謝　　誌

隨著碩士論文的完成，此時的心中雖有著些許的激動，但卻願意以更多的感謝來與所有關心我的人一同分享。恩師—林永松教授不管是對於我在論文的寫作上或是生活的關懷上，老師諄諄的解惑及提醒總讓我受益良多，尤其是在我遭逢身心俱疲的時刻，老師的體諒更是使我能重新再出發的重要動力，從老師身上所學習到的，不僅是其面對挑戰般的努力不懈，在為人處事上的虛懷若谷更將會是我未來追求最佳化的過程中的努力目標。

感謝諸位口試委員：台科大電子系陳郁堂教授、台大資管系李瑞庭教授、世新大學資管系顏宏旭教授寶貴的指導與建議，也感謝旭成學長在程式重建上的幫忙及協助，喜恩、茂原在百忙中幫我校正及潤飾內容，都讓論文更加完備。感謝同門麒安、晉華、茂原、至鈞與佳宏，這段期間的砥礪與互動，讓我印象深刻；也感謝實驗室的學弟妹，在忙碌的課程之餘，仍然幫忙諸多繁雜的工作，讓我們無後顧之憂地去準備口試。至於教會中的好友:依芸、陳豪、女友一家人、二姑一家人、三姑一家人、及許多很照顧我的學長姐和貼心的學弟妹，感謝你們的關懷之情，很開心能共度這段黃金歲月，相信這份情誼會是一輩子的連結。

最後，我要感謝我的父母：黃仁德先生與王秀絹女士，因為您們無怨無悔的付出及一心一意的關愛，給予我在求學過程中最有力的支持，才造就了今日的我，也感謝大哥克聖、弟弟克先及嫂嫂佳賢的關心、打氣，讓我更能勇往直前。感謝我的女友喜慧，在我茫然不知所措的那段日子裡，是妳以細心溫柔的傾聽及爽朗有力的笑聲伴我走過瀕臨崩潰的黑夜，有妳的感覺，真好！謹將此論文獻與我最親愛的人，願榮耀歸與和我同行至今的那位神。

黃克行 謹識

2003.11.18.於台大資管所

論 文 摘 要

本研究乃是在群播網路的架構中，提供有效率且具彈性的多媒體傳輸機制之設計方法，基於當前網路應用環境之中的終端連結裝置(例如：analog modem, cable modem, xDSL, etc.)及使用者本身對於傳輸品質(例如高畫質,低畫質)所存在的異質性，透過近來在網路交換及傳輸裝置上的發展，例如：progressive coder 或是 video gateway 的應用，使得在傳送資料的過程當中，接收端不但可以依照本身所處在的網路狀況去作頻寬的選擇，同時在傳送端，僅需傳送接收端中的最大頻寬量即可。群播服務群組根據此一網路傳輸特性，計算出群播服務群組可使用的最小成本樹(minimum cost tree)去達到有效率的傳輸；而為了提高傳輸服務上的彈性，考慮部分允入控制（Partial Admission Control）的機制，所謂部分允入控制是有別於傳統的全有或全無允入模式，乃用於網路經營業者決定是否接受任何一組中的個別用戶群之服務要求，以達成在有限資源的服務提供之下，能為業者本身獲取最佳的經營利潤。

本研究採行之方法如下：首先將所擬研究之問題數學模式化為一數學規劃（Mathematical Programming）問題，其中目標函數（Objective Function）分別為最小化系統之傳輸成本及最大化系統之整體收益，同時需滿足諸如容量限制、群播樹限制等條件限制，再者就此數學模式中所具有之特性，研擬出以最佳化技巧為基礎（Optimization-based）之演算法以解決此一複雜之規劃問題，為此我們發展了以拉格蘭氏鬆弛法（Lagrangean Relaxation）為基礎的修改版 T-M 解題程序和幾個調整程序，這些解題程序分別應用在考慮兩個不同網路應用層面的二個模組上，分別是（I）最小成本樹（Minimum Cost Tree）模組、（II）部分允入控制（Partial Admission Control）模組，最後在數個著名的網路拓僕上導入大規模的數據測試以驗證這些解題程序之效果及效率。

根據上述實驗的結果，在最小成本樹模組平均達到 6.42%的誤差比率，而在部分允入控制模組也可達到平均 13.86%的誤差比率，可以證明本研究所提出的方法在不同的測試條件下，如不同的網路拓僕、群播服務數量等，能求得近似最佳解（Near Optimal Solution）之優異表現。相對於直覺性的經驗法則(Heuristic)，亦有明顯的改進，在最小成本樹模組平均可提升 14.42%的整體收益，部分允入控制模組亦有 12.06%的整體收益提升。

多媒體網路的應用已成為現代人生活中不可或缺的一部分，群播服務可以提供更好的網路效能使用。本研究在基於在效率及彈性的雙重考量下，配合傳輸設備的日新月異，提供可行且優異的演算法，同時兼顧了實用性與效能表現，也就是說本論文具有相當程度的實用與學術價值。

關鍵詞：群播服務、多媒體傳輸、最小成本樹、部分允入控制、最佳化、拉格蘭氏鬆弛法

THESIS ABSTRACT

GRADUATE INSTITUTE OF INFORMATION MANAGEMENT
NATIONAL TAIWAN UNIVERSITY
NAME: KO-HSING HUANG MONTH/YEAR: NOVEMBER, 2003
ADVISER: YEONG-SUNG LIN

In this thesis, we intend to solve the problem of supporting efficient and flexible mechanisms for multimedia distribution on multicast networks. Under multimedia application environments, it is characterized by large bandwidth variations due to heterogeneous access-technologies of the networks (e.g., analog modem, cable modem, xDSL, etc.) and receivers (e.g., high resolution, low resolution). Taking advantage of recent advances in switching and transmission technologies, either by a progressive coder, or video gateway, destinations can request different bandwidth requirement from the source. The source only needs to transmit signals that are sufficient for the highest bandwidth destination. The minimum cost tree of multicast service is thus calculated by this property of transmission mechanisms to achieve the efficiency. Furthermore, we also consider about partial admission control mechanism. For network operators, the function of partial admission control is to determine whether a user service request can be granted so that the requested bandwidth of the new user can be satisfied and the total revenue can be maximized under limited resources.

We formulate the problems as mathematical models firstly. According to different models, we focus on the minimization of total transmission cost and the maximization of total system revenues separately by satisfying the capacity constraint and multicast tree constraints. The basic approach to solve the problem is Lagrangean Relaxation and the subgradient method. The Lagrangean based modified T-M heuristic and several adjustment procedures are developed to get primal feasible solutions. According to the computational experiments, solutions from the proposed algorithm are within a few percent of the optimal solutions on networks with 9-26 nodes; both in the minimum cost tree model and partial admission control model (error difference in the minimum cost tree model is 6.42% on average, error difference in the partial admission control model is 13.86% on average). In terms of performance, our Lagrangean Relaxation based solution has more significant improvement than simple heuristics. The improvement on the total cost can reach 14.42% on the average in the minimum cost tree model, and 12.06% on the average in the partial admission control model.

Keywords: Multicast Service, Multimedia distribution, Minimum cost tree, Partial admission control, Optimization, Lagrangean Relaxation

Contents

List of Tables

List of Figures

Chapter 1 Introduction

1.1 Overview

Many of the emerging applications in the Internet are farther comprehensive, such as Internet TV, distance learning, replicated database update, command and control systems, and distributed games [10]. They all fall in the category of group communications as opposed to the classical one-to-one conversations. In general, these applications may have several sources and a huge number of destinations, which could be up to millions in the case of Internet TV for example. The importance of group communications lies in the new applications that will be generated by adding networking capabilities to multimedia devices, and hopefully gains in efficiency and holding cost when multimedia resources are part of distributed computing systems. These applications drive the development of the multicast service. As a result, the Internet is becoming increasingly multicast capable. Basically, multicast routing establishes a tree that connects the source with the destinations. The multicast tree is rooted at the source and the leaves are the destinations. Multicast delivery sends data across this tree toward the destinations. Comparing with the use of several unicast channels is impractical in terms of network resources and processing power of end destinations [28] [29].

Multimedia applications such as digital video and audio often have strict quality-of-service (QoS) requirements [38]. The QoS guarantee is of utmost importance for the development of future networks. Recent developments in switching and transmission technologies allow the implementation of high-speed networks that carry vast amounts of traffic that is generated by applications that are more sensitive to data quality (such as video or audio), and at the same time less predictable than current fixed rate sources. In the next information age, it will be possible to support new multimedia applications in a global environment and design new services on flexible platforms without upgrading the physical infrastructure. This requires new network architectures capable of offering computation services to communication applications with stringent QoS requirements. A key issue is the provision of well-designed network structures and mechanisms so as to meet these requirements [8].

There are several different kinds of network design problems. In the most instances, one may evaluate the quality of a network in some ways; typical quality measures include the weight (total length of all edges in a network), diameter (longest network distance between two sites), and dilation (largest ratio of network distance to Euclidean distance). The problem may be static or various types of dynamic changes to the collection of sites. Much of the research about the network design problems has involved problems in which the network to be designed is a tree. Such problems include the minimum spanning tree, maximum spanning tree, minimum diameter spanning tree, bounded degree spanning trees (such as the traveling salesman path), and the k-point minimum spanning tree [12].

In another way, we consider the fundamental components of the network design process: whether admit the request or not. The admission control decision may be as straightforward as a simple inequality calculation: if the sum of the bandwidth usage of the current flows and a new flow is greater than the network's total bandwidth, it could reject the flow. These QoS guarantees, having no tolerance for violations, are called "hard" guarantees, and some flows demand this guaranteed service. Other flows, however, may accept some amount of QoS guarantee violation that usually bounded by some probability values. This is called predictive service, and such statistical, or "soft," guarantees provide more flexibility for the admission control algorithm, leading to increased network utilization [1][4]. From the operator's viewpoint, the target is to admit maximum users through multicasting service to earn more revenues. However, supporting a global network application environment that can provide performance guarantees to group communication is still the most concerned issue from the viewpoint of users. Seeking the balance point between operators and users has always been an important issue, and apparently it will be to continue even more in the coming days.

1.2 Motivation

Multimedia distribution over the Internet is getting more and more popular. Since the Internet was designed for computer data communication, how to reach the necessary requirements for the efficient delivery of multimedia streams becomes a great challenge. For example, the Internet is characterized by large bandwidth variations due to heterogeneous access-devices of the destinations (e.g. analog modem, cable modem, xDSL, etc.). In video multicast the heterogeneity of the networks and destinations makes it difficult to achieve bandwidth efficiency and service flexibility. There are many challenging issues that need to be addressed in designing architectures and mechanisms for data transmission [31] [43].

Unicast and multicast delivery of video have important building blocks for many Internet applications. Unicast video distribution uses multiple point-to-point connections in Figure 1.1; classic multicast video distribution uses point-to-multipoint transmission with commonly intermediate devices in Figure 1.2. Figure 1.3 shows the intelligent intermediate devices by using progressive coder [25] or by converting between format encoders such as video gateways[5][40]. This is also similar to destination-initiated reservations and packet filtering used in RSVP [44]. For applications such as video conferencing and Internet TV, using multicast can achieve high bandwidth efficiency while the destinations can share links. The efficiency of multicast is achieved at the cost of losing the service flexibility of unicast, because in unicast each destination can individually negotiate the service contract with the source. The heterogeneity of the networks and destinations makes it difficult to design an efficient and flexible mechanism for servicing all multicast group users [42] [43]. As a result, we want to make the equilibrium by adapting the advanced intermediate devices.

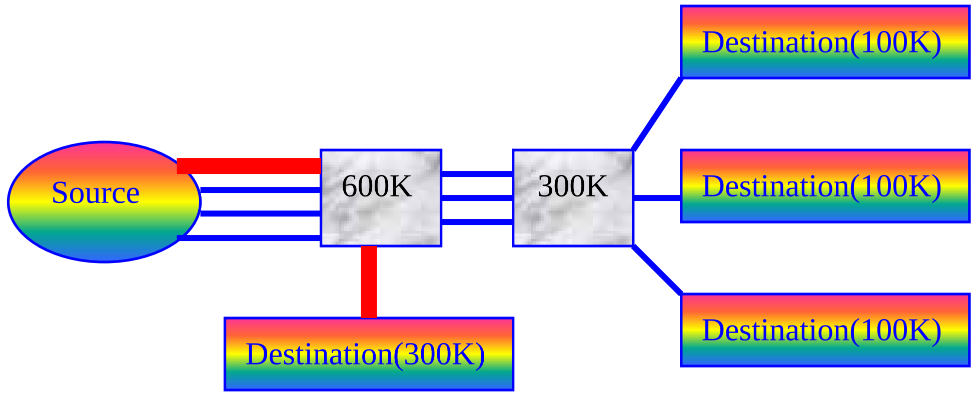

Figure 1-1: Unicast video distribution using point-to-point transmission

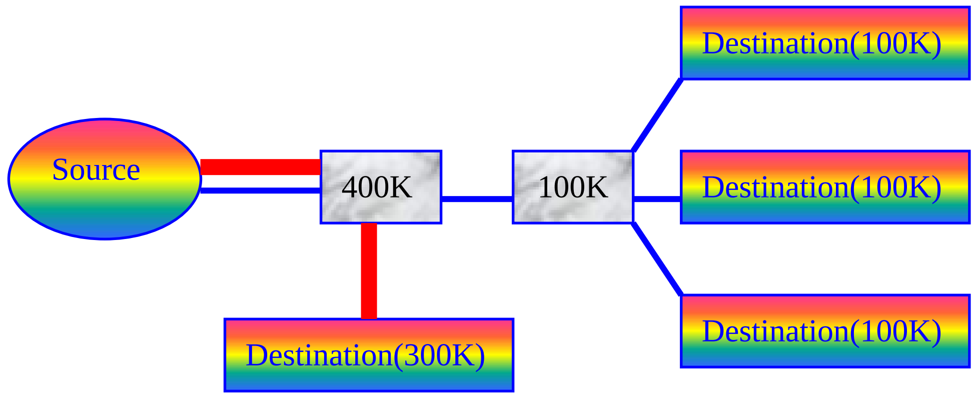

Figure 1-2: Multicast video distribution using point-to-multipoint transmission with common intermediate devices

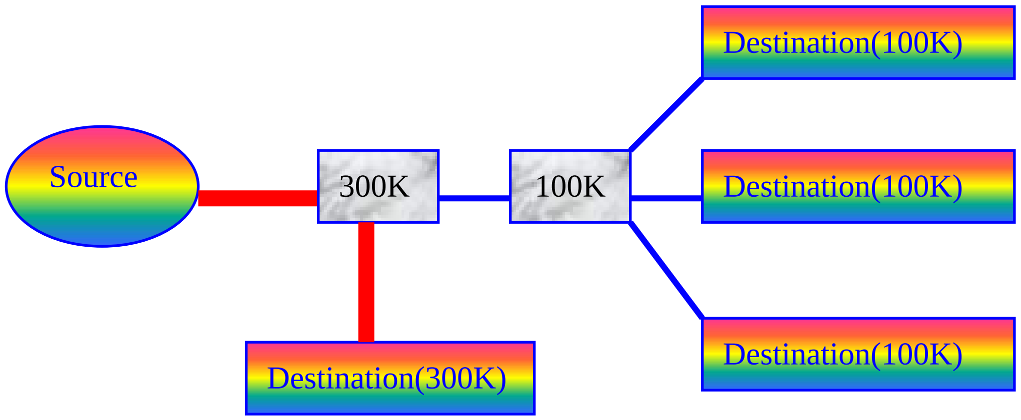

Figure 1-3: Multicast video distribution using point-to-multipoint transmission with intelligent intermediate devices

In another way, for providing their service to satisfy these traffic requirements of multicast groups, operators under some circumstances where the benefit of a multicast service may not justify its cost to the users. These occasions only occur when some group members want to be serviced but others do not wish to pay more. For example, if a CABLE TV operator within a residential area was to use multicasting, as everyone knows that not every group member would like all provided programs. Adopting traditional admission policy is not beneficial for operators and members. This creates additional bandwidth usage on the access links where resource availability is often most limited [15] [23]. Avoiding bandwidth being wasted and improving the flexibility, enhancing traditional admission control strategy or combining other components is necessary for multimedia distribution on multicast networks.

This paper is to discuss how to appropriately design the efficient and flexible methods for multimedia distributions on multicast networks. We consider different models according to the current network application environment. In the first model, given the network topology and to serve all group destinations, we consider constructing a minimum cost tree to establish a network that may ensure the source provides the bandwidth requested by the destinations. Destinations can request different bandwidth requirement from the source. The source transmits only signal that is sufficient for the highest bandwidth destination. In the second model, without knowing any information of the network topology, we establish a network which allows the admission strategy of the multicast group doesn't base on traditional "all or none" approach, instead of considering to accept partial destinations of the requested multicast group.

1.3 Literature Survey

1.3.1 IP multicast

IP multicast traffic for a particular (source/destinations) pair is transmitted from the source to the destinations via a multicast tree that connects all the nodes in the group. In generally, there are three basic approaches to construct multicast trees [1] [9]. They can be characterized as centralized or distributed and are designed to support dense or sparse mode of multicast group membership among the destinations.

(1) Source-based routing: Reverse path forwarding (RPF) has been widespread use in IP multicast. It is optimized for dense mode, but it does not take into consideration group membership; improved RPF was proposed to solve this weakness, it used "flood and prune" method. However, as the number of sources and groups grows too large, memory could be saturated in the routers.

(2) Center-based trees: "Center-based trees" is the most recent routing approach. It is aimed at multiple sources/multiple destinations. Core-based tree (CBT) algorithm is the most famous. It is a totally destination based approach that limits the diffusion of packets naturally to group members and is suitable for sparsely distributed destinations. However, suffering from traffic concentration, as the traffic from all sources of a given group will converge to the center.

(3) Hybrid: This type of algorithms may combine the properties of (1) and (2). Furthermore, it will be impossible to make any guarantee if the algorithm is not capable of maintaining a multicast route with chosen properties.

In accordance with above description of the protocols, the following parameters are taken into account [22].

(1) Connection: Roughly speaking, multicast protocols are either based on a single tree shared by all the members, or based on several trees. That is, we make a distinction between shared tree and source-based tree.

(2) Aggregation: The process of aggregation is said greedy if a joining node connects to the closest node already in the group. And the process of aggregation is said RPF if a joining node is connected to the group by an optimal path to the source.

(3) Quality of service: Some protocols aim to optimize parameters such as bandwidth、delay……etc. Although one cannot formally consider all QoS parameters, some are more concerned by certain aspects of QoS than others.

(4) Construction: Some are based on the underlying unicast protocol; some others use Breadth-First Search technique; others use a pruned spanning tree of the network; others explore multiple paths and keep the best path.

(5) Loop: Some protocols can theoretically avoid loop. For the others, either there exist situations for which loops would occur or there is non-exist proof to be ensured.

	Connection	Aggregation	QoS	Construction	Loop
DVMRP	Source-based	RPF	No	Broadcast/Pruning	
MOSPF	Source-based	RPF	No	OSPF + Group	
CBT	Shared	RPF	No	Unicast	
PIM-SM	Shared & S-B	RPF	No	Unicast/Pruning	
YAM	Shared	Greedy	Yes	Multiple paths	
BGMP	Shared	RPF	No	BGP	
SM	Shared	RPF	No	Unicast	No
QoSMIC	Shared & S-B	Greedy	Yes	Multiple paths	No

Table 1-1: Properties of multicast protocols

1.3.2 QoS Routing

QoS routing is an important element for supporting multimedia applications. The goal of QoS routing is to select the network routes with sufficient resources for the requested QoS parameters. It is to satisfy the QoS requirements for every admitted connection, as well as to achieve efficiency in resource utilization. Many QoS routing algorithm have been proposed recently with a variety of constraints considered, and a survey of recent development in this area was presented in Figure 1.4 [9] [3].

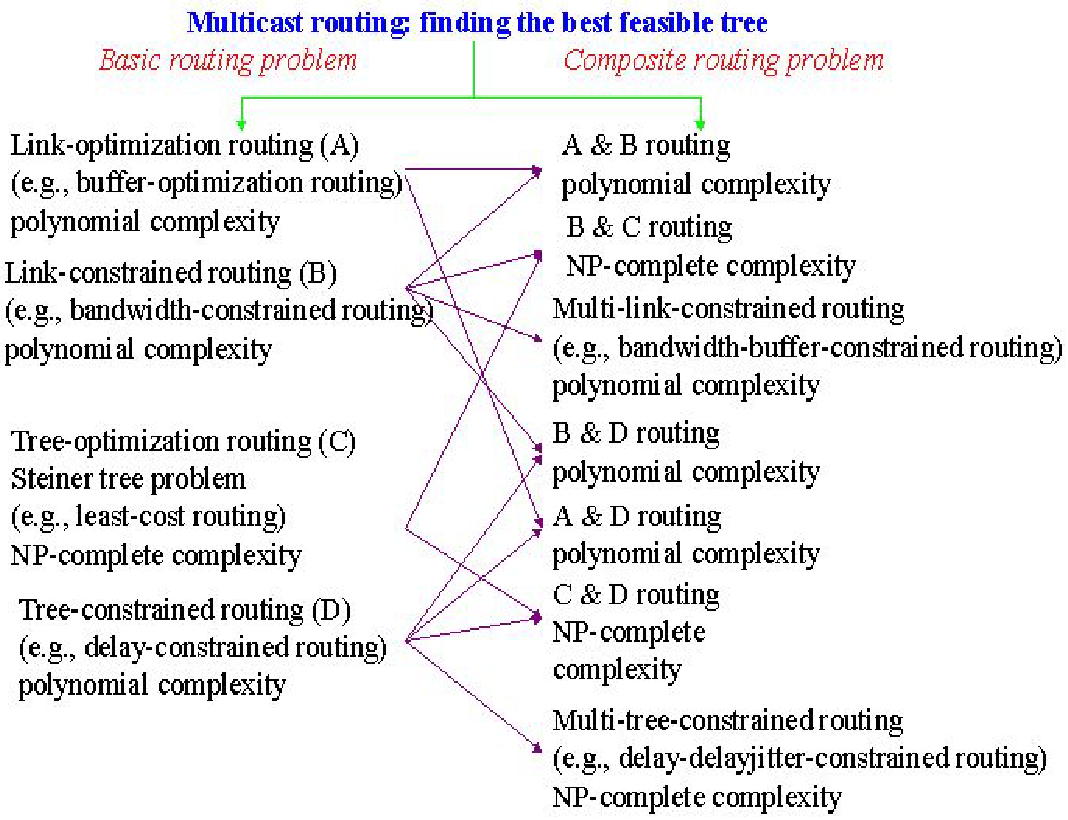

Figure 1-4: The category of multicast routing problems

Wang and Crowcroft [40] consider a number of issues in QoS routing. They try to evaluate the basic component of QoS routing, namely, finding a path that satisfies multiple constraints and its implications on routing metric selection. Moreover, they propose three path computation algorithms for source routing and hop-by-hop routing. However, QoS routing is an integrated part of a resource management system, it should be jointly considered with other components in resource management architectures, such as admission control.

Ergun, Sinha, and Zhang [13] examine a network model in which each link is associated with a set of delays and costs. The aim is to choose a path for each O-D pair and determine a set of per link delay guarantees along this path so as to satisfy the requested constraint while minimizing the total cost. In the case where the O-D path is known, they try to optimally partition the end-to-end delay constraint into link constraints along the path. They present approximations algorithms for both problems. Although the first polynomial-time ε-approximations are presented. However, for PARTION problems, the authors use the heuristics to solve. Besides, they do not consider the more complicated structures, such as multicast trees.

Fang and Ellen [14] specifically focus on topology aggregation, which can reduce overhead by orders of magnitude. They also investigate the interaction of topology aggregation with other important factors that contribute to performance, such as routing algorithms and network configuration. They consider five common route selection methods and propose two methods of aggregating routing information. As a result, for multimedia application, we can adopt this scalable concept to adjust the above route selection methods and different network configuration for satisfying our efficient and flexible principles.

Most QoS routing algorithms consider the optimization of resource utilization measured by an abstract metric such as cost, George, Roch, Sanjay, and Satish [16] study complexity and frequent computations costs and propose solutions that achieve good performance with reduced cost, such as higher level admission control in heavy load environment. It is called “trunk reservation”. However, from a network operator’s point of view, it would be beneficial to develop a generic algorithm in advance instead of implementing the approaches in execution time.

Chen and Nahrstrdt [9] discuss the QoS requirement of a connection. It is a set of constraints, which can be link constraints, path constraints, or tree constraints. The basic function of QoS routing is to find such a feasible path (tree). Therefore, constructing multicast trees in our model will jointly consider the above constraints.

To sum up, for the lack of consideration of other equivalently important components and applicable levels, the QoS mathematical formulations are proposed in this work including minimum spanning tree, and partial admission control.

1.3.3 Minimum spanning tree

For finding a minimum spanning tree, many papers are published. In generally, they include two parts, theoretical and practical aspects [23]. What we attempt to solve in this thesis is the latter. However, for the purpose of whole, we give a brief introduction.

In the theoretical side, the authors consider how to design algorithms with the best taking linear time in a randomized expected case model. Graham and Hell give a complete survey of results from the earliest known algorithm of Bor°uvka to the invention of Fibonacci Heaps, which were central to the algorithms in Fredman and Tarjan and Gabow et al. Chazelle presented an MST algorithm based on the Soft Heap having complexity $O(m\alpha(m; n) \log \alpha(m; n))$, where α is a certain inverse of Ackermann's function, m is the number of edges and n is the number of vertices. Subsequently, Chazelle modified the algorithm in Chazelle to bring down the running time to $O(m\alpha(m; n))$. Later a similar algorithm of the same running time was presented by Pettie, which gives an alternate exposition of the $O(m\alpha(m; n))$ result. Pettie and Ramachandran present a deterministic algorithm to find a minimum spanning tree of a graph with n vertices and m edges that runs in time $O(T^*(m; n))$ where $T^*(m;n)$ is the minimum number of edge-weight comparisons needed to determine the MST [32][23].

In the practical side, finding minimum cost spanning trees is the most fundamental and well-studied network design problem. For instance, one may wish to connect components of a VLSI circuit by networks of wires, in a way that uses little surface area on the chip, draws little power, and propagates signals quickly. Similar problems come up in other applications such as telecommunications, road network design, and medical imaging [12].

Przytycka and Higham [35] assume that associated with each link is a positive weight representing the cost of sending one message along the link and the cost on a weighted network is the sum of the costs of all messages sent during its execution. They present an efficient cost-sensitive, asynchronous, distributed that finds a minimum spanning tree in a weighted connected network of processors. However, the link costs in the network are fixed. In our variant of the problem, the link costs are dependent upon the set of destinations that share the link.

Salama, Reeves, Viniotis [37] formulate the problem of constructing broadcast trees for real-time traffic with delay constraints as a delay-constrained minimum spanning tree (DCMST) problem in directed networks. They propose delay-constrained minimum Steiner tree heuristic to solve the problem. Simulation results indicate that the fastest delay-constrained minimum Steiner tree heuristic, DMCT, is not as efficient as the heuristic they propose, while the most efficient delay-constrained minimum Steiner tree heuristic, BSMA, is much slower than their proposed heuristic and does not construct delay-constrained broadcast trees of lower cost. However, in many applications, multicast will be more suitable and we can consider other constrained factors such as bandwidth.

Maxemchuk [29] discuss the issue of video distribution on multicast networks. This type of application has a greater demand for network bandwidth than e-mail or most information retrieval functions on the WWW. His goal is to construct a minimum cost tree from the source to every destination. Destinations can request different bandwidth signals from the source. The source transmits only one signal that is sufficient for the highest bandwidth destination. However, the author’s solution approach is heuristic-based. Obviously, in his work, it could be further optimized. Charikar, Naor, and Schieber [8] extend this concept to present heuristics with provable performance guarantees for the Steiner tree problem in the rate model and the priority model. However, no simulation results are reported to justify the proposed approaches.

A minimum cost multicast tree is also referred to as a Steiner tree. That is to say, a Steiner tree is to construct a minimum cost tree for a subset of the nodes in a network with fixed costs on the corresponding network links. The problem of determining a Steiner tree is known to be NP-complete [17]. The problem that we intend to solve is related to the Steiner tree problem but it is complicated, due to not every receiver requires the same bandwidth. Therefore, using any deterministic algorithms would not be capable of solving the problems in polynomial time. In other words, adopting other non-deterministic algorithms to solve the variant of the problem is necessary, such as Lagrangean method.

From the above mentioned, according to different applications, constructing different minimum spanning tree to satisfy different constraints is required and practical.

1.3.4 Admission control

The objective of admission control is to control the operation of a network in such a way to ensure the uninterrupted service provision to the existing connections and at the same time to accommodate in an optimum way the new connection requests. In other words, the decision is based on (1) does the new connection affect the QoS of the connections currently being provided by the network? (2) can the network provide the QoS requested by the new connection? Once a request is accepted, the required resources must be guaranteed [33].

Cetinkaya and Knightly [7] propose the solutions to perform admission control based on passive measurements. Routers monitor the passing traffic. While they receive a set-up request, they decide about the admission of that service based on the collected estimates about the current resource usage. This technique is less precise than the active measurement approach in the estimation of the available resources, and it requires that each router is able to perform admission control mechanism. Lai and Baker [26] adopt the active measurement technique, which is used to estimate the capacity of the bottleneck link along a path. However, all those solutions do not involve mechanisms to deal with multicast communications.

Firoiu and Towsley [15] consider the problem of admission control is decomposed into several subproblems that include: the division of end-to-end QoS requirements into local QoS requirements, the mapping of local QoS requirements into resource requirements, and the reclaiming of the resources allocated in excess. They solve the independent subproblems by a set of mechanisms and policies that can be used to provide admission control and resource reservation for multicast connection establishment. However, route establishment is an important part of connection process. The solution of the problem will be better if jointly considered the routing and admission control problems.

Jia, Zhang, Pissinou, and Makki [23] present a real-time multicast connection setup mechanism, which integrates multicast routing with real-time admission control and performs the real-time admission experiments on a cost optimal tree (COT) and a shortest path tree (SPT) in parallel, aiming at optimizing network cost of the routing tree under the real-time constraints. It has the following important features: (1) it is fully distributed; (2) it achieves sub-optimal network cost of routing trees; (3) it takes less time and less exchanged messages for a connection setup. However, the link costs in the network are fixed. In our model, the link costs are dependent upon the set of the destinations that share the link.

Pagani and Rossi [32] propose the call admission multicast protocol (CAMP) to provide bandwidth guarantees to multicast applications with dynamic changes of the destination group membership. They prove that the protocol terminates, and that it avoids the destination making the uncorrected decision. Simulation results show that the devised mechanism effectively performs admission control. However, without considering the property of heterogeneous destinations in their proposed methods is the weakness. It makes room for space.

Tang, Tsui, and Wang [38] describe three basic components of admission control schemes in Figure 1.5: traffic descriptors, admission criteria, and measurement processes. A request will be accepted or not depends on the three factors. However, most of the researches focus on “measurement processes”. We will make some changes on “admission criteria” instead of current usage strategy for another breakthrough.

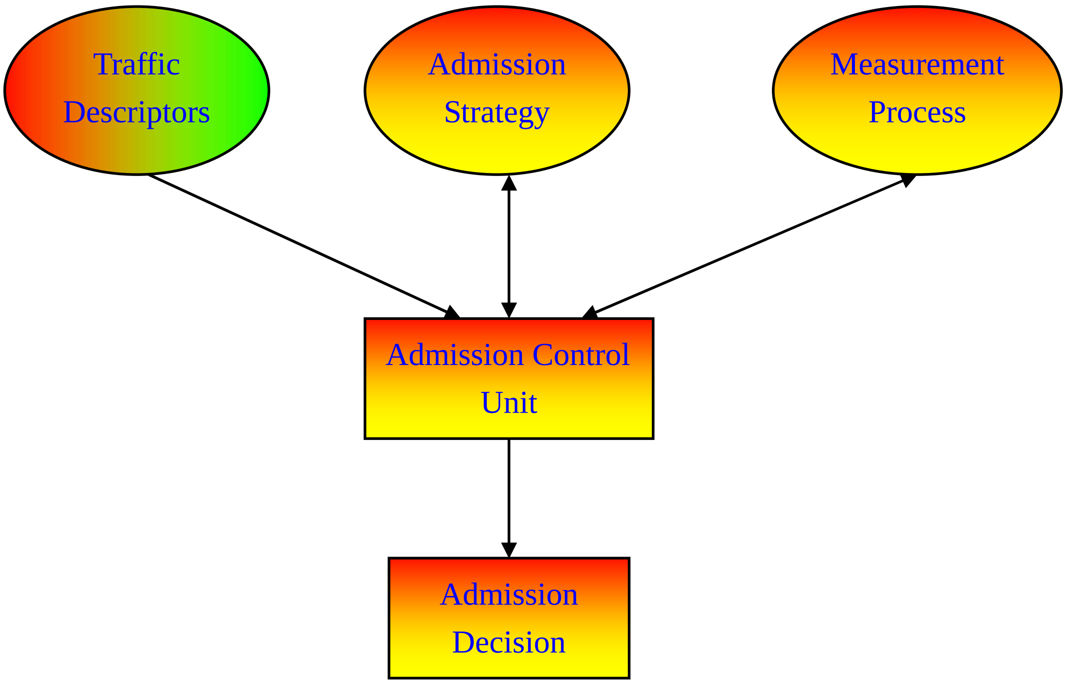

Figure 1-5: The Relationship between basic components of admission control schemes

We may learn lessons from the above introductions to know that in order to consider the

QoS assurance issue for the broadband Internet, the three closely-related mechanisms, admission control, routing and resource reservation, should be considered jointly [1]. Furthermore, an alternative admission control mechanism such as "partial admission control" collocate other components may bring about the wonderful effect.

1.3.5 Lagrangean Relaxation

Lagrangean methods were used in scheduling and the general integer programming problems at first. But as time went by, it has become one of the best tools for optimization problems such as integer programming, linear programming combinatorial optimization, and non-linear programming [16]. Adopting Lagrangean relaxation as our approach has several advantages [4].

1. Lagrangean relaxation is a highly flexible approach since it is often possible to divide and conquer models in several ways and properly apply Lagrangean relaxation to each different subproblem.
2. In decomposing problems, Lagrangean relaxation solves primal problems as individual components; consequently, the solution approach permits us to exploit any known methodology or algorithm for solving the problem.
3. We can use Lagrangean relaxation methods to devise effective heuristic solution methods for solving complex combinatorial optimization problems and integer problems.

Lagrangean relaxation method permits us to remove constraints and instead place them in the objective function with associated Lagrangean multipliers. The optimal value of the relaxed problem is always a lower bound (for minimization problems) on the objective function value of the problem. By adjusting the multiplier of Lagrangean relaxation, we can get the upper bound and the lower bound of this problem in Figure 1.6. We can solve the Lagrangean multiplier problem in a variety of ways. The subgradient optimization technique is possibly the most popular technique for solving the Lagrangean multipliers problem [16] [18].

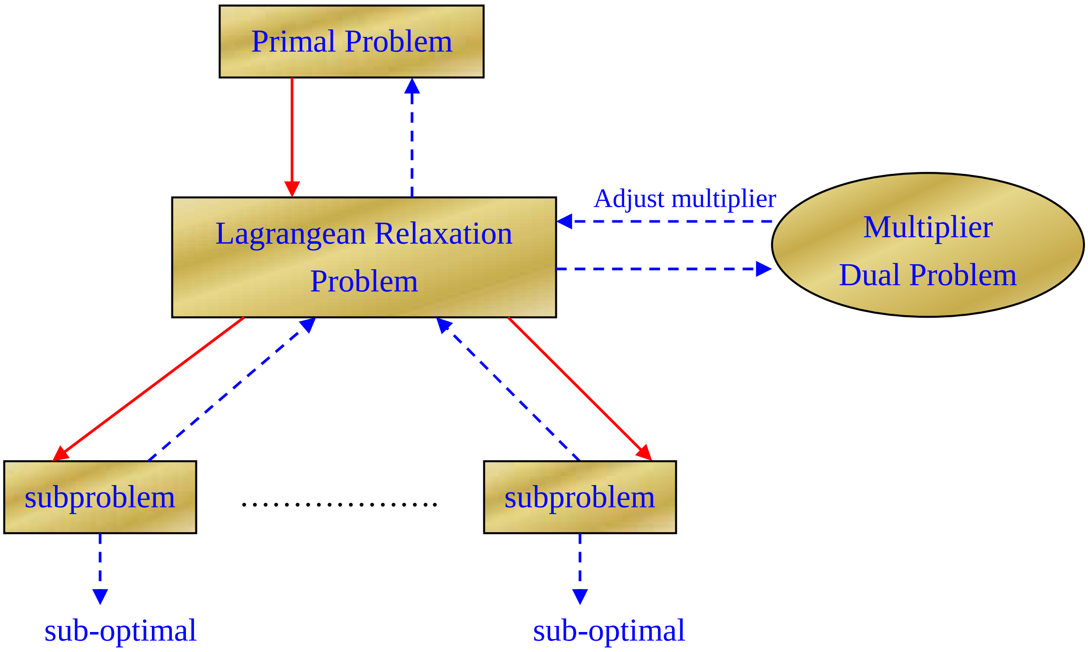

Figure 1-6: Using branch and bound approach to get feasible solution

1.4 Proposed Approach

For multicasting multimedia distributions, we aim to design the efficient and flexible mechanisms. We use different models to represent the static application environment and the dynamic application environment.

We model the problems as optimization problems. In the structure of mathematics, they undoubtedly have the properties of nonlinear programming problems, which are belonging to nonlinear programming problems. We will apply the Lagrangean relaxation method and the subgradient method to solve these problems [3] [4] [16].

1.5 Thesis Organization

The organization of this thesis is as following: Chapter 2 provides two problems and their mathematical formulations—minimum cost tree model and partial admission control model. Chapter 3 provides the Lagrangean Relaxation approach, the problem decomposition, and the optimal solution to each subproblem. Chapter 4 describes how to get primal feasible solutions and its heuristics of each problem. Chapter 5 is our computational experiments and results for each problem. Finally, Chapter 6 is the summary of this thesis and also suggests some direction for the future works.

Chapter 2 Problem Formulation

2.1 Problem I: Minimum cost tree model

2.1.1 Problem Description

The network is modeled as a graph where the switches are depicted as nodes and the links are depicted as arcs. A user group is an application requesting for transmission in this network, which has only one source but more than one destination. Given the network topology, the capacity of links and the bandwidth requirement of every user group's destination, we want to jointly determine the following decision variables: (1) the routing assignment (a tree for multicasting or path for unicasting) of each user group; and (2) the maximum allowable traffic requirement of each multicast user group through each link.

In this model, the problem of constructing a multicast tree for multimedia distribution is a variant of the Steiner tree problem. The link costs in Steiner tree problem are invariant. However, in our first model, the link costs are dependent upon the set of destinations that share the link. So, we propose a minimum cost tree model to establish a network that may ensure the source provides the bandwidth requested by the destinations. Destinations can request different bandwidth requirements from the source. The source transmits only signal that is sufficient for the highest bandwidth destination in Figure 2.1. This model is especially suitable for supporting video and audio applications that require stricter quality of service guarantees. By formulating the problem as a mathematical programming problem, we intend to solve this mathematical problem optimally by obtaining a network fitting into our goal, that is, to make sure the network operator would spend minimum cost on constructing the multicast tree.

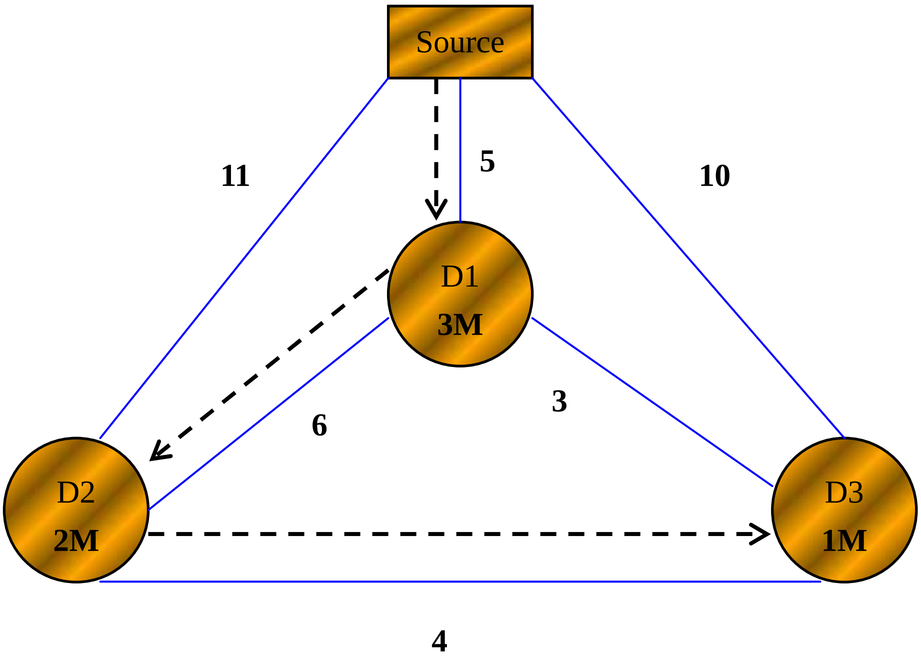

Figure 2-1: The diagram of minimum cost tree model

2.1.2 Notation

Given Parameters	
Notation	**Descriptions**
a_l	The transmission unit cost associated with link l
α_{gd}	The traffic requirement of destination d of multicast group g
G	The set of multicast groups in the network
V	The set of nodes in the network
L	The set of links in the network
D_g	The set of destinations of multicast group g
h_g	The minimum number of hops to the farthest destination node in multicast group g
C_l	The capacity of link l
I_v	The incoming links to node *v*
r_g	The multicast root of multicast group *g*
I_{r_g}	The incoming links to node r_g
P_{gd}	The set of paths destination d of multicast group g may use
δ_{pl}	The indicator function which is 1 if link l is on path p and 0 otherwise

Table 2-1: Notation of Problem I Given Parameters

Decision Variables	
Notation	**Descriptions**
x_{gpd}	1 if path p is selected for group g destined for destination d and 0 otherwise
y_{gl}	1 if link *l* is on the subtree adopted by multicast group *g* and 0 otherwise

m_{gl}	The maximum traffic requirement of the destinations in multicast group g that are connected to the source through link l

Table 2-2: Notation of Problem I Decision Variables

2.1.3 Problem I Formulation

Optimization Problem:

Objective function:

$$\min \sum_{g\in G}\sum_{l\in L} a_l\ m_{gl} \qquad \text{(IP1.1)}$$

subject to:

$$\sum_{p\in P_{gd}} x_{gpd}\alpha_{gd}\delta_{pl} \le m_{gl} \qquad \forall g\in G, d\in D_g, l\in L \qquad (1.1)$$

$$\sum_{g\in G} m_{gl} \le C_l \qquad \forall l\in L \qquad (1.2)$$

$$m_{gl} \in [0, \max_{d\in D_g} \alpha_{gd}] \qquad \forall l\in L, g\in G \qquad (1.3)$$

$$\sum_{l\in L} y_{gl} \ge \sum_{d\in D_g}\sum_{p\in P_{gd}} x_{gpd} \qquad \forall g\in G \qquad (1.4)$$

$$y_{gl} = 0 \text{ or } 1 \qquad \forall l\in L, g\in G \qquad (1.5)$$

$$\sum_{l\in L} y_{gl} \ge \max\{h_g, |D_g|\} \qquad \forall g\in G \qquad (1.6)$$

$$\sum_{d\in D_g}\sum_{p\in P_{gd}} x_{gpd}\delta_{pl} \le |D_g| y_{gl} \qquad \forall g\in G, l\in L \qquad (1.7)$$

$$\sum_{l\in I_v} y_{gl} \le 1 \qquad \forall g\in G, v\in V-\{r_g\} \qquad (1.8)$$

$$\sum_{l\in I_{r_g}} y_{gl} = 0 \qquad \forall g\in G \qquad (1.9)$$

$$\sum_{p\in P_{gd}} x_{gpd} = 1 \qquad \forall d\in D_g, g\in G \qquad (1.10)$$

$x_{gpd} = 0$ or 1 $\quad\quad \forall d \in D_g, g \in G, p \in P_{gd} \quad\quad$ (1.11)

The objective function of (IP1.1) is to minimize the total transmission cost of servicing the maximum bandwidth requirement destination on all links L for all multicast groups G, where L is the set of links and G is the set of user groups requesting for connection. The maximum bandwidth requirement on a link in the specific group m_{gl} can be viewed to that the source would be required to transmit in a way for matching the most constrained destination.

Constraint (1.1) and (1.2) is referred to the capacity constraint, which requires that the aggregate flows on each link l does not exceed its physical capacity C_l. In constraints (1.1), a variable m_{gl} is introduced. In this model, the variable m_{gl} can be interpreted as the "estimate" of the aggregate flows. Since the objective function is strictly increasing with m_{gl} and (IP1.1) is a minimization problem, each m_{gl} will equal the aggregate flow in an optimal solution. Constraint (1.3) is a redundant constraint which provides upper and lower bounds on the maximum traffic requirement for multicast group g on link l. Constraints (1.4) requires that if one path is selected for group *g* destined for destination *d*, it must also be on the subtree adopted by multicast group g. Constraint (1.5) and (1.6) require that the number of links on the multicast tree adopted by the multicast group g be at least the maximum of h_g and the cardinality of D_g. The h_g and the cardinality of D_g are the legitimate lower bounds of the number of links on the multicast tree adopted by the multicast group g. Constraint (1.7) is referred to as the tree constraint, which requires that the union of the selected paths for the destinations of user group *g* forms a tree. Constraints (1.8) and

(1.9) are both redundant constraints. Constraint (1.8) requires the number of selected incoming links y_{gl} to node is 1 or 0. Constraint (1.9) requires there is no selected incoming links y_{gl} to node that is the root of multicast group *g*. As a result, the links we select can form a tree. Constraint (1.10) and (1.11) requires that exactly one path is selected for each multicast source/destination pair.

In this basic model, bandwidth is the only one QoS requirements that we consider. However, enhance or modify some constraints would be able to deal with other issues, such as the delay requirement. The left side of constraint 1.7 represents the number of usage for selected link l on multicast group g. Remove the first summation signal makes the remainder $\sum_{p \in P_{gd}} x_{gpd} \delta_{pl}$ stands for the number of usage for selected link l through destination d on multicast group g. We assume D_l as the average delay on link $l \in L$, L_{gd} as the end-to-end mean delay requirement for destination d of multicast group g. Combine the above terms will become the delay constraint. The left side of the constraint represents the end-to-end delay of destination d and the right side of the constraint represents the maximum allowable end-to-end delay requirement of destination d. $\sum_{l \in L} \sum_{p \in P_{gd}} x_{gpd} \delta_{pl} D_l \le L_{gd}$ requires the end-to-end average delay should be no longer than the maximum allowable end-to-end average delay requirement for destination d. Take the delay constraint into account is especially suitable for multimedia applications, such as video conferencing. As a result, through proper extension will enable our basic model to handle other QoS requirements is undoubted.

2.2 Problem II: Partial admission control model

2.2.1 Problem Description

The network is modeled as a graph where the switches are depicted as nodes and the links are depicted as arcs. A user group is an application requests for transmission in this network, and it has only one source but more than one destination. Given the network topology, the capacity of links and the bandwidth requirement of every user group's destination, we want to jointly determine the following decision variables: (1) the routing assignment (a tree for multicasting or path for unicasting) of each admitted destination; and (2) the admitted number of destinations of each partial admitted multicast group. In this model, for simplicity and comparison reason, we assume the same traffic requirement for all destinations in one group.

In this partial admission control model, we establish a network that allows the admission policy of the multicast group does not base on traditional "all or none" strategy. Instead of considering that accepts partial portions of destinations for the requested multicast group. This flexible model is suitable for supporting services that is characteristic of divergence, because destinations of a group may vary significantly in their interests. By formulating the problem as a mathematical programming problem, we intend to solve this mathematical problem optimally by obtaining a network fitting into our goal, which means to make sure the network operator can earn maximum revenue by servicing the partial admitted destinations.

This model is based on the following viable assumptions.

- The revenue on each partial admitted group could be fully characterized by two parameters: the entire admitted revenue of the group and the number of admitted destinations.

- The revenue on each partial admitted group is a monotonically increasing function with respect to the number of admitted destinations.
- The revenue function on each partial admitted group is a concave function with respect to the entire admitted revenue of the group and the number of admitted destinations. But the entire admitted revenue and the number of admitted destinations jointly may not be a concave function.
- The revenue on each partial admitted group is independent.

2.2.2 Notation

Given Parameters	
Notation	**Descriptions**
F_g	The revenue generated from admitting partial users of multicast group g , which is a function of f_g and a_g
a_g	The revenue generated from admitting multicast group g
α_g	The traffic requirement of multicast group g
G	The set of multicast groups in the network
V	The set of nodes in the network
L	The set of links in the network
D_g	The set of destinations of multicast group g
C_l	The capacity of link l
I_v	The incoming links to node v
r_g	The multicast root of multicast group g

I_{r_g}	The incoming links to node r_g
P_{gd}	The set of paths user d of multicast group g may use
δ_{pl}	The indicator function which is 1 if link l is on path p and 0 otherwise

Table 2-3: Notation of Problem II Given Parameters

Decision Variables	
Notation	**Descriptions**
x_{gpd}	1 if path p is selected for group g destined for destination d and 0 otherwise
y_{gl}	1 if link l is on the subtree adopted by multicast group g and 0 otherwise
f_g	The number of admitted destinations in multicast group g

Table 2-4: Notation of Problem II Decision Variables

2.2.3 Problem II Formulation

Optimization Problem:

Objective function:

$$\min \; -\sum_{g \in G} F_g(a_g, f_g) \qquad \text{(IP2.1)}$$

subject to:

$$\sum_{g\in G}\alpha_g y_{gl} \le C_l \qquad \forall l \in L \qquad (2.1)$$

$$\sum_{d\in D_g}\sum_{p\in P_{gd}} x_{gpd} \le \sum_{l\in L} y_{gl} \qquad \forall g \in G \qquad (2.2)$$

$$\sum_{d\in D_g}\sum_{p\in P_{gd}} x_{gpd}\delta_{pl} \le \left|D_g\right| y_{gl} \qquad \forall g \in G, l \in L \qquad (2.3)$$

$$\sum_{l\in I_v} y_{gl} \le 1 \qquad \forall g \in G, v \in V - \{r_g\} \qquad (2.4)$$

$$\sum_{l\in I_{r_g}} y_{gl} = 0 \qquad \forall g \in G \qquad (2.5)$$

$$x_{gpd} = 0 \text{ or } 1 \qquad \forall g \in G, p \in P_{gd}, d \in D_g \qquad (2.6)$$

$$\sum_{d\in D_g}\sum_{p\in p_{gd}} x_{gpd} = f_g \qquad \forall g \in G \qquad (2.7)$$

$$f_g \in \{0,1,2,......,\left|D_g\right|\} \qquad \forall g \in G \qquad (2.8)$$

The objective function of (IP2.1) is to maximize the total "revenue" F_g of servicing the partial admitted destinations of multicast groups g, where $g \in G$ and G is the set of user groups requesting for transmission. F_g can be viewed to reflect the priority of partial users belonging to group g, while different choices of F_g may provide different physical meanings of the objective function. For example, if F_g is chosen to be the mean traffic requirement of partial users belonging to group g, then the objective function is to maximize the total system throughput. If F_g is chosen to be the earnings of servicing partial users belonging to group g, then the objective function is to maximize the total system revenue. In general, if a user group g is to be given a higher priority, then the corresponding F_g may be assigned a larger value.

Constraint (2.1) is referred to the capacity constraint, which requires that the aggregate flows on each link *l* not exceed its physical capacity C_l. Constraints (2.2) requires that if one path is selected for group g destined for destination d, it must also be on the subtree adopted by multicast group g. Constraint (2.3) is referred to as the tree constraint, which requires that the union of the selected paths for the destinations of user group *g* forms a tree. Constraint (2.4) requires the number of selected incoming links y_{gl} to node is 1 or 0. Constraint (2.5) requires there is no selected incoming links y_{gl} to node that is the root of multicast group *g*. As a result, the links we select can form a tree. Constraints (2.6) and (2.7) requires that exactly one path is selected for each admitted multicast source/destination pair and Constraint (2.7) relates the routing decision variables x_{gpd} to the auxiliary variables f_g. The introduction of the auxiliary variables f_g may facilitate the decomposition in the Lagrangean relaxation problem to be discussed later. Constraint (2.8) requires that the number of admitted destinations in multicast group g are the set of integers. Thus, we can easily decide the value of them.

Obviously, this model is a simplified version without considering priority. This effect influences the revenue function F_g. Moreover, we assume the revenue function is a concave function so that the Lagrangean Relaxation can easy to work. Nevertheless, adopt appropriate transformation would make this model more generic in spite of viability. Take an example from the objective function $\min - \sum_{g \in G} F_g(a_g, f_g)$. The objective function of (IP2.1) is to maximize the total "revenue" of servicing the partial admitted destinations of multicast groups g. However, through the left side of constraint 2.7 $\sum_{d \in D_g} \sum_{p \in p_{gd}} x_{gpd}$, we get the information of the total admitted destinations of multicast group g. And we introduce a_{gd} as the revenue generated from admitting destination d of multicast group g. The revised objective function is becoming $\sum_{g \in G} \sum_{d \in D_g} \sum_{p \in P_{gd}} x_{gpd} a_{gd}$. This modified representation strengthens our original revenue function to improve the generality and practicability.

Chapter 3 Lagrangean Relaxation

3.1 Problem I: Minimum cost tree model

3.1.1 Solution Approach

By using the Lagrangean Relaxation method, we can transform the primal problem (IP1.1) into the following Lagrangean Relaxation problem (LR1.1) where constraints (1.1), (1.4) and (1.7) are relaxed.

3.1.2 Lagrangean Relaxation

For a vector of non-negative Lagrangean multipliers, a Lagrangean Relaxation problem of (IP1.1) is given by

Optimization problem (LR1.1):

$$Z_{D1.1}(\beta,\lambda,\theta)=\min \sum_{g\in G}\sum_{l\in L}a_l m_{gl}+\sum_{g\in G}\sum_{d\in D_g}\sum_{l\in L}\sum_{p\in P_{gd}}\beta_{gdl}x_{gpd}\alpha_{gd}\delta_{pl}-\sum_{g\in G}\sum_{d\in D_g}\sum_{l\in L}\beta_{gdl}m_{gl}$$
$$+\sum_{g\in G}\sum_{l\in L}\lambda_g y_{gl}-\sum_{g\in G}\sum_{d\in D_g}\sum_{p\in P_{gd}}\lambda_g x_{gpd}+\sum_{g\in G}\sum_{d\in D_g}\sum_{l\in L}\sum_{p\in P_{gd}}\theta_{gl}x_{gpd}\delta_{pl}-\sum_{g\in G}\sum_{l\in L}\theta_{gl}\left|D_g\right|y_{gl}$$

(LR1.1)

subject to:

$\sum_{g\in G} m_{gl}\le C_l$	$\forall l\in L$	(1.2)
$m_{gl}\in[0,\max_{d\in D_g}\alpha_{gd}]$	$\forall l\in L, g\in G$	(1.3)
$y_{gl}=0$ or 1	$\forall l\in L, g\in G$	(1.5)
$\sum_{l\in L} y_{gl}\ge \max\{h_g,\left|D_g\right|\}$	$\forall g\in G$	(1.6)
$\sum_{l\in I_v} y_{gl}\le 1$	$\forall g\in G, v\in V-\{r_g\}$	(1.8)
$\sum_{l\in I_{r_g}} y_{gl}=0$	$\forall g\in G$	(1.9)
$\sum_{p\in P_{gd}} x_{gpd}=1$	$\forall d\in D_g, g\in G$	(1.10)
$x_{gpd}=0$ or 1	$\forall d\in D_g, g\in G, p\in P_{gd}$	(1.11)

Where $\beta_{gdl},\lambda_{g,}\theta_{gl}$ are Lagrangean multipliers and $\beta_{gdl},\theta_{gl}\ge 0$. To solve (LR1.1), we can decompose (LR1.1) into the following three independent and easily solvable optimization subproblems.

Subproblem 1.1: (related to decision variable x_{gpd})

$$Z_{Sub1.1}(\beta,\lambda,\theta) = \min \sum_{g\in G}\sum_{d\in D_g}\sum_{p\in P_{gd}}(\sum_{l\in L}\beta_{gdl}\alpha_{gd}\delta_{pl} + \sum_{l\in L}\theta_{gl}\delta_{pl} - \lambda_g)x_{gpd}$$

subject to:

$\sum_{p\in P_{gd}} x_{gpd} = 1$	$\forall d \in D_g, g \in G$	(1.8)
$x_{gpd} = 0$ or 1	$\forall d \in D_g, g \in G, p \in P_{gd}$	(1.9)

The Subproblem 1.1 can be further decomposed into $|G||D_g|$ independent shortest path problems with nonnegative arc weights. Each shortest path problem can be easily solved by the Dijkstra's algorithm.

Subproblem 1.2: (related to decision variable y_{gl})

$$Z_{Sub1.2}(\lambda,\theta) = \min \sum_{g\in G}\sum_{l\in L}(\lambda_g - \theta_{gl}\left|D_g\right|)y_{gl}$$

subject to:

$y_{gl} = 0$ or 1	$\forall l \in L, g \in G$	(1.3)
$\sum_{l\in L} y_{gl} \ge \max\{h_g, \left|D_g\right|\}$	$\forall g \in G$	(1.4)
$\sum_{l\in I_v} y_{gl} \le 1$	$\forall g \in G, v \in V - \{r_g\}$	(1.6)
$\sum_{l\in I_{r_g}} y_{gl} = 0$	$\forall g \in G$	(1.7)

The algorithm to solve Subproblem 1.2 is stated as follows [3]:

Step 1. Compute $\max\{h_g, \left|D_g\right|\}$ for multicast group *g*.

Step 2. Compute the number of negative coefficient $\lambda_g - \theta_{gl} \left| D_g \right|$ for all links on multicast group *g*.

Step 3. If the number of negative coefficient is greater than $\max\{h_g, \left| D_g \right|\}$ for multicast group *g*, then assigns the corresponding negative coefficient of y_{gl} to 1 and 0 otherwise.

Step 4. If the number of negative coefficient is no greater than $\max\{h_g, \left| D_g \right|\}$ for multicast group *g*, then assign the corresponding negative coefficient of y_{gl} to 1. Then, assigns [$\max\{h_g, \left| D_g \right|\}$ – the number of negative coefficient of y_{gl}] numbers of smallest positive coefficient of y_{gl} to 1 and 0 otherwise.

Subproblem 1.3: (related to decision variable m_{gl})

$$Z_{Sub1.3}(\beta) = \min \sum_{g \in G} \sum_{l \in L} (a_l - \sum_{d \in D_g} \beta_{gdl}) m_{gl}$$

subject to:

$\sum_{g \in G} m_{gl} \le C_l$	$\forall l \in L$	(1.2)
$m_{gl} \in [0, \max_{d \in D_g} \alpha_{gd}]$	$\forall l \in L, g \in G$	(1.3)

First, we decompose Subproblem 1.3 into $\left| L \right|$ independent problems. For each link $l \in L$:

$$Z_{Sub1.3.1}(\beta) = \min \sum_{g \in G} (a_l - \sum_{d \in D_g} \beta_{gdl}) m_{gl} \qquad \text{(Subproblem 1.3.1)}$$

subject to:

$\sum_{g \in G} m_{gl} \le C_l$	$\forall l \in L$	(1.2)
$m_{gl} \in [0, \max_{d \in D_g} \alpha_{gd}]$	$\forall l \in L, g \in G$	(1.3)

The algorithm to solve Subproblem 1.3.1 is stated as follows:

Step 1. Compute $a_l - \sum_{d \in D_g} \beta_{gdl}$ for link *l* of multicast group *g*.

Step 2. Sort the negative coefficient $a_l - \sum_{d \in D_g} \beta_{gdl}$ from the smallest value to the largest value

Step 3. According to the sorted sequence. <i> assigns the corresponding m_{gl} to the maximum traffic requirement in multicast group and add to the sum value until the total amount of maximum traffic requirement on link *l* is less than the capacity of link *l*. <ii>assigns the boundary negative coefficient of m_{gl} to the difference between the capacity on link *l* and the sum value of m_{gl} <iii> assigns the others' coefficients of m_{gl} to 0.

3.1.3 The Dual Problem and the Subgradient Method

According to the weak Lagrangean duality theorem [18], for any $\beta_{gdl}, \theta_{gl} \ge 0$, $Z_{D1.1}(\beta_{gdl}, \lambda_g, \theta_{gl})$ is a lower bound on $Z_{IP1.1}$. The following dual problem (D1.1) is then constructed to calculate the tightest lower bound.

Dual Problem (D1.1):

$$Z_{D1.1} = \max Z_{\mathrm{D1.1}}(\beta_{gdl}, \lambda_g, \theta_{gl})$$

subject to:

$$\beta_{gdl}, \theta_{gl} \geq 0$$

There are several methods to solve the dual problem (D1.1). Among them is the most popular method, the subgradient method, which is employed here [20]. Let a vector *s* be a subgradient of $Z_{D1.1}(\beta_{gdl}, \lambda_{g,} \theta_{gl})$. Then, in iteration *k* of the subgradient optimization procedure, the multiplier vector is updated by $\omega^{k+1} = \omega^k + t^k s^k$. The step size t^k is determined by $t^k = \delta \dfrac{Z^h{}_{IP1.1} - Z_{D1.1}(\omega^k)}{\left\| s^k \right\|^2}$. $Z^h{}_{IP1.1}$ is the primal objective function value for a heuristic solution (an upper bound on $Z_{IP1.1}$). δ is a constant, $0 < \delta \leq 2$.

3.2 Problem II: Partial admission control model

3.2.1 Solution Approach

By using the Lagrangean Relaxation method, we can transform the primal problem (IP2.1) into the following Lagrangean Relaxation problem (LR2.1) where Constraints (2.1), (2.2), (2.3), (2.7) are relaxed.

3.2.2 Lagrangean Relaxation

For a vector of non-negative Lagrangean multipliers, a Lagrangean Relaxation problem of (IP2.1) is given by

Optimization problem (LR 2.1):

$$\begin{aligned} Z_{D2.1}(\beta,\lambda,\theta,\varepsilon) = \min\ & -\sum_{g\in G} F_g(a_g, f_g) + \sum_{g\in G}\sum_{l\in L}\beta_l\alpha_g y_{gl} - \sum_{l\in L}\beta_l C_l + \sum_{g\in G}\sum_{l\in L}\lambda_g y_{gl} \\ & -\sum_{g\in G}\sum_{d\in D_g}\sum_{p\in P_{gd}}\lambda_g x_{gpd} + \sum_{g\in G}\sum_{l\in L}\sum_{d\in D_g}\sum_{p\in P_{gd}}\theta_{gl}x_{gpd}\delta_{pl} - \sum_{g\in G}\sum_{l\in L}\theta_{gl}\left|D_g\right|y_{gl} \\ & +\sum_{g\in G}\sum_{d\in D_g}\sum_{p\in P_{gd}}\varepsilon_g x_{gpd} - \sum_{g\in G}\varepsilon_g f_g \end{aligned}$$

(LR2.1)

subject to:

$$\sum_{l\in I_v} y_{gl} \le 1 \qquad \forall g\in G, v\in V-\{r_g\} \qquad (2.4)$$

$$\sum_{l\in I_{r_g}} y_{gl} = 0 \qquad \forall g\in G \qquad (2.5)$$

$x_{gpd} = 0$ or 1 $\quad\quad \forall g \in G, p \in P_{gd}, d \in D_g \quad\quad$ (2.6)

$f_g \in \{0,1,2,......,|D_g|\}$ $\quad\quad \forall g \in G \quad\quad$ (2.8)

where $\beta_l, \lambda_g, \theta_{gl}, \varepsilon_g$ are Lagrangean multipliers and $\beta_l, \theta_{gl} \ge 0$. To solve (LR2.1), we can decompose (LR2.1) into the following five independent and easily solvable optimization subproblems.

Subproblem 2.1: (related to decision variable x_{gpd})

$$Z_{Sub2.1}(\lambda, \theta, \varepsilon) = \min \sum_{g \in G} \sum_{d \in D_g} \sum_{p \in P_{gd}} (\varepsilon_g - \lambda_g + \sum_{l \in L} \theta_{gl} \delta_{pl}) x_{gpd}$$

subject to:

$x_{gpd} = 0$ or 1	$\forall g \in G, p \in P_{gd}, d \in D_g$	(2.6)

The Subproblem 2.1 is to determine x_{gpd}. There are three cases to consider:

Case 1. If $\varepsilon_g - \lambda_g < 0$ and the absolute value of $\varepsilon_g - \lambda_g > \sum_{l \in L} \theta_{gl} \delta_{pl}$, then assigns the corresponding x_{gpd} to 1

Case 2. If $\varepsilon_g - \lambda_g < 0$ and the absolute value of $\varepsilon_g - \lambda_g \le \sum_{l \in L} \theta_{gl} \delta_{pl}$, then assigns the corresponding x_{gpd} to 0

Case 3. If $\varepsilon_g - \lambda_g > 0$, then assigns the corresponding x_{gpd} to 0

Subproblem 2.2: (related to decision variable y_{gl})

$$Z_{Sub2.2}(\beta,\lambda,\theta) = \min \sum_{g\in G}\sum_{l\in L}(\beta_l\alpha_g + \lambda_g - \theta_{gl}\left|D_g\right|)y_{gl}$$

subject to:

$\sum_{l\in I_v} y_{gl} \le 1$	$\forall g \in G, v \in V - \{r_g\}$	(2.4)
$\sum_{l\in I_{r_g}} y_{gl} = 0$	$\forall g \in G$	(2.5)

The Subproblem 2.2 can be decomposed into $|G|$ independent problems. For each multicast group $g \in G$:

$$Z_{Sub2.2.1}(\beta,\lambda,\theta) = \min \sum_{l\in L}(\beta_l\alpha_g + \lambda_g - \theta_{gl}\left|D_g\right|)y_{gl} \qquad \text{(Subproblem 2.2.1)}$$

subject to:

$\sum_{l\in I_v} y_{gl} \le 1$	$\forall g \in G, v \in V - \{r_g\}$	(2.4)
$\sum_{l\in I_{r_g}} y_{gl} = 0$	$\forall g \in G$	(2.5)

The algorithm to solve to Subproblem 2.2.1 is stated as follows:

Step 1. Compute the coefficient $\beta_l\alpha_g + \lambda_g - \theta_{gl}\left|D_g\right|$ for each group.

Step 2. If the value is less than zero, assigns the corresponding negative coefficient of y_{gl} to 1 and 0 otherwise.

Subproblem 2.3: (related to decision variable f_g)

$$Z_{Sub2.3}(\varepsilon) = \min \ -\sum_{g\in G}(F_g(a_g, f_g) + \varepsilon_g f_g)$$

subject to:

$f_g \in \{0,1,2,......,\lvert D_g \rvert\}$	$\forall g \in G$	(2.8)

We may easily solve Subproblem 2.3 optimally by exhaustively searching from the known set of f_g.

3.2.3 The Dual Problem and the Subgradient Method

According to the weak Lagrangean duality theorem [18], for any $\beta_l, \theta_{gl} \ge 0$, $Z_{D2.1}(\beta_l, \lambda_g, \theta_{gl}, \varepsilon_g)$ is a lower bound on $Z_{IP2.1}$. The following dual problem (D2.1) is then constructed to calculate the tightest lower bound.

Dual Problem (D2.1):

$$Z_{D2.1} = \max Z_{D2.1}(\beta_l, \lambda_g, \theta_{gl}, \varepsilon_g)$$

subject to:

$$\beta_l, \theta_{gl} \ge 0$$

There are several methods to solve the dual problem (D2.1). Among them is the most popular method, the subgradient method, which is employed here [20]. Let a vector s be a subgradient of $Z_{D2.1}(\beta_l, \lambda_g, \theta_{gl}, \varepsilon_g)$. Then, in iteration k of the subgradient optimization procedure, the multiplier vector is updated by $\omega^{k+1} = \omega^k + t^k s^k$. The step size t^k is determined by $t^k = \delta \frac{Z^h{}_{IP2.1} - Z_{D2.1}(\omega^k)}{\left\| s^k \right\|^2}$. $Z^h{}_{IP2.1}$ is the primal objective function value for a heuristic solution (an upper bound on $Z_{IP2.1}$). δ is a constant, $0 < \delta \le 2$.

Chapter 4 Getting Primal Feasible Solution

After using the Lagrangean relaxation and the subgradient method to solve these problems, not only getting a theoretical lower bound of primal feasible solution, but also providing some hints to help us to find our primal feasible solution under each iteration of solving the dual problem.

Although taking advantage of the above methods, we cannot guarantee that the results of Lagrangean dual problems will be a feasible solution to the primal optimization problem, since there are some constraints relaxed by Lagrangean relaxation.

If the decision variables calculated happen to satisfy the relaxed constraints, then a primal feasible solution is found. Otherwise, to obtain primal feasible solutions, properly modify such infeasible primal solutions is necessary.

4.1 Heuristics for Minimum Cost Tree Model

To calculate the primal feasible solutions of the minimum cost tree model, the solutions to the Lagrangean Relaxation problems are considered. The set of $\left\{x_{gpd}\right\}$ obtained by solving (Subproblem 1.1) may not be a valid solution to problem (IP 1.1) because the capacity constraint is relaxed. The capacity constraint may be violated for some links. The set of $\left\{y_{gl}\right\}$ obtained by solving (Subproblem 1.2) may not be a valid solution. It is because of the link capacity constrain and the union of $\left\{y_{gl}\right\}$ may not be a tree.

Thus, we need additional heuristics to obtain a primal feasible solution. In this section, we propose the comprehensive method. It is composed of two parts, including Lagrangean based modified T-M heuristic and adjustment procedures. Lagrangean based modified T-M heuristic is the beginning task and adjustment procedure is the following task. We describe the detail of the heuristics in the following.

Particularly mentioned one point, this model is supposed to serve all requested multicast groups. Then we assume the multicast group's traffic demand is lightly-load. In other words, this model's goal is only to reduce the total cost since all groups have to be satisfied their needs without considering the admission control issues.

4.1.1 Lagrangean Based Modified T-M Heuristic

Maxemchuk [29] modified the proposed heuristics by Takahashi and Matsuyama (T-M heuristics) to solve the variant Steiner tree problem. However, M.Charikar, J.Naor, and B.Schieber [8] argued that Maxemchuk presented a heuristic for computing a minimum cost Steiner tree, but provided only experimental evidence for its performance. From their results, the cost of the multicast tree generated by modified T-M heuristics was no more than 4.214 times the cost of an optimal multicast tree. Apparently, it was not pretty good. As a result, how to reduce the gap became the main concern. Intuitively, properly integrate the modified T-M heuristics and the results of Lagrangean dual problems may be useful to improve the drawback. We named this method as Lagrangean Based Modified T-M Heuristics. This scenario was composed of two parts: arc weight choices and modified T-M heuristics. In the following section, we describe the detail of the heuristics.

4.1.1.1 Modified T-M Heuristics

Specifically, this heuristic operates as follows:

Step 1.	Separate the receivers into subsets according to rate.
Step 2.	Run the T-M heuristic on the subset with the highest requirements [38].
Step 3.	Once the tree with the subset of receivers with the highest requirements has been constructed, repeat the heuristic with this tree as the starting tree for the subset of receivers with the next highest set of requirements.
Step 4.	Repeat the procedure until all subsets of receivers are connected to the tree.

Table 4-1: The steps of modified T-M heuristics

4.1.1.2 Arc Weight Choices Scheme

While solving the Lagrangean relaxation dual problem, we got some multipliers related to each OD pair and each link. According to the information, we can make our routing more efficient. Therefore, there are several options about how to decide which multipliers to represent the arc weights of the links. In model I, we use three types of multipliers to assign the relative arc weight as follow:

i. For each multicast group, link *l*'s arc weight is equal to $\sum_{d\in D_g}\beta_{gdl}$.

ii. For each multicast group, link *l*'s arc weight is equal to $\sum_{d\in D_g}\{\beta_{gdl}\times\alpha_{gd}$ (Each OD pair's Traffic Demand)}.

iii. For each multicast groups, link *l*'s arc weight is equal to θ_{gl} .

4.1.2 Adjustment Procedures

Initially, through the integration of modified T-M heuristics and arc weights choices to get a primal feasible solution. However, to make the overall performance is better than better and to process some exceptions in non-regular environment. We propose some adjustment procedure options as the following tasks. Nevertheless, redundantly check actions may cause serious performance decline even if the total cost is down. Therefore, we consider the most usual occurrence to reduce the total cost and control the used resources in an acceptable range. We developed three "hop count based" schemes to adjust the initial multicast tree. In this section, the detail of three "hop count based" schemes described.

i.

➢ Compute the number of hops from the source to the destinations

- Choose the node that its downlinks' traffic is maximum to adjust in every hop's iteration
- Consider the following possible criteria and set the best one to be the final tree (connect the others' node having the same hops、connect the source node、connect the nodes having the hop counts is larger or smaller by 1)

ii.

- Compute the number of hops from the source to the destinations
- Choose the node that its out-degree is maximum to adjust in every hop's iteration
- Consider the following possible criteria and set the best one to be the final tree (connect the others' node having the same hops、connect the source node、connect the nodes having the hop counts is larger or smaller than 1)

iii.

- Compute the number of hops from the source to the destinations
- Choose the node that its incoming traffic divides its own traffic demand is maximum to adjust in every hop's iteration
- Consider the following possible criteria and set the best one to be the final tree (connect the others' node having the same hops、connect the source node、connect the nodes having the hop counts is larger or smaller than 1)

4.2 Heuristics for Partial Admission Control Model

To calculate primal feasible solutions of the partial admission control model, the solutions to the Lagrangean Relaxation problems are considered. The set of $\{x_{gpd}\}$ obtained by solving (Subproblem 2.1) may not be a valid solution to problem (IP 2.1) because the capacity constraint is relaxed. The capacity constraint may be violated for some links. The set of $\{y_{gl}\}$ obtained by solving (Subproblem 2.2) may not be a valid solution. It is because of the link capacity constrain and the union of $\{y_{gl}\}$ may not be a tree.

Thus, we need additional heuristics to obtain a primal feasible solution. In this section, we describe the detail of the heuristics. First, the overall algorithm steps are introduced. Next, according to some information from the solved Lagrangean dual problem, we propose the Lagrangean based method to achieve more revenue. Finally, as the deliberation of the above-mentioned, apply adjustment procedures to make the original work better.

4.2.1 The Steps of Proposed Algorithm

Step 1.	**Use the information from the subproblem 2.3:** If $f_g = D_g$, it represents admit all destinations in group g Else, it represents admit partial destinations in group g
Step 2.	**Tree construction (I):**

	Use Lagrangean based modified T-M heuristics to construct the multicast tree for each admitted multicast group.
Step 3.	**Tree construction (II):** Apply the adjustment procedure to tune constructed multicast tree and choose the best one to be the final tree.
Step 4.	**Check capacity constraint for all links:** If it violates, drop some groups by specific criteria order until all links satisfy the rule, go to step 5 Else, we get a feasible solution.
Step 5.	**Drop procedure:** Sort all groups by using the subgradient $-F_g(a_g, f_g) - \varepsilon_g f_g$ from smallest to largest, drop the largest subgradient's group and check capacity constraint until all links are satisfied.
Step 6.	**Add procedure:** Sort all groups by using the subgradient $-F_g(a_g, f_g) - \varepsilon_g f_g$ from smallest to largest, add one destination at a time from the smallest subgradient's group and check capacity constraint until all links are satisfied
Step 7.	**Reconstruct multicast tree and Final checks:** Adopt Lagrangean based modified T-M heuristics and adjustment procedures to reconstruct the final admitted groups. Check the capacity constraint for all links. If it's all-safe, then done. Else, go to step 4

Table 4-2: The steps of proposed algorithm for model II

4.2.2 Lagrangean Based Modified T-M Heuristic

Maxemchuk [29] modified the proposed heuristics by Takahashi and Matsuyama (T-M heuristics) to solve the variant Steiner tree problem. However, M.Charikar, J.Naor, and

B.Schieber [8] argued that Maxemchuk presented a heuristic for computing a minimum cost Steiner tree, but provided only experimental evidence for its performance. From their results, the cost of the multicast tree generated by modified T-M heuristics was no more than 4.214 times the cost of an optimal multicast tree. Apparently, it was not pretty good. As a result, how to reduce the gap became the main concern. Intuitively, properly integrate the modified T-M heuristics and the results of Lagrangean dual problems may be useful to improve the drawback. We named this method as Lagrangean Based Modified T-M Heuristics. This scenario was composed of two parts: arc weight choices and modified T-M heuristics. In the following section, we describe the detail of the heuristics.

4.2.2.1 Modified T-M Heuristics

Specifically, this heuristic operates as follows:

Step 1.	Separate the receivers into subsets according to rate.
Step 2.	Run the T-M heuristic on the subset with the highest requirements [38].
Step 3.	Once the tree with the subset of receivers with the highest requirements has been constructed, repeat the heuristic with this tree as the starting tree for the subset of receivers with the next highest set of requirements.
Step 4.	Repeat the procedure until all subsets of receivers are connected to the tree.

Table 4-3: The steps of modified T-M heuristics

4.2.2.2 Arc Weight Choices Scheme

While solving the Lagrangean relaxation dual problem, we got some multipliers related to each OD pair and each link. According to the information, we can make our routing more efficient. Therefore, there are several options about how to decide which multipliers to represent the arc weights of the links. In model II, we use three types of multipliers to assign the relative arc weight as follow:

i. For all multicast group, link *l*'s arc weight is equal to β_l.

ii. For each multicast group, link *l*'s arc weight is equal to $\{\beta_l \times \alpha_{gd}$ (Each OD pair's Traffic Demand)$\}$.

iii. For each multicast groups, link *l*'s arc weight is equal to θ_{gl}.

4.2.3 Adjustment Procedures

Initially, through the integration of modified T-M heuristics and arc weights choices to get a primal feasible solution. However, to make the overall performance is better than better and to process some exceptions in non-regular environment. We propose some adjustment procedure options as the following tasks. Nevertheless, redundantly check actions may cause serious performance decline even if the total cost is down. Therefore, we consider the most usual occurrence to reduce the total cost and control the used resources in an acceptable range. We developed three "hop count based" schemes to adjust the initial multicast tree. In this section, the detail of three "hop count based" schemes described.

i.

- Compute the number of hops from the source to the destinations
- Choose the node that its downlinks' traffic is maximum to adjust in every hop's iteration
- Consider the following possible criteria and set the best one to be the final tree (connect the others' node having the same hops、connect the source node、 connect the nodes having the hop counts is larger or smaller by 1)

ii.

- Compute the number of hops from the source to the destinations
- Choose the node that its out-degree is maximum to adjust in every hop's iteration
- Consider the following possible criteria and set the best one to be the final tree (connect the others' node having the same hops、connect the source node、 connect the nodes having the hop counts is larger or smaller than 1)

iii.

- Compute the number of hops from the source to the destinations
- Choose the node that its incoming traffic divides its own traffic demand is maximum to adjust in every hop's iteration
- Consider the following possible criteria and set the best one to be the final tree (connect the others' node having the same hops、connect the source node、 connect the nodes having the hop counts is larger or smaller than 1)

4.2.4 Drop Heuristic

Each group constructs its multicast tree, but there is no guarantee that link capacity constraint is not being violated. Consequently, we check the traffic flow of each links. If the capacity constraint is satisfied, we get a feasible solution. Otherwise, we will drop some admitted groups by some specific order we define. The steps of drop heuristics are as fallow:

1. Sort on all user groups by the subgradient $-F_g(a_g, f_g) - \varepsilon_g f_g$ used in (Subproblem 2.3) to represent the order.

2. Pick the largest value one, and remove it from network.

3. Check all links. If the capacity constraint is satisfied, stop the drop action.

   Otherwise, repeat step 1 and 2.

4.2.5 Add Heuristic

After drop heuristics, we get a feasible solution since the traffic of violated links are being re-arranged through some user groups were dropped. Nonetheless, those groups need another mechanism for being admitted and improve the total revenue of network. The steps of add heuristics are as follow:

1. Sort on all user groups by the subgradient $-F_g(a_g, f_g) - \varepsilon_g f_g$ used in (Subproblem

2.3) to represent the order.

2. Pick the smallest value, and add one destination at a time from the corresponding group.

3. Check all links. If the capacity constraint is satisfied, repeat step 1 and 2. Otherwise, stop the add heuristic.

Chapter 5 Computational Experiments

For the purposes of proving our heuristics and showing the difference between the results of our Lagrangean relaxation method and other primal heuristics, we implement two simple algorithms to compare with our heuristics.

5.1 Simple Algorithm of Minimum Cost Tree Model

Step 1.	Let each link's arc weight equal to (1 / the sum of its connected nodes' traffic demand).
Step 2.	According to the link set metrics, every user group constructs its multicast tree by the proposed method (modified T-M heuristic + adjustment procedures).

Table 5-1: Simple Algorithm of model I

5.2 Simple Algorithm of Partial Admission Control Model

Step 1.	Let each link's arc weight equal to (1 / the sum of its connected nodes' traffic demand).
Step 2.	According to the link set metrics, every user group constructs its multicast tree by proposed method (modified T-M heuristic+ adjustment procedures).
Step 3.	Check the capacity constraint. If all links are satisfied, stop it. Otherwise, drop the user group as the sequence in their group ID.
Step 4.	According to the sequence of group ID and destination ID, add destinations until all links' traffic flow could not increase and reconstructs its corresponding multicast tree.

Table 5-2: Simple Algorithm of model II

5.3 Assumptions, Parameters, and Cases

Number of Nodes	9~26
Cost Unit	5
Number of Iteration	2000
Maximum Unimprovement Counter	50
Begin to Tune	100
Initial Upper Bound	Cost of transmitting maximum demand
Initial Scalar of Step Size	2
Test Platform	Windows XP, 1.6 Hz CPU, 512M RAM

Table 5-3: The testing parameters for model I

Number of Nodes	9~26
Cost Unit	5
Number of Iteration	2000
Maximum Unimprovement Counter	20
Begin to Tune	50
Initial Upper Bound	0
Initial Scalar of Step Size	2
Test Platform	Windows XP, 1.6 Hz CPU, 512M RAM

Table 5-4: The testing parameters for model II

These models and algorithms were coded in Python and run on a Pentium 4 1.6 G PC with 512 MB RAM. The maximum number of iteration was set to 2000 iterations, but it is flexible to reduce the number of iterations in program for some special cases. In our implementation, $Z^h_{IP1.1}$ was initial chosen as the maximum cost when all links transmitted the maximum traffic demand and $Z^h_{IP2.1}$ was initial chosen as 0, which means the worst case of rejecting all user groups. For the two models, the choice of the initial values of the multipliers was 0.

We have tested the algorithms on four networks – Mesh, GTE, NSF, and OCT, with 9, 12, 14, 26 nodes. These topologies are shown in Figure 5.1, 5.2, 5.3, and 5.4. For each test network, several distinct cases are considered which have different pre-determined capacity of links and traffic requirement of users. The traffic demand for each user group is drawn from a random variable uniformly distributed in a pre-specified range, which is shown in the second column of Table 5-21 through Table 5-28. The third column specifies the capacity of each link. The fourth column shows the number of new user groups. The fifth column is the number of users admitted after applying our model. The sixth column gives the computed value by the simple algorithm. The seventh column gives the best objective function value calculated for (IP2.1) by the proposed heuristics. The eighth column is the smallest upper bound on $Z_{IP2.1}$ calculated by solving (D2.1). The ninth column presents the error difference: [(Upper Bound – Lower Bound) / Lower bound].

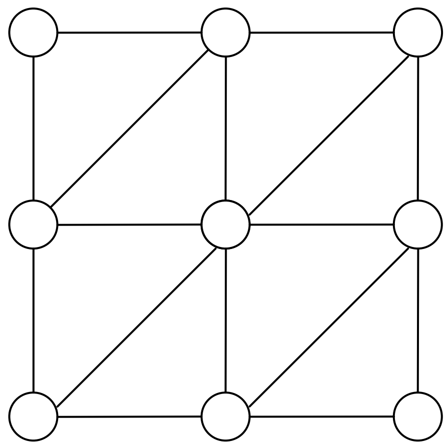

Figure 5-1: 9-node 16-link Mesh network

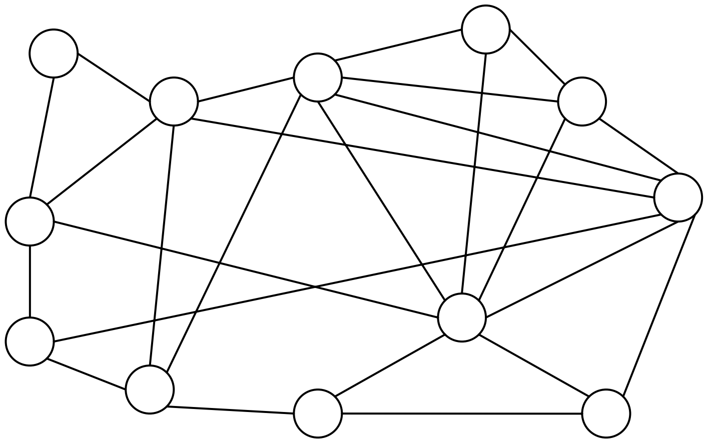

Figure 5-2: 12-node 25-link GTE network

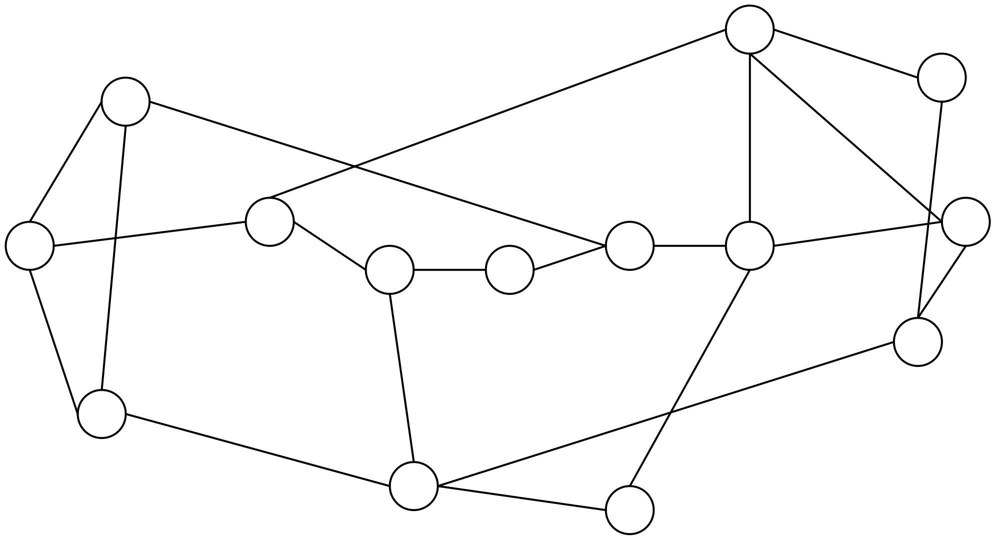

Figure 5-3: 14-nodes 21-links NSF Network

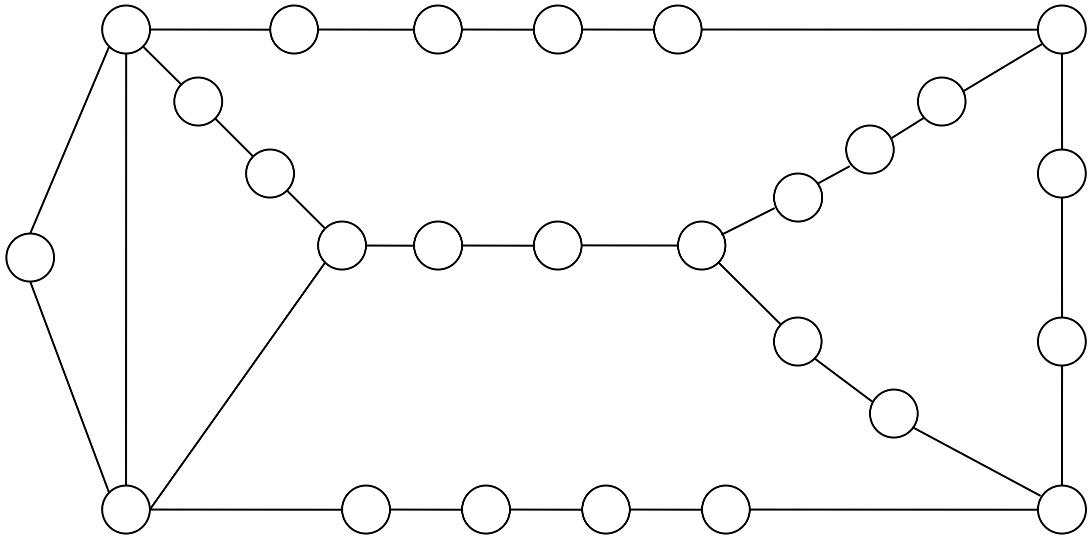

Figure 5-4: 26-node 30-link OCT network

5.4 Heuristic Comparisons for Getting Primal Feasible Solutions

Before expressing the implementation of the experiments, there are 3 heuristics about the adjustment procedures and 3 heuristics about the arc weight choices for each model to determine the multicast routing, and it's only necessary to find the best one to do the experiments. Therefore, we run four cases by these heuristics, and choose the heuristic, which have the best result.

5.4.1 Experiment Result of Model I

Case 1:

Mesh Network (9 nodes, 16 links)

Number of Requested Multicast Group: 5

Range of Requested Bandwidth: 1 ~ 5 (Mbps)

Number of Iterations: 2000

Result:

	UB	LB	Gap
Modified T-M	110.00	79.3519	38.62%
Modified T-M+Beta	91.00	79.3519	14.68%
Modified T-M+Alpha	89.00	79.3519	12.16%
Modified T-M+Theta	99.00	79.3519	24.76%

Table 5-5: The experiment results of model I with arc weight choices and without arc weight choices on Mesh network

	UB	LB	Gap
Modified T-M+Alpha+AP1	82.00	79.3519	3.34%
Modified T-M+Alpha+AP2	85.00	79.3519	7.12%
Modified T-M+Alpha+AP3	81.00	79.3519	2.08%

Table 5-6: The experiment results of model I with the best arc weight choice and adjustment procedures on Mesh network

Case 2:

GTE Network (12 nodes, 25 links)

Number of Requested Multicast Group: 10

Range of Requested Bandwidth: 10 ~ 20(Mbps)

Number of Iterations: 2000

Result:

	UB	LB	Gap
Modified T-M	6569.00	4133.085	58.94%
Modified T-M+Beta	5091.00	4133.085	23.18%
Modified T-M+Alpha	5006.00	4133.085	21.12%
Modified T-M+Theta	5408.00	4133.085	30.85%

Table 5-7: The experiment results of model I with arc weight choices and without arc weight choices on GTE network

	UB	LB	Gap
Modified T-M+Alpha+AP1	4391.00	4133.085	6.24%
Modified T-M+Alpha+AP2	4708.00	4133.085	13.91%
Modified T-M+Alpha+AP3	4306.00	4133.085	4.18%

Table 5-8: The experiment results of model I with the best arc weight choice and adjustment procedures on GTE network

<u>Case 3:</u>

NSF Network (14 nodes, 21 links)

Number of Requested Multicast Group: 10

Range of Requested Bandwidth: 1 ~ 20(Mbps)

Number of Iterations: 2000

Result:

	UB	LB	Gap
Modified T-M	1809.00	1106.778	63.45%
Modified T-M+Beta	1412.00	1106.778	27.58%
Modified T-M+Alpha	1386.00	1106.778	25.23%
Modified T-M+Theta	1588.00	1106.778	43.48%

Table 5-9: The experiment results of model I with arc weight choices and without arc weight choices on NSF network

	UB	LB	Gap
Modified T-M+Alpha+AP1	1169.00	1106.778	5.62%
Modified T-M+Alpha+AP2	1255.00	1106.778	13.39%
Modified T-M+Alpha+AP3	1142.00	1106.778	3.18%

Table 5-10: The experiment results of model I with the best arc weight choice and

adjustment procedures on NSF network

<u>Case 4:</u>
OCT Network (26 nodes, 30 links)
Number of Requested Multicast Group: 15
Range of Requested Bandwidth: 5 ~ 20(Mbps)
Number of Iterations: 2000
Result:

	UB	LB	Gap
Modified T-M	42116.00	19220.531	119.12%
Modified T-M+Beta	27503.00	19220.531	43.09%
Modified T-M+Alpha	26441.00	19220.531	37.57%
Modified T-M+Theta	33545.00	19220.531	74.53%

Table 5-11: The experiment results of model I with arc weight choices and without arc weight choices on OCT network

	UB	LB	Gap
Modified T-M+Alpha+AP1	20911.00	19220.531	8.80%
Modified T-M+Alpha+AP2	24081.00	19220.531	25.29%
Modified T-M+Alpha+AP3	20737.00	19220.531	7.89%

Table 5-12: The experiment results of model I with the best arc weight choice and adjustment procedures on OCT network

From the above tests for model I, it can be obviously that using the Lagrangean multiplier as the link arc weight is better than without adapting it. In addition, after the link arc weight to be determined as $\sum_{d \in D_g} \beta_{gdl} \times \alpha_{gd}$, integrating the three types of adjustment procedures have different results. We find the adjustment procedure 3 will bring the best performance. As a result, applying adjustment procedure 3 and letting link arc weight equal to $\sum_{d \in D_g} \beta_{gdl} \times \alpha_{gd}$ become the best arrangement for model I.

Furthermore, the ratio of the reduced amount by modified T-M heuristic plus arc weight choices divided the original value by pure modified T-M heuristic is from 0.19 to 0.37. And the ratio of the reduced amount by Lagrangean based modified T-M heuristic plus adjustment procedures divided the original value by pure Lagrangean based modified T-M heuristic is from 0.09 to 0.22. Therefore, while using modified T-M heuristic, considering corresponding link arc weight and adjustment procedure is equally necessary and important.

5.4.2 Experiment Result of Model II

Case 1:

Mesh Network (9 nodes, 16 links)

Number of Requested Multicast Group: 5

Range of Requested Bandwidth: 1 ~ 10 (Mbps)

Range of link capacity: 30

Number of Iterations: 2000

Result:

	UB	LB	Gap
Modified T-M	-305.00	-395.0291	29.52%
Modified T-M+Beta	-329.00	-395.0291	20.07%
Modified T-M+Alpha	-326.00	-395.0291	21.17%
Modified T-M+Theta	-336.00	-395.0291	17.57%

Table 5-13: The experiment results of model II with arc weight choices and without arc weight choices on Mesh network

	UB	LB	Gap

Modified T-M+Theta+AP1	-383.00	-395.0291	3.14%
Modified T-M+Theta+AP2	-358.00	-395.0291	10.34%
Modified T-M+Theta+AP3	-376.00	-395.0291	5.06%

Table 5-14: The experiment results of model II with the best arc weight choice and adjustment procedures on Mesh network

Case 2:

GTE Network (12 nodes, 25 links)

Number of Requested Multicast Group: 10

Range of Requested Bandwidth: 1 ~ 10(Mbps)

Range of link capacity: 30

Number of Iterations: 2000

Result:

	UB	LB	Gap
Modified T-M	-3167.00	-4709.110	48.69%
Modified T-M+Beta	-3589.00	-4709.110	31.21%
Modified T-M+Alpha	-3402.00	-4709.110	38.42%
Modified T-M+Theta	-3855.00	-4709.110	22.16%

Table 5-15: The experiment results of model II with arc weight choices and without arc weight choices on GTE network

	UB	LB	Gap
Modified T-M+Theta+AP1	-4502.00	-4709.110	4.60%
Modified T-M+Theta+AP2	-4093.00	-4709.110	15.05%
Modified T-M+Theta+AP3	-4352.00	-4709.110	8.21%

Table 5-16: The experiment results of model II with the best arc weight choice and adjustment procedures on GTE network

Case 3:

NSF Network (14 nodes, 21 links)
Number of Requested Multicast Group: 10
Range of Requested Bandwidth: 1 ~ 20(Mbps)
Range of link capacity: 30
Number of Iterations: 2000
Result:

	UB	LB	Gap
Modified T-M	-3913.00	-6012.301	53.65%
Modified T-M+Beta	-4638.00	-6012.301	29.63%
Modified T-M+Alpha	-4517.00	-6012.301	33.10%
Modified T-M+Theta	-4909.00	-6012.301	22.48%

Table 5-17: The experiment results of model II with arc weight choices and without arc weight choices on NSF network

	UB	LB	Gap
Modified T-M+Theta+AP1	-5703.00	-6012.301	5.42%
Modified T-M+Theta+AP2	-5036.00	-6012.301	19.39%
Modified T-M+Theta+AP3	-5508.00	-6012.301	9.16%

Table 5-18: The experiment results of model II with the best arc weight choice and adjustment procedures on NSF network

<u>Case 4:</u>
OCT Network (26 nodes, 30 links)
Number of Requested Multicast Group: 15
Range of Requested Bandwidth: 1 ~ 20(Mbps)
Range of link capacity: 30
Number of Iterations: 2000
Result:

	UB	LB	Gap
Modified T-M	-12805.00	-22079.108	72.43%
Modified T-M+Beta	-15479.00	-22079.108	42.64%
Modified T-M+Alpha	-15283.00	-22079.108	44.47%

Modified T-M+Theta	-16512.00	-22079.108	33.72%

Table 5-19: The experiment results of model II with arc weight choices and without arc weight choices on OCT network

	UB	LB	Gap
Modified T-M+Theta+AP1	-20193.00	-22079.108	9.34%
Modified T-M+Theta+AP2	-16918.00	-22079.108	30.51%
Modified T-M+Theta+AP3	-18772.00	-22079.108	17.62%

Table 5-20: The experiment results of model II with the best arc weight choice and adjustment procedures on OCT network

From the above tests for model II, it can be obviously that using the Lagrangean multiplier as the link arc weight is better than without adapting it. In addition, after the link arc weight to be determined as θ_{gl}, integrating the three types of adjustment procedures have different results. We find the adjustment procedure 1 will bring the best performance. As a result, applying adjustment procedure 1 and letting link arc weight equal to θ_{gl} become the best arrangement for model II.

Furthermore, the ratio of the reduced amount by modified T-M heuristic plus arc weight choices divided the original value by pure modified T-M heuristic is from 0.1 to 0.29. And the ratio of the reduced amount by Lagrangean based modified T-M heuristic plus adjustment procedures divided the original value by pure Lagrangean based modified T-M heuristic is from 0.14 to 0.23. Therefore, while using modified

T-M heuristic, considering corresponding link arc weight and adjustment procedure is equally necessary and important.

5.5 Experiment Results

5.5.1 Experiment Result of Minimum Cost Tree Model

Case No.	Traffic Range	Link Capacity	No. of Requested Groups	SA	Upper Bound	Lower Bound	Gap
1	0.5~1.0	40	5	201	189	189	0.00%
2	1.0~2.0	40	5	572	517	510	1.37%
3	0.5~1.0	40	10	368	331	330	0.30%
4	1.0~2.0	40	10	846	750	727	3.16%
5	0.5~1.0	40	15	693	601	596	0.84%
6	1.0~2.0	40	15	1206	1032	985	4.77%
7	0.5~1.0	40	20	1044	883	868	1.73%
8	1.0~2.0	40	20	1863	1539	1456	5.70%

Table 5-21: Summary of computational results of Minimum Cost Tree Model by using Lagrangean Relaxation method on Mesh network

Case No.	Traffic Range	Link Capacity	No. of Requested Groups	SA	Upper Bound	Lower Bound	Gap
1	0.5~1.0	40	5	273	249	247	0.81%
2	1.0~2.0	40	5	624	553	534	3.56%
3	0.5~1.0	40	10	521	471	466	1.07%
4	1.0~2.0	40	10	1577	1400	1316	6.38%
5	0.5~1.0	40	15	1031	887	865	2.54%
6	1.0~2.0	40	15	2492	2124	1929	10.11%
7	0.5~1.0	40	20	1807	1506	1443	4.37%
8	1.0~2.0	40	20	3842	3177	2735	16.16%

Table 5-22: Summary of computational results of Minimum Cost Tree Model by using Lagrangean Relaxation method on GTE network

Case No.	**Traffic Range**	**Link Capacity**	**No. of Requested Groups**	**SA**	**Upper Bound**	**Lower Bound**	**Gap**
1	0.5~1.0	40	5	359	322	319	0.94%
2	1.0~2.0	40	5	742	657	626	4.95%
3	0.5~1.0	40	10	683	586	577	1.56%
4	1.0~2.0	40	10	1448	1265	1174	7.75%
5	0.5~1.0	40	15	937	794	772	2.85%
6	1.0~2.0	40	15	2214	1855	1660	11.77%
7	0.5~1.0	40	20	1428	1167	1121	4.10%
8	1.0~2.0	40	20	2964	2450	2074	18.13%

Table 5-23: Summary of computational results of Minimum Cost Tree Model by using Lagrangean Relaxation method on NSF network

Case No.	Traffic Range	Link Capacity	No. of Requested Groups	SA	Upper Bound	Lower Bound	Gap
1	0.5~1.0	40	5	1339	1124	1105	1.72%
2	1.0~2.0	40	5	4762	3883	3601	7.83%
3	0.5~1.0	40	10	2897	2416	2353	2.68%
4	1.0~2.0	40	10	10527	8610	7622	12.96%
5	0.5~1.0	40	15	8604	7047	6737	4.60%
6	1.0~2.0	40	15	15347	12644	10384	21.76%
7	0.5~1.0	40	20	13677	10906	10187	7.06%
8	1.0~2.0	40	20	26958	21772	16508	31.89%

Table 5-24: Summary of computational results of Minimum Cost Tree Model by using Lagrangean Relaxation method on OCT network

From the computational results, it is observed that excellent results can be obtained by the minimum cost tree model for the Mesh, GTE, NSF, and OCT network. For the four tested network, the average error difference are respectively 2.23%, 5.63%, 6.51%, and 11.31%, which means, the solutions of using the Lagrangean Relaxation method are near-optimal.

Besides the above effect, our algorithm performs better than the simple algorithm heuristic for minimum cost tree model. For the test networks, our algorithm achieves up to 12.19% to 18.07% (average 14.42%) improvement in the total cost over the simple algorithm heuristic.

The reason that LR works better than SA is that LR has multipliers to provide hints about the extent of constraint violating. Moreover, it can help to make decision variables more effective and accurate. SA's hints are rare, since it only uses the traffic demand of nodes to reflect the link's importance. Furthermore, LR can improve the result iteration by iteration; since the result of SA is the same no matter how many iterations it runs.

5.5.2 Experiment Result of Partial Admission control Model

Case No.	Traffic Range	Link Capacity	No. of Requested Groups	No. of Admitted Groups	SA	Upper Bound	Lower Bound	Gap
1	1.0~2.0	10	5	5	-535	-562	-566	0.71%
2	2.0~4.0	10	5	5	-725	-764	-783	2.49%
3	1.0~2.0	10	10	10	-912	-991	-1004	1.31%
4	2.0~4.0	10	10	$8\frac{1}{5}$	-1378	-1479	-1570	6.15%
5	1.0~2.0	10	15	$14\frac{1}{3}$	-1757	-1915	-1954	2.04%
6	2.0~4.0	10	15	$12\frac{5}{7}$	-2244	-2452	-2725	11.13%
7	1.0~2.0	10	20	$15\frac{1}{4}$	-2665	-2910	-3013	3.54%
8	2.0~4.0	10	20	$13\frac{5}{8}$	-3073	-3511	-4187	19.25%

Table 5-25: Summary of computational results of Partial Admission Control Model by using Lagrangean Relaxation method on Mesh network

Case No.	Traffic Range	Link Capacity	No. of Requested Groups	No. of Admitted Groups	SA	Upper Bound	Lower Bound	Gap
1	1.0~2.0	10	5	5	-779	-822	-833	1.34%
2	2.0~4.0	10	5	5	-1285	-1368	-1449	5.92%
3	1.0~2.0	10	10	$8\frac{3}{4}$	-1238	-1352	-1380	2.06%
4	2.0~4.0	10	10	$8\frac{2}{7}$	-1880	-2108	-2433	15.42%
5	1.0~2.0	10	15	$14\frac{3}{9}$	-1788	-2005	-2077	3.59%
6	2.0~4.0	10	15	$13\frac{1}{5}$	-2239	-2553	-3266	27.93%
7	1.0~2.0	10	20	$15\frac{3}{4}$	-2378	-2694	-2858	6.09%
8	2.0~4.0	10	20	$14\frac{5}{6}$	-2381	-2796	-3997	42.95%

Table 5-26: Summary of computational results of Partial Admission Control Model by using Lagrangean Relaxation method on GTE network

Case No.	**Traffic Range**	**Link Capacity**	**No. of Requested Groups**	**No. of Admitted Groups**	**SA**	**Upper Bound**	**Lower Bound**	**Gap**
1	1.0~2.0	10	5	5	-838	-898	-909	1.22%
2	2.0~4.0	10	5	5	-1385	-1507	-1608	6.70%
3	1.0~2.0	10	10	$9\frac{1}{6}$	-1361	-1503	1547	2.93%
4	2.0~4.0	10	10	$7\frac{2}{4}$	-2015	-2291	-2602	13.57%
5	1.0~2.0	10	15	$12\frac{2}{5}$	-2059	-2296	-2422	5.49%
6	2.0~4.0	10	15	$10\frac{3}{6}$	-2481	-2893	-3732	29.00%
7	1.0~2.0	10	20	$13\frac{2}{8}$	-2479	-2815	-3057	8.60%
8	2.0~4.0	10	20	$11\frac{1}{3}$	-2902	-3518	-5188	47.47%

Table 5-27: Summary of computational results of Partial Admission Control Model by using Lagrangean Relaxation method on NSF network

Case No.	Traffic Range	Link Capacity	No. of Requested Groups	No. of Admitted Groups	SA	Upper Bound	Lower Bound	Gap
1	1.0~2.0	10	5	5	-1752	-1932	-1980	2.48%
2	2.0~4.0	10	5	$4\frac{1}{8}$	-2787	-3138	-3652	16.38%
3	1.0~2.0	10	10	$7\frac{2}{7}$	-2530	-2840	-3041	7.08%
4	2.0~4.0	10	10	$6\frac{3}{8}$	-4488	-5104	-6220	21.81%
5	1.0~2.0	10	15	$11\frac{3}{6}$	-4296	-4881	-5473	12.13%
6	2.0~4.0	10	15	$11\frac{2}{3}$	-7743	-9207	-12836	39.42%
7	1.0~2.0	10	20	$13\frac{2}{9}$	-5896	-6836	-7906	15.65%
8	2.0~4.0	10	20	$12\frac{4}{6}$	-12200	-15527	-25081	61.53%

Table 5-28: Summary of computational results of Partial Admission Control Model by using Lagrangean Relaxation method on OCT network

From the computational results, it is observed that excellent results can be obtained by the partial admission control model for the Mesh, GTE, NSF, and OCT network. For the four tested network, the average error difference are respectively 5.83%, 13.16%, 14.37%, and 22.06%, which means, the solutions of using the Lagrangean Relaxation method are near-optimal.

Besides the above effect, our algorithm performs better than the simple algorithm heuristic for partial admission control model. For the test networks, our algorithm achieves up to 8.51% to 15.57% (average 12.06%) improvement in the total revenue over the simple algorithm heuristic.

The reason that LR works better than SA is that LR has multipliers to provide hints about the extent of constraint violating. Moreover, it can help to make decision variables more effective and accurate. SA's hints are rare, since it only uses the traffic demand of nodes to reflect the link's importance and runs add/drop heuristics by group ID numbers. Furthermore, LR can improve the result iteration by iteration; since the result of SA is the same no matter how many iterations it runs.

Chapter 6 Conclusion

6.1 Summary

In this thesis, we attempt to solve the problem of supporting efficient and flexible mechanisms for multimedia distribution on multicast networks. Under multimedia application environments, it is characterized by large bandwidth variations due to heterogeneous access-technologies of the networks (e.g., analog modem, cable modem, xDSL, etc.) and receivers (e.g., high resolution, low resolution). Because of recent advances in switching and transmission technologies, either by a progressive coder, or video gateway, destinations can request different bandwidth requirement from the source. The source transmits only signal that is sufficient for the highest bandwidth destination. The minimum cost tree of multicast service is calculated by this property of transmission mechanisms to achieve the efficiency. Furthermore, we also consider partial admission control mechanism. For network operators, the function of partial admission control is to determine whether a user service request can be granted such that the requested bandwidth of the new user can be satisfied and the total revenue can be maximized under limited resources.

Consequently, we deal with the problems as mathematical models. The first focuses on the minimizing total transmission cost, while the second focuses on maximizing the total system revenues and satisfy capacity constraint and multicast tree constraints. The basic approach to the algorithm is Lagrangean Relaxation and the subgradient method. We develop the Lagrangean based modified T-M heuristic and several adjustment procedures to get primal feasible solutions. From computational experiments, the proposed algorithm determines solutions that are within a few percent of an optimal solution with 9-26 nodes both in the minimum cost tree model and partial admission control model (error difference in the minimum cost tree model is 6.42% on average, error difference in the partial admission control model is 13.86% on average). In terms of performance, our Lagrangean Relaxation based solution has more significant improvement than simple heuristics. The improvement on the total cost can reach 14.42% on the average in the minimum cost tree model, and 12.06% on the average in the partial admission control model.

6.2 Future Work

At first, in model II-- Partial Admission Control Model-- the destination's traffic demand in one multicast user group is set the same, but every destination may have different requirements. Therefore, it could be modified to adapt for supporting multimedia services that is characteristic of divergence, because destinations of a group may vary significantly according to their interests. In other words, the concept of priority would be introduced. This concept can be extended from our model II.

Secondly, adjustment procedures play an important role of the entire getting primal feasible solution process in this paper. To break an even point between performance and cost, we proposed the hop count based heuristic to adjust one node in every iteration. But, it lacks of broader rules to understand all nodes' condition in the network. In the future, we can consider alternative methods for related algorithms development.

Finally, throughout this thesis, the multicast environment we are concerned about is quasi-dynamic. Thus, how to take into account the real-time membership action including join and leave is more complex. It is, of course, also an interesting and meaningful research issue.

References

[1] 宋之揆，*服務品質保證限制下群播開放最短路徑優先路由演算法之研究*，國立台灣大學資訊管理學研究所碩士學位論文，民國九十一年六月。

[2] 葉慧嫻，*高速網路之綜合性最佳化允入控制路由指定及資源預留策略以確保即時性群撥服務之服務品質要求*，國立台灣大學資訊管理學研究所碩士學位論文，民國八十八年六月。

[3] 顏宏旭，*電腦網路與後勤網路之規劃與容量管理*，國立台灣大學資訊管理學研究所博士學位論文，民國九十年六月。

[4] Ahuja, R. K., Magnanti, T. L., and Orlin, J. B., *Network Flows—Theory, Algorithms, and Applications*, Prentice Hall, 1993.

[5] Amir, E., McCanne, S., and Zhang, H., “An application level video gateway,” *Proc. of the third ACM international conference on Multimedia*, pp. 255-265, 1995.

[6] Casetti, C., Kurose, J., and Towsley, D., “An Adaptive Algorithm for Measurement-based Admission Control in Integrated Services Packet Networks,” *Proc. of the Protocols for High Speed Networks Workshop*, October 1996.

[7] Cetinkaya, C. and Knightly E. W. “Egress Admission Control,” *Proc. of INFOCOM 2000*, Vol. 3, pp. 1471-1480, March 2000.

[8] Charikar, M., Naor, J., and Schieber, B., “Resource optimization in QoS multicast routing of real-time multimedia,” *Proc. of INFOCOM 2000*, Vol. 3, pp. 1518-1527, March 2000.

[9] Chen, S. and Nahrstedt, K., "An overview of Quality of Service routing for next-generation high-speed networks: problems and solutions," *IEEE Network*, Vol. 12, no. 6, pp. 64-79, Nov./Dec. 1998.

[10] Cheng, K. T. and Lin, F. Y. -S. "Minimax end-to-end delay routing and capacity assignment for virtual circuit networks," *IEEE Proc. of GLOBECOM '95*, Vol. 3, pp.2134-2138, 1995.

[11] Diot, C., Dabbous, W., and Crowcroft, J., "Multipoint communication: a survey of protocols, functions, and mechanisms," *IEEE Journal on Selected Areas on Communications*, Vol.15, no.3, pp. 277-290, April 1997.

[12] Eppstein David, "Spanning trees and Spanners," *Tech. Report 96-16 in Handbook of Computational Geometry*, pp. 425-461, Elsevier, 1999.

[13] Ergun, F., Sinha, R., and Zhang, L. "QoS Routing with Performance-Dependent Costs," *Proc. of INFOCOM 2000*, Vol. 1, pp. 137-146, March 2000.

[14] Fang, H. and Zegura, E. W., "On Scalable QoS Routing: Performance Evaluation of Topology Aggregation," *Proc. of INFOCOM 2000*, Vol. 1, pp. 147-156, March 2000.

[15] Firoiu, V. and Towsley, D., "Call Admission and Resource Reservation for Multicast Sessions," *Proc. of INFOCOM '96*, Vol.1, pp. 94-101, April 1996.

[16] Fisher, M. L., "The Lagrangian relaxation method for solving integer programming problems," *Management Science*, vol. 27, pp.1-18, 1981.

[17] Garey, M. R. and Johnson, D. S., "Computers and Intractability—a Guide to the Theory of NP-Completeness," Freeman: New York, 1979.

[18] Geoffrion, A. M., "Lagrangian relaxation and its uses in integer programming," *Math. Programming Study*, Vol. 2, pp. 82-114, 1974.

[19] Apostolopoulos, G.., Guérin, R., Kamat, S., and Tripathi, S. "Quality of service based routing: a performance perspective," *ACM SIGCOMM Computer Communication Review*, Vol.28, no.4, pp.17-28, Oct. 1998.

[20] Held, M. and Karp, R.M. "The Traveling salesman problem and minimum spanning trees: Part I," *Operations Research*, Vol. 18, pp. 1138-1162, 1970.

[21] Hill, R. and Kung, H.T., "A Diff-Serv enhanced admission control scheme," *Proc. of GLOBECOM '01*, Vol. 4, pp. 2549 -2555, 2001.

[22] Alvarez-Hamelin, J. I. and Fraigniaud, P., "A survey of multicast protocols for Internet," *Proc. of Argentine Symposium on Computing Technology*, 2001.

[23] Irit .K, Peter .S, and Jesper .T.L, "a practical minimum spanning tree algorithm using the cycle property", *Tech. Report*, October 2002.

[24] Jia, X., Zhang, Y., Pissinoun, N., and Makkis, S., "An Integrated Routing and Admission Control Mechanism for Real-time Multicast Connection Establishment," *International Journal of Communication Systems*, vol. 14, no.3, pp 287-303, April 2001.

[25] Karlsson, G. and Vetterli, M., "Packet video and its integration into the network architecture," *IEEE Journal on Selected Areas in Communications*, Vol. 7, pp. 739–751, 1989.

[26] Lai, K. and Baker, M., "Measuring Link Bandwidths Using a Deterministic Model of Packet Delay," *Proc. of SIGCOMM'00*, pp 283–294, 2000.

[27] Lin, Y. -S. and Yee, J. R., “Three Algorithms for Routing and Flow Control in Virtual Circuit Networks,” *IEEE Proc. of GLOBECOM ‘90,* Vol.1, pp.339-343, 1990.

[28] Legout, A., Nonnenmacher, J., and Biersack, E.W. “Bandwidth allocation policies for unicast and multicast flows,” *IEEE/ACM Transaction on Networking*, Vol. 9, no. 4, pp.464-478, August 2001.

[29] Maxemchuk, N. F. “Video Distribution on Multicast Networks,” *IEEE JSAC*, Vol. 15, no. 3, pp. 357-372, April 1997.

[30] Merabti, M. and Taylor, A., “A Review of Multimedia Networking,“ *HEFECE JISC Committee Report*, January 1995.

[31] Mrak, M., Sprljan, N., and Grgic, S., “Video Coding Schemes for Transporting Video over the Internet,” *18th International Symposium on Information and Communication Technologies*, 2001.

[32] Pagani, E. and Rossi, G. P., “Distributed Bandwidth Broker for QoS Multicast Traffic,” *Proceedings of 22nd IEEE International Conference on Distributed Computing Systems (ICDCS 2002)*, pp. 319-326, July 2002.

[33] Perros, H. G.. and Elsayed, K. M., “Call admission control schemes: A review,” *IEEE Communications Magazine*, pp. 82-91, Nov. 1996.

[34] Pettie, S. and Ramachandran, V., “An optimal minimum spanning tree algorithm,” *J. ACM*, Vol. 49, no.1, pp. 16-34, Jan. 2002.

[35] Przytycka, T, and Higham, L., “Optimal Cost Sensitive Distributed Minimum Spanning Tree Algorithm,” *Fifth Scandinavian Workshop on Algorithm Theory*, Reykjavik, Iceland, July, 1996.

[36] Roca, V., Costa, L., Vida, R., Dracinschi, A., and Fdida, S., "A Survey of Multicast Technologies," *Technical Report*, September 2000.

[37] Salama, H. F., Reeves, D. S., and Viniotis, Y., "The delay-constrained minimum spanning tree problem," *Proceedings Second IEEE Symposium on Computer and Communications*, pp.699-703, July 1997.

[38] H. Takahashi, A. Matsuyama, "An Approximate Solution for the Steiner Problem in Graphs," *Math. Japonica*, 1980, vol. 6, pp. 573-577.

[39] Tang N., Tsui, S., and Wang, L., "A survey of admission control algorithms," *Technical Report*, December 1998.

[40] Turletti, T. and Bolot, J. C., "Issues with multicast video distribution in heterogeneous packet networks," *Proceedings of the 6th International workshop on packet video*, 1994.

[41] Wang, Z. and Crowcroft, J., "Quality-of-service routing for supporting multimedia applications," *IEEE Journal on Selected Areas in Communication*, Vol. 14, pp. 1228-1234, Sept. 1996.

[42] Wu, D., Hou, T., and Zhang, Y. Q., "Transporting Real-time Video over the Internet: Challenges and Approaches," *Proceedings of the IEEE*, vol. 88, no. 12, pp. 1855 -1875, Dec. 2000.

[43] Wu, D., Hou, Y. T., Zhu, W., Zhang, Y., and Peha, J., "Streaming video over the Internet: approaches and directions," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 11, no. 3, pp. 282-300, March. 2001.

[44] Zhang, L., Deering, S., Estrin, D., Shenker, S., and Zappala, "RSVP: A new resource reservation protocol," *IEEE Network*, Vol. 7, pp.8–18, 1993.