1) performance assessment of the GA model; (2) comparisons among the GA model, 3-Phase Model (3PM) proposed in [11], and ILP; (3) comparisons among 4 mutation heuristics. It is obvious that the number of population size and the number of generations will affect the elapsed execution time when evolution is in progress. The effects of population size (PS), generations, mutation probability (MP), crossover probability (CP), average multicast cost, and average elapsed execution time for the two networks net_1 and net_2 will discuss in the first part. According to experimental results in the first part, we will give the suggestion about these suitable values of PS , generations, MP , and CP which will be referred in the second part and the third part.

7.4.1 Performance Assessment of the GA model

Because the studied problem is NP-hard [35], the efficiency of GA model is hard to be discussed but comparing with the well-known Minimum Steiner Tree problem (MSTP). The studied problem can be reduced to the MSTP by setting delay bound to be infinity, the splitting capacity of each node to be infinity, and $\alpha = 0$; therefore, the candidate with the minimum multicast cost could be equivalent to the Steiner Tree with the minimum communication cost. The four types of comparison between the GA model and the *Minimum Distance Network Heuristic (MDNH)* [39] for finding the MSTP would be discussed by choosing $\alpha = 0$, $\beta = 1$, and $\hat{\beta} = 1$.

Effect of Population size

The average demanded generations to obtain the multicast trees with less or equivalent multicast cost than the multicast trees obtained by MDHN [39] for different population sizes are shown in Figures 7-5 and 7-6 by choosing q from the set $\{4, 5, \dots, 10\}$, PS from the set $\{100, 200, \dots, 500\}$ for net_1 , and PS from the set $\{300, 400, \dots, 700\}$ for net_2 on the probabilities $MP = 0.3$ and $CP = 1.0$. From different criteria, we make several observations:

- (1) The GA model can give a better or equivalent multicast tree than MDHN's first 500 generations.
- (2) The greater number of PS can reduce the number of generations implicitly but increases the elapsed execution time. Therefore, for the two networks, net_1 and net_2 , the population sizes are chosen 200 and 300 such that the GA model can give a better solution elapsed less execution time.

Effect of Generations

Because the experiments in the GA model are time-consuming, it is necessary to

examine the effects of generations. We set $MP = 0.3$, $CP = 1.0$, maximum of generations to be 1000 ($MG = 1000$), and q to be selected from the set $\{4, 5, 6, 7, 8, 9, 10\}$. The results of net_1 on $PS = 200$ and net_2 on $PS = 300$ shown in Figures 7-7 and 7-8 describe the relationship between generations and multicast costs for different requests. The improvement percentage of multicast cost for comparing the GA model with MDHN is shown in Figure 7-9. We make several observations:

- (1) The best fitness value converges sharply. For example, the best fitness values for $q = 4, 5, 6, 7, 8, 9, 10$ appear in 60, 360, 60, 380, 700, 100, 620 generations on net_1 , and appear in 420, 320, 540, 820, 620, 360, 580 generations on net_2 , respectively. When the number of generations is higher than 700, the multicast costs are near static such that the evolution proceeding is not conducive to obtain a better solution. Setting the maximum generations to terminate the GA procedure as 700 generations seems to be reasonable. Therefore, 2000 is chosen as the maximum of generations ($MG = 2000$) for net_1 and net_2 in the following experiments.
- (2) Comparing with the solutions by MDHN in Figure 7-9, the solutions found by the GA model are better. Although the improvement is not significant, the GA model can be used to solve the reduced problem, MSTP. The observation may help account for applying the GA model to examine the MRP-DC-WWC-SR.

Effect of Mutation Probability

To examine the effect of the mutation probabilities (MPs) in the GA model to obtain the multicast trees with less or equivalent multicast cost than the multicast trees obtained by MDHN [39], the same primitive individuals in initialization are used to simulate different test cases with different MPs . Let we set $PS = 200$, $CP = 1.0$, $MG = 1000$, and the value of MP is selected from the set $\{0.0, 0.05, 0.10, 0.20, 0.30, 0.40, 0.50, 0.6, 0.7, 1.0\}$. After 5 runs, the

experimental results shown in Figures 7-10 and 7-11 describe the elapsed execution times and average generations for different MP s in net_1 , respectively. We make several observations:

- (1) When the value of MP is set near to 1, each chromosome will have a good probability to mutate which causes extra concatenating and repairing operations required. The effect of MP value to execution time seems apparently for the request with many destinations.
- (2) The experimental result of comparing the effect between MP s and demanded generations is similar to the effect of execution time. Therefore, $MP = 0.2$ chosen in the following experiments will help the GA model to gain better fitness values but elapse less execution time than large value of MP .

Effect of Crossover Probability

To examine the effect of the crossover probabilities (CP s) in the GA model in obtaining the multicast trees with less or equivalent multicast cost than the multicast trees obtained by MDHN [39], we set $PS = 200$, $MP = 0.2$, $MG = 1000$, and the value of CP is selected from the set $\{0.0, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0\}$. After 5 runs, the experimental results shown in Figures 7-12 and 7-13 describe the elapsed execution times and generations for different CP s in net_1 , respectively. We make several observations:

- (1) According to these experimental results, the better value of CP is different for different requests and we found that $CP = 1.0$ is conducive to gain better performance. Therefore, in our following experiments, $CP = 1.0$ is chosen.
- (2) In brief, the population sizes chosen 200 and 300 for net_1 and net_2 , $MG = 2000$, $MP = 0.2$, and $CP = 1.0$ can be determined. Nevertheless, these setups may not be suitable for any network.

7.4.2 Comparisons between GA, 3PM, and ILP

The experimental results of multicast costs for different generations and comparisons of multicast costs among three methods, GA, 3-Phase Model (3PM) [11], and ILP proposed in Chapter 5, for different requests are shown in Figures 7-14 and 7-15 for net_1 , and Figures 7-16 and 7-17 for net_2 , respectively. The experimental results of average promotion percentages of multicast cost for the GA model and 3PM are shown in Figure 7-18. For the elapsed execution times of GA, 3PM, and ILP, they are shown in Figure 7-19 for different requests. In implementing the ILP by CPLEX, the optimal solution for routing a request with more than 8 destinations in net_1 or more than 7 destinations in net_2 may not attain in an affordable execution time. According to these experimental results, we make several observations:

- (1) The light-forest found by the GA model, whose multicast cost is approximated to the multicast cost of light-forest found by ILP and is better than the multicast cost of light-forest found by 3PM model. Besides, GA model can gain the average promotion percentages of multicast cost more than 19.86% for net_1 and 29.94% for net_2 than 3PM model.
- (2) Although GA model is more time-consuming than 3PM model, the elapsed execution time is far smaller than the elapsed execution time for ILP. Nevertheless, because the elapsed execution time is proportional to the numbers of destinations in requests, the consumption of execution time is an important challenge in GA.

7.4.3 Comparisons among Four Mutation Heuristics

In the GA model, four mutation heuristics are proposed to examine the MRP-DC-WWC-SR. Using the same primitive individuals in the stage of population initialization, the comparisons of multicast costs for different generations and elapsed execution times for

different requests among the four mutation heuristics are shown from Figures 7-20 to 7-22 for net_1 . According to these experimental results for elapsed execution time, we observe that the convergence of *MCFM* is quicker than the other three heuristics and *HM* needs less execution time.

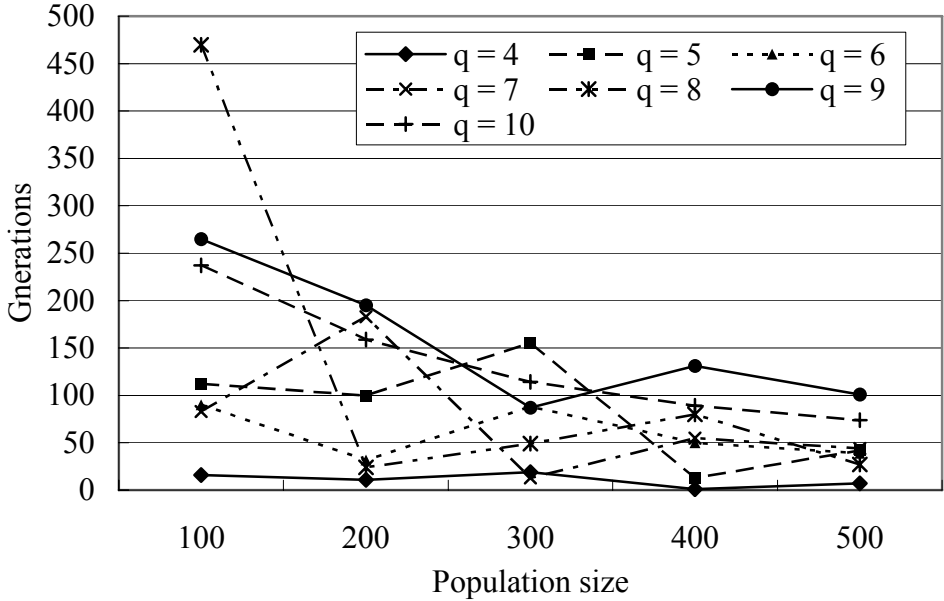


Figure 7-5: Demanded generations for different PSs and different requests in net_1

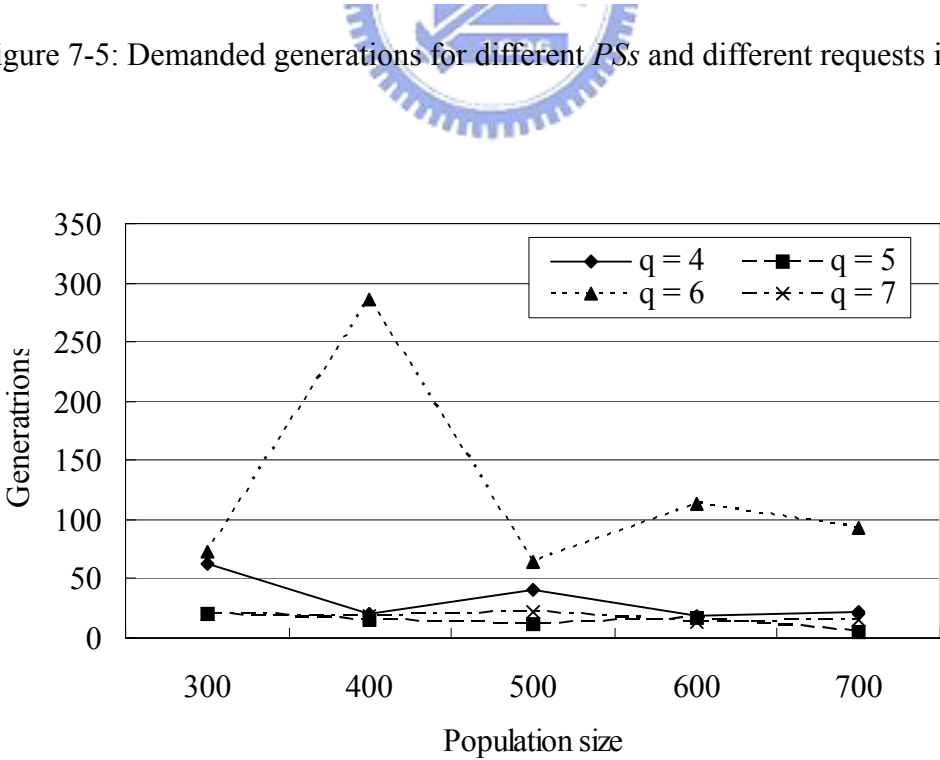


Figure 7-6: Demanded generations for different PSs and requests in net_2

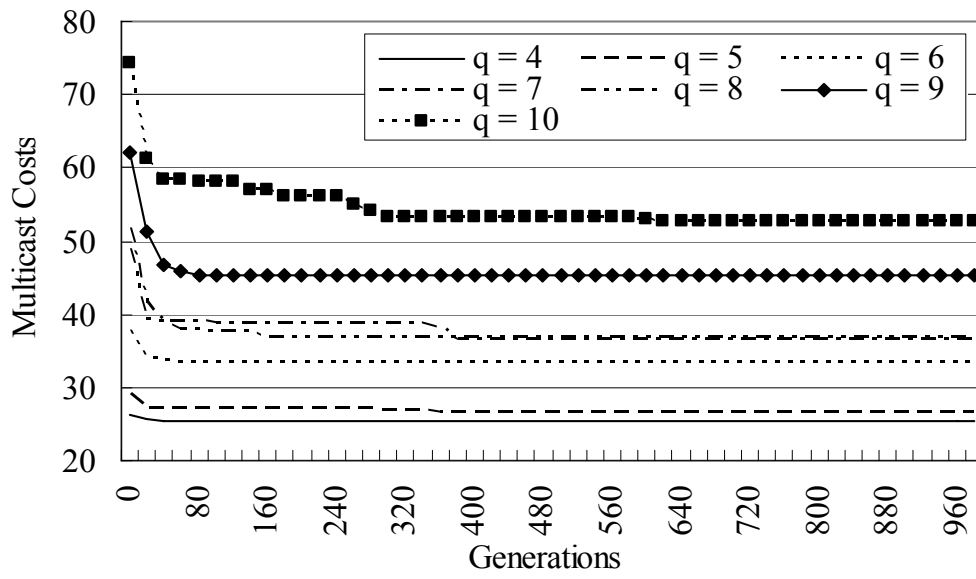


Figure 7-7: Multicast costs for different generations in net_1

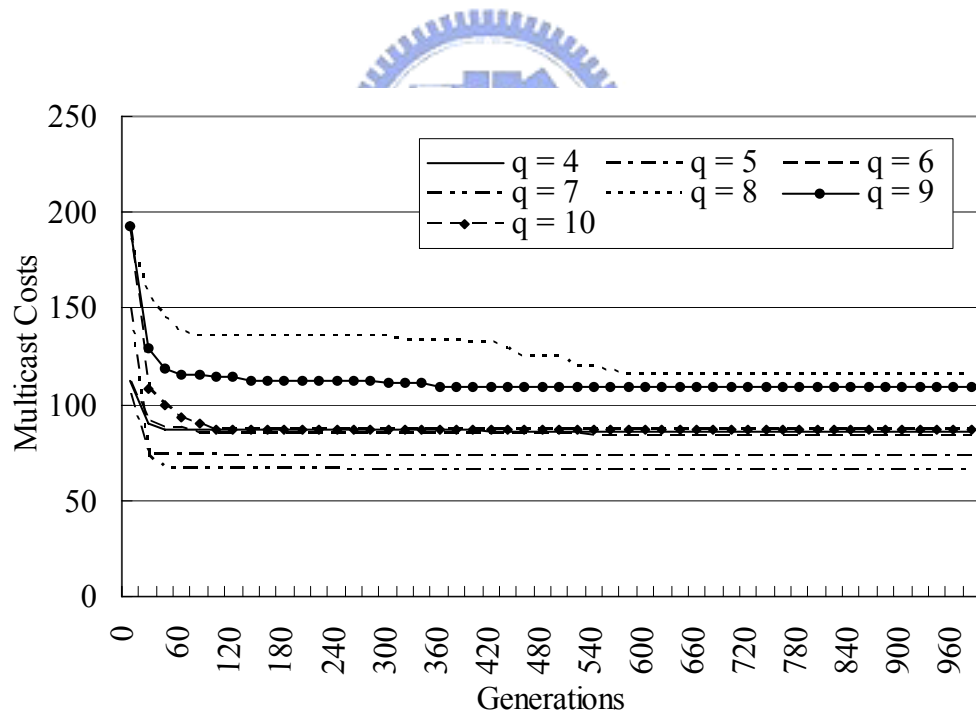


Figure 7-8: Multicast costs for different generations in net_2

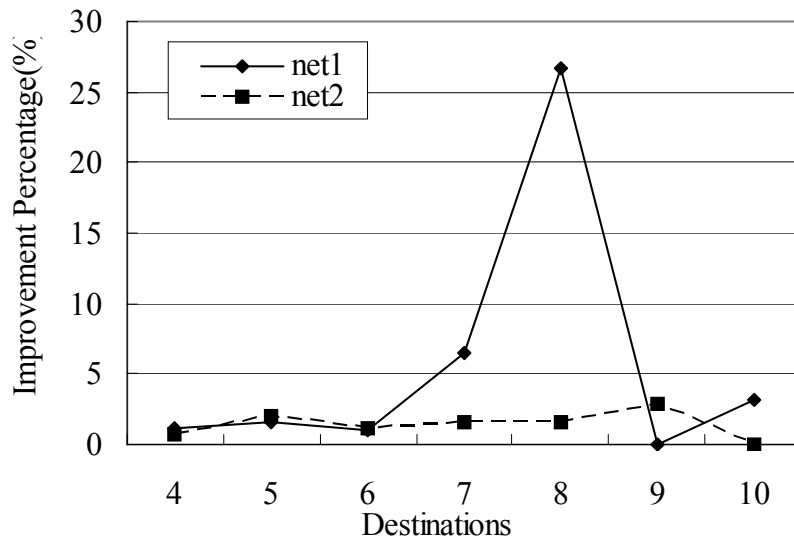


Figure 7-9: Promotion Percentages of multicast costs for different requests

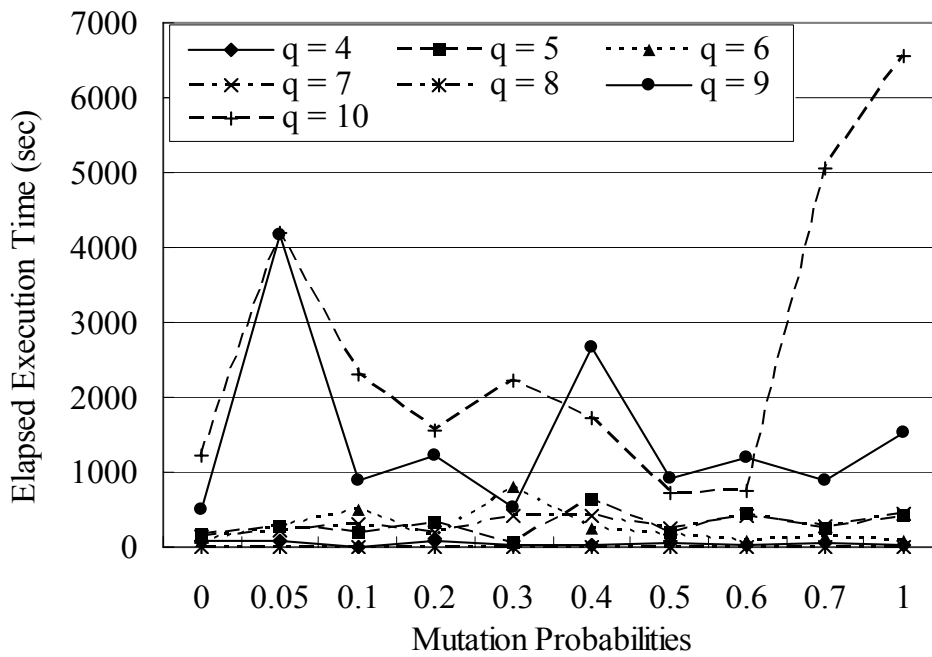


Figure 7-10: Elapsed execution times for different mutation probabilities (MPs) in net_1

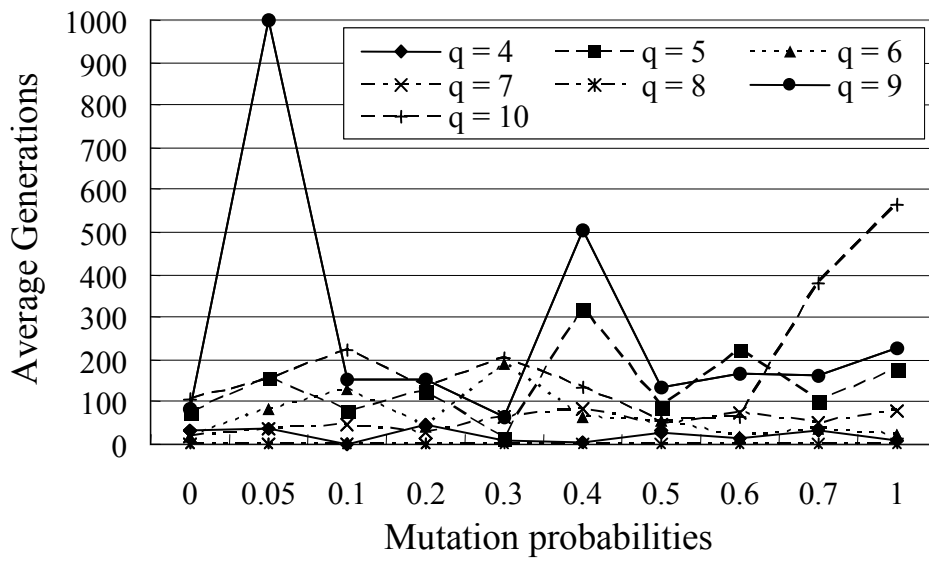


Figure 7-11: Average generations for different mutation probabilities (MPs) in net_1

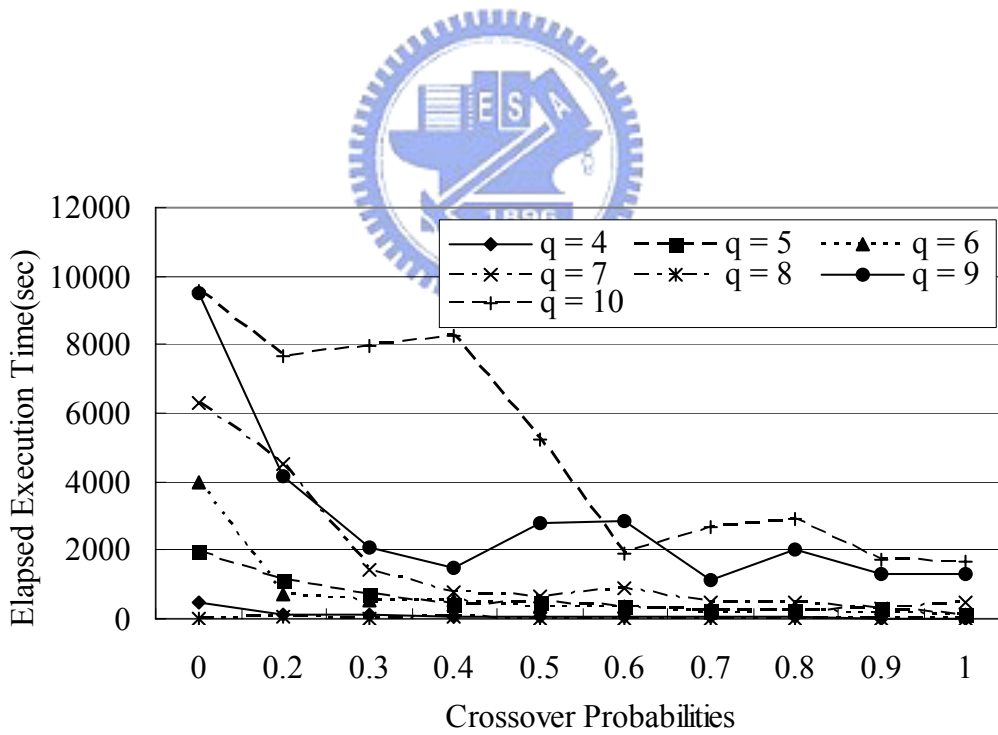


Figure 7-12: Elapsed execution times for different crossover probabilities (CPs) in net_1

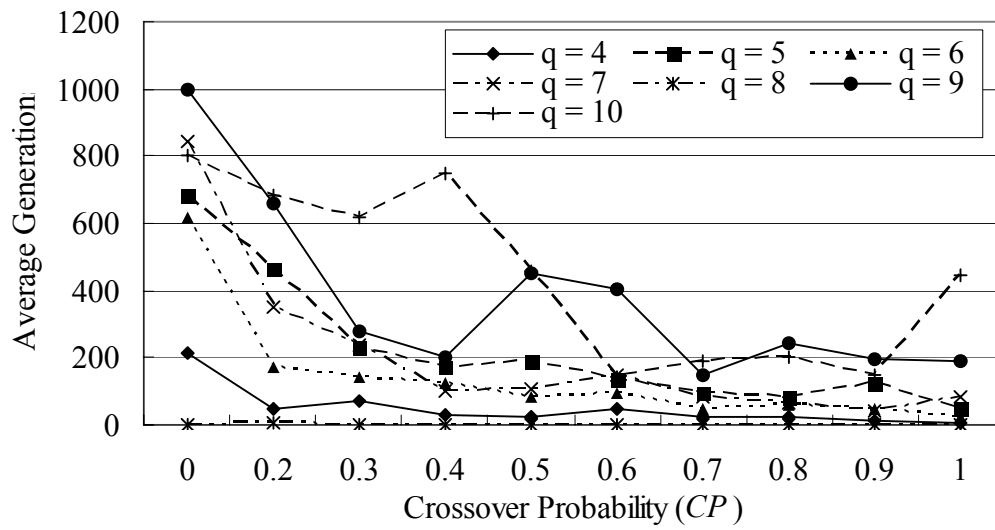


Figure 7-13: Average generations for different crossover probabilities (CPs) in net_1

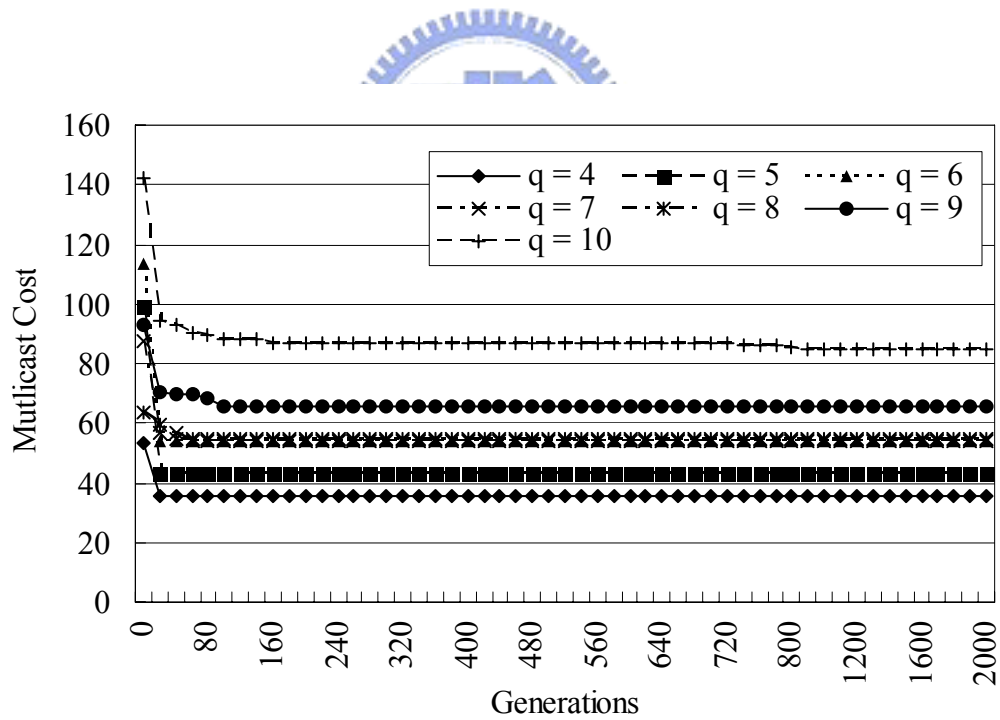


Figure 7-14: Multicast costs for different requests in net_1

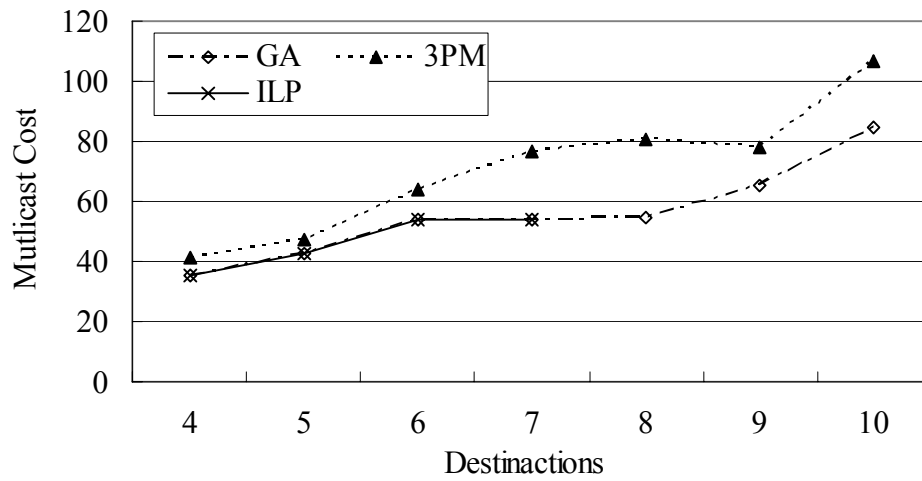


Figure 7-15: Multicast costs for different requests in net_1

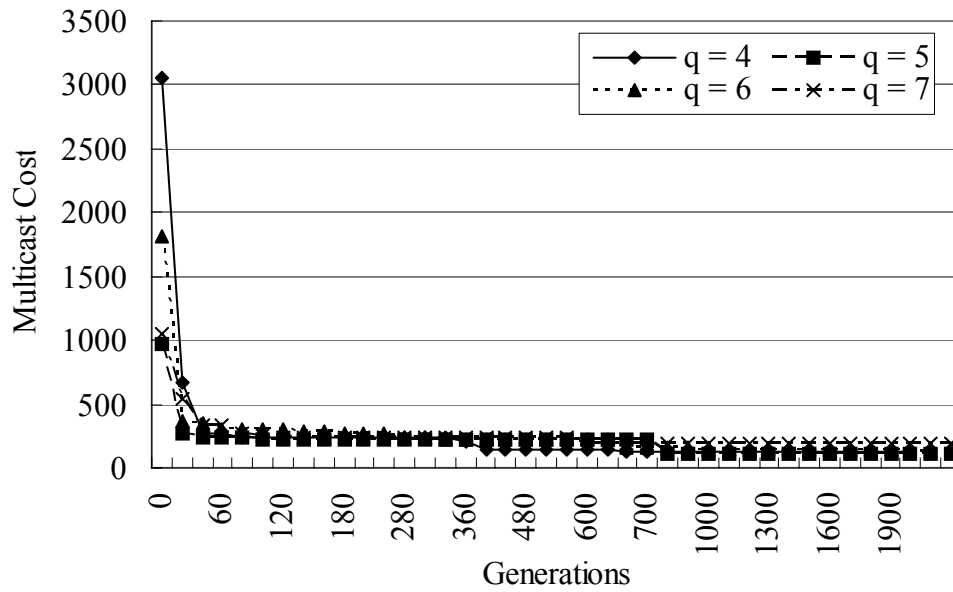


Figure 7-16: Multicast costs for different generations in net_2

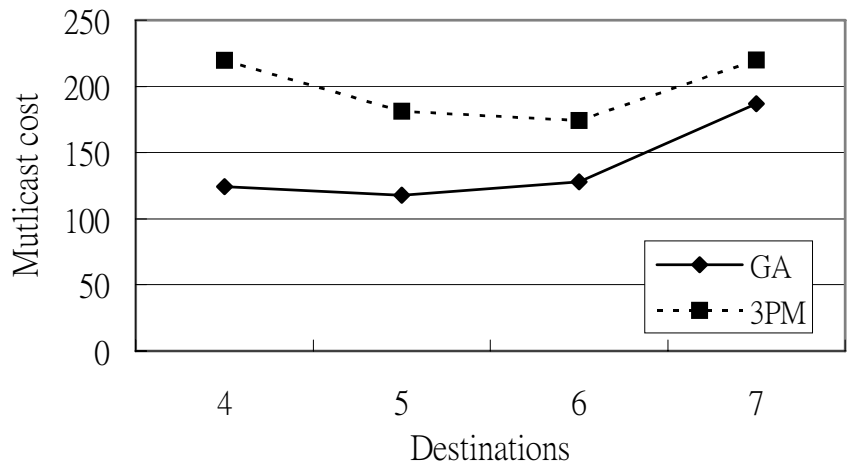


Figure 7-17: Multicast costs for different requests in *net₂*

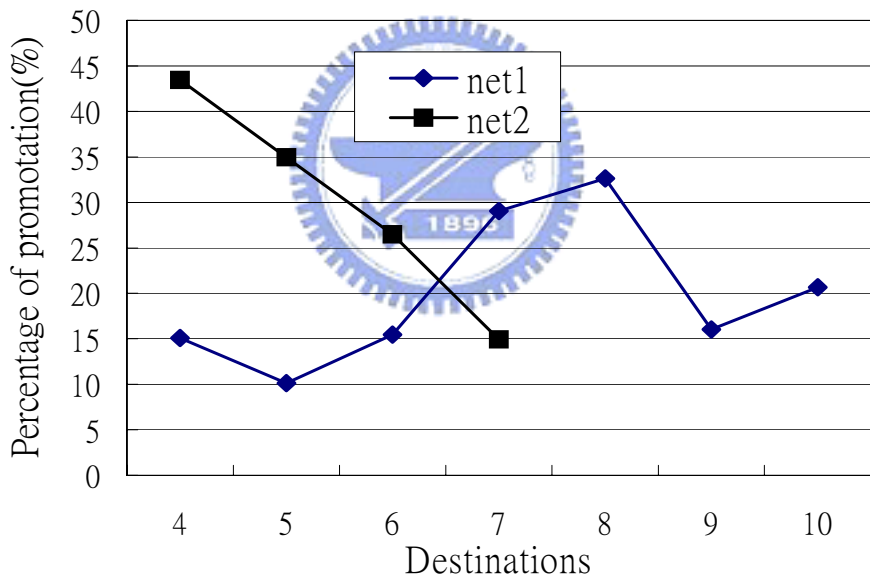


Figure 7-18: Average promotion percentages of multicast cost for different requests

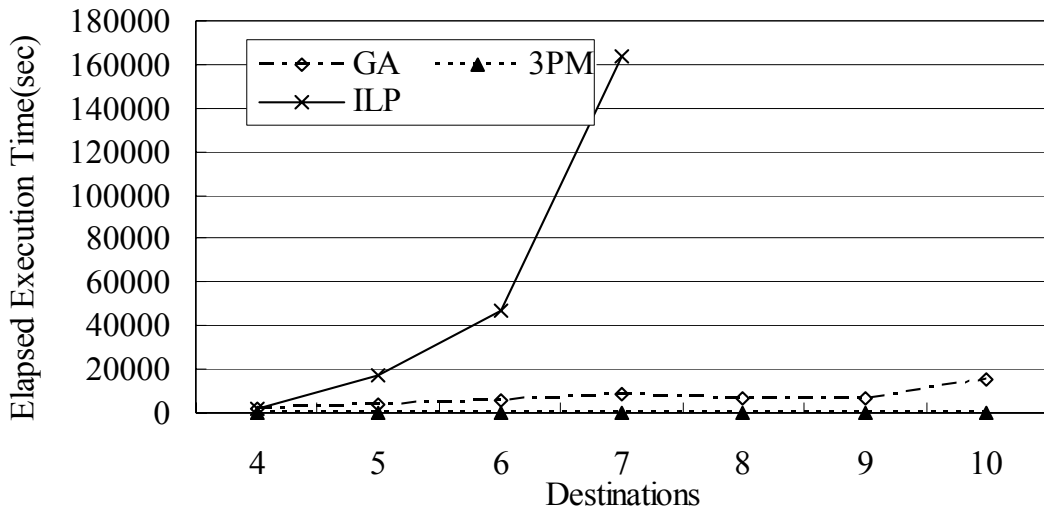


Figure 7-19: Elapsed execution times for different requests

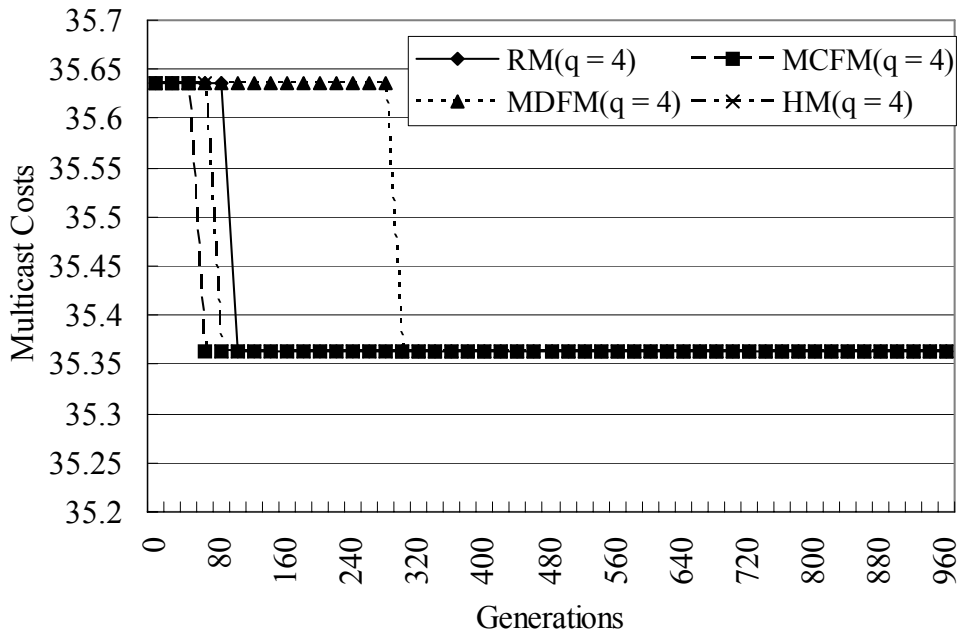


Figure 7-20: Multicast costs for different generations in net_1

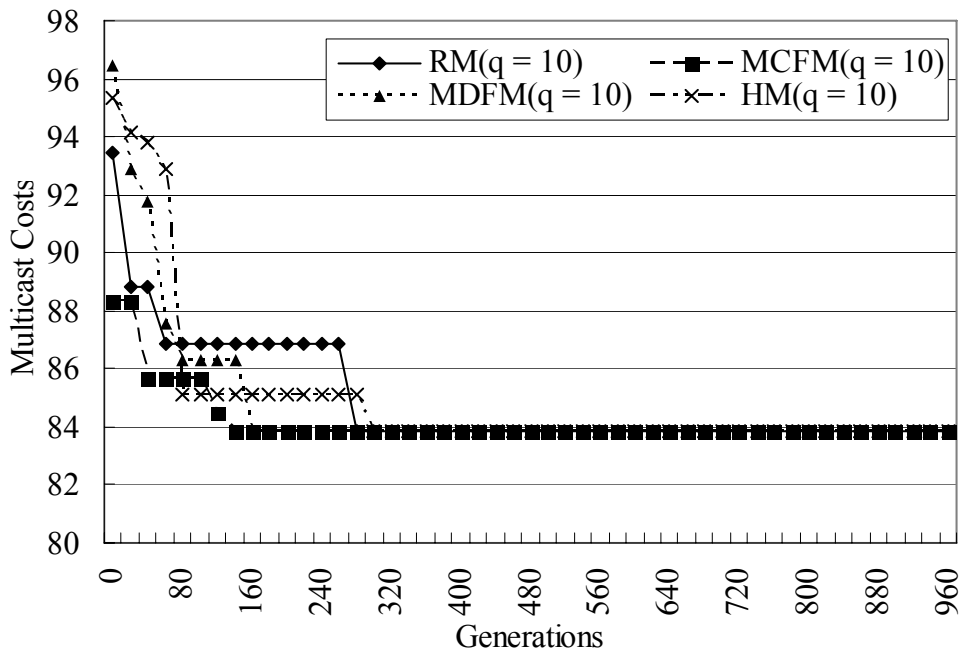


Figure 7-21: Multicast costs for different generations in net_1

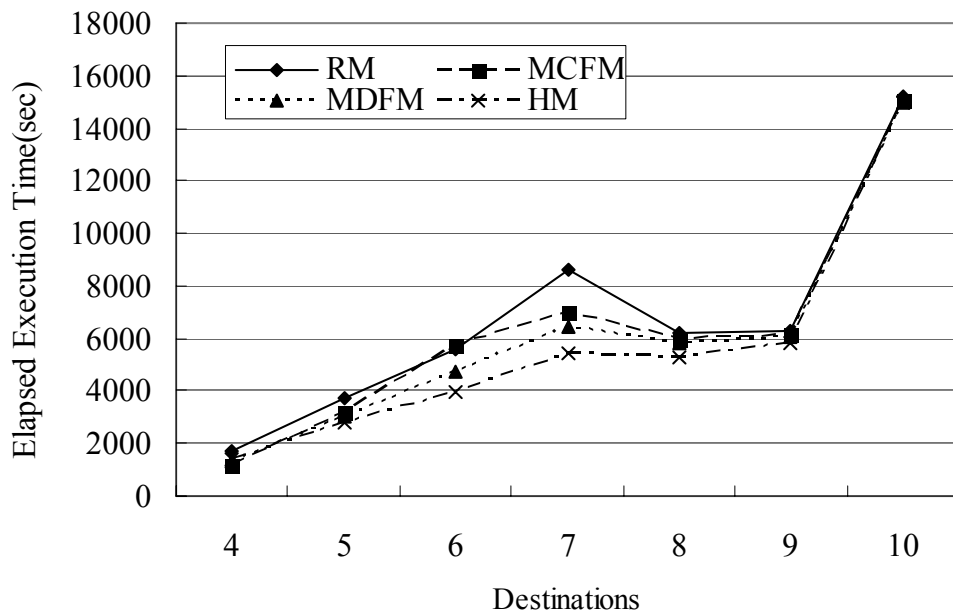


Figure 7-22: Elapsed execution times for different requests in net_1

7.5 Conclusion

In this chapter, the MRP-DC-WWC-SR is studied and a GA model is designed to examine the problem. In the GA model, not only a destination-oriented encoding scheme used to represent polyploid chromosomes in an individual but also multicast tree is presented in the phenotype of an individual. Three general genetic operators, selection, crossover, and mutation, are well formulated such that the GA model is conducive to solve the studied problem. For promoting the efficiency in proceeding of evolution in GA, four types of operators (Chromosome Crossover, Individual Crossover, Chromosome Mutation, and Individual Mutation) and four mutation heuristics (*Random Mutation (RM)*, *Minimum Cost First Mutation (MCFM)*, *Minimum Delay First Mutation (MDFM)*, and *Hybrid Mutation (HM)*) are employed in our model. Experimental results indicate that a solution obtained by GA method is not only better than 3PM but also approximated to the solution found by ILP.



Chapter 8 Heuristics for MRWAP-DC-WWC-SR

For routing requests in the large-scale networks, the networks providing more wavelengths, the requests with enormous number of destinations, or enormous number of requests in the request set, the MRWAP-DC or the URWAP-DC may not be solved in an affordable execution time by using ILP. ACO and GA methods may be applied to find a feasible solution in less elapsed execution time than ILP, but these methods are not suitable to the dynamic MRWAP-DC or URWAP-DC due to the time-consuming execution time. In this chapter, two efficient heuristics Near- k -Shortest-Path-based Heuristic (*NKSPH*) and Iterative Solution Model (*ISM*) proposed to improve the searching efficiency or to diminish searching space will be contributive to find a feasible solution in no time; Furthermore, Due to the improvement, it is conceivable to find a near optimal solution. The elementary concept of the *NKSPH* is based on the two heuristics, finding k -shortest light-paths between the source and each destination may diminish the searching space and finding the optimal solution from all combinations of these k -shortest paths may approximate to the optimal solution. Using the adjustment in the value of k , increasing the value will enlarge the searching space cause to find near optimal solution and to reduce the failure opportunity, but it will consume more execution time than setting k to be a smaller value. Moreover, the *ISM* is a greedy approach applying the minimum spanning tree algorithm to find a spanning tree with minimum transmission delay which results in find a feasible solution in short time. Refining the sub-

light-paths between two nodes to reduce the multicast cast will conduce to obtain a feasible solution under a given execution time criterion. The two heuristics will be introduced in detail as follows.

8.1 NKSPH for MRWAP-DC-WWC-SR

Suppose a *wavelength-based graph* of λ , $G^\lambda = (V, W, E^\lambda, \theta, c, d, \hat{c}, \hat{d}, w)$, is defined as a graph obtained by removing all edges which are not available in λ , where $E^\lambda = \{e \mid e \in E, w(e, \lambda) = 1\}$. For each node pair of the source and each destination, called as a *source-destination pairs*, a light-path whose transmission delay is abiding by the delay bound can be found in G^λ . We consider the graph obtained by the union of the light-paths for all source-destination pairs. Infeasible nodes or cycles may exist in the graph. Here, infeasible nodes mean that the number of outbound edges of a node outnumbers the light splitting capacity of the node. Removing some outbound edges from each infeasible node to become a feasible node and cutting the edge from the cycle to become a tree, the remainder will be a light-tree referred as T . It is should be said with some emphasis that the transmission delay of the light-tree may exceed the given delay bound (i.e., the delay constraint is violated). When the delay constraint is satisfied in T , T will be an assigned light-tree for the request. For these destinations which are not routed by T , another light-tree is found by executing the procedure again for the wavelength-based graph of other wavelength. When the procedure is executed iteratively till all destinations are routed by assigned light-trees, the set of assigned light-trees will be a solution of the problem. Nevertheless, three issues: (1) how to decide the order of chosen wavelengths; (2) how to refine the infeasible nodes to be feasible nodes; (3) how to minimize the number of required wavelengths and the communication cost for each assigned light-tree such that the multicast cost is minimal, are not discussed here.

The above iterative procedure is a greedy approach so an optimal solution may not be found. Extending the iterative procedure, the NKSPH proposed in this chapter does not focus on how to find an optimal assigned light-tree in iteration but on providing a heuristic to find a feasible solution in a large-scale adjustable search space. To avoid the heuristic from heavy loads, three simple strategies include selecting wavelengths in random, preserving the edges connecting to maximum destinations, and selecting the k near shortest light-paths that satisfy the delay bound will be used in the NKSPH. Recall that there are q destinations in D and k light-paths are demanded for each destinations. Each iteration of the above iterative process consists of the following seven steps:

- (1) Choose a wavelength λ and construct the wavelength-based graph G^λ .
- (2) For each destination in G^λ , find at most k light-paths from the source subject to the delay bound.
- (3) For each destination, select one of the k derived light-paths. Unify the q light-paths for all destinations to form a graph. Corresponding to k^q possible combinations of light-paths for q destinations, at most k^q graphs will be constructed.
- (4) For any node violating its capacity constraint (i.e., infeasible node) in each of the q^k graphs, reserve the outbound edges whose number is equal to the light splitting capacity of the node and which covers more number of destinations than the eliminated to form a light-tree.
- (5) For any node violating the input constraint in each graph, reserve the inbound edge with the minimum transmission delay to eliminate the cycle.
- (6) Choose the light-tree with a minimum communication cost out of the q^k light-trees to be a candidate T^λ and remove the destinations in T^λ from D .
- (7) Repeat the previous six steps until D is empty.

The details of the NKSPH procedure are given below.

```

NKSPH( $G, r = (s, D, \Delta), k$ )
{
1.  $\Gamma = \text{NULL}, T^{opt} = \emptyset, C^{opt} = \infty, Dest^{opt} = 0$ 
2. While  $D \neq \emptyset$ 
3.   If  $M = \emptyset$  // No wavelength available for routing the data
4.     Return NULL
5.   Randomly select a wavelength  $\lambda$  from  $M$ 
6.   For each  $\zeta_l \in D$ ,
7.      $P(s, \zeta_l, k_l) = \text{Finding-}k\text{-Near-Shortest-Path}(G^\lambda, s, \zeta_l, k, \Delta)$ 
       //  $k_l, k_l \leq k$ , is the number of light-paths found between  $s$  and  $\zeta_l$ 
8.   End For-loop
9.   For each  $T, T = \bigcup_{1 \leq l \leq q} Path_{\zeta_l}, Path_{\zeta_l} \in P(s, \zeta_l, k_l)$  for all  $l, 1 \leq l \leq q$ 
10.    For each node  $v$  in  $T$  in the breadth first order
11.      If ( $out(T, v) > \theta(v)$ ) //  $v$  is infeasible with respect to the capacity constraint
12.        delete the first  $out(T, v) - \theta(v)$  edges according to the increasing order of
           the numbers of connected destinations.
13.      If ( $in(T, x) > 1$ ) //  $x$  violates the input constraint
14.        delete the first  $in(T, v) - 1$  edges according to non-increasing order of
           the transmission delay
15.      End For-loop
16.      If ( $D(T) > Dest^{opt}$  and  $c(T) < MD(T)$ )
           or ( $D(T) = Dest^{opt}$  and  $c(T) < C^{opt}$ ) //  $MD(T) = \sum_{\zeta_l \in V(T) \cap D} \min_{P \in P(s, \zeta_l, k_l)} c(P)$ 
17.         $C^{opt} = c(T)$ 
18.         $Dest^{opt} = D(T)$ 
19.         $T^{opt} = T$ 
20.    End For-loop
21.     $\Gamma = \Gamma \cup \{(T^{opt}, h^{T^{opt}})\}, D = D - V(T^{opt}),$  where  $h^{T^{opt}} : E(T^{opt}) \rightarrow \{\lambda\}$ 
22.  End While-loop
23.  Return  $\Gamma$ 
}

```

In Step 7 *Finding- k -Near-Shortest-Path*($G^\lambda, s, \zeta_l, k, \Delta$) is a procedure to call for finding k near shortest light-paths with minimum transmission delays between s and ζ_l subject to the delay bound Δ . Applying these k -shortest path algorithms developed in [23] and [56] can find

the k shortest light-paths with a minimum communication cost. However, these algorithms cannot be used directly to find constrained shortest light-paths with a minimum communication cost under the delay bound. Furthermore, it is difficult to revise these algorithms for deriving constrained shortest light-paths. The facts that these algorithms are complicated to implement or modify to meet the need and that optimal light-paths are not necessarily optimal suggest us to deploy a simple approach to finding the shortest paths in a timely fashion. The implementation of *Finding-k-Near-Shortest-Path* develops to an iterative procedure consisting of three steps: (1) apply *Dijkstra's* shortest path algorithm [19] to find a light-path with minimum transmission delay (i.e., the light-path is a constrained shortest light-path); (2) keep the light-path if its transmission delay satisfies the delay constraint; and (3) delete the edge whose transmission delay is minimum in this light-path from G^λ . The procedure terminates when k near shortest light-paths have been attained or no more light-path can be found.

Assume the number of light-paths found for the destination ζ_l is k_l , $k_l \leq k$. Let $P(s, \zeta_l, k_l)$ be the set of the constructed light-paths. A graph is formed by selecting one light-path $Path_{\zeta_l}$ from $P(s, \zeta_l, k_l)$ for each $\zeta_l \in D$ and then unifying these selected light-paths. Denote this graph by $T = \bigcup_{1 \leq l \leq q} Path_{\zeta_l}$. The derived graph may contain cycles or infeasible nodes. In Steps 11 to

12 and 13 to 14, all nodes in T are examined to verify whether the capacity constraint and input constraint are violated or not. For each node x violating the capacity constraint, the first $out(T, v) - \theta(v)$ edges in increasing order of the numbers of connections to destinations will be eliminated in Step 12 such that each internal node is feasible. In Step 14, the edge with the minimum transmission delay will be reserved when there is more than one edge connected to the node. It implies that finding a feasible solution is more important than finding an optimal solution. All cycles can be detected and removed in Step 13~14. After these steps, a light-tree will be obtained. In Step 16, $D(T)$ represents the number of destinations contained in T and

$MD(T) = \sum_{\zeta_l \in \mathcal{V}(T) \cap D} \min_{P \in \mathcal{P}(s, \zeta_l, k_l)} c(P)$ represents the sum of minimum communication costs of the destinations in T . The case $c(T) < MD(T)$ implies that T will include at least an MC node to reduce the communication cost. Therefore, the checking in Step 16 implies the heuristic that the light-tree rerouting more destinations and consuming less communication will prefer being chosen. In Steps 16 to 19, a light-tree T^{opt} consuming less communication cost or covering more destinations will be kept as a local optimal light-tree by using λ . Therefore, T^{opt} using λ represents an assigned light-tree denoted by $(T^{opt}, h^{T^{opt}})$, where $h^{T^{opt}} : E(T^{opt}) \rightarrow \{\lambda\}$. Steps 2 to 22 will be executed iteratively until no wavelength is available or the destination set D is empty. The former case leads to the report that the request cannot be successfully routed by our algorithm.

8.2 ISM for MRWAP-DC

Based on the formulation in Chapter 7, the Iterative Solution Model (ISM) proposed to solve the MRWAP-DC-WWC-SR consists of two procedures: (1) *Selecting Wavelength Procedure (SWP)* to choose a wavelength to construct a wavelength-based graph, and (2) *Finding Assigned Light-tree Procedure (FALP)* to find a light-tree under delay bound from the wavelength-based graph. In the *SWP*, the order of selected wavelengths is anticipated by two heuristics *Maximum W-Feasible Edges Assigning First (MaxE)* and *Minimum Requests Assigning First (MinR)*. In the *FALP*, two simple heuristics *Maximum Depth Reserving First (MaxDepth)* and *Maximum Destinations Reserving First (MaxDest)* are proposed to construct a light-tree to cover destinations.

8.1.1 Selecting Wavelength Procedure (SWP)

A light-tree with minimum communication cost is desired to be found in iteration, and it

will not be modified again. The parts of destinations are routed by the assigned light-tree, but it is possible that no light-tree can be found to route the parts of the remaining destinations or that a found light-tree may far from the optimal light-tree. To avoid that routing the request is unsuccessfully or needs more multicast cost, it is reasonable to propose an evaluation function such that a wavelength with maximum evaluated values is chosen first. A dramatic or perfect evaluation function benefits that all used wavelengths will be selected exactly and that the found assigned light-forest may be approximately equal to the optimal solution in high possibility. Nevertheless, finding the evaluation function may be a miracle or may consume unaffordable execution time.

In SWP, an evaluation function $Eval(\lambda)$ for λ is defined to reflect the objective that the order of selecting wavelength is exact, where the wavelength λ with high value of $Eval(\lambda)$ means that using λ to route the request may be conducive to find an assigned light-forest with less multicast cost. Nevertheless, there is no doubt about that $Eval(\lambda)$ is very hard to define accurately or that it can be computed to get the evaluation value in unaffordable elapsed execution time. Two simple greedy heuristics, *Maximum W-Feasible Edges Assigning First (MaxE)* and *Minimum Requests Assigning First (MinR)* are proposed as follows.

(1) *MaxE heuristic*

The *MaxE* heuristic is based on the assumption that the wavelength-based graph with more edges benefits to find a light-tree with less communication cost in higher possibility. For a set of unassigned wavelengths M' , $M' \subseteq M$, the wavelength $\lambda^{opt} \in M'$ satisfying $|E^{\lambda^{opt}}| \geq |E^\lambda|$ for all $\lambda \in M'$ will be chosen first and the $Eval(\lambda)$ is defined as :

$$Eval(\lambda) = |E^\lambda|, \text{ where } E^\lambda = \{e | e \in E, w(e, \lambda) = 1\}$$

(2) *MinR heuristic*

A wavelength which has been used to route minimum number of requests represents its utility rate is lowest among all wavelengths. The lowest wavelength selected first benefits to

balance the transmission load of wavelength, which can reduce the blocking rate of routing requests. Therefore, the *MinR* heuristic is proposed and defined as:

$$Eval(\lambda) = 1 / (\text{the number of requests routed by using the wavelength } \lambda)$$

8.1.2 Finding Assigned Light-Tree Procedure (FALP)

In this procedure, the optimal light-tree with minimum communication cost for using the selected wavelength is expected to be found. Nevertheless, finding the optimal light-tree is NP-hard such that proposing an efficient heuristic to find a near optimal light-tree in polynomial time is more important than to find the optimal light-tree. The FALP divided into three Steps, *generating a weak candidate*, *refining the weak candidate*, and *eliminating infeasible nodes* will be described as follows.

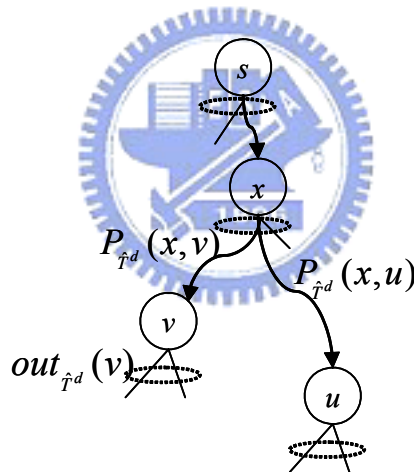


Figure 8-1: A weak candidate \hat{T}^d

(1) *Generating a Weak Candidate*

All MDLPs between the source and all destinations need to be checked the condition that their transmission delay must be smaller than or equal to the delay bound. For only one light-path with minimum transmission delay between two nodes, the graph constructed by merging these MDLPs must be a tree. The delay constraint is not involving in constructing the tree, so the tree is called a *weak candidate*. Dissimilar to the definition in Chapter 7.1 that a *candidate*

is a multicast tree satisfying the delay constraint and the destination constraint, the *weak candidate* satisfies the delay constraint only such that it maybe is neither a light-tree nor an optimal multicast tree. Therefore, the major objective in the Step is to find a feasible solution.

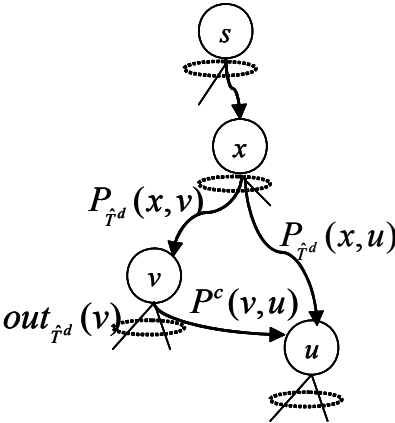


Figure 8-2: $\hat{T}^d \cup P^c(v, u)$

(2) *Refining the Weak Candidate*

The refining process is also an iterative process which refines the light-path between two nodes to reduce multicast cost. The iterative process consists of three sub-steps, choosing a node-pair (u, v) to be refined, rerouting u to v , and checking the new weak candidate to have less multicast cost. The second sub-step is described first and the others are described as follows.

For the weak candidate \hat{T}^d shown in Figure 8-1, suppose that u is rerouted to v and x is the nearest common predecessor node. The second sub-step consists of applying the *Prim's* algorithm [59] to clean up the cycles contained in the graph which is formed by concatenating $P^c(v, u)$ to \hat{T}^d shown in Figure 8-2, and deleting all leaf nodes which do not belong to the destination set. The former is implemented by $Prim^c(\hat{T}^d \cup P^c(v, u))$, where $Prim^c$ is an operation of finding *Minimum Spanning Tree (MSpT)* with communication cost from $\hat{T}^d \cup P^c(v, u)$.

Nevertheless, the order and the number of node-pairs which need to be rerouted are hard to be recognized. The *cost-difference* (CD) of a node-pair (u, v) , $CD(\hat{T}^d, u, v)$, is proposed in the first sub-step to predict the expectation of the multicast cost promotion on rerouting u to v . The node-pair (u, v) having highest cost-difference value indicates that u is rerouted to v may reduce multicast cost more efficient than others. To predict the cost-difference value of node-pair accurately, $CD(\hat{T}^d, u, v)$ will be well defined as follows.

Suppose that \hat{T}^n is obtained after the second sub-process. It can be observed that the number of outbound edges of v in \hat{T}^n can be increased by 1 to be $out_{\hat{T}^d}(v)+1$. For the case $out_{\hat{T}^d}(v) < \theta(v)$, $out_{\hat{T}^n}(v)$ cannot be greater than $\theta(v)$ such that the wavelength consumption of \hat{T}^n is changeless. Therefore, the anticipation of multicast cost promotion for rerouting u to v is the difference between the communication cost of the eliminated path $P_{\hat{T}^d}(x, u)$ and the communication cost of the concatenated path $P^c(v, u)$; that is, $\alpha(c(P_{\hat{T}^d}(x, u)) - c(P^c(v, u)))$. Otherwise, the promotion needs to subtract the cost of one extra wavelength (equal to β) and the communication cost of the partial light-path connecting s and v (i.e., $c(P_{\hat{T}^d}(s, v))$) for using the extra wavelength. Therefore, the $CD(\hat{T}^d, u, v)$ of the node-pair (u, v) is defined as :

$$CD(\hat{T}^d, u, v) = \begin{cases} \alpha(c(P_{\hat{T}^d}(x, u)) - c(P^c(v, u))), & \text{if } out_{\hat{T}^d}(v) < \theta(v) \\ \alpha(c(P_{\hat{T}^d}(x, u)) - c(P^c(v, u)) - c(P_{\hat{T}^d}(s, v))) - \beta, & \text{otherwise} \end{cases}, \quad (8-1)$$

where x is the nearest common predecessor node of u and v . The *best node* of u in \hat{T}^d , $\delta(u)$, is defined as the node in V satisfying $CD(\hat{T}^d, u, \delta(u)) \geq CD(\hat{T}^d, u, v)$ for all $v \in V - \{u\}$. In the first sub-step, the node-pair $(u, \delta(u))$ with maximum cost-difference is chosen first. Nevertheless, it may not be guaranteed that the transmission delay of \hat{T}^n is under the delay bound nor that the multicast cost of \hat{T}^n is reduced. Therefore, the final sub-step checks two

conditions $d(\hat{T}^n) \leq \Delta$ and $f(\hat{T}^n) < f(\hat{T}^d)$ to confirm that \hat{T}^n is a better weak candidate than \hat{T}^d . The Step will be terminated when no better weak candidate can be found again.

(3) *Eliminating Infeasible Nodes*

We know that a weak candidate may contain some infeasible nodes such that it is not a light-tree. In the step, some outbound edges in infeasible nodes will be eliminated from the weak candidate to form a light-tree. Because the solution model is a greedy approach, how to eliminate edges is important and complex. To reach the objective of the problem, the *reservation weight function*, *r-weight*, is defined to give a *reservation weight* for each node. The greedy method, reserving the edges which connect the nodes with high reservation weight may benefit to reduce the multicast cost or the blocking rate of routing requests, will be adopted in the step. By implementing the *r-weight* according to different heuristics, different light-trees can be attained due to eliminating different edges. Suppose v be an infeasible node (i.e., $out_{\hat{T}^n}(v) > \theta(v)$) in \hat{T}^n . It is necessary to eliminate $out_{\hat{T}^n}(v) - \theta(v)$ outbound edges whose reservation weights are smaller than others from $out_{\hat{T}^n}(v)$ outbound edges of v such that v becomes a feasible node. There are two different heuristics, *Maximum Depth Reserving First (MaxDepth)* and *Maximum Destinations Reserving First (MaxDest)*, proposed for the *r-weight*. Routing the request to one destination by using the light-path with maximum depth require more links than other light-paths implies that the request routed to the destination is blocked in high probability for needing more w-feasible edges. Therefore, the *MaxDepth* heuristic applies the observation that these edges connecting these sub-trees with maximal depth are reserved first. Nevertheless, in the *MaxDest* heuristic, it assumes that the sub-tree covering more destinations may indicate routing the request with fewer wavelengths.

To summarize the discussion above, the skeleton of the *FALP* will be described as follows.

$FALP(G, r = (s, D, \Delta))$

{

Step 1. generating a weak candidate

1.1. $\hat{T}^d = \emptyset$

1.2. **for** all d_i in D

1.3. **if** $(d(P^d(s, d_i)) \leq \Delta)$ // $P^d(s, d_i)$ is the MDLP from s to d_i in $G(V, E^\lambda)$

$$\hat{T}^d = \bigcup_{d_i \in D} P^d(s, d_i)$$

1.4. **end for-loop**

1.5. **if** $\hat{T}^d = \emptyset$ // If \hat{T}^d is null, wavelength λ cannot be used to reroute the request

1.6. **return** \emptyset

Step 2. refining the weak candidate to reduce multicast cost

2.1. **for** each $u \in \hat{T}^d$

2.2. **keep** $(u, \mathcal{A}(u))$ in the array K by sorting $CD(\hat{T}^d, u, \mathcal{A}(u))$ in decreasing order

2.3. **end for-loop**

2.4. **for** each node-pair (u, v) in K

2.5. $\hat{T}^n = Prim^c(\hat{T}^d \cup P^c(u, v))$ // $Prim^c$ is a operation of finding *Minimum*

// *Spanning Tree (MSpT)* with minimum communication cost

2.6. **delete** all leaf nodes in \hat{T}^n which does not belong to D

2.7. **if** $(d(\hat{T}^n) \leq \Delta)$ // checking the transmission delay of \hat{T}^n

2.8. **if** $(f(\hat{T}^n) < f(\hat{T}^d))$

2.9. $\hat{T}^d = \hat{T}^n$

2.10. **go to** 2.1

2.11. **end-for loop**

Step 3. eliminating infeasible nodes

3.1. **check** all node x in \hat{T}^n from leaves bottom up

3.2. **while** $(x$ is not leaf node)

if $(out_{\hat{T}^n}(x) > \theta(x))$ // x is a infeasible node

3.3. **eliminate** $out_{\hat{T}^n}(x) - \theta(x)$ edges which with minimum r -weight(x)

3.4. **return** \hat{T}^n // will be a light-tree

}

8.3. Experiments

The approach used in the experiments to evaluate the performance of the two proposed heuristics has been introduced in Section 4.3 or is referred to Waxman [78]. We assume that Δ equals to 1.2 time of the maximum in the following experiments (i.e., $\chi = 1.2$ discussed in Chapter 5.3). Eight types of networks introduced in Chapter 5.3 were tested: 30, 40, 50, 60, 70, 80, 90, 100 switches ($n = 30, 40, 50, 60, 70, 80, 90,$ and 100), for each of which 60 different requests were randomly generated. Each 60 requests are categorized into 3 groups corresponding to 2 destinations ($q = 2$), 3 destinations ($q = 3$), and 4 destinations ($q = 4$).

The experiments consisting of four parts: comparisons for different values of k in NKSPH, comparisons among ILP, NKSPH, and ISM, comparisons between MaxDepth and MaxDest, and comparisons between MaxE and MinR.

8.3.1 Comparisons for Different Values of k in NKSPH

Next we proceed to discuss the efficiency of NKSPH for different values of k . Numerical results are summarized in Table 8-1. For the column corresponding to each request group ($q = 2, q = 3,$ and $q = 4$), we keep track of feasible solutions found ($\#Succ$), average multicast cost (MC), and average elapsed execution time (ET) over every 20 requests. From the numerical results, we have the following observations:

- (1) The elapsed execution time of the NKSPH is related to the value of k , the number of destinations, and the number of nodes in the network. For example, $ET = 0.01$ seconds for $n = 40, q = 2,$ and $k = 2$ and $ET = 3.24$ seconds for larger values of $n = 100, q = 4,$ and $k = 20$.
- (2) Increasing the value of k usually reduces the multicast cost. Such a phenomenon is significant when the request is associated with more destinations or the network

has more nodes. For example, it can reduce multicast cost 4.18% ($\frac{119.65 - 114.64}{119.65} \times 100\%$) for $n = 100$, $q = 4$ from $k = 4$ to $k = 20$, but there is no improvement for $n = 100$, $q = 2$ from $k = 4$ to $k = 20$. Nevertheless, in some cases the increase multicast cost is higher when the request is associated with more destinations and the network with more nodes. For example, $MC = 117.35$ for $n = 100$, $m = 4$, and $k = 8$, but $MC = 118.30$ when $k = 10$. It seems reasonable to conclude that choosing a suitable value of k cannot only reduce the multicast cost but also save the execution time.

8.3.2 Comparisons among ILP, NKSPH, and ISM

In order to compare ILP, NKSPH, and ISM, only these requests solved successfully in ILP are addressed in the experiments. For each combination of network types and different method (ILP vs. ISM/NKSPH with different values of k), all experimental results shown in Table 8-2 are summarized into three blocks for three groups of requests ($q = 2$, $q = 3$, and $q = 4$). Each block displays the number of optimal solutions ($\#Opt$) produced by our heuristic, average multicast cost deviation (Dev) of the found solutions from the optimal ones found by the ILP, MC , and ET , where Dev is defined as:

$$Dev = \frac{\sum_r \frac{f(\Gamma_r^k) - f(\Gamma_r^{opt})}{f(\Gamma_r^{opt})} \times 100\%}{\#Opt_{ILP}}, \quad (8-2)$$

where $f(\Gamma_r^k)$ and $f(\Gamma_r^{opt})$ are the multicast cost of the feasible solution Γ_r^k found for k by NKSPH or ISM and the multicast cost of the optimal solution Γ_r^{opt} found by the ILP formulation for request r , and $\#Opt_{ILP}$ is the number of requests solved successfully by the ILP. In the first row, $\#Opt = 20$, $MC = 63.72$ and $ET = 816.83$ dictate the number of requests that

were successfully solved, the average multicast cost, and the average elapsed execution time for $n = 30$, $q = 2$ by using the ILP. For different network types and three groups of requests ($q = 2$, $q = 3$, and $q = 4$), the deviation (Dev) of the worst cases in each group are shown in Table 8-3. From the numerical results, we make the following observations:

- (1) The elapsed execution time of NKSPH is much shorter than that of the ILP, but is much longer than that of the ISM. For example, $ET = 0.01$ seconds or $ET = 0.00$ (which the elapsed execution time is small than 0.01 seconds) in $n = 30$ and $q = 4$ by using ISM, $ET = 0.10$ seconds or $ET = 0.18$ seconds in $n = 30$, $q = 4$, and $k = 8$ or $k = 20$ by using NKSPH, but $ET = 7,094.21$ seconds by using the ILP.
- (2) Although NKSPH cannot always find the optimal solution, the solutions it has produced are close to optimal ones. Nevertheless, ISM seldom finds the optimal solution. For large-scale networks, the greater the value of k is, the more the multicast cost is reduced. For example, $Dev = 9.93\%$ for $n = 60$, $q = 4$, and $k = 8$, and $Dev = 4.08\%$ by using NKSPH, which is much smaller for $k = 15$ or 20. $Dev = 33.90\%$ for $n = 60$ and $q = 4$ by using ISM.

8.3.3 Comparisons between MaxDepth and MaxDest

The experimental results of execution times and multicast cost distances of two heuristics (*MaxDepth* and *MaxDest*) for different requests are shown in Figures 8-3 and 8-4 for the network with 30 nodes ($n = 30$), where x is defined as a value of β divided by α ; that is, $x = 1, 10, 50$ are the cases of choosing $\alpha = 1$ and $\beta = 1$, choosing $\alpha = 1$ and $\beta = 10$, and choosing $\alpha = 1$ and $\beta = 50$, respectively. According to these experimental results, we make the following observations:

- (1) The elapsed execution times of both are proportional to the numbers of destinations associated in requests. Moreover, *MaxDest* needs less execution time

and increase of execution time is gentler than *MaxDepth*. For some requests, the execution time increases sharply for *MaxDepth*; for example, $q = 13$ and 14 in Figure 8-3.

- (2) *MaxDepth* may have high probability to find a light-forest with less multicast cost but spend more execution time. The multicast cost distance in Figure 8-4 is the value of subtracting multicast cost by using *MaxDepth* from multicast cost by using *MaxDest*; that is, the multicast cost distance using positive value means that *MaxDepth* can find a light-forest with less multicast cost. For different x with high value, the *MaxDepth* finding a light-forest will have less multicast cost in the case of requests with fewer destinations, but it may not be obviously in the case of request with more destinations.

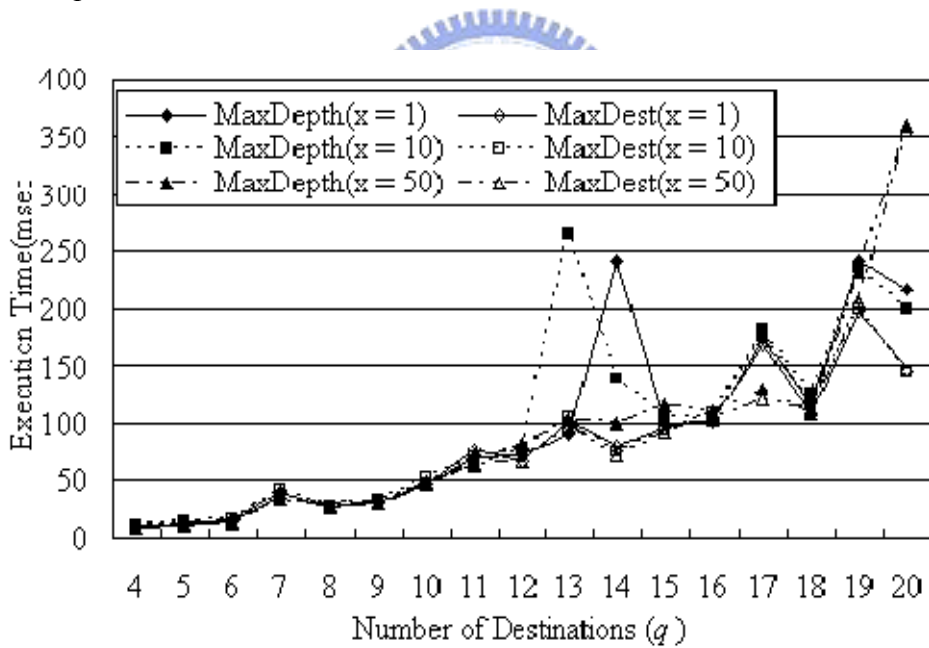


Figure 8-3: Comparisons of execution time for different requests

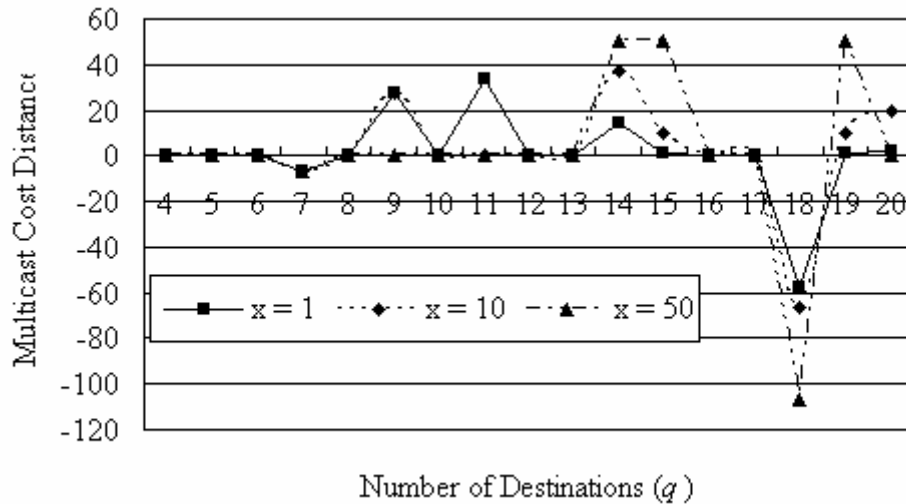


Figure 8-4: Comparisons of multicast cost distance for different requests

8.3.4 Comparisons between MaxE and MinR

In the experiments, the two heuristics (*MaxE* and *MinR*) are applied to reroute the set of 100 different requests with 5 destinations in the networks with 4 wavelengths and with different numbers of nodes ($n = 100, 90,$ and 80). According to the experimental results shown in Table 8-4, the third and the fourth rows, present the total requests which can be routed successfully and total light-trees in these light-forests found successfully, where these request are named as *success*. The next two rows are used to describe the total communication cost and total multicast cost of success, respectively. *ETS* (elapsed execution time of success requests), *ETF* (execution time of failure requests), and total execution time which is a sum of *ETS* and *ETF* are described in following. The final row presents the sum of edges of light-trees in success. We may make the following observations:

- (1) In the phase of routing 100 requests, there are about 30% of requests rerouted successfully and the success rates are proportional to the numbers of nodes in network. We can derive that it will need more wavelengths to rerouted more requests concurrently in network with less nodes.

- (2) The performances of the two heuristics *MaxE* and *MinR* cannot be distinguished. Nevertheless, according to the comparisons between the *ETS* and the *ETF*, the *MaxE* needs less execution time to reroute these requests successfully than the *MinR*.
- (3) According to the ratio between *ETS* and *ETF*, the *MaxE* seems to be suitable to be applied to the routing problem with more nodes and more concurrent requests to reroute successfully; on the contrary, the *MinR* may be utilized.

8.4. Conclusion

In this chapter, an extended multicast routing and wavelength assignment problem with delay constraints in WDM networks with heterogeneous light splitting capabilities (MRWAP-DC-WWC-SR) is studied. Because the ILP formulation proposed in Chapter 5 cannot solve instances of the studied problem in large-scale networks, statistics from computational experiments evince that the two proposed heuristics can produce near-optimal solutions in an acceptable execution time.

Table 8-1: Experimental results for different values of k in NKSPH

n	k	$q = 2$			$q = 3$			$q = 4$		
		#Succ	MC	ET	#Succ	MC	ET	#Succ	MC	ET
40	2	20	52.75	0.01	20	93.80	0.02	20	103.15	0.02
	4	20	50.10	0.02	20	89.60	0.03	20	95.75	0.04
	6	20	50.10	0.02	20	88.35	0.04	20	93.20	0.06
	8	20	50.10	0.02	20	88.35	0.05	20	92.35	0.09
	10	20	50.10	0.03	20	90.20	0.06	20	91.95	0.10
	15	20	50.10	0.03	20	90.20	0.10	20	93.15	0.25
	20	20	50.10	0.03	20	90.20	0.08	20	93.15	0.28
60	2	20	56.85	0.03	20	80.30	0.06	20	105.40	0.13
	4	20	53.00	0.05	20	74.00	0.17	20	102.35	0.21
	6	20	52.45	0.07	20	72.75	0.22	20	96.75	0.42
	8	20	52.25	0.15	20	73.35	0.30	20	95.65	0.56
	10	20	52.25	0.11	20	73.35	0.18	20	92.50	0.38
	15	20	52.20	0.11	20	72.65	0.24	20	91.55	0.89
	20	20	52.20	0.11	20	72.65	0.48	20	91.40	1.19
80	2	20	57.05	0.06	20	89.40	0.09	20	116.80	0.20
	4	20	54.40	0.19	20	86.95	0.23	20	112.75	0.32
	6	20	54.40	0.15	20	85.25	0.27	20	111.10	0.68
	8	20	53.95	0.31	20	83.55	0.61	20	108.15	0.83
	10	20	53.95	0.30	20	83.05	0.59	20	107.40	1.07
	15	20	53.95	0.20	20	82.65	0.64	20	109.15	0.92
	20	20	53.95	0.21	20	82.65	0.47	20	108.45	1.15
100	2	20	60.05	0.11	20	94.65	0.19	20	125.20	0.32
	4	20	53.00	0.25	20	91.90	0.44	20	119.65	0.66
	6	20	53.00	0.32	20	89.95	0.56	20	117.90	1.24
	8	20	53.00	0.58	20	88.90	0.71	20	117.35	1.42
	10	20	53.00	0.45	20	88.90	1.03	20	118.30	1.84
	15	20	53.00	0.86	20	88.40	1.13	20	117.15	2.13
	20	20	53.00	0.54	20	88.40	1.14	20	114.65	3.24

Table 8-2: Experimental results among ILP, NKSPH, and ISM

<i>n</i>	<i>method</i>	<i>q</i> = 2				<i>q</i> = 3				<i>q</i> = 4			
		# <i>Opt.</i>	<i>Dev</i>	<i>MC</i>	<i>ET</i>	# <i>Opt.</i>	<i>Dev</i>	<i>MC</i>	<i>ET</i>	# <i>Opt.</i>	<i>Dev</i>	<i>MC</i>	<i>ET</i>
30	<i>ILP</i>	20	-	63.72	816.83	19	-	76.68	263.18	19	-	84.53	7,094.21
	<i>ISM</i>	12	10.93%	70.68	0.01	6	12.22%	86.05	0.00	3	17.71%	99.50	0.00
	<i>NKSPH</i> <i>k</i> = 8	9	5.38%	68.50	0.02	7	9.47%	83.63	0.04	8	12.44%	94.42	0.10
	<i>k</i> = 10	9	5.38%	68.50	0.02	7	9.47%	83.63	0.05	9	12.21%	94.21	0.12
	<i>k</i> = 15	9	5.38%	68.50	0.02	7	9.47%	83.63	0.05	9	8.64%	91.42	0.17
	<i>k</i> = 20	9	5.38%	68.50	0.02	7	9.47%	83.63	0.05	9	8.64%	91.42	0.18
40	<i>ILP</i>	20	-	49.25	2,682.85	16	-	70.44	3,406.43	14	-	77.70	6,960.79
	<i>ISM</i>	18	0.36%	49.43	0.01	3	18.27%	83.31	0.02	1	17.12%	91.00	0.02
	<i>NKSPH</i> <i>k</i> = 8	17	1.22%	50.10	0.02	10	8.79%	76.06	0.05	5	6.66%	82.60	0.09
	<i>k</i> = 10	17	1.22%	50.10	0.03	10	11.76%	78.38	0.06	6	5.67%	81.80	0.10
	<i>k</i> = 15	17	1.22%	50.10	0.03	10	11.76%	78.38	0.10	7	3.79%	80.10	0.25
	<i>k</i> = 20	17	1.22%	50.10	0.03	10	11.76%	78.38	0.08	7	3.79%	80.10	0.28
50	<i>ILP</i>	20	-	57.20	2,257.08	19	-	68.68	2,144.92	18	-	84.44	7,096.11
	<i>ISM</i>	5	15.22%	65.90	0.03	2	30.43%	89.58	0.03	0	29.38%	109.25	0.03
	<i>NKSPH</i> <i>k</i> = 8	13	8.60%	62.95	0.04	8	12.42%	77.26	0.12	5	17.30%	99.89	0.22
	<i>k</i> = 10	13	8.60%	62.95	0.09	8	12.42%	77.26	0.19	5	16.13%	98.56	0.33
	<i>k</i> = 15	13	8.60%	62.95	0.05	8	11.62%	76.79	0.14	5	15.38%	97.78	0.44
	<i>k</i> = 20	13	8.60%	62.95	0.06	8	11.62%	76.79	0.18	5	15.38%	97.78	0.43
60	<i>ILP</i>	20	-	50.20	272.01	18	-	63.33	1,8042.98	6	-	63.33	22,266.95
	<i>ISM</i>	4	17.25%	58.86	0.05	3	21.41%	76.89	0.06	0	33.90%	84.80	0.07
	<i>NKSPH</i> <i>k</i> = 8	12	3.59%	52.25	0.15	6	8.27%	71.67	0.34	4	9.93%	72.33	1.87
	<i>k</i> = 10	12	3.59%	52.25	0.11	6	8.27%	71.67	0.20	4	7.76%	70.67	1.26
	<i>k</i> = 15	13	3.50%	52.20	0.11	6	7.20%	70.89	0.26	5	4.08%	67.83	2.98
	<i>k</i> = 20	13	3.50%	52.20	0.11	6	7.20%	70.89	0.53	5	4.08%	67.83	3.96
70	<i>ILP</i>	18	-	53.33	5,488.86	8	-	69.10	1,7561.86	-	-	-	-
	<i>ISM</i>	6	7.87%	57.53	0.08	0	32.90%	91.83	0.10	-	-	124.50	0.11
	<i>NKSPH</i> <i>k</i> = 8	11	2.00%	54.56	0.28	4	8.44%	74.50	0.91	-	-	118.65	2.02
	<i>k</i> = 10	11	2.00%	54.56	0.26	4	8.16%	74.30	0.58	-	-	115.80	2.48
	<i>k</i> = 15	11	2.00%	54.56	0.20	4	8.16%	74.30	0.68	-	-	113.90	3.29
	<i>k</i> = 20	11	2.00%	54.56	0.37	4	8.16%	74.30	1.27	-	-	113.90	3.76

Table 8-3: Worst cases of NKSPH and ISM among test groups

n	$method$	$q = 2$	$q = 3$	$q = 5$
		Dev	Dev	Dev
30	<i>ISM</i>	20.93%	22.22%	37.71%
	<i>NKSPH</i> $k = 8$	18.38%	19.47%	22.44%
	$k = 10$	16.38%	13.47%	13.21%
	$k = 15$	15.38%	12.47%	10.64%
	$k = 20$	15.38%	9.47%	10.64%
40	<i>ISM</i>	30.36%	20.27%	29.52%
	<i>NKSPH</i> $k = 8$	21.22%	18.79%	12.56%
	$k = 10$	14.73%	15.76%	11.67%
	$k = 15$	11.12%	13.76%	8.70%
	$k = 20$	8.34%	12.76%	5.45%
50	<i>ISM</i>	25.22%	30.43%	29.38%
	<i>NKSPH</i> $k = 8$	23.40%	12.42%	17.30%
	$k = 10$	22.34%	12.42%	16.13%
	$k = 15$	18.63%	11.62%	15.38%
	$k = 20$	10.30%	11.62%	15.38%
60	<i>ISM</i>	27.25%	21.41%	33.90%
	<i>NKSPH</i> $k = 8$	20.59%	15.22%	19.93%
	$k = 10$	13.59%	12.32%	13.76%
	$k = 15$	12.50%	10.56%	10.08%
	$k = 20$	7.50%	9.12%	9.08%
70	<i>ISM</i>	32.87%	39.90%	-
	<i>NKSPH</i> $k = 8$	19.40%	20.45%	-
	$k = 10$	13.48%	19.36%	-
	$k = 15$	11.45%	15.19%	-
	$k = 20$	9.76%	11.45%	-

Table 8-4: Comparisons of *MaxE* and *MinR*

	$n = 80$		$n = 90$		$n = 100$	
	<i>MaxE</i>	<i>MinR</i>	<i>MaxE</i>	<i>MinR</i>	<i>MaxE</i>	<i>MinR</i>
No. of success	16	17	30	29	31	32
Total light-trees	46	52	76	81	87	91
Communication cost	2571.05	2635.68	4106.64	4161.96	5261.22	5312.66
Multicast cost	2617.05	2687.68	4182.64	4242.96	5348.22	5403.66
<i>ETS</i>	1005.47	1029.41	2391.68	2520.55	4305.21	3669.24
<i>ETF</i>	2432.26	3554.59	1997.10	2499.14	2702.03	4840.71
Total execution time	3437.73	4584.01	4388.78	5019.69	7007.24	8509.95
Total used edges	403	417	639	657	811	831

Chapter 9 Conclusion and Future Work

In this dissertation, an extended routing and wavelength assignment problem called a multicast routing and wavelength assignment problem with delay bound (MRWAP-DC) in WDM networks with heterogeneous light splitting capacities is studied. For a set of requests, the objective is to find a set of light-forests whose sum of multicast costs, which is a linear combination of communication cost and wavelength consumption, is minimal. In the MRWAP-DC, five features in WDM networks including the OXCs with or without wavelength conversion capability, the OXCs with or without slight splitting capability (i.e., the light splitting capacity of OXCs may be different), single request or multiple requests demanding to be transmitted, requests with or without delay bounds, requests being unicast or multicast, are discussed in MRWAP-DC such that it is well formulated to define as a general problem. Nevertheless, due to this generalization, the MRWAP-DC is hard to solve in an affordable execution time. The variants, including the MRWAP-DC-WWC, URWAP-DC-SR, MRP-DC-WWC-SR, and MRWAP-DC-WWC-SR, are special cases of MRWAP-DC are explored by different methods.

According to the characteristics of the variants, first, the ILP method to solve the MRWAP-DC-WWC is formulated by two constraint categories, light-paths constraint category and light-tree constraint category. The former (including source constraints, target constraints, wavelength continuity constraints, and delay constraints) is used to find an assigned light-path under the given delay bound from the source to each destination. The

latter (including input constraints, capacity constraints, link usage constraints, wavelength usage constraint) is used to find an assigned light-tree, constructed by merging these assigned light-paths using the same wavelength, and to compute the communication cost of the light-tree and the wavelength consumption of routing a request. The proposed formulation has been not only shown to find optimal solution but also demonstrated by the CPLEX. Nevertheless, the execution time is not affordable for large-scale networks. Results from our computational study show that the ILP formulation can be used to solve the MRWAP-DC-WWC in networks of a limited number of nodes and in routing requests of a limited number of destinations.

Secondly, to extend the practicability of solving the variants of MRWAP-DC in larger-scale networks, two meta-heuristics, ant colony optimization (ACO) and genetic algorithm (GA), have been developed to derive approximate solutions in a reasonable time for the URWAP-DC-SR and the MRP-DC-WWC-SR. In the ACO formulation, the initial pheromones and the static desirability measures of edges are computed according to their communication costs and transmission delays. In the GA formulation, the destination-oriented encoding scheme to encode the phenotype of an individual to be a multicast tree is efficient to formulate the MRP-DC-WWC-SR. The individual with minimum fitness will be a near optimal solution. The simulation results obtained by the ILP method will be viewed as a baseline for the comparison with the two meta-heuristics. Therefore, our study will not only extend the application areas of the ACO approach and GA approach but also suggest a new viable method for coping with the complex optimization problems arising from the WDM domain. For routing a set of requests in large-scale networks, where the network provides more wavelengths, more requests are issued and the requests have enormous destinations, the ILP, ACO and GA are all time-demanding in solving the MRWAP-DC-WWC, URWAP-DC-SR, or MRP-DC-WWC-SR. It is not suitable to apply these three methods to solve the variants of dynamic MRWAP.

Finally, two efficient heuristics, Near- k -Shortest-Path-based Heuristic (NKSPH) and

Iterative Solution Model (ISM), have been proposed to find feasible approximate solutions for solve MRWAP-DC-WWC-SR. In the NKSPH, adjusting the value of k will encourage the heuristic to find better solutions or to consume less elapsed execution times. Moreover, the ISM seems to find poorer solutions in less elapsed execution times than the NKSPH.

Although these proposed methods, ILP, ACO, GA, NKSPH, and ISM have been proposed to examine the variants of MRWAP-DC, it still lack for a general method to solve the general MRWAP-DC. Moreover, several works shown in Table 9-1 worthy to research further includes:

- (1) Extending the ILP formulation to explore the general MRWAP-DC, a new ILP formulation involves in wavelength conversion cost and wavelength conversion delay. There still exist several challenges in the number of nodes, the number of wavelengths, and the number of destinations.
- (2) Relaxing these integral variables in the ILP formulation to be real variables, a new relaxed ILP (R-ILP) formulation may find approximate solutions but consume less elapsed execution time. It may be conceivable to solve the MRWAP-DC and MRWAP-DC-WWC.
- (3) Like the proposed GA method, simulated annealing (SA) algorithm which emulates the process of annealing may be useful to explore the variants of MRWAP-DC. The simulated annealing begins at a very high temperature where the input values are allowed to assume a great range of random values. As the temperature is allowed to fall in the training progresses, it restricts the degree to which the inputs are allowed to vary. This often leads the simulated annealing algorithm to a better solution, just as a metal achieves a better crystal structure through the actual annealing process. The SA algorithm may be conceivable to MRP-DC-WWC-SR or MRWAP-DC-WWC-SR.
- (4) Extending the formulations of ACO, GA, NKSPH, and ISM, applying these

formulations to explore the other variants may be an attractive research topic. For example, because the route of an ant indicates a light-path, it is conceivable to group ants into ant sets and the objective of each labeled ant in an ant set is to find a light-path from the source to a labeled destination. The phenotype of an ant set will be formulated to be a light-tree. Therefore, the extended ACO formulation may solve the MRWAP-DC-SR or the MRWAP-DC-WWC-SR.

Table 9-1: The further works in extending proposed methods or proposing new methods

	ILP	R-ILP	SA	ACO	GA	NKSPH	ISM
MRP-DC-WWC-SR	Yes		FW		Yes	Yes	Yes
URWAP-DC-SR	Yes		FW	Yes	FW	Yes	Yes
MRWAP-DC-WWC-SR	Yes		FW	FW	FW	Yes	Yes
MRWAP-DC-WWC	Yes	FW					
MRWAP-DC-SR	FW	FW		FW		FW	FW
MRWAP-DC	FW	FW					

Note :

“Yes” means that the problem has been examined by the method proposed in this dissertation.

“FW” means that the problem can be examined by extending the proposed method or by proposing a new method.

References

- [1]. M. Ali and J. Deogun, Power-efficient design of multicast wavelength-routed networks, *IEEE Journal of Selected Areas in Communications*, Vol. 18, Oct. 2000, pp. 1852–1862.
- [2]. A.S. Arora, S. Subramaniam, Converter placement in wavelength-routing mesh topologies, in *Proc. of ICC*, Jun. 2000, pp. 1282-1288.
- [3]. K. Bala, K. Petropoulos, and T. E. Stern. Multicasting in a linear lightwave network. *Proc., IEEE INFOCOM*, 1993, pp. 1350-1358.
- [4]. K. Bala, T. E. Stern, D. Simchi-Levi, and K. Bala, Routing in a linear lightwave network, *IEEE/ACM Transactions on Networking*, Vol. 3, No. 4, Aug. 1995, pp. 459-469.
- [5]. T. Ballardie, P. Francis and J. Crowcroft, Core Based Trees (CBT): an architecture for scalable inter-domain multicast routing, *Proc., ACM SIGCOMM '93*, Sep. 1993.
- [6]. E. J. Berglund and D. Cheriton, Amaze: A multiplayer computer game, *IEEE Software*, Vol. 2, May 1985, pp. 30–39.
- [7]. K. Bharath-Kumar and J. M. Jaffe, Routing to multiple destinations in computer networks, *IEEE Transactions on Communications*, Vol. COM-31, Mar. 1983, pp. 343–351.
- [8]. T.A. Birks, R.P. Kenny, K.P. Oakley, and C.V. Cryan, Elimination of water peak in optical fibre taper components, *Electronics Letters*, Vol 26, Issue 21, Oct. 1990, pp. 1761-1762.
- [9]. K. Birman and T. Joseph, Reliable communication in the presence of failures, *ACM Trans. Comput. Syst.*, Vol. 5, No. 1, Feb. 1987, pp. 47–76.
- [10]. R.C. Bray and D.M. Baney, Optical networks: backbones for universal connectivity,

Hewlett Packard Company, 1997.

- [11].M.T. Chen, S.S. Tseng, Multicast routing under delay constraint in WDM network with different light splitting, *Proc. International Computer Symposium 2002 (ICS 2002)*, Taiwan R.O.C.
- [12].B. Chen and J. Wang, Efficient routing and wavelength assignment for multicast in WDM networks, *IEEE Journal of Selected Areas in Communications*, Vol 20, Jan. 2002, pp. 97-109.
- [13].D. Cheriton and W. Zwaenepoel, Distributed process groups in the V-kernel, *ACM Trans. Comput. Syst.*, Vol. 3, No. 2, May 1985, pp. 77–107.
- [14].I. Chlamtac, A. Ganz, and G. Karmi, Lightpath communications: Approach to high bandwidth optical WANs, *IEEE Transactions on Communication*, Vol. 40, 1992, pp.1171-1182.
- [15].E. C. Cooper, Circus: A replicated procedure call facility, *Proc., 4th Symp. Reliability in Distributed Software and Database Systems*, Silver Spring, MD, Oct. 1984, pp. 11–24.
- [16].L. Davis, (Ed) handbook of genetic Algorithms, Van Nostrand Reinhold, 1991, New York.
- [17].X. Deng, G. Li, and W. Zang, Wavelength allocation on trees of rings, *Networks*, Vol. 35, No. 4, , 2000, pp. 248–252.
- [18].T. Dennis, and P.A. Williams, Tech. Digest, Chromatic dispersion measurement error caused by source amplified spontaneous emission, *IEEE Photonic Technology Letters*, Vol. 16, No. 11, Nov. 2004, pp. 2532-2534.
- [19].E.W. Dijkstra, A node on two problems in connexion with graph, *Numerische Mathematik*, Vol. 1, 1959, pp. 269-271.
- [20].D.R. Din and S.S. Tseng, A genetic algorithm for solving dual-homing cell assignment problem of the two-level wireless ATM network, *Computer Communications* 25, p.1536-1547, 2002.

- [21].M. Dorigo, V. Maniezzo, and A. Colorni, Positive feedback as a search strategy, *Technical Report*, 91-016, Politecnico di Milano, IT, 1991.
- [22].M. Dorigo, Optimization, Learning and natural algorithms, Ph.D. Dissertation, Dipartimento di Elettronica, Politecnico di Milano, Italy (in Italian), 1992.
- [23].D. Eppstein, Finding the k shortest paths, *SIAM Journal on Computing*, Vol. 28(2), 1998, pp. 652–673.
- [24].H. Eriksson, MBONE: The multicast backbone, *Communications of the ACM*, Aug. 1994, pp. 54–60.
- [25].T. Erlebach, K. Jansen, C. Kaklamanis, and P. Persiano, An optimal greedy algorithm for wavelength allocation in directed tree networks, *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, Vol. 40, 1998, pp. 117–129.
- [26].D. J. Farber, J. Feldman, F. R. Heinrich, M. D. Hopwood, K. C. Larson, D. C. Loomis, and L. A. Rowe, The distributed computing system, *Proc., IEEE COMPCON*, 1973, pp. 31–34.
- [27].R.M. Garlick and R.S. Barr, Dynamic wavelength routing in WDM networks via ant colony optimization, *Proc., 3th International Workshop on Ant Algorithms*, Brussels, Belgium, Sep. 2002, pp. 250-255.
- [28].A.M. Hamad and A. E. Kamal, A Survey of multicasting protocols for broadcast-and-select single-hop networks, *IEEE Network*, July/August 2002, pp. 36-48.
- [29].Q. D. Ho, M. S. Lee, and Converter-Aware, Wavelength assignment in WDM networks with limited-range conversion capability, *IEICE Transactions on Communications*, Vol. E89–B, No.2 Feb. 2006, pp. 436-445.
- [30].M. E. Houmaidi and M. A. Bassiouni, Dominating set algorithms for sparse placement of full and limited wavelength converters in WDM optical networks, *Journal of Optical networking*, Vol. 2, No. 6, Jun. 2003, pp. 162-177.
- [31].J. Iness, Efficient use of optical components in WDM based optical networks, Ph.D.

Dissertation, University of California, Davis, Nov, 1997.

- [32]. J. Iness and B. Mukherjee, Sparse wavelength conversion in wavelength-routed WDM networks, *Photonic Network Communications Journal*, Vol. 1, No. 3, Nov. 1999, pp. 183-205.
- [33]. X.H. Jia, D.Z. Du and X.D. Hu, Integrated algorithm for delay bounded multicast routing and wavelength assignment in all optical networks, *Computer Communications* 24, 2001, pp. 1390-1399.
- [34]. X. Jia, A distributed algorithm of delay-bounded multicast routing for multimedia applications in wide area networks, *IEEE/ACM Trans. Networking*, Vol. 6, Dec. 1998, pp. 828-837.
- [35]. R.M. Karp, Reducibility among combinational problems, In R.E. Miller, J. W. Thatcher (Eds.), *Complexity of Computer Computations*, Plenum Press, New York, 1972, pp. 85-103.
- [36]. S. Khuller, B. Raghavachari, and N. Young, Balancing minimum spanning trees and shortest-path trees, *Algorithmica*, Vol. 14, No. 4, 1995, pp. 305-321.
- [37]. V. P. Kompella, J. C. Pasquale, and G. C. Polyaos, Optimal multicast routing with quality of service constraints, *Journal of Network and Systems Management*, Vol. 4, No. 2, 1996, pp. 107-131.
- [38]. M. Kovacevic and A. Acampora, Benefits of Wavelength Translation in All-Optical Clear Channel Networks, *IEEE Journal of Selected Areas in Communication*, vol. 14, no. 5, 1996, pp. 868-880.
- [39]. L. Kou, G. Markowsky, L. Berman, A fast algorithm for Steiner trees, *Acta Informatica*, Vol. 15, p. 141-145, 1981.
- [40]. M.S. Kumar, P. Sreenivasa Kumar, Static lightpath establishment in WDM network – New ILP formulation and heuristic algorithms, *Computer Communication*, 25, 2002, pp. 109-114.

- [41]. M.S. Kwang and H.S. Weng, Ant colony optimization for routing and load-balancing: survey and new directions, *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Human*, Vol. 33, No. 5, Sep. 2003, pp. 560-572.
- [42]. K. C. Lee and V. O. K. Li, A wavelength-convertible optical network, *IEEE/OSA Journal of Lightwave Technology*, Vol. 11, No. 5-6, May-Jun. 1993, pp. 962-970.
- [43]. V. T. Lei, X. Jiang, S. H. Ngo, and S. Horiguchi, Dynamic RWA based on the combination of mobile agents technique and genetic algorithms in WDM Networks with Sparse wavelength conversion, *IEICE Transactions on Information and Systems*, E88-D/9, Sep. 2005, pp. 2067-2078.
- [44]. J. Leuthold, D. M. Marom, S. Cabot, J. J. Jaques, R. Ryf, and C. R. Giles, All-optical wavelength conversion using a pulse reformatting optical filter, *Journal of Lightwave Technology*, Vol. 22, No. 1, Jan. 2004, pp. 186-192.
- [45]. D. Li, X. Du, X. Hu, L. Ruan, and X. Jia, Minimizing number of wavelengths in multicast routing trees in WDM networks, *Networks*, Vol. 35, 2000, pp. 260-265.
- [46]. W. Liang and H. Shen, Multicasting and broadcasting in large WDM networks, *Proc.*, 12th, *International Parallel Processing Symposium. (IPPS/SPDP)*, Orlando, FL, 1998, pp. 516-523.
- [47]. R. Libeskind-Hadas, R. Melhem, Multicast routing and wavelength assignment in multihop optical networks, *IEEE/ACM Transactions on Networking*, Vol. 10, No. 5, Oct. 2002 pp. 621-629.
- [48]. H. C. Lin, and C. H. Wang, A hybrid multicast scheduling algorithm for single-hop WDM networks, *Journal of Lightwave Technology*, Vol. 19, No. 11, Nov. 2001, pp. 1654-1664.
- [49]. Y.P. Liu, M.G. Wu, and J. Qian, The distributed multicast routing scheme with delay constraint using ant colony optimization, *Proc. of the 6th World Congress on Intelligent Control and Automation*, June, 2006, Dalian, Chin.

- [50].R. Malli, X. Zhang, and C. Qiao, Benefit of multicasting in all-optical networks, *Proc., SPIE Conf. All-Optical Networks*, Vol. 3531, Boston, MA, Nov. 1998, pp. 209–220.
- [51].K. Marzullo and F. Schmuck, Supplying high availability with a standard network file system, *Proc., 8th Distributed Computing Systems*, 1988, pp. 447–453.
- [52].N. F. Mir, A Survey of data multicast techniques, architectures, and algorithms, *IEEE Communications*, Sep. 2001, pp. 164-170.
- [53].B. Mukherjee, *Optical Communication Networks*, McGraw-Hill, 1997.
- [54].A. E. Ozdaglar, D. P. Bertsekas, Routing and wavelength assignment in optical networks, *IEEE/ACM Transactions on Networking*, Vol. 11, No. 2, Apr. 2003, pp. 259-273.
- [55].R. K. Pankaj, Wavelength requirements for multicasting in all-optical networks, *IEEE/ACM Trans. Networking*, Vol. 7, Jun. 1999, pp. 414–424.
- [56].M. Pascoal, E. Martins, and J. Santos, A new improvement for a k shortest paths algorithm, *Investigação Operacional*, Vol. 21, Jan., 2001, pp. 47-60.
- [57].S. Paul, K.K. Sabnani, Reliable Multicast Transport Protocol (RMTP), *IEEE Journal of Selected Areas in Communications, Special Issue on Network Support for Multipoint Communication*, Vol.15, No.3, Apr 1997, pp. 407-421.
- [58].R. Pickholtz, D. Schilling, and L. Milstein, Theory of spread spectrum communications – A tutorial, *IEEE Transactions on Communications*, Vol. COM-30, No.5, 1982, pp. 855–884.
- [59].R.C. Prim, Shortest connection networks and some generations, *Bell System Technical Journal*, Vol. 36, 1957, pp. 1389-1401.
- [60].P. Prucnal, E. Harstead, and S. Elby, Low-loss, high-impedance integrated fiber-optic tap, *Optical Engineering*, Vol. 29, Sept. 1990, pp. 1136–1142.
- [61].T. Pusateri, DVMRP version 3, IETF Internet draft, draft-ietf-idmrdvmp-v3-07, obsoletes RFC 1075, Aug. 1998.
- [62].C. Qiao, M. Jeong, A. Guha, X. Zhang, and J. Wei, WDM multicasting in IP over WDM

- networks, *Proc., IEEE Int. Conf. on Network Protocols (ICNP)*, 1999.
- [63].B. Ramamurthy and B. Mukherjee, Wavelength conversion in WDM networking, *IEEE Journal of Selected Areas in Communications*, Vol. 16, Sep. 1998, pp. 1061–1073.
- [64].R. Ramaswami and K. Sivarajan, Routing and wavelength assignment in all-optical networks, *IEEE/ACM Transactions on Networking*, Vol. 3, Jun. 1995, pp. 489-500.
- [65].E. Rolland, Abstract heuristic search methods for graph partitioning, Ph.D. Dissertation, Ohio State University, Columbus, Ohio. 1991.
- [66].K.K. Sabnani, Error and flow control performance of a high-speed protocol, *IEEE Transactions on Communications*, May 1993, pp. 707-720.
- [67].L.H. Sahasrabudde and B. Mukherjee, Light-trees: Optical multicasting for improved performance in wavelength-routed networks, *IEEE Communications Magazine*, Vol. 37, 1999, pp. 67-73.
- [68].G. Sahin and M. Azizoglu, Multicast routing and wavelength assignment in wide-area networks,” *Proc., SPIE Conf. All-Optical Networks*, Vol. 3531, Boston, MA, Nov. 1998, pp. 196–208.
- [69].S.J. Shyu, P.Y. Yin and B.M.T. Lin, An ant colony optimization algorithm for the minimum weight vertex cover problem, *Annals of Operations Research*, Vol. 131, 2004, pp. 283–304.
- [70].C. Siva Ram Murthy and M. Gurusamy, *WDM optical networks*, Prentice-Hall, 2002.
- [71].N. Sreenath, N. Krishna Mohan Reddy, G. Mohan, C. Siva Ram Murthy, Virtual source based multicast routing in DWM networks with sparse light splitting, *IEEE, Proc., Workshop on High Performance Switching and Routing*, 2001, pp. 141-145.
- [72].S. Subramaniam, M, Azizoglu, and A. K. Somani, All optical networks with sparse wavelength conversion, *IEEE/ACM Transactions on Networking*, Vol. 4, No. 4, Aug 1996, pp. 544-557.
- [73].M. Sust, Code division multiple access for commercial communications, *Review of*

Radio Science, 1992–1994, pp. 155–179.

- [74]. B. D. Theelen, J. P. M. Voeten, P. H. A. van der Putten, and P. M. J. Stevens, Concurrent support of higher-layer protocols over WDM, *Photonic Network Communications*, 4:1, 2002, pp. 47-62.
- [75]. Y. Ueno, S. Nakamura, K. Tajima, S. Kitamura, 3.8-THz wavelength conversion of picosecond pulses using a semiconductor delayed-interference signal-wavelength Converter (DISC), *IEEE Photonics Technology Letters*, Vol. 10, No. 3, Mar. 1998, pp. 346-348.
- [76]. G.N. Varela and M.C. Sinclair, Ant-colony optimization for virtual wavelength-Path routing and wavelength allocation, *Proc., Congress on Evolutionary Computation (CEC'99)*, Washington DC, USA, Jul. 1999, pp. 1809-1816.
- [77]. Roca Vincent, Costa Luis, Vida Rolland, Dracinschi Anca, and Fdida Serge, A survey of multicast technologies, *Technical Report RP-LIP6-2000-09-05*, Universit Pierre et Marie Curie, September 2000.
- [78]. B.M. Waxman, Routing of multipoint connections, *IEEE Journal of Selected Areas in Communications*, Vol. 6, 1988, pp. 1617-1622.
- [79]. R. Wittmann and M. Zitterbart, *Multicast communication: protocols, programming, and applications*. San Francisco, CA: Morgan Kaufman, 2000.
- [80]. D.N. Yang, W. Liao, Design of light-tree based logical topologies for multicast streams in wavelength routed optical networks, *Proc., IEEE INFOCOM'03*, San Francisco, CA, 2003.
- [81]. Y. Yang, J. Wang, and C. Qiao, Nonblocking WDM multicast switching networks, *IEEE Transactions on Parallel and Distributed Systems*, Vol. 11, Dec. 2000, pp. 1274–1287.
- [82]. J. Yates, J. Lacey, D. Everitt, and M. Summerfield, Limited range wavelength translation in all optical networks, *Proc., IEEE INFOCOM'96*, San Francisco, CA, Vol. 3, Mar. 1996, pp.954-961.

- [83].S. Yoshima, K. Onohara, N. Wada, F. Kubota, and K.I. Kitayama, Multicast-capable optical code label switching and its experimental demonstration, *Journal of Lightwave Technology*, Vol. 24, No. 2, Feb. 2006, pp. 713-722.
- [84].H. Zang, J. Jue, and B. Mukherjee, A review of routing and wavelength assignment approaches for wavelength-routed optical WDM networks, *Optical Networks Magazine*, Vol. 1, No. 1, Jan. 2000, pp. 47-60.
- [85].X. Zhang, J. Wei, and C. Qiao, Constrained multicast routing in WDM networks with sparse light splitting, *Journal of Lightwave Technology*, Vol. 18, 2000, pp. 1917-1927.
- [86].W. D. Zhong, J. P. R. Lacey, R. S. Tucker, Wavelength cross-connect for optical transport networks, *Journal of Lightwave Technology*, Vol. 14, 1996, pp. 1613-1620.

