MODELING AND ANALYSIS OF ISSUES IN HUB LOCATION PROBLEM

A THESIS
SUBMITTED TO THE DEPARTMENT OF
INDUSTRIAL ENGINEERING
AND THE INSTITUTE OF ENGINEERING AND SCIENCE
OF BILKENT UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

By
Bahar Yetiş Kara
September, 1999

THESIS
HB
199
·K37
1999

MODELING AND ANALYSIS OF ISSUES IN HUB LOCATION PROBLEM

A THESIS
SUBMITTED TO THE DEPARTMENT OF
INDUSTRIAL ENGINEERING
AND THE INSTITUTE OF ENGINEERING AND SCIENCE
OF BILKENT UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

By
Bahar Yetiş Kara
September, 1999

HB
199
.K37
1999
B049438

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of Doctor of Philisophy.

Assoc. Prof. Barbaros Ç. Tansel(Supervisor)

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of Doctor of Philisophy.

Asst. Prof. Emre Berk

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of Doctor of Philisophy.

Assist. Prof. Oya Karaşan

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of Doctor of Philisophy.

Assoc. Prof. Osman Oĝuz

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of Doctor of Philisophy.

Assoc. Prof. Canan Sepil

Approved for the Institute of Engineering and Sciences:

Prof. Mehmet Baray
Director of Institute of Engineering and Sciences

ABSTRACT

MODELING AND ANALYSIS OF ISSUES IN HUB LOCATION PROBLEM

Bahar Yetiş Kara
Ph.D. in Industrial Engineering
Supervisor: Assoc. Prof. Barbaros Ç. Tansel
September, 1999

The hub location problem has been around for more than 10 years. The first mathematical model was formulated by O'Kelly (1986) which is a quadratic integer program. Since then, nearly all of the researchers in this area have concentrated on developing 'good' linearizations. However, there are many aspects of the problem that need to be analyzed. In this dissertation, we investigate some of these issues. We first study the application areas of the hub location problem and clarify the underlying assumptions of the real world problems which lead to the customarily defined hub location problem. We identify a certain problem characteristic of cargo delivery systems, which is one of the major application areas of the hub location problem, which is not satisfactorily modeled by means of the customarily defined hub location models. We propose a new hub location model that captures the specific requirements that are particular to cargo delivery systems. Another issue that we concentrate on is the identification, modeling and analysis of the hub location problem under different performance measures, namely minimax and covering criteria. We propose new integer programming models for the hub location problem under minimax and covering objectives. Both of the new models are the result of a different way of approaching the problem and their computational performance is far more superior than the performance of the various linearizations of the basic models proposed for these problems in the literature.

Key words: Hub Location Problem, Modeling, Complexity, Linearizations

ÖZET

ANA DAĞITIM ÜSSÜ YERSEÇİMİ PROBLEMİNİN İNCELENMESİ

Bahar Yetiş Kara
Endüstri Mühendisliği Bölümü Doktora
Tez Yöneticisi: Doç. Dr. Barbaros Ç. Tansel
Eylül, 1999

Ana dağıtım üssü yer seçimi problemi 10 yıldan daha uzun bir süredir literatürde yer almaktadır. Bu konudaki ilk matematiksel model 1986 yılında O'Kelly tarafından ortaya konulan quadratik tamsayılı programlamadır.
O'Kelly'nin bu çalışmasından sonra ana dağıtım üssü yer seçimi problemi üzerinde çalışan araştırmacıların büyük bir kısmı bu temel modelin linearizasyonu üzerinde yoğunlaşmışlardır. Oysa ki ana dağıtım üssü yer seçimi probleminin incelenmesi gereken daha pek çok boyutu bulunmaktadır.. Biz bu doktora çalışmasında bu boyutların bazılarını inceledik. Öncelikle ana dağıtım üssü probleminin uygulama alanları üzerinde bir araştırma yaptık. Bu araştırma sonucunda gerçek hayattaki problemlerin literatürde tanımlanan ana dağıtım üssü modeline dönüşebilmesi için gereken varsayımları ortaya çıkardık. Bu çalışma sırasında, ana dağıtım üssü problemlerinin önemli bir uygulama alanı olan kargo dağıtım sistemlerinin önemli bir özelliğini modellemede temel ana dağıtım üssü modelinin yetersiz kaldığı bazı durumları keşfettik ve bu özelliği de modelleyen yeni bir ana dağıtım üssü modeli geliştirdik. Bu doktora çalışmasında üzerinde durduğumuz bir diğer konu da ana dağıtım üssü modelinin farklı

performans ölçütleri altında incelenmesidir. Ana dağıtım üssü problemi icin minimax ve kaplama (cover) ölçütleri için yeni modeller gelistirdik. Her iki problem için geliştirmiş olduğumuz yeni modeller ana dağıtım üssü problemlerinin farklı yaklaşımlarla incelenmesi sonucu ortaya çıkmış modeller olup, literürde bu ölçütler için geliştirilmiş olan modellerden çok daha iyi performans göstermişlerdir.

Anahtar sözcükler: Ana dağıtım üssü problemi, Modelleme, Optimizasyon, Linearizasyon teknikleri

Bana her zaman destek olan

babam İmdat Kara

ve eşim Kadri Yetiş'e

ACKNOWLEDGEMENT

I would like to express my sincere gratitute to the numerous people who have supported me in various ways during the process of this dissertation research. I am mostly grateful to my supervisor Barbaros Tansel for suggesting a research topic full of enthusiasm, and who has been supervising me with patience and everlasting interest and being helpful in any way during my studies.

I am also indebted to members of my dissertation committee: Osman Oĝuz, Emre Berk, Canan Sepil, Oya Karaşan and Levent Kandiller for showing keen interest to the subject matter and accepting to read and review this thesis. Their remarks and recommendations have been invaluable.

I wish to express my deepest gratitude Vedat Verter for his invaluable guidance and for hosting my visit to Canada, during the final stages of this research.

I would like to express my deepest thanks to my family, without whom this study would not have been possible. To my father, İmdat Kara for his valuable suggestions and encouragements, to my husband Kadri Yetiş for his love, understanding, patience and moral support and finally to my mother Günay Kara, my sister Gonca Kara and my mother-in-law Pınar Yetiş for their prays. My family was always there with an everlasting support when I need them.

Last but not least, I owe special thanks to my "best friend" Muhittin Hakan Demir. This study could have take much longer without his continuous moral support and encouragement. I will never forget our small chats and discussions.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This thesis is on modeling and analysis of issues on hub location problems. The generic hub location problem can be stated as follows: There are n demand nodes, of any kind, each of which generates and/or absorbs demands. Examples include cities with passenger or cargo flows between cities, and computers with flow of data packets/messages in between. The main problem involves determining the locations of hubs and the allocation of demand nodes to hubs so as to carry the total traffic from origins to destinations via hubs to minimize a cost function. The hubs are consolidation and dissemination centers. Flows from the same origin with different destinations are consolidated on their route to a hub facility and are combined with the flows with different origins but same destinations. Thus, hubs replace direct flows with indirect ones.

The process of consolidating and disseminating flows is referred to as 'hubbing' in the literature. Hub location is a rather new research area which started with a quadratic integer programming (IP) formulation of O'Kelly (1987). Since then, the literature is focused on developing 'good' linearizations for the O'Kelly's model and some heuristics. However, real world applications require the consideration of different phenomena which have not been covered in the original model. Thus, the literature suffers from lack of different modeling issues. In this thesis, we analyze the problem structure of real world applications to identify their requirements. The analysis of cargo delivery systems (which is an application area of hub location problems) leads to a different problem that we call the *Latest Arrival Hub Location* problem. Another issue that deserves attention is on the modeling and analysis of the hub location problem under different performance criteria, e.g. minmax.

In this thesis, we first analyze the problem structure of the application areas of hub location problem : airline systems, cargo delivery systems, and telecommunications network design. We search answers to such questions as why these systems have the hub structure, what the basic assumptions or restrictions are, and under what circumstances these problems can be unified. We also identify the factors affecting the problem parameters and costs in each application area. This analysis is presented in Chapter 2.

In Chapter 3, we first identify the 'transportation network' aspect of the hub location problem as a distinguishing feature. We then provide a combinatorial formulation which takes into account the transportation network on which the cross traffic is carried. The existing studies in the literature on hub location have almost exclusively focused on the *p-hub median* problem which involves the minimization of total cost. In Chapter 3, we also present the literature on the p-hub median problem which involves an initial IP and different linearizations. We then provide 3 new linearizations for the initial model which are also explained in the same chapter.

Once the locations of hubs are known, the p-hub median problem turns into the hub allocation problem which is a provably difficult problem. We show in Chapter 4 that the allocation problem is NP-Hard by first proving that it is equivalent to a well known location problem, the restricted multimedian location problem with mutual communication (restricted MMC). A byproduct of this analysis is a strengthened version of a previous theorem of Tamir (1993) on the complexity of the restricted MMC problem. Additionally, we identify solvable cases of the allocation problem. These cases are in two categories: the ones utilizing the structure of the flow graph, and the ones utilizing the structure of the transport network. The definition and utilization of the transport network in this problem area is the result of our proof of equivalence of the allocation problem with the restricted MMC. The use of a transport network leads to interesting results for the allocation problem which are also discussed in Chapter 4.

Other criteria different than the total cost criterion is essentially unstudied in the literature. We analyze the hub location problem under the minimax criterion. The minimax criterion is traditionally used in location applications to minimize the adverse effects of worst case scenarios in providing emergency service. In hub location, even though emergency service protection does not seem to be an issue, the minimax criterion is still important from the viewpoint of minimizing the maximum dissatisfaction of passengers in air travel and minimizing the worst case delivery time in cargo delivery systems. The latter case is particularly important for delivery of perishable or time sensitive items. In Chapter 5, we analyze the hub location problem under minimax criterion, the *p-hub center* problem. We first prove that this problem is NP-Hard. We then focus on different linearizations of the basic model, proposed by Campbell (1994a) as well as a new model that we propose for this problem. We study the computational performances of these models. It is known from location literature that covering problems have an inverse relationship to center problems. Campbell (1994a) also defines the *hub covering problem* as it may prove to be a useful model in solving the p-hub center problem. In Chapter 5, we also analyze the hub covering problem. Utilizing its close relation to the p-hub center problem, we first prove that it is NP-Hard. We then focus on different linearizations of the basic model proposed by Campbell as well as a new model of the problem and study their computational performances. Both in the p-hub center and hub covering problems, the computational performance of the new models that we propose is far more superior than the linearizations of the basic models in terms of both CPU times and core storage requirements. This shows that it is sometimes more important to devise a new model for a given problem than focusing solely on improvements that come from different linearizations of the basic model.

We realize in our analysis of cargo delivery systems that the structure of the customarily defined hub location problem is not appropriate for the requirements of the real problem, especially for overnight delivery firms. Overnight delivery firms typically require a minimax type objective. Even though the p-hub center problem has a minimax objective, it is not a realistic model for overnight delivery since it focuses solely on the travel distances from origins to destinations without paying attention to how this travel actually takes place in the real world. One particular aspect that has been overlooked is the fact that any carrier departing from a hub must wait for the arrival of all incoming units that will be loaded onto that carrier. When there are hard constraints on the maximum delivery time, as in the case of cargo delivery, it seems necessary to pay attention to waiting times at hubs to correctly compute the maximum delivery time (Hall 1989). In chapter 6, we formulate a new model which correctly computes the delivery times. We call the resulting model the *Latest Arrival Hub Location* model. We study various aspects of the latest arrival hub location problem including modeling variations, computational aspects, and the analysis and interpretation of the model output, meanwhile investigating answers to various what-if type questions in the same chapter.

The last chapter is a short summary of the thesis and highlights our contributions to the existing literature.

Chapter 2

Hubbing in Real Life

In this chapter, we analyze the structure of three real world applications in which hubbing is most often encountered.

2.1 Airline Systems

Consider an airline company which gives flight services between pairs of, say, n cities. It is possible that there will be cross-traffic between every city pair. A crude approach to provide the required flight services between the city pairs is to assign a direct flight between each pair of cities. This would result in $\binom{n}{2}$ flight segments which results in a highly complex and expensive network structure (Yu 1998). For example, Turkish Airlines (THY) gives flight service in Turkey between 29 cities. If THY assigns a direct flight between each pair of cities, it would end up with 406 flights segments! However, THY provides the required service with only 38 segments using the hub structure shown in Figure 2.1.

Notice that instead of providing direct flights between every city pair, all flights are consolidated at 3 cities: Ankara, İstanbul, and İzmir. For example, any passenger flying from Ağrı to Edremit needs to follow the following route:

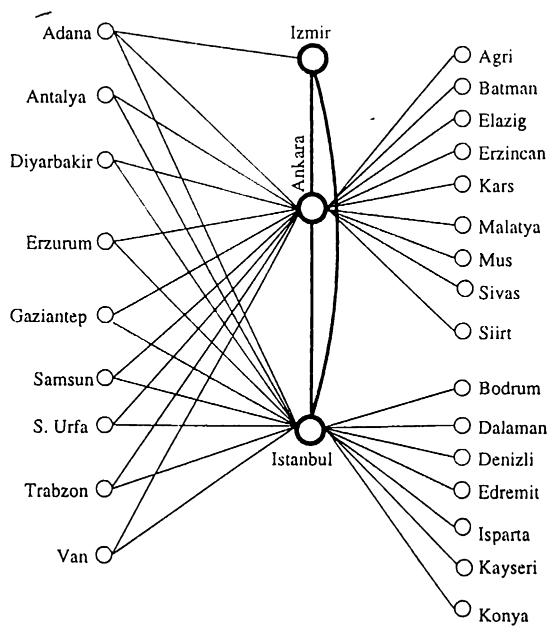


Figure 2.1: Segment Structure of Turkish Airlines

Ağrı ⟶ Ankara ⟶ İstanbul ⟶ Edremit

Toh and Higgins (1985) analyze the economical profitability of hubs in airline systems. They classify the advantages of hubbing in two categories: Operational and Marketing.

- Operational:
  - Hubbing allows indirect connections between city pairs
    - which cannot generate enough volume of traffic
    - which are too far from each other

    for direct flights.
  - Indirect flights lead to serving all city pairs with minimum aircraft availability
  - Shorter nonstop flows allow the use of smaller aircraft with greater frequency and higher utilization rates.
  - Hubbing allows smaller processing costs since maintenance, servicing, and apron services are centralized at a hub city.

- Marketing:
  - Flight arrivals/departures can be synchronized to elevate passenger load factors.
  - Hubbing establishes regional identification and domination as a result of more frequent flights.

Kanafani and Ghobrial (1985) analyze the impacts of hubbing on airport economies. The authors also provide regional air networks for Delta Airlines for 1960, 1970, and 1982. Hubbing structure has begun to emerge in the United States in 1970 and it has become dominant as of 1982. Ceha and Ohta (1994) as well as Ghobrial and Kanafani (1995) analyze hubbing in airline systems. O'Kelly (1986a) is the first thorough study of this problem. After analyzing Civil Aeronautics Board Sample Survey 1970 Report of the United States Intercity Passenger Stream, the author points out that "... several air carriers operate a highly simplified sparse network organized around hubs...".

In view of the summarized advantages of hubbing, it is clear that the network of the segment structure of airline systems will be in the hub network structure: Certain nodes will be selected as hubs, and cross-traffic between the nodes will be routed via these hubs.

We now analyze the cost parameter in airline systems. We base our analysis on Meyer and Oster (1984) and O'Connor (1989).

The total cost is usually aggregated from two factors:

- costs of opening hubs,
- costs of assigning flights between pairs of cities.

- The opening cost at city i, $f(i)$: Once a node is selected as a hub, then the airport corresponding to that node will be the place where most of the 'processings' regarding the aircrafts and passengers are done. The sum of the costs of these processes constitutes the cost $f(i)$. They include

  - Maintenance burden: overhead costs related to the upkeep and repair of flight equipment and other property such as the administering of stockrooms, the keeping of maintenance records, scheduling and servicing of maintenance operations.
  - Operating costs related to
    * Reservations and sales.
    * Advertising and publicity.
    * Traffic servicing including ticketing and baggage handling.
    * Terminal gate and lounge facilities.
  - Cost of servicing aircraft which includes the routine services such as washing the aircraft and cleaning the passenger cabin.
  - Overhead costs which includes the expenses of maintaining the organization such as personal functions, planning and general management.

- Cost of assigning a flight between two cities, $c(i,j)$, corresponds to the sum of costs which result from having a flight scheduled between cities i and j. It will incur the following types of costs:
  - Fuel cost: Each flight consumes a fixed amount of fuel for taxiing, take-off and landing. The rest of the fuel consumption varies by the distance or time flown.
  - Crew cost includes the salaries of aircraft crew. This cost is dependent on the duration of the flight since people are paid according to that.
  - Direct maintenance cost: This cost includes the cost of labor and material directly attributable to the maintenance and repair of aircraft including the periodic overhauls and other flight equipment. This cost is also dependent on the duration of the flight.
  - Miscellaneous flying and Oil cost: Miscellaneous flying and oil expenses includes the cost for other kinds of material which are needed for a flight. They are also dependent on the flight duration.

- Passenger service: This includes the cost of food and of providing cabin attendants. This cost is also dependent on the flight duration.
- Landing fee: This cost is independent of the flight duration. This fee is to be paid for every landing of an aircraft and so it is a fixed cost.

Note that all components of $c(i, j)$ except landing fee are travel time dependent and they are customarily defined as \$ per unit time costs.

Since the cost components are independent of the amount of traffic carried, this will lead to a model in which the cross-traffic between origin-destination pairs is not apparent in the objective function. If that sort of a model were applied to, say, Turkish Airlines, it may result in choosing, say, Edremit, Siirt, and Trabzon as the hub cities. Making a trip by passing through Ankara may seem reasonable for passengers, but they may question the situation if they were forced to their destination by passing through, say, Edremit. In fact, even though the costs that we have explained in $c(i, j)$ are not defined as per passenger costs in the optimization models that seek to find the location of hubs and allocation of nodes, they are transformed into per passenger costs. The main reason for that is to reflect the marketing issue in the models. Revenue of an airline system is based on the passenger fees and the volume of traffic. This may also be a reason to incorporate per passenger costs into the models. There may be ways to transform the explained costs to the per passenger cost on each arc. For simplicity, researchers usually use the length of an arc as the cost of carrying one passenger over that arc. Since all the significant cost components of $c(i, j)$ are dependent on the time of the flight duration which is directly transformable to the length of the traveled arc, taking the length as the cost seems reasonable.

2.2 Cargo Delivery Systems

Consider a firm which carries cargo between n demand centers. If the firm assigns a carrier (aircraft/truck) for each demand center pair, it will require $\binom{n}{2}$ carriers and most of the carriers will not be fully utilized (Yu 1998). Thus, as in the airline case, it is again economically infeasible to give direct service between each demand center pair and so hubbing is encountered in cargo delivery systems.

Another factor which is utilized in the service network structure of cargo delivery firms is the usage of stopovers.

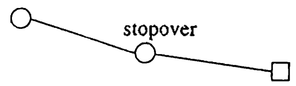


Figure 2.2: A stopover

If stopovers exists, the carrier stops at each stopover city and collects its cargo before reaching the hub. On the way back, the cargo of the stopover cities are delivered by the same carrier. Stopovers save the companies investment in carriers and also on labor and fuel and so used frequently by cargo delivery firms. Kuby and Gray (1993) give an analysis for the case of Federal Express which show that a hub network with stopovers is economically more profitable. They also point out that the cargo should arrive at its destination in 15 hours (between 5:30 p.m. and 8.30 a.m. the next morning). Another study for cargo delivery systems is conducted by Marsten and Muller (1980). The authors analyze the case of The Flying Tiger Line which used to operate under a single hub strategy. They proposed new hubs.

The design problem of cargo delivery systems is, again, that of deciding on the locations of hub nodes and determining the route for cargo between every city pair depending on the locations of these hub nodes. In the delivery business service, time is more important than its cost. As pointed out by Air Cargo World Magazine, "delivery is time-sensitive rather than price-sensitive... customers are willing to pay for time ...". Thus, the objective may be to minimize the total cost, or to minimize the maximum delivery time between an origin/destination pair (especially for overnight delivery firms), depending on the firm's strategy. So, we consider the cargo delivery firms in two categories: cost sensitive and time sensitive. We first analyze the cost parameters for the cost sensitive case. This analysis is based on O'Connor (1989). Total cost is again aggregated from two factors: cost of declaring a city as a hub, $f(i)$, and cost of providing direct service between two nodes, $c(i,j)$.

- The setup cost for 'processings' operations, $f(i)$, includes :
  - Cost of sorting and allocating. Cargo from many demand centers come to the hub at batches. Sorting and allocating these cargo according to their destinations is a major cost component.
  - Cost for unloading arriving carriers and loading the leaving ones.
  - Cost for storing, guarding, and providing proper protection for the waiting cargo.
  - Cost for paperwork which is used to tell shippers where their cargo is, or to keep statistics.
- Cost of providing service between two nodes i and j, $c(i,j)$, includes:
  - Cost of allocating aircraft/truck.
  - Fuel cost for aircraft/truck depending on the travel time.
  - Crew cost, which is the salary for the driver or aircraft crew.
  - Direct maintenance cost of aircraft/truck. It includes both labor and equipment cost used for maintenance.
  - Equipment cost, i.e. the cost of aircraft/truck depreciated to unit time of usage.
  - Other miscellaneous and oil costs which are encountered by carriers in order to 'move'.

If the objective is to minimize the maximum delivery time, then the 'cost' parameter is, in fact, the duration of the corresponding journey. In order to identify the factors which constitute the cost, we need to analyze how the delivery actually takes place. We base our analysis on Sigafoos and Easson (1988). The typical overnight delivery firm picks up packages from customers at a local station by 5:30 p.m. with a promise to deliver them to their destinations by 8:30 a.m. the next morning. Each incoming package at the local station is labeled (e.g. fragile, hazardous, flammable) and assigned a bar code that includes the zip code of the destination. The processed units are loaded onto an aircraft and are delivered to the hub which serves that local station. There are two major operations at any hub: unloading the arriving aircrafts and loading the departing ones. The packages that are unloaded from arriving aircrafts are fed into a conveyor system that is equipped with manual or automatic bar code readers. The bar code readers at the feeder lines read the zip code information and route the packages to the specific area of the hub where they can be reloaded onto the correct cargo containers. The outgoing aircraft is ready to depart when all the cargo for its destination is loaded onto it. If a departing aircraft from a hub is destined to go to a nonhub city, then it is unloaded at the local station of its final destination and the unloaded packages are delivered to the consignees by 8:30 a.m. An aircraft that goes from a hub to another hub goes through the unloading, reloading, and the associated sorting/routing operations at the second hub to have its cargo delivered to the final destination cities that are serviced from that hub.

As is evident from the above description, the delivery time from an origin i to a destination j consists of two components: flight times and the transient times spent at hubs between flights. Then 'for a time sensitive cargo delivery firm' the problem is to decide on the locations of hubs and the route structure for each pair of nodes so as to minimize the latest arrival time at any consignee.

2.3 Communication Network Design

Communications network is the general name given to networks which are installed to satisfy communication between 'devices' which 'communicate' (Stallings 1991). The terms 'communication' and 'communicating device' are specified according to the usage area. For example, if the devices are computers and if communication is data transfer or program execution, then we have computer communication networks. On the other hand, if devices are telephones and if communication is phone-talk, then we have telephone networks. The basic structure of all communication networks is the same. Considering the fact that the number of devices which communicate is typically very large, it is clear that providing direct connection between each pair of devices is impossible. Communication between any pair of devices is satisfied through a 'communication network'.

There are some special devices which are used to satisfy communication needs in a communication network. Some examples from computer communications are: multiplexer, concentrator, router, bridge, gate, switch, hub, repeater etc. They all have different special characteristics, but their main purpose in the communication network is to allow data to pass on its way to its destination. Multiplexers and concentrators are used to utilize link capacities, whereas switchers are used for switching the flowing data. Bridges and routers are used for interconnecting geographically distant devices. Hubs and repeaters repeat the incoming information on their output line. Depending on the use of these special devices, the topological structure of communication networks, and the hierarchy of the network change.

The topological structure of communication networks is hierarchical. The number of hierarchy levels depends on the size (both geographical, and the number of devices) of the communication network. The lowermost layer is composed of communicating devices, e.g. computers, telephones etc., and the upper layer(s) is composed of special devices, e.g. concentrators, hubs, etc.

The topological design problem of communication networks is customarily defined for a two-level hierarchical structure. In fact, any problem which require more than two levels of hierarchy can be analyzed by taking into account two levels at a time sequentially, from down up (Stallings 1991). In a two layered network, the upper layer is called the 'backbone network' and the lower layer is called the 'local access network'. Topological design problem has three phases (Gavish 1991, 1992):

- locating backbone nodes
- designing the backbone network
- designing local access network and connecting local access network to the backbone network.

Each device in the local access network sends its message using the backbone nodes. Depending on the position of the destination device, backbone arcs may or may not be used. In order to satisfy communication between each pair, a path must exist from each device to at least one backbone node. Physical links are being established to satisfy this communication. In communication networks there are no physical carriers. Message from any device travels along the network until it reaches the destination device. In order to utilize the established links, 'multidrop lines' are used which means several devices are attached on a line and they use the same entrance to the backbone node. A message will flow through a link only if that link is empty at that moment. If the link is busy, the message either waits, or continues along its way by using another link (if exists).

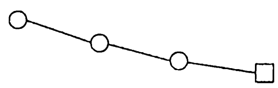


Figure 2.3: A Multidrop Line

When installing a communication network, the set of potential locations for backbone nodes is given. The problem is to select which subset of this potential location set will be used as backbone nodes and to determine the route structure of any communicating pair. Note that the set of potential locations for backbone nodes is not a subset of devices. This is one of the major differences between communication networks and airline or cargo delivery networks. In the first two application areas, hub locations are selected from among the demand centers whereas, in communication networks, hub (backbone node) locations are selected from a different set that does not include any demand centers (devices). In fact, this difference can be handled by enlarging the set of demand centers by adding to it the potential hub location set. The nodes corresponding to these locations will have zero demand, i.e. the flows originating or ending at these nodes will be zero.

For communication networks there are two types of costs: communication costs and cost of delay. The delay costs include the costs that result from queuing. In the literature, there are some models which include these costs in the form of expectations based on some underlying distribution. In this thesis we mainly concentrate on the communication costs.

Communication costs are of two types: setup costs and movement costs. As discussed in Klincewicz (1998) and in Altınkemer and Yu (1992),

- setup costs of hubs, $f(i)$, include
  - investment for acquiring land for the multiplexers, concentrators etc.
  - all equipment costs which depend on the type of backbone node (concentrator, switch etc.)
- arc cost, $c(i,j)$, includes
  - investment for acquiring land for the link
  - material and equipment cost such as cost of fiber, repeater etc.
  - if the line is leased, this cost represents the fixed charge paid to the company for using arc (i,j).

For determining the route structure, the researchers prefer to specify the structures of local access and backbone networks, and analyze the problem according to that specification. Most commonly used structures are: complete, tree, star or ring structures for backbone networks and star or tree structures for local access networks.

For example, if backbone network is complete and local access network is a star tree, then any device will have connection with exactly one backbone node, and data will travel along 2 or 3 links before reaching the destination.

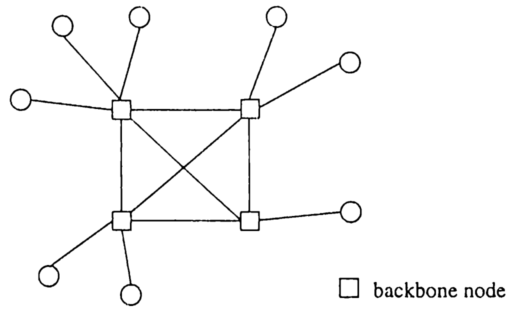


Figure 2.4: Example of a Complete-star Topology

Problems which are in other structures can be handled by writing appropriate constraints to satisfy the required network structure in the final network. For example, Kim and Tcha (1992) analyze the case when the backbone network is a tree and the local access network is a star tree. Lee, Ro, and Tcha (1993) analyze ring structured backbone networks connected to a star type local access networks. Chung, Myung, and Tcha (1992) analyze the complete-star topology.

2.4 Structural Similarities and Differencies

Upto now we have identified 3 different real world application areas : airline systems, cargo delivery systems, and communication networks. The common properties of these systems are:

1) There are n demand centers and there is some sort of flow between these demand centers (people, cargo, data packets).

2) It is economically infeasible to give direct service between each pair since this would require $\binom{n}{2}$ connections which is too costly and results in a complicated network.

3) The flow between each pair is required to pass through some specific nodes which are called hubs. In airline or cargo delivery systems hubs are nodes for consolidating and disseminating flows whereas in communication networks they are used for switching or multiplexing data packets.

Besides these common properties, there are some area specific properties:

1) For cargo delivery systems there are two different problem types: cost sensitive and time sensitive. Both of them satisfy the 3 common properties explained above, but the design problem of the time sensitive cargo delivery systems has additional requirements.

2) Stopovers may be used in cargo delivery systems and multidrop lines may be encountered in communication networks. Such issues are not considered in airline systems.

3) In airline and cost sensitive cargo delivery systems, per passenger and per cargo costs are used which are called usage-based costs in the literature whereas in communication network design problems hub and link setup costs are considered.

The 3 common properties of the three problem areas are the basics of a problem known in the literature as the hub location problem. The hub location problem can be stated as follows: Given n demand centers with known cross traffic, and arc costs satisfying triangle inequality, determine the locations of hubs and the allocations of nonhub nodes to hubs so as to minimize the total cost. In terms of the allocation of nonhub nodes to hubs, there are two variates of the problem: the single-assignment allocation and the multi-assignment allocation. In the single-assignment allocation, each node is assigned to exactly one hub whereas in the multi-assignement allocation each node can be assigned to many different hubs. A typical flow between an origin/destination pair (i, j) is then: $i \to$ hub1 $\to$ hub2 $\to j$ where hub1 = hub2 allowed. $i \to$ hub1, hub2 $\to j$, and hub1 $\to$ hub2 are direct links . Having a direct link between $i \to$ hub1, hub2 $\to j$ is a problem requirement (allocation phase) whereas the direct link between hub1 $\to$ hub2 is justified by the triangle inequality assumption on arc costs.

Because of this route convention of the hub location problem, we conclude that the customarily defined hub location problem has application in airline, or cargo delivery systems and in communications network design only under certain assumptions.

- Airline systems: Restricts the route between each origin destination pair to at most 3 segments on each route, stopping only at hub nodes.
- Cargo delivery systems: Applicable to cost sensitive cargo delivery systems when stopovers are not allowed. The route restriction explained for airline systems is also valid here. For time sensitive cargo delivery systems, even though the basic requirements of hubbing are the same (open p hubs, provide service to each node from hubs), there are additional restrictions to correctly compute transport times. This special problem has not been analyzed in the literature. We propose a new model for this problem which will be explained in Chapter 6.
- Communication network design: Applicable to systems with complete backbone network without any multidrop lines.

Chapter 3

The p-Hub Median Problem

Most of the literature on the hub location problem is devoted to the *p-hub median* problem which is the problem of locating a fixed number, p, of hubs and finding the allocation(s) for each node based on the selected set of hubs while minimizing the total cost of travel. We provide combinatorial formulations of the p-hub median problem for both single and multi assignment cases.

3.1 Combinatorial Formulation

In the literature, the p-hub median problem is generally posed by its integer programming formulations. We first give a combinatorial formulation which is more compact and provides additional insights.

Let $G = (N', E)$ be a connected *transportation network* with node set $N' = \{1, ..., n'\}$ and arc set E. Without loss of generality, we assume that the nodes $1, ..., n$ are the demand centers each of which generates and/or absorbs a positive flow from the rest of the nodes. The remaining nodes $n + 1, ..., n'$ are intersection points of the transport network, and neither generate nor absorb a flow, but act as transshipment points which allow the passage of flows. Let $N = \{1, ..., n\}$ and refer to this set as the demand set. The arc set E is composed of the links of the transportation network. For a surface transportation system, these arcs are the road segments whereas, for air transport, the arcs correspond to nonstop flight routes. Associated with each arc $(i, j) \in E$ is a weight $d_{ij} > 0$ which represents the cost of carrying a unit flow between its endpoints. In most cases, we may interpret this cost to be the physical distance between the endpoints of the arc. The length of a path in G is the sum of the weights of its edges. For each pair of nodes $i, j \in N'$, let c_{ij} be the length of a shortest path connecting i and j. Note that, under the assumption of a connected network, c_{ij} is always finite even if $(i, j) \notin E$. Note also that, $0 \leq c_{ij} \leq d_{ij} \forall i, j \in N', c_{ij} = 0$ iff $i = j, c_{ij} = c_{ji}$ and $c_{ij} + c_{jk} \geq c_{ik} \forall i, j, k$. Let w_{ij} denote the flow from node i to node j with w_{ij} or $w_{ji} > 0$ for $i, j \in N$ and $w_{ij} = 0$ otherwise. We define the *cost network* to be the auxiliary graph $K = (N, A)$ with node set N and undirected arc set $A = \{(i, j) : i, j \in N\}$. Assign the weight c_{ij} to each arc $(i, j) \in A$. Note that the cost network is a complete graph on n nodes. In order to incorporate economies of scale resulting from the increased traffic between the hub nodes, the least cost of travel between the hub nodes is discounted by a factor α $(0 \leq \alpha \leq 1)$. Hence, the cost of carrying unit flow between two hub nodes k and r is αc_{kr}. The p-hub median problem can be defined using the transport network, G, or the cost network, K. Most of the models in the literature do not differentiate between these two networks and define the problem with respect to a 'cost matrix'. However, how the entries of the cost matrix are defined is not very clear. In the papers that we are aware of in the literature, the problem is posed on a complete graph whose arc costs satisfy the triangle inequality. This graph corresponds to our cost network. The arc costs of the cost network, K, are induced by the shortest path lengths of an underlying transportation network, G. We prefer to make a distinction between these two networks because the structure of the transportation network has a role in determining the difficulty of the problem. For example, as will be proved in the sequel, the allocation problem is NP-Hard for $p = 3$ even if the transport network is a star tree whereas it is polynomially solvable for general p if the transport network is a path. The transportation network aspect of this problem is first distinguished by Kara and Tansel (1997).

The p-hub median problem is to select a subset $H = \{h_1, ..., h_p\}$ of N' and allocate the rest of the demand nodes to the hub nodes $h_1, ..., h_p$. As discussed in Chapter 2, $H \subseteq N$ in airline or cargo delivery systems, and $H \subseteq N' - N$ in communication network design. Because our focus is primarily on airline or cargo delivery systems, we follow the customary convention in the literature and assume that the hubs are selected from among the demand centers; that is, $H \subseteq N$.

Under the single assignment restriction, all the incoming and outgoing traffic of a node passes through the same hub. Define $a(i) \in H$ to be the hub to which node i is assigned. Then the cost of carrying a unit flow from i to j is :

$$c_{i\,a(i)} + \alpha c_{a(i)\,a(j)} + c_{a(j)\,j} \qquad (1').$$

The single assignment p-hub median problem is, then:

$$\min_{\substack{H \subset N, \\ |H|=p}} \min_{a \in H^n} \sum_{i,j \in N} w_{ij}(c_{i\,a(i)} + \alpha c_{a(i)\,a(j)} + c_{a(j)\,j}) \qquad (2)$$

For the multiple allocation case, once the hubs are fixed, the optimal allocations of nodes can be found directly by taking into account each pair of demand nodes separately (under the assumption that the arcs are uncapacitated). The allocation for the pair (i, j) is determined by solving:

$$\min_{h_1, h_2 \in H} (\, c_{i\,h_1} + \alpha c_{h_1\,h_2} + c_{h_2\,j}) \qquad (3)$$

For a fixed value of i, the result of the minimization in (3) for different values of j may cause demand center i to be allocated to different hubs. In fact, it is possible for a demand center to be allocated to all of the hubs. The multiple assignment p-hub median problem is then :

$$\min_{\substack{H \subset N, \\ |H|=p}} \sum_{i,j \in N} w_{ij} \min_{h_1, h_2 \in H} \{c_{i\,h_1} + \alpha c_{h_1\,h_2} + c_{h_2\,j}\} \qquad (4)$$

3.2 Literature on Hub Location Problems

The hub location problem is first posed by O'Kelly (1986a) in which the author gives real world examples which operate under one-hub or two-hub strategies. O'Kelly points out the fact that $\binom{n}{2}$ segment requirements would drop to $(n-1)$ if hubbing exists. Later, O'Kelly (1986b) analyzes the cost terms in a hub network and identifies the quadratic structure in hubbing. O'Kelly (1987) presents the first model of this problem which happens to be a binary program with a quadratic objective function. He considers the case with single allocation and provides the following quadratic integer programming formulation, (QP), which is later considered to be the basic model for hub location problems in the literature.

Let X_{jk} be a zero/one variable which takes on the value 1 if node j is assigned to hub k and 0 otherwise. Note that $X_{kk} = 1$ means there is a hub at node k and $X_{kk} = 0$ means there is no hub at node k. With the parameters N, w_{ij}, c_{ij}, p, and α as defined in Section 3.1, the formulation provided by O'Kelly (1987) is :

$$(\mathrm{QP}) \quad \min \sum_{i,j} w_{ij} \{ \sum_k c_{ik} X_{ik} + \sum_k \sum_m \alpha c_{km} X_{ik} X_{jm} + \sum_m c_{jm} X_{jm} \}$$

s.t

$$\sum_k X_{ik} = 1 \qquad \forall\, i \in N \qquad (5)$$

$$\sum_k X_{kk} = p \qquad (6)$$

$$X_{ik} \leq X_{kk} \qquad \forall\, i, k \in N \qquad (7)$$

$$X_{ik} \in \{0,1\} \qquad \forall\, i, k \in N \qquad (8)$$

Constraints (5) and (8) ensure that each node is assigned to exactly one hub whereas constraint (7) allows assignments to hub nodes only. The restriction on the number of hubs being p is satisfied by constraint (6). This model is linearized by many other researchers. Initial linearizations of Aykın (1995a), Campbell (1996), and Skorin-Kapov et al. (1996) are based on defining four indexed variables, namely X_{ijkm} to replace the product terms $X_{ik}X_{jm}$. Among the three, that of Skorin-Kapov et al.'s happens to perform best in terms of solution time requirements on standard optimization tools such as CPLEX. We will analyze their model later in sections 3.3.1 and 3.3.2. The best integer programming formulation (in terms of CPU time) for the p-hub median problem is due to Ernst and Krishnamoorthy (1996), (1998a) which is a multicommodity flow based formulation and will also be analyzed in sections 3.3.1 and 3.3.2.

O'Kelly and Miller (1994) develop a categorization for hub location problems. They identify three criteria as a basis of their categorization.

- the structure of the connection between hub nodes
- the allocation structure (single/multi)
- the existence or nonexistance of direct connections between nonhub origin/destination pairs

Note that with our formulation of the cost matrix, we do not need to make the first distinction. The standard hub location problem corresponds to a hub connection structure which is a complete graph, single allocation structure, and nonexistence of direct connections between nonhubs. Aykın (1995a) develops an integer programming formulation for both single and multiple allocations where there is a subset A of demand centers for which direct service is allowed.

Aykın (1994), Campbell (1996), Klincewicz (1991), (1992), (1996), O'Kelly (1992a), O'Kelly et al. (1994), Skorin-Kapov et al. (1994) and Ernst and Krishnamoorthy (1996), (1998a), (1998c) develop several heuristics for the p-hub median problem. Lower bounds for the p-hub median problem are analyzed by O'Kelly (1995) and Ernst and Krishnamoorty (1998b).

Since the literature on the hub location problem is mainly devoted to the p-hub median problem, the literature on other hub location problems are sparse and so we will discuss them here. Campbell (1994a) is the only study which discusses performance measures other than the total sum criterion. He proposes integer programming formulations for the p-hub median, p-hub center, uncapacitated hub location, and hub covering problems. Among these problems, the p-hub center and the hub covering problems have not received any attention in the literature. We analyze these problems in Chapter 5.

O'Kelly (1992b) introduces the planar p-hub median problem and Aykın (1995b) develops heuristics for this problem. Campbell (1994b) presents a survey of hub location papers.

3.3 Different Linearizations of the Basic Model

In this section, we analyze the computational performance of different linearizations of the basic model (QP) both for single and multi assignment cases. Apart from the linearizations of the literature, we also propose new linearizations for the single assignment case.

3.3.1 Single Allocation

In the literature, there are mainly four different linearizations for (QP); the ones provided by Aykın (1995a), Campbell (1996), Skorin-Kapov et al. (1996), and Ernst and Krishnamoorthy (1996).

The first three linearizations are similar to each other. They define $X_{ijkm} = X_{ik} * X_{jm}$ and develop integer programming models which differ in forcing that equality linearly. Among all, the best one (in terms of CPU requirement for CPLEX 5.0) is the one provided by Skorin-Kapov et al. (1996). Their model is also the one whose linear programming relaxation is tightest.

With $C_{ijkm} = c_{ik} + \alpha c_{km} + c_{jm}$ the model provided by Skorin-Kapov et al. is:

$$
\begin{array}{lll}
\text{(SK)} & \min & \sum_i \sum_j \sum_k \sum_m w_{ij} C_{ijkm} X_{ijkm} \\
 & \text{s.t} & \\
 & & \begin{array}{rcll}
\sum_m X_{ijkm} & = & X_{ik} & \forall\, i,j,k \in N \qquad (9) \\
\sum_k X_{ijkm} & = & X_{jm} & \forall\, i,j,m \in N \qquad (10) \\
X_{ijkm} & \in & \{0,1\} & \forall\, i,j,k,m \in N \qquad (11) \\
(5)-(8) & & &
\end{array}
\end{array}
$$

The constraints (9) - (11) are to satisfy that $X_{ijkm} = 1$ if and only if $X_{ik} = X_{jm} = 1$. The authors report that the LP relaxation of their linearization has usually ended in all integer solutions (74 of 80 in CAB2 data test which will be explained later).

The linearization of Ernst and Krishnamoorthy (1996), on the other hand, is a different approach. They utilize a multicommodity flow structure in their model. They consider output flow of each demand center as a different commodity and model the related multicommodity problem. The authors define Z^i_{kl} to be the amount of flow of commodity i (flow emanating from node i) that is routed through hubs k and l. With $O_i = \sum_j w_{ij}$ and $D_i = \sum_j w_{ji}$ their model is:

$$
\begin{array}{lll}
\text{(EK)} & \min & \sum_i \sum_k (O_i + D_i) X_{ik} c_{ik} + \sum_i \sum_k \sum_l \alpha c_{kl} Z^i_{kl} \\
 & \text{s.t.} & \\
 & & \begin{array}{rcll}
\sum_l Z^i_{kl} - \sum_l Z^i_{lk} & = & O_i X_{ik} - \sum_j w_{ij} X_{jk} & \forall\, i,k \in N \qquad (12) \\
Z^i_{kl} & \geq & 0 & \forall\, i,k,l \in N \qquad (13) \\
(5)-(8) & & &
\end{array}
\end{array}
$$

where constraint (12) is the flow balance equation.

As seen in Table 3.2, the computational performance of this model solved via CPLEX is very good. The CPU time of (EK) is even better than the CPU time required to solve the LP relaxation of (SK).

At the time of our study of the p-hub median problem, the EK model was not in existence, so we have independently developed three different linearizations for this problem:

- The first linearization is an adaptation of the linearization of Kettani and Oral (1993) for the Quadratic Assignment Problem (QAP) to the p-hub median problem.
- The second linearization is based on a multicommodity flow structure where the flow between each pair of nodes is taken as a different commodity.
- The third one is based on a reinterpretation of the quadratic objective function.

The first linearization which is based on Kettani and Oral's approach defines a new variable $\epsilon_{ik} = X_{ik}\sum_j \sum_m C_{ijkm}X_{jm}$ and forces this equality by appropriate lower and upper bounds. The IP model of the hub location problem linearized via this approach, (KT1), is:

$$
\begin{array}{llll}
\text{(KT1)} & \min & \sum_i \sum_k \{D_{ik}^- X_{ik} + \epsilon_{ik}\} & \\
& \text{s.t.} & & \\
& \epsilon_{ik} & \geq \sum_j \sum_m C_{ijkm}X_{jm} - D_{ik}^- X_{ik} + D_{ik}^+(1 - X_{ik}) \quad \forall\, i, k, \in N & (14) \\
& \epsilon_{ik} & \geq 0 \quad \forall\, i, k \in N & (15) \\
& & (5) - (8) &
\end{array}
$$

where D_{ik}^- and D_{ik}^+ are lower and upper bounds for the variable ϵ_{ik}. The authors suggest that these bounds can be determined by using the procedure suggested by Kaufman and Broeckx (1978). This results in:

$$
D_{ik}^- = \min \sum_j \sum_m C_{ijkm}X_{jm} \quad \text{s.t. } (5) - (8) \qquad \text{and} \qquad D_{ik}^+ = \max \sum_j \sum_m C_{ijkm}X_{jm} \quad \text{s.t.} (5) - (8)
$$

While solving the model (KT1), the calculation of the lower and upper bounds D_{ik}^- and D_{ik}^+ requires solving $2n^2$ IP's via CPLEX. Once the D_{ik}^- and D_{ik}^+ for all n^2 variables are calculated, the CPU time requirement of (KT1) would be very small. The major part of the CPU time is due to the preprocessing required to form D_{ik}^- and D_{ik}^+ values. It is possible to use different and easy to calculate methods to find correct bounds for the variable ϵ_{ik}, but then, the bounds will not be tight enough and the solution time of (KT1) would increase. As seen in Table 3.2, the computational performance of (KT1) is not satisfactory.

In the second linearization that we propose, we interpret flow between each city pair as a different commodity. We define Z_{ijkm} to be the amount of commodity (i, j) that is routed through hubs k and m. The resulting multi-commodity model is:

$$
\begin{array}{llrcll}
\text{(KT2)} & \min & \multicolumn{4}{l}{\sum\limits_i \sum\limits_k (O_i + D_i) X_{ik} + \sum\limits_i \sum\limits_j \sum\limits_k \sum\limits_m C_{ijkm} Z_{ijkm}} \\
& \text{s.t.} & & & & \\
& & (w_{ij} + w_{ji}) X_{ik} - (w_{ij} + w_{ji}) X_{jk} & = & \sum\limits_m Z_{ijkm} & \forall\, i, j, k \in N \quad (16) \\
& & Z_{ijkm} & \geq & 0 & \forall\, i, j, k, m \in N \quad (17) \\
& & (5) - (8) & & &
\end{array}
$$

where (16) is the flow balance equation. The structure of the model (KT2) is similar to that of flow with gains model. Thus, we initially expected that the CPU time requirement of (KT2) would be better than the other linearizations. However, the linearization of Ernst and Krishnamoorthy (EK) is better than that of (KT2). The structure of (KT2) is similar to that of (SK) and the computational performance of (KT2) is competitive with that of (SK) as seen in Table 3.2.

The third linearization that we propose in this report happens to be the best linearization of the p-hub median problem in terms of core storage requirements. For that linearization, we first define

$$Z_j = \sum_i w_{ij} \sum_k \left[c_{kj} + \sum_r (c_{ir} + \alpha c_{rk}) X_{ir} \right] X_{jk} \quad (18)$$

Since $\sum\limits_k X_{ik} = 1$, we have $\sum\limits_j Z_j = \sum\limits_j \sum\limits_i w_{ij} \left[\sum\limits_k \sum\limits_r (c_{ir} + \alpha c_{rk} + c_{kj}) X_{ir} X_{jm} \right]$ and so the objective of the p-hub median problem can be written as $\sum\limits_j Z_j$ where Z_j is determined by (18). Observe that the second summation operator (over k) in (18) can be replaced with a maximand operator since there exists exactly one k for which $X_{jk} = 1$ for every j. Then we have:

$$Z_j \geq \sum_i w_{ij} \left[\sum_r (c_{ir} + \alpha c_{rk}) X_{ir} + c_{kj} \right] X_{jk} \ \forall k \quad (19)$$

Consequently, the p-hub median problem can be stated as:

$$\begin{array}{lll} & \min & \sum_j Z_j \\ (\text{KT3})' & \text{s.t.} & (5)-(8),(19), Z_j \geq 0 \end{array}$$

Note that, in (KT3)' the nonlinearity is in the constraints.

Observation 1: $Z_j \geq \sum_i w_{ij} \left[\sum_r (c_{ir} + \alpha c_{rk}) X_{ir} - c_{jk} + 2c_{kj} X_{jk} \right]$ (20) correctly linearizes the constraint (19).

Proof : There are two cases to consider depending on the value of X_{jk}

- Case 1: If $X_{jk} = 1$, then the right hand sides of inequalities (19) and (20) are the same. $\surd$.
- Case 2: If $X_{jk} = 0$, then (20) provides $Z_j \geq \sum_i w_{ij} \left[\sum_r (c_{ir} + \alpha c_{rk}) X_{ir} - c_{jk} \right]$ whereas (19) provides $Z_j \geq 0$. For ease of computation, we define 2 auxiliary variables:

$$\begin{array}{ll} \text{Let} & L_{ik} = \sum_r (c_{ir} + \alpha c_{rk}) X_{ir} \\ \text{and} & S_j^k = \sum_i w_{ij} [L_{ik} - c_{jk} + 2c_{jk} X_{jk}] \\ \text{with} & Z_j \geq S_j^k \ \forall k \end{array}$$

Since $X_{jk} = 0$ in this case, due to constraint (5), there exists $k'(\neq k)$ such that $X_{jk'} = 1$.

$$\begin{array}{ll} \text{Then} & S_j^{k'} = \sum_i w_{ij} (L_{ik'} + c_{jk'}) \\ \text{and} & S_j^k = \sum_i w_{ij} (L_{ik} - c_{jk}). \end{array}$$

We prove the observation by showing that the inequality provided by (20) for the case with $X_{jk} = 0$ is ineffective. For this we show that

$$S_j^{k'} \geq S_j^k \ \forall k \quad (21).$$

We show that $L_{ik'} + c_{jk'} \geq L_{ik} - c_{jk} \; \forall \; k$ (22) which is stronger than (21). Using triangle inequality, (22) can be rewritten as

$$L_{ik'} + c_{jk'} + c_{kj} - c_{kj} \geq L_{ik'} + c_{kk'} - c_{kj} = \sum_r (c_{ir} + \alpha c_{rk'}) X_{ir} + \alpha c_{kk'} - c_{jk}.$$

Without loss of generality, let $X_{is} = 1$. Then we have

$$L_{ik'} + c_{jk'} \geq c_{is} + \alpha c_{sk'} + \alpha c_{kk'} - c_{jk} \geq c_{is} + \alpha c_{sk} - c_{jk} = L_{ik} - c_{jk}.$$

This proves the inequality (22) which is stronger than (21). So, we have shown that $S_j^{k'} \geq S_j^k \forall k$. This means that even if (20) for $X_{jk} = 0$ case provide an incorrect constraint, that constraint is ineffective since there will be another k' for which $X_{jk'} = 1$. That is, (20) always provides a correct constraint which is tighter than any incorrect one. □.

Then a new formulation for the p-hub median problem is:

$$\text{(KT3)} \quad \begin{array}{ll} \min & \sum_j Z_j \\ \text{s.t.} & (20), (5) - (8), Z_j \geq 0 \end{array}$$

Note that in this new formulation, there are n^2 binary and n real variables with $2n^2 + n + 1$ constraints. Table 3.1 provides the number of variables and constraints for all the linearizations.

	Variables			
Model	Binary	Real	Total	Constraints
(SK)	$n^4 + n^2$	-	$n^4 + n^2$	$2n^3 + n^2 + n + 1$
(KT1)	n^2	n^4	$n^4 + n^2$	$n^3 + n^2 + n + 1$
(EK)	n^2	n^3	$n^3 + n^2$	$2n^2 + n + 1$
(KT2)	n^2	n^2	$2n^2$	$n^3 + n^2 + n + 1$
(KT3)	n^2	n	$n^2 + n$	$2n^2 + n + 1$

Table 3.1: Core Storage Requirement of the Linearizations of (QP)

Note that, except for (SK), all the linearizations require n^2 binary variables. In terms of storage requirements, the best model is our third linearization, (KT3).

The computational performances of all the linearizations are compared by solving the models via CPLEX 5.0 using the CAB Data set. That data set is generated from the Civil Aurenatics Board Survey of 1970 passenger data in the United States. It contains the passenger flow between 25 cities and the distance between the cities. This data set is considered as benchmark by all the researchers in hub location area and any heuristic or exact solution procedures are tested via this data set. The researchers take northwest 10, 15, 20 and 25 nodes as different sets of nodes. The number of hubs, p, is taken from the set $\{2, 3, 4\}$ and the discount factor α is taken from the set $\{0.2, 0.4, 0.6, 0.8, 1.0\}$. The combinations generate $4 * 3 * 5 = 60$ instances. We enlarge the standard test set for p by adding the case $p = 5$ and compare the performance of different linearizations on the resulting 80 instances. We make a distinction about the data sets and call the standard set consisting of 60 instances CAB1 and call the enlarged set of 80 instances CAB2. Table 3.2 gives average and maximum CPU usage of CPLEX for each value of n with CAB2.

We have identified 5 different linearizations of the basic model, (QP). Skorin-Kapov et al. (1996) claimed that their linearization is the best among the one provided by Campbell (1996) and Aykın (1995a) in terms of solution time and quality. The common structure in all the three linearizations is that they all define X_{ijkm} variables. The resulting models are really huge and it is nearly impossible to solve them as integer programs. Instead, the authors concentrate on LP relaxations and add integrality restrictions only when the LP solutions have noninteger variables. In the CAB2 data set, the LP relaxation of (SK) found the optimal integer solution in 74 out of 80 instances which is better than both Campbell's and Aykın's. However, when compared to our four indexed multicommodity flow based linearization (KT2), the performance of (KT2) is nearly 30 times faster than that of (SK) in average CPU time requirements. (KT2) gives the same performance in terms of solution quality as (SK). It finds integer solutions to the same set of 74 instances out of 80 as does (SK). The 6 instances that result in noninteger solutions have the same cost at LP optimality in both (KT2) and (SK).

The three indexed multicommodity flow based linearization provided by

		CPU	
	n	Avg.	Max
(SK) as LP	10	15.8 sec.	21.8 sec.
	15	4.9 min.	6.1 min.
	20	56.9 min.	1.3 hrs.
	25	4.75 hrs.	8.56 hrs.
(KT2) as LP	10	0.9 sec.	1.2 sec.
	15	9.87 sec.	15.9 sec.
	20	2 min.	4 min.
	25	9.6 min.	19.5 min.
(EK)	10	0.9 sec.	1.7 sec.
	15	10 sec.	21 sec.
	20	3.5 min.	14.8 min.
	25	28.2 min.	4.8 hrs.
(KT1)	10	13.3 sec.	14.5 sec.
	15	3.6 min.	4.7 min.
	20	20.05 min.	50 min.
	25	1.6 hrs.	9 hrs.
(KT3)	10	5.1 sec.	9 sec.
	15	1.9 min.	4.4 min.
	20	19.4 min	49.3 min.
	25	1.5 hrs.	8.83 hrs.

Table 3.2: Computational Performance of the Linearizations

Ernst and Krishnamoorthy (EK) can be solved as IP in CPLEX. Both the storage requirements and solution times as IP are very low when compared with those of (SK) and (KT2) as IP's. In fact, the (EK) model obtains the IP solution in a few seconds even if you do not specify any starting solution. The authors also suggest a heuristic algorithm whose solution can be used as a starting solution for CPLEX. The solution quality of their heuristic is also very good. The heuristic finds the optimal solution in 59 out of 60 instances of the CAB1 data set.

It is also possible to solve our linearization (KT1) as IP in CPLEX. However, the solution times are higher than that of (EK). In fact, the highest portion of the solution times are due to the computation of the lower bounds D_{ik}^- and D_{ik}^+ for each (i, k) pair. Nearly 70% of the total time is devoted to this bound calculation. Once these values are known, the solution time of (KT1) is very low (even lower than that of (EK)). However, those bounds have to be calculated and that increases the solution time of (KT1).

For our last linearization (KT3), the CPU times are much better than that of (SK), (KT1), and (KT2). However, in terms of average CPU time, (EK) is about 10 times faster than (KT2) for $n = 10$ and 15, while it is about 5.5 times faster for $n = 20$ and about 3 times faster for $n = 25$. The gap in average CPU time between (EK) and (KT3) seems to be decreasing with increased n.

In conclusion, we may say that (EK) gives the best CPU times as IP on CPLEX followed by (KT3) which is about 3 to 10 times slower while (KT3) is the best one in terms of core storage requirements with (EK) also being nearly as competitive in this respect as (KT3). The state of the art is determined by (EK).

3.3.2 Multiple Allocation

O'Kelly's original formulation (QP) is for the single allocation case. For multiple allocation, there are some mixed integer programming formulations provided by Aykın (1995a), Campbell (1996), Skorin-Kapov et al. (1996), and Ernst and Krishnamoorthy (1998a). In the first three models the authors relax the integrality requirement on X_{ijkm} variables and define X_{ijkm} to be the fraction of flow from i to j that is routed through hubs k and m. The basic structure of the model, as represented in Campbell (1996), is:

$$\min \sum_i \sum_j \sum_k \sum_m w_{ij} C_{ijkm} X_{ijkm}$$

s.t

$$\begin{aligned} \sum_k \sum_m X_{ijkm} &= 1 && \forall\, i,j \in N && (23) \\ X_{ijkm} &\leq X_{kk} && \forall\, i,j,k,m \in N && (24) \\ X_{ijkm} &\leq X_{mm} && \forall\, i,j,k,m \in N && (25) \\ X_{ijkm} &\geq 0 && \forall\, i,j,k,m \in N && (26) \\ (6) - (8) & && && \end{aligned}$$

Note that, in the solution of the above model the real valued variables X_{ijkm} will have values of 0 or 1. Since there is no capacity restriction, the minimizing C_{ijkm} will be selected for each (i, j) pair. There is a formal proof of this result in Ernst and Krishnamoorthy (1998a).

For the solution of the above model, Aykın (1995a) and Ernst and Krishnamoorthy (1998a) provide enumeration based algorithms. The IP model of Ernst and Krishnamoorthy (1998a) is again different than the basic model. They use the same approach as they did in the single assignment case, and define:

Z^i_{kl} : the amount of flow of commodity i (traffic emanating from node i) that is routed between hubs k and l,

U^i_{lj} : the amount of flow of commodity i flowing from hub l to node j,

P_{ik} : flow from node i to hub k.

The corresponding model is:

$$\min \quad \sum_i [\sum_k P_{ik} c_{ik} + \sum_k \sum_l \alpha c_{kl} Z^i_{kl} + \sum_l \sum_j c_{lj} U^i_{lj}]$$

s.t

$$\begin{aligned}
\sum_k P_{ik} &= O_i && \forall\, i \in N \\
\sum_l U^i_{lj} &= w_{ij} && \forall\, i, j, \in N \\
\sum_l Z^i_{kl} + \sum_j U^i_{kj} &= \sum_l Z^i_{lk} + P_{ik} && \forall\, i, k, \in N \\
P_{ik} &\leq O_i X_{kk} && \forall\, i, k \in N \\
U^i_{lj} &\leq w_{ij} X_{ll} && \forall\, i, j, l \in N \\
Z^i_{kl}, U^i_{kl}, P_{ik} &\geq 0 && \forall\, i, k, l \in N \\
(12), X_{kk} = 0/1 & &&
\end{aligned}$$

This new formulation requires $2n^3 + n^2 + n$ variables out of which n are binary and $1+n+3n^2+n^3$ linear constraints whereas traditional models require $(n^4 + n)$ variables and $1 + n^2 + 2n^3$ linear constraints. The authors also point out that, in terms of CPU time usage in CPLEX, the new formulation is much better than the existing models.

Chapter 4

Allocation Problem

In this chapter, we concentrate on the allocation phase of the p-hub median problem. Recall that the allocation problem is the problem of determining which demand node must be assigned to which hub(s) given that the hubs are at a fixed set of locations.

Having a good methodology to solve the allocation problem is extremely useful in searching for an optimal solution to the hub location problem since the allocation problem must be repeatedly solved for each choice of hub locations that is encountered at the nodes of an enumeration tree which seeks to optimize the location and allocation aspects jointly. Another reason for our focus on the allocation phase is that, in many real world problems, especially in airline or cargo delivery systems, the decisions on the hub locations is a top management decision which takes a considerable amount of time and money to change once implemented. If the problem parameters change in time, an immediate but less expensive reaction is to reallocate the demand centers without changing the locations of the hubs. Hence, there is ample justification to consider the allocation problem as a problem in its own right.

The complexity status of the allocation problem depends on which assignment rule is used. The allocation problem in the multiple assignment case is polynomially solvable. In the single assignment case the problem is NP-Hard for $p \geq 3$ while the case with $p = 2$ reduces to a minimum cut problem which is polynomially solvable (Sohn and Park 1997, 1998b). We strengthen the NP-Hardness result by proving that the allocation problem in single assignment case is NP-Hard even if the transport network is a star tree.

4.1 Solvability of the Allocation Problem

In this section, we analyze the solvability of the allocation problem for multiple assignment and single assignment cases.

First, we briefly discuss the polynomial solvability of the multi assignment case. The Multiple Assignment Allocation Problem can easily be solved by decomposing the problem for each pair of nodes and solving a shortest path problem for each pair where the paths are restricted to those that visit at least one hub. This can easily be done, as observed by Sohn and Park (1998a), by employing Floyd's (1962) all pairs shortest path algorithm and terminating at the pth step. Let H denote the set of nodes which are selected as hubs. The method begins with an initial n by n matrix $C^0 = [c^0_{ij}]$ where $c^0_{ij} = \infty$ if $i, j \notin H$; $c^0_{ij} = c_{ij}$ if either i or j is in H but not both; $c^0_{ij} = \alpha c_{ij}$ if $i, j \in H$. The initial matrix gives the path lengths when no intermediate nodes are visited by the path. Assume without loss of generality that the first p nodes $1, ..., p$ are the hub nodes. The kth step computes $C^k = [c^k_{ij}]$ from C^{k-1} by $c^k_{ij} = \min\{c^{k-1}_{ik} + c^{k-1}_{kj}, c^{k-1}_{ij}\}$. That is, c^k_{ij} is the length of a shortest path from i to j if the path is allowed to use any subset (including the null set) of the first k nodes as intermediate or end nodes, but not allowed to use any of the nodes $k+1, ..., p$ as intermediate nodes. Matrix C^p gives the all pairs shortest path lengths when all paths visit at least one hub. The time bound of the method is $O(pn^2)$. Note that the algorithm works on the cost network and the time bound of obtaining the cost network (i.e. c_{ij}'s for $i, j \in N$) is $O(nn'^2)$ if one uses Dijkstra's method on the transportation network once for each demand center.

For the single assignment case the problem is trivial if $p = 1$ and can be solved polynomially if $p = 2$. In the latter case, the problem becomes a minimum cut problem on a graph with $n + 2$ nodes with the two hubs denoting the source and the sink (Sohn & Park 1997). In the rest of this chapter we focus on the single assignment case for $p \geq 3$. We realize that this problem is equivalent to a restricted version of the Multimedian Location problem with Mutual Communication (MMC) which is a well known problem from the literature.

4.1.1 Multi-median Location Problem with Mutual Communication

In this section, we first define a location problem well known in the literature as the multi-median or the m-median problem with mutual communication (MMC) (discussed in Tansel et al. 1983b). Our interest in this problem comes from the fact that the single assignment hub allocation problem is equivalent to a special case of problem MMC as will be shown in the sequel. Problem MMC is defined as follows: Suppose given a connected undirected transportation network $\bar{G} = (\bar{V}, \bar{A})$ with node set $\bar{V} = \{\bar{v}_1, ..., \bar{v}_{\bar{n}}\}$ and arc set $\bar{A}$ where each arc $(\bar{v}_i, \bar{v}_j) \in \bar{A}$ is assigned a positive weight l_{ij} which represents its length. For each pair of nodes $\bar{v}_i$ and $\bar{v}_j$, let $d(\bar{v}_i, \bar{v}_j)$ denote the length of a shortest path connecting $\bar{v}_i$ and $\bar{v}_j$. Nodes in set $\bar{V}$ represent existing facilities. New facilities will be opened in order to serve these existing facilities and $f_{ji} \geq 0$ denotes the flow per period between new facility j and existing facility i. New facilities will also have interaction between themselves. Let $v_{jk} = v_{kj} \geq 0$ denote the flow per period between new facilities j and k. The problem is to choose the locations $x_1, ..., x_m$ of m new facilities on the nodes of $\bar{G}$ to serve existing facilities as well as other new facilities with minimum total transportation cost. An instance of this problem is defined by the data $\bar{G} = (\bar{V}, \bar{A}), \bar{V} = \{\bar{v}_1, ..., \bar{v}_{\bar{n}}\}, \{l_{ij} : (\bar{v}_i, \bar{v}_j) \in \bar{A}\}, \{f_{ij}\}, \{v_{ij}\}, M = \{1, ..., m\}, \bar{N} = \{1, ..., \bar{n}\}$ and usually represented by an auxiliary graph as in Figure 4.1. The graph has m nodes on the left, $\bar{n}$ nodes on the right, and arcs correspond to positive flows

$f_{ij}, i \in M, j \in \bar{N}$, and positive flows $v_{ij}, i, j \in M$.

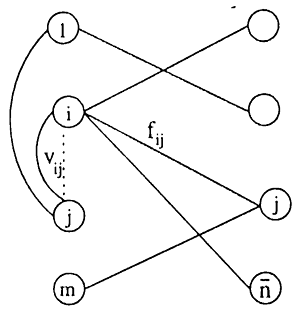


Figure 4.1: Flow Structure of MMC

The problem is formulated as :

$$\min_{x_1,\ldots,x_m \in \bar{V}} \sum_{j=1}^{m} \sum_{i=1}^{\bar{n}} f_{ji} d(x_j, \bar{v}_i) + \sum_{1 \le j < k \le m} v_{jk} d(x_j, x_k)$$

Kolen (1986) proves the problem is NP-Hard on general networks but gives necessary and sufficient conditions for optimality for the case of a tree network. Kolen's result leads to a polynomial time algorithm for MMC when the transport network is a tree.

Tamir (1993) defines the restricted MMC as the version of MMC where the locations of the new facilities are restricted to a subset of the node set instead of the whole node set. Let $Q \subseteq \bar{V}$ denote the set of available locations of new facilities. Then the restricted MMC is as follows:

$$\min_{x_1,\ldots,x_m \in Q} \sum_{j=1}^{m} \sum_{i=1}^{\bar{n}} f_{ji} d(x_j, \bar{v}_i) + \sum_{1 \le j < k \le m} v_{jk} d(x_j, x_k)$$

Tamir proves that the restricted MMC is NP-Hard even if $\bar{G}$ is a tree on 4 nodes. Thus, even though MMC on tree networks is polynomially solvable, the restricted MMC is NP-Hard on tree networks with $\bar{n} \ge 4$. However, the case in which the transport network is a path is still polynomially solvable in $O(m^3 + m^2 log q + \bar{n}m)$ time, where $q = |Q|$(Tamir 1993). The restricted MMC forms one of the rare cases in which polynomial solvability of the path problem does not extend to the tree network case.

Next, we define a special version of the restricted MMC which has not been defined in the literature before. We call this special version the Balanced

Restricted MMC and abbreviate it to BR-MMC. In BR-MMC, $m = \bar{n}$. In addition, $w_{ij} = 0$ for every i, j such that $i \neq j$ and $w_{ii} \geq 0$ for $i = 1, ..., \bar{n}$. The flow data $v_{jk}(= v_{kj})$ among new facilities are nonnegative constants which are required to satisfy the condition $w_{ii} = C \sum_{k=1}^{\bar{n}} v_{ik}$ where C is a positive constant. An instance of BR-MMC is defined by $\bar{G} = (\bar{V}, \bar{A}), \bar{V} = \{\bar{v}_1, ..., \bar{v}_{\bar{n}}\}, \{l_{ij} > 0 : (\bar{v}_i, \bar{v}_j) \in \bar{A}\}, \{v_{jk} \geq 0 : 1 \leq j < k \leq \bar{n}\}, C > 0, Q$ (the values of w_{ii} are induced by $w_{ii} = C \sum_k v_{ik}$).

Lemma 1: The allocation problem is polynomial time transformable to BR-MMC.

Proof : Given an instance of the allocation problem, we obtain an instance of BR-MMC as follows. Take $\bar{n} = n'$ and $C = \frac{1}{\alpha}$. The transport network $\bar{G} = (\bar{V}, \bar{A})$ of BR-MMC is taken to be the transport network $G = (N', E)$ of the allocation problem with $l_{ij} = c_{ij}$ and $d(\bar{v}_i, \bar{v}_j) = d_{ij}$. The flows v_{jk} are determined as: $v_{jk} = \alpha(f_{kj} + f_{jk}) \forall j, k \in N, v_{jk} = 0$ otherwise. Take $Q = H$. Then the restricted MMC obtained from the allocation problem becomes

$$\min_{x_1,...,x_n \in H} \sum_{i \in N} w_{ii} d(i, x_i) + \sum_{j,k \in N, j<k} v_{jk} d(x_j, x_k)$$

where $w_{ii} = \sum_{j \in N} (f_{ij} + f_{ji}) \forall i \in N$, and $w_{ii} = 0$ for $i \in N' \setminus N \square$.

Lemma 2: BR-MMC is polynomial time transformable to the allocation problem.

Proof: Given an instance of BR-MMC, an instance of the allocation problem is obtained as follows. Take $n' = \bar{n}$. Renumber the nodes of $\bar{G}$ so that $w_{ii} > 0$ for $i = 1, ..., n$ and $w_{ii} = 0$ for $i = n + 1, ..., n'$ where $n \leq n'$. The transport network of BR-MMC forms the transport network of the allocation problem with $c_{ij} = l_{ij} \forall (i, j) \in \bar{A}$. Let $\alpha = \frac{1}{C}$. The flows f_{ij} are determined by taking f_{ik} and f_{ki} to be any values that satisfy $\alpha(f_{ik} + f_{ki}) = v_{ik}$, e.g. $f_{ik} = f_{ki} = (1/2)(1/\alpha)v_{ik}$. Finally, we take $H = Q \square$.

Lemmas 1 and 2 imply that the problems BR-MMC and the allocation problem are special cases of each other. Then the following theorem follows:

Theorem 1 : The allocation problem is equivalent to BR-MMC.

4.1.2 Complexity of the Allocation Problem

Even though the NP-Hardness of the restricted MMC is well established (Tamir 1993), the complexity status of BR-MMC is not known (Kara and Tansel 1998 is the first study that considers BR-MMC due to its relation to the allocation problem). We first establish the NP-Hardness of BR-MMC then deduce the complexity of the allocation problem from it.

Theorem 2 : BR-MMC is NP-Hard even if the transportation network is a star tree with $|Q| = 3$.

Proof : We prove that BR-MMC is NP-Hard by showing that it is a generalization of the 3-multiway cut problem which is shown to be NP-Hard by Dalhaus et al. (1992). The 3-multiway cut problem is defined as follows: Given a graph $G^* = (V^*, E^*)$ and a set of 3 specified nodes x, y, and z in V^*, find a minimum cardinality subset of edges, $\bar{E}^*$, such that the removal of $\bar{E}^*$ from E^* disconnects each of the above three nodes from the other two.

Consider the 3-multiway cut problem. Suppose G^* has $l+4$ nodes and $V^* = \{v_1, v_2, v_3, y, z_1, ..., z_l\}$. The three specified nodes are v_1, v_2, and v_3. Every 3-multiway cut corresponds to a feasible solution to an instance of BR-MMC defined by the following:

- $\bar{V} = \{v_1, v_2, v_3, y, z_1, ..., z_l\}$, $Q = \{v_1, v_2, v_3\}$
- For $j, k \in N, v_{jk}$ is 1 iff nodes j and k are connected by an edge in G^* and 0 otherwise.
- Take $C = 1$. Then $w_{ii} = \sum_{j \in N} v_{ij}$ = the degree of node i.
- Transportation network $\bar{G} = (\bar{N}, \bar{A})$ is a star tree with y at the center with arc lengths defined by $l_{v_i y} = 0.5$ for $i = 1, 2, 3$ and $l_{z_i y} = \theta$ for $i = 1, ..., l$, where θ is an arbitrary positive constant.

With these parameter settings, a feasible solution to BR-MMC corresponds to a 3-multiway cut in the $l+4$ node graph G^*. In a feasible solution of BR-MMC, the set of new facilities $\{v_1, v_2, v_3, y, z_1, ..., z_l\}$ is partitioned into three subsets, say M_1, M_2 and M_3, such that each new facility in subset M_i is located at $v_i \in Q, i = 1, 2, 3$. From this solution of BR-MMC problem the corresponding solution to the 3-multiway cut problem is obtained as follows: Color the nodes in M_i (in the linkage network of BR-MMC) with color $i, (i = 1, 2, 3)$. Then, the arcs which have end points at different colored nodes are the ones that need to be removed from G^* to have the required disconnection in the 3-multiway cut problem. An optimum solution to the BR-MMC problem is also an optimal solution to the 3-multiway cut problem because the objective functions of both of the models differ by a constant which comes from the cost of the interaction between new and existing facilities in BR-MMC. That constant is $(0.5 \text{ degree}(y) + \sum_{z_i \in \{z_1,\dots,z_l\}} \text{degree}(z_i)(\theta + 0.5))$.

Then BR-MMC defined on a star tree with $|Q| = 3$ is a generalization of the 3-multiway cut problem which proves that BR-MMC is NP-Hard □.

This theorem mildly strengthens an earlier result of Tamir (1993) on the NP-Hardness of the restricted MMC.

From Theorems 1 and 2, the following corollary is established.

Corollary 2.1 : The allocation problem is NP-Hard for $p = 3$ even if the transportation network is a star tree.

This corollary strengthens the result of Sohn and Park (1998b) on the complexity of the allocation problem.

4.2 Polynomially Solvable Cases

The polynomial solvable cases are in two categories: the ones utilizing the flow data and the ones utilizing the transport network structure.

4.2.1 Utilizing Flow Data

Lemma 1 shows that the allocation phase of the p-hub location problem is reducible to the restricted MMC. Thus, we can make use of the results and algorithms which have been developed for the restricted MMC in the solution process of the allocation phase of the hub location problem. Even though the classical MMC is a well studied problem, the restricted version of it is recently defined by Tamir (1993) and has not received as much attention.

There are two polynomial algorithms for MMC problems which have a special structured 'linkage network' defined by v_{jk}'s. The *linkage network* is the auxiliary network $LN = (M, I)$ where $M = \{1, ..., m\}$ and $I = \{(j,k) : v_{jk} > 0\}$. One of the algorithms is applicable to series-parallel linkage networks (Chajjed and Lowe 1992) and the other one is applicable to linkage networks that are k-trees (Chajjed and Lowe 1994, Fernandez-Baca 1989). A *k-tree* is either a k-clique (i.e. a complete graph on k nodes) or a graph recursively constructed as follows: Given a k-tree and a subgraph of the k-tree which is a k-clique, the graph obtained by introducing a new node and connecting it to every node of the k-clique is again a k-tree. A partial k-tree is a subgraph of a k-tree. Series-parallel graphs are partial 2-trees.

These algorithms are enumeration based methods which optimize the location of a node conditional on all possible ways of locating its neighbors. Given a k-tree, it is possible to obtain an "elimination ordering", which sequentially eliminates nodes together with its incident arcs to obtain a k-clique. The algorithms proceed by eliminating nodes in this elimination ordering. For every new facility, there is a set of available positions for locations, (e.g the set $Q = H$ or the whole node set). At the elimination of each node, the best location for that node is found by enumerating all the alternatives for every possible combination of the positions of the node's neighbors. The complexity of such an algorithm is $O(m\bar{n}^{k+1})$ plus the time needed to obtain an elimination ordering. Since allocation phase of the p-hub location problem in the single assignment case is equivalent to this restricted MMC, that problem is also polynomially solvable in $O(np^{k+1})$ if the flow graph of w_{jk}'s is a k-tree.

4.2.2 Utilizing Transport Network Structure

Tamir (1993) gives the only result in the literature regarding polynomial solvability of the restricted MMC focusing on the structure of the transport network. He presents a polynomial time algorithm for the case when transport network is a path. He proves that the restricted MMC is polynomially solvable if the transport network is a path and gives an algorithm of $O(m^3 + m^2 logq + \bar{n}m), q = |Q|$. Since the allocation problem is polynomial time transformable to the restricted MMC, the allocation problem is also polynomially solvable if the transport network is a path. The nodes which are in set H can be renumbered as $1, ..., p$ in the order as they appear in the transport network starting from one end of the path. The algorithm is based on solving min-cut problems between hub nodes $l-1$ and l. A direct min-cut approach would require $O(np^3)$. Tamir (1993) improves this bound to $O(n^3+n^2logp+n^2)$ by applying a parametric approach for the solution of consecutive min-cut problems.

We further analyze the transport network structure in the allocation problem in the next section and identify more general cases which are also polynomially solvable.

4.3 Transport Network Structure in Solvability

Even though Corollary 2.1. proves that the allocation problem is NP-Hard on tree networks, we identify certain structures of the transport network for which the allocation problem is polynomially solvable. We also develop a decomposition theorem, again, based on the structure of the transportation network. All these results utilize a new definition, the so called allocation set.

The allocation set, A_i, of each node i, is to be constructed in such a way that, keeping the allocations of the nodes other than node i as constant, for each solution in which node i is allocated to a hub not in A_i, there exist another solution in which node i is allocated to a hub in A_i with objective value at least as good as the previous one. Then, in solving the allocation problem for the set of hubs among which the hub that node i is allocated to will be selected, we take into account the sets A_i for each node i, instead of set H.

In the rest of this chapter, we call the solutions in which any node is allocated to a hub which is not in its allocation set as "dominated solutions".

We now present a procedure to compute the *Allocation Sets* for the nonhub nodes.

- Apply Dijkstra's algorithm to find the shortest paths between each pair of hubs. Let $SPL(h_i, h_j)$ denote the length of a shortest path between hubs h_i and h_j.
- For each non hub node i,
  - Apply Dijsktra by taking node i as the source node. This will result in shortest paths from node i to every other node. Let $SPL(i, j)$ denote the length of a shortest path from node i to node j
  - Take any $h_j \in H$. If $SPL(i, h_j) < SPL(i, h_t) + SP(h_t, h_j)\ \forall\, h_t \in H$ then put h_j in set A_i. Repeat this step for each $h_j \in H$.

Observe that, the above procedure requires $O(p^2)$ for the first step and $O(n^2 + p^2)$ for each non hub node in the second step where $O(n^2)$ is needed for Dijkstra and $O(p^2)$ is for the last step. Thus, the time bound for computing all the A_i's is $O(n^3)$.

Example 1: For the transport network given in Figure 4.2, $A_i = \{h_1, h_2\}$

Next we prove that the solutions in which the allocation sets are formed by the above procedure are nondominated.

Theorem 3 : The solutions in which any demand center is allocated to a hub which is not in its allocation set are "dominated solutions".

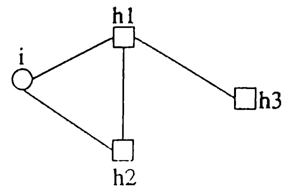
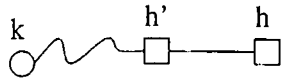


Figure 4.2: Example for Allocation Sets

Proof : We prove the result by taking a solution in which the allocation of some node, k is to a hub $h \notin A_k$. We construct another solution in which the allocation of node k is changed to a hub in its allocation set which has at least as good objective value as the old one.

Let I be a solution in which node k is allocated to $h \notin A_k$. Since $h \notin A_k$, there is a node $h' \in H$ such that h' is on a shortest path between node k and h (see figure below).



Now consider the solution, II, in which node k is allocated to h' and all the other allocations are the same as in I.
Let Δ = Cost(I) - Cost(II), and let $a(i)$ denote the hub to which i is allocated.

$$\Delta = \sum_{j \in N} (w_{jk} + w_{kj})\{c_{k\,h} + \alpha c_{h\,a(j)} + c_{a(j)\,j} - c_{k\,h'} - \alpha c_{h'\,a(j)} - c_{a(j)\,j}\}$$

Since $c_{k\,h} = c_{k\,h'} + c_{h'\,h}$,

$$\Delta = \sum_{j \in N} (w_{jk} + w_{kj})(c_{h'\,h} + \alpha c_{h\,a(j)} - \alpha c_{h'\,a(j)})$$

Since triangle inequality is satisfied and $\alpha \geq 0$ we have:

$$\alpha c_{h'\,a(j)} \leq \alpha c_{h'\,h} + \alpha c_{h\,a(j)}$$

Then

$$\Delta \geq \sum_{j \in N} (w_{jk} + w_{kj})\{c_{h\,h'} + \alpha c_{h\,a(j)} - \alpha c_{h'\,h} - \alpha c_{h\,a(j)}\}$$

which is

$$\geq \sum_{j \in N} (w_{jk} + w_{kj})(1 - \alpha) c_{h\,h'} \geq 0.$$

Instead of solution I, we can use II whose objective is at least as good as I. Then the solutions in which demand centers are allocated to hubs which are not in their allocation sets are "dominated solutions". □

Corollary 3.1 : Each demand center j can be allocated to a hub in its allocation set without loss of optimality.

In view of Corollary 3.1, we identify some structures for the transport network for which the allocation problem is polynomially solvable. We start the analysis with the simplest case: transport network being a path.

4.3.1 Transport Network Path

First note that, if transport network is a path the allocation set of each demand center contains 1 or 2 hubs.

We explain our results by going over an example. Take an example with 10 nodes 3 of which are selected as hubs. The transport network is given in Figure 4.3.

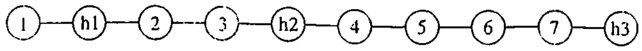


Figure 4.3: Example With Path Transport Network

$A_1 = \{h_1\}, A_2 = A_3 = \{h_1, h_2\}, A_4 = A_5 = A_6 = A_7 = \{h_2, h_3\}$.

The allocation of node 1 is trivial. It will be allocated to h_1 (Corollary 3.1).

Now, consider node 2. According to Corollary 3.1. it will be allocated to either h_1 or h_2.

Case 1: Node 2 is allocated to h_1. The cost of the allocation induced by node 2 is:

$$\sum_{j\in N} w_{2j}c_{2h_1} + w_{23}\alpha c_{h_1\,a(3)} + \sum_{j\in\{4,5,6,7\}} w_{2j}\alpha c_{h_1\,a(j)} \quad (27')$$

Note that for $j \in \{4,5,6,7\}$, $c_{h_1\,a(j)} = c_{h_1\,h_2} + c_{h_2\,a(j)}$. Then, the cost of this allocation is:

$$\sum_{j \in N} w_{2j} c_{2\,h_1} + w_{23} \alpha c_{h_1\,a(3)} + \sum_{j \in \{4,5,6,7\}} w_{2j} \alpha c_{h_1\,h_2} + \sum_{j \in \{4,5,6,7\}} w_{2j} \alpha c_{h_2\,a(j)} \quad (27)$$

Case 2: Node 2 is allocated to h_2, the cost of the allocation induced by node 2 is:

$$\sum_{j \in N} w_{2j} c_{2\,h_2} + w_{21} \alpha c_{h_1 h_2} + w_{23} \alpha c_{h_2\,a(3)} + \sum_{j \in \{4,5,6,7\}} w_{2j} \alpha c_{h_2\,a(j)} \quad (28)$$

Note that the components $w_{23} \alpha c_{h_1\,a(3)}$ and $w_{23} \alpha c_{h_2\,a(3)}$ are either 0 or $w_{23} \alpha c_{h_1\,h_2}$ depending on the allocations of nodes 2 and 3.

We define a minimum-cut problem by assigning cost components given in (27) and (28) to arc capacities. The resulting graph is given in Figure 4.4.

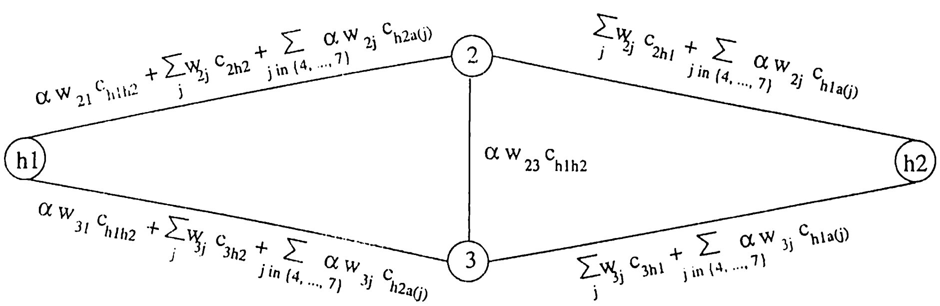


Figure 4.4: Min-Cut Structure for 2 Nodes

If we solve a minimum-cut problem between h_1 and h_2 on the graph given in Figure 4.4, the solution will provide us with the optimum allocations of nodes 2 and 3. The nodes will be allocated to the hub that they are connected to after the removal of cut edges. However, we need to know the values of $a(j)$ in order to solve the constructed min-cut problem since the arc capacities are dependent on $a(j)$. Consider the residual graph of the graph given in Figure 4.4 when

$\sum_{j \in \{4,\ldots,7\}} w_{2j} \alpha c_{h_2\,a(j)}$ is sent through path $h_1 \to 2 \to h_2$

$\sum_{j \in \{4,\ldots,7\}} w_{3j} \alpha c_{h_2\,a(j)}$ is sent through path $h_1 \to 3 \to h_2$

In that case, $a(j)$ for $j \in \{4,5,6,7\}$ disappears from the arc capacities. Thus, we can find the allocations of nodes 2 and 3 without knowing the exact allocation of nodes in set $\{4,5,6,7\}$. This leads to the following theorem.

Theorem 4 : The allocation problem decomposes into independent minimum cut problems if the transport network is a path.

Let L_j denote the nodes on the *left* of node j (nodes 1 to $j-1$), and R_j denote the nodes on the *right* of node j (nodes j+1 to n). Without loss of generality we assume that nodes are renumbered in the order that they appear in the transport network from 1 to n. Then, the generic minimum-cut problem to solve the allocation problem between hubs h_l and h_{l+1} is given in Figure 4.5.

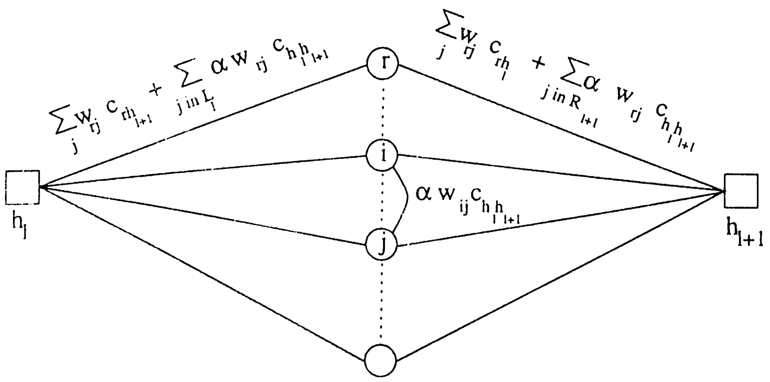


Figure 4.5: Structure of the Generic Min-Cut Problem

The polynomial solvability of the allocation problem on a path transport network can also be deduced from Theorem 4. However, Theorem 4 can be utilized in more general structures as will be shown in the next section.

4.3.2 First Generalization : (Block Graph Path)

Suppose that the transport network has the structure shown in Figure 4.6.

The allocation sets of the nodes $1, ..., 7$ are the same as the ones given in Figure 4.3 (transport network path case). If we were to write the cost of allocating node 2 to h_1 or h_2, we will end up with the same expressions given

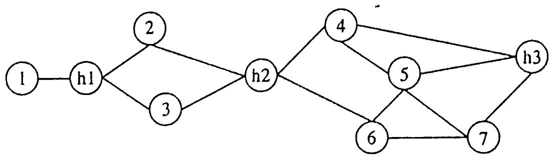


Figure 4.6: Transport Network Structure

in (27) and (28). Thus we can make use of the same idea that we have used for the path case in solving the allocation problem defined on a transport network given in Figure 4.6. We, first, need to define the concept, block.

A *block* is a maximal set of nodes whose allocation sets are the same. In the example, we have 3 blocks $B_1 = \{1\}, B_2 = \{2,3\}, B_3 = \{4,5,6,7\}$.

We define a *block node*, b_i for each block B_i, and let $B = \{b_1, b_2, ..., b_r\}$ where r is the number of blocks.

We define a *block graph* $G_B = (N_B, A_B)$ corresponding to a transport network $G = (N, E)$ as follows.
We take the block nodes and hubs as the nodes of the block graph. We put arcs between two hub nodes if that arc already exists in the transport network and we put arcs between a block node and a hub node if the hub node is in the allocation set of the nodes in the corresponding block. i.e. $G_B = (N_B, A_B)$ where

$$N_B = H \cup B \quad \text{and} \quad A_B = \{(i,j) : \begin{array}{ll} & i,j \in H \text{ and } (i,j) \in E \\ \text{or} & i \in H, j \in B \text{ and } i \in A_j \\ \text{or} & i \in B, j \in H \text{ and } j \in A_i\} \end{array}$$

The block graph of the transport network given in Figure 4.6 is:

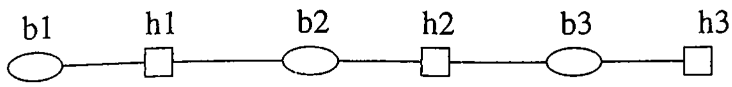


Figure 4.7: Block Graph of Figure 4.6

Note that if we draw the block graph of the transport network given in Figure 4.3., we would end up with the same graph.

Our min cut based decomposition is valid for the cases where the block graph of the transportation network is a path. This leads to the following theorem:

Theorem 5 : The allocation problem decomposes into independent minimum cut problems if the block graph of the transport network is a path.

Proof: Directly follows from Theorems 3 and 4.

The following corollary immediately follows.

Corollary 5.1: The allocation problem is polynomially solvable if the block graph is a path.

We remark here that, if the transport network is a path then so is the block graph but the reverse is not true.

In the next subsection we take into account the cases where the block graph is a tree.

4.3.3 Generalization 2: (Block Graph Tree)

We first need to verify the validity of the decomposition idea. Can we solve the allocation problem on each block separately? The answer is yes due to the following theorem.

Theorem 6 : Let $N = B_1 \cup B_2 \cup \ldots B_r$. Let $A(B_i)$ denote the set of hubs in the allocation sets of nodes in B_i $((A(B_i) = \{\cup A_k : k \in B_i\}))$. The problem decomposes into subproblems each associated with a given block if for each pair of blocks B_i and B_j there is a cut vertex $t_{ij} \in B$ between $A(B_i)$ and $A(B_j)$ so that the following condition is satisfied:

$$c_{a(k)\,a(l)} = c_{a(k)\,t_{ij}} + c_{t_{ij}\,a(l)} \quad k \in B_i, l \in B_j \qquad (29)$$

Proof: With $O_i = \sum_j w_{ij}$, and $D_i = \sum_j w_{ji}$, the allocation problem can be written as:

$$(P) \quad \min_{a(i)\in H, i\in N} \sum_{l=1}^{r} \Big[\sum_{i\in B_l} (O_i + D_i) c_{i\,a(i)} + \sum_{i,j\in B_l} \alpha w_{ij} c_{a(i)\,a(j)} +$$

$$\sum_{k=1,k\neq l}^{r} \sum_{i\in B_l, j\in B_k} \alpha w_{ij} c_{a(i)\,a(j)} \big) \Big]$$

where the first term is the linear cost of allocating node i to $a(i)$, the second term is the cost of interaction between two nodes of the same block, and the third term is the cost of interaction between two nodes of different blocks. Let

$$F(B_l) = \sum_{i\in B_l} (O_i + D_i) c_{i\,a(i)} + \sum_{i,j\in B_l} \alpha w_{ij} c_{a(i)\,a(j)}.$$

Using (29), (P) can be rewritten as:

$$(P') \quad \min_{a(i)\in H, i\in N} \Big[\sum_{l=1}^{r} \Big(F(B_l) + \sum_{k=1,k\neq l}^{r} \sum_{i\in B_l, j\in B_k} \alpha w_{ij} (c_{a(i)\,t_{lk}} + c_{t_{lk}\,a(j)}) \Big) \Big]$$

which is equivalent to:

$$\min_{a(i)\in H \forall i\in N} \Big[\sum_{l=1}^{r} \Big(F(B_l) + \sum_{k=1,k\neq l}^{r} \sum_{i\in B_l} \alpha c_{a(i)\,t_{lk}} \sum_{j\in B_k} (w_{ij} + w_{ji}) \Big) \Big].$$

Observe here that, the above form of the cost function is; as if there is a fictitious hub node at the point t_{lk} and all the flow between node $i \in B_l$ and any node $j \in B_k$ is processed at this fictitious hub. Thus, as long as we know the location of the fictitious hub, which is t_{lk}, we can find the optimal allocations of nodes in set B_l without knowing the exact allocations of the nodes in the other blocks. Let

$$G(B_l) = F(B_l) + \sum_{k=1,k\neq l}^{r} \sum_{i\in B_l} \alpha c_{a(i)\,t_{lk}} \sum_{j\in B_k} (w_{ij} + w_{ji})$$

Then, we have:

$$(P'') \quad \min_{a(i)\in H, i\in N} \sum_{l} G(B_l).$$

Since $N = B_1 \cup B_2 \cup \ldots B_r$ we can write (P'') as:

$$(P^*) \quad \min_{a(i)\in A(B_t)\forall i\in B_t; t=1,\ldots,r} \sum_{l=1}^{r} G(B_l)$$

which is equivalent to

$$(P^*) \quad \sum_{l=1}^{r} \min_{a(i)\in A(B_l)\forall i\in B_l} G(B_l).$$

Thus, we have shown that, instead of minimizing (P), we can solve (P^*) which minimizes the cost of each subset B_l seperately and adds up the result. This completes the proof □.

Corollary 6.1 : The decomposition theorem is satisfied if the block graph is a tree.

Proof: The result immediately follows since each node of a tree is a cut vertex and a cut vertex exists between each pair of nodes. □.

Example 2: Consider the allocation problem defined on a transport network which lead to the block graph given in Figure 4.8.

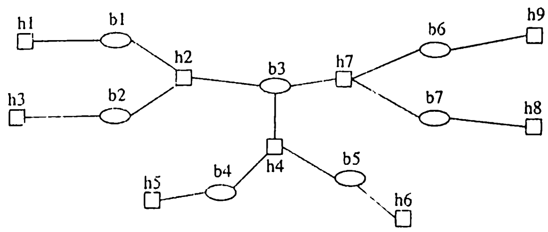


Figure 4.8: Example for Decomposition

Take B_1 and B_7. We have $A(B_1) = \{h_1, h_2\}$ and $A(B_7) = \{h_7, h_8\}$ and a cut vertex between $A(B_1)$ and $A(B_7)$ is h_2. Since a seperating cut vertex can be found for every pair of blocks, the problem decomposes. Thus, we solve the allocation problem for subproblems corresponding to each block, $B_1, ..., B_7$ separately. Note that the allocation sets for nodes in all blocks except B_3 have two hubs. The subproblems for those nodes can be solved via minimum-cut algorithms by using the generic graph defined in Figure 4.5.

However, the allocation sets of the nodes in B_3 have 3 hubs h_2, h_4, h_7. Thus, for the nodes in this set, we need to solve a 3-way cut problem.

As a result, for block graph being a tree, the decomposition theorem applies and each block can be solved independently. For each block B_i, we need to solve a k-way cut problem where k is the degree of node b_i in the block graph. Thus, with a tree structured block graph, the allocation problem is polynomially solvable if the degree of all block nodes are less than or equal to 2. (Since 2-way cut problem is polynomially solvable.) Otherwise, decomposition still applies but polynomial solvability is no longer valid since k-way cut problem is NP-Hard for $k \geq 3$ (Dalhaus et al. 1992).

We remark here that the proof of theorem 6 still applies, if we change the condition (29) to (29′) given below.

$$c_{a(k)a(l)} = c_{ka(k)} + \gamma_{kl} + c_{la(l)} \quad \forall k \in B_i, l \in B_j \quad (29')$$

where γ_{kl} is a constant. The requirement is, in fact, to be able to write the cost of going from one hub to the other as the sum of the costs of going from nodes to hubs and a constant term, which can be any number. In the proof, we need to add a constant term which is $\sum_{l}\sum_{k \neq l} w_{kl}\gamma_{kl}$ for the expression of $G(B_l)$ which does not change the results. The following corollary is then established.

Corollary 6.2: The decomposition theorem is satisfied if condition (29) is replaced with (29′).

The following theorem summarizes the obtained results.

Theorem 7: If the decomposition theorem is satisfied, for each block B_i, we need to solve a k_i-way cut problem where k_i is the degree of the block node b_i.

For the cases when the problem cannot be solved polynomially, the decomposition theorem still helps a lot since it breakdowns the initial problem into many smaller sized subproblems. The following example highlights that property.

Example 3 : Suppose that $n = 38, p = 9$ with the following transport network.

When certain nodes are selected as hubs, we have the structure given in Figure 4.10 whose block graph is given in Figure 4.11:

Note that Theorem 6 is satisfied and the problem decomposes for each

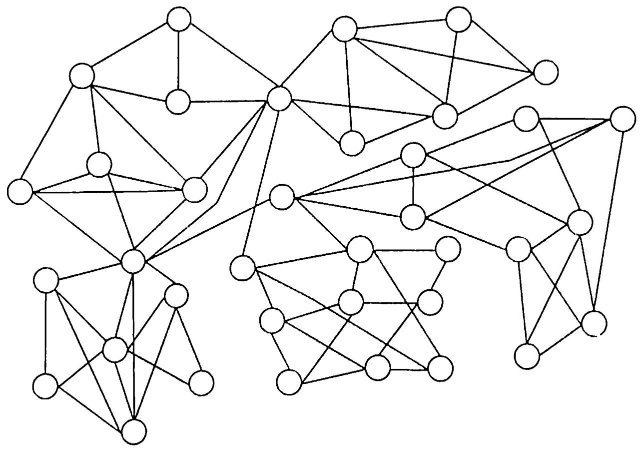


Figure 4.9: Transport Network of Example 3

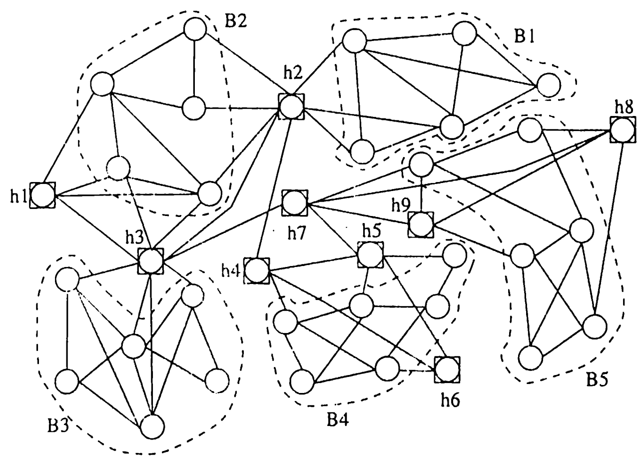


Figure 4.10: Blocks of Example 3

block. For example the cut vertex between $A(B_3)$ and $A(B_4)$ is $h4$ and that of $A(B_1)$ and $A(B_5)$ is $h7$.

Then, instead of solving an allocation problem with 38 nodes and 9 hubs we need to solve 5 independent subproblems:

$$SP1 : B1 : n = 5, p = 1 \quad SP2 : B2 : n = 5, p = 3$$
$$SP3 : B3 : n = 6, p = 1 \quad SP4 : B4 : n = 6, p = 2$$
$$SP5 : B5 : n = 6, p = 3$$

The allocation problem of SP1 and SP3 are trivial and SP4 is polynomially

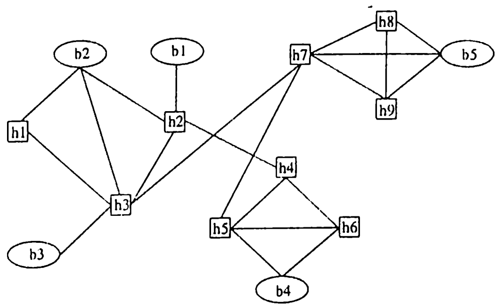


Figure 4.11: Block Graph of Example 3

solvable. Thus, we end up with 2 smaller sized problems ($n = 5, 6, p = 3$) instead of ($n = 38, p = 9$).

This example highlights the importance of the decomposition theorem. Even if the whole problem cannot be solved to optimality in polynomial time, decomposition is still important as it may greatly reduce the size of the initial problem. Since the decomposition theorem utilizes the structure of the transport network, we conclude that the distinction of the transport network in hub location problem leads to interesting and fruitful results.

Chapter 5

p-Hub Center and Hub Covering Problems

As mentioned before, the literature on hub location is mainly focused on the problem with the total cost criterion. There are only two studies in the literature that deal with other performance measures. O'Kelly and Miller (1991) give initial motivation for the minimax criterion in hub location problem in the context of cargo delivery. In the same paper, the special case with $p = 1$ is shown to be equivalent to the well known 1-center location problem in which a single facility is to be located to minimize the maximum distance to the users of the facility. The second paper that deals with different performance measures in hub location is Campbell (1994a) in which the author gives integer programming formulations for 4 different hub location problems: the p-hub median, uncapacitated hub location, p-hub center, and hub covering problems. These four problems are defined analogous to the well known facility location problems: p-median, uncapacitated facility location, p-center, and covering problems. Among these four, the literature is mainly focused on the p-hub median problem. The structure of the uncapacitated hub location problem is very similar to the p-hub median problem and the models that are developed for the p-hub median problem can be applied to the uncapacitated hub location problem with minor changes. However, the p-hub center and hub covering problems are completely different and deserve special attention. The p-hub center problem involves locating a fixed number, p, of hubs to minimize the maximum travel time between origin destination pairs. The hub covering problem involves the decision on the number of hubs and their locations so that the time of journey between each origin destination pair is within a specified bound. Campbell (1994a) gives quadratic binary programs for the p-hub center and hub covering problems which we refer to as the basic models in the sequel. Campbell also gives linearizations for the basic models, but he does not report any computational results. In this chapter, we focus on these problems in the single assignment case. We first provide combinatorial formulations of the p-hub center and hub covering problems and prove that both are NP-Hard. We then develop new models for these problems whose computational performance is far more superior than the linearizations of the basic models provided by Campbell (1994a).

5.1 Complexity

Campbell (1994a) observes that minimax is a well justified performance measure when 'time' is of concern. We also follow his convention and interpret c_{ij} as time.

The *p-hub center problem* involves the decision of locations of hubs and the allocation of demand nodes to hubs so as to minimize the maximum travel time between any origin destination pair. Thus, the p-hub center problem is:

$$\min_{\substack{H \subset N, \\ |H|=p}} \min_{a \in H^n} \max_{\substack{i,j \in N, \\ i<j}} (c_{i\,a(i)} + \alpha c_{a(i)\,a(j)} + c_{a(j)\,j}) \quad (30)$$

The *hub covering* problem is closely related to the p-hub center problem. An origin-destination pair, i, j, is said to be *covered* by hubs $a(i)$ and $a(j)$ if the travel time from i to j via the hubs $a(i)$ and $a(j)$ is less than a predetermined upper bound, say β. The hub covering problem seeks to minimize the number of hubs so that each origin destination pair is covered. That is, the hub covering problem is:

$$\begin{aligned} \min_{\substack{H \subset N, \\ a \in H^n}} \quad & |H| \\ \text{s.t.} \quad & c_{i\,a(i)} + \alpha c_{a(i)\,a(j)} + c_{a(j)\,j} \leq \beta \quad \forall i, j \in N, i < j \quad (31) \end{aligned}$$

We now state the recognition form of the p-hub center problem which is also the same as the recognition form of the hub covering problem :
Given an undirected network $G = (N, E)$ with node set $N = \{1, ..., n\}$ and with edge lengths $d_{ij} > 0, (i, j) \in E$, a rational α in the unit interval, a positive rational β, and a positive integer $p\,(1 \leq p \leq n-1)$, does there exist a subset H of N consisting of at most p nodes and an assignment vector $a = (a(1), ..., a(n)) \in H^n$ such that $c_{i\,a(i)} + \alpha c_{a(i)\,a(j)} + c_{a(j)\,j} \leq \beta$ for $1 \leq i < j \leq n$?

Theorem 8: The recognition form of the p-hub center problem for $p < n - 1$ is NP-Complete even if $\alpha = 0$ and $G = (N, E)$ is a planar graph with unit arc lengths and maximum degree three.

Proof: The theorem will be proved by reduction from the dominating set problem.

Dominating Set Problem: Given a connected undirected graph $\bar{G} = (\bar{N}, \bar{E})$ and a positive integer $k \leq |\bar{N}|$, does there exists a subset X of $\bar{N}$ with $|X| \leq k$ such that every node not in X is adjacent to at least one node in X, i.e. $\forall u \in \bar{N} \backslash X \; \exists \; v \in X$ for which $(u, v) \in \bar{E}$?

We note that the dominating set problem is NP-Complete even if $\bar{G}$ is planar with maximum degree 3 (Garey and Johnson 1979).

Clearly, the recognition form of the p-hub center problem is in class NP. Consider an instance of the dominating set problem. We reduce it to the p-hub center problem as follows: Take $N = \bar{N}, E = \bar{E}, d_{ij} = 1 \; \forall (i, j) \in E, p = k, \alpha = 0, \beta = 2$.

We first prove that if X solves the dominating set problem, then X also solves the created instance of the p-hub center problem. To prove the claim, take $H = X$ and construct an assignment vector $a = (a(1), ..., a(n))$ where, for each $i \in N$, $a(i)$ is a closest node in H to i. The constructed solution (H, a) satisfies $|H| \leq k = p$ and $c_{i\,a(i)} + \alpha c_{a(i)\,a(j)} + c_{a(j)\,j} \leq 2$ since $\alpha = 0$ and H is a dominating set so that $c_{i\,a(i)} \leq 1 \; \forall i \in N$. Conversely, if (H, a) solves the created instance of the p-hub center problem, then $X = H$ solves the dominating set problem. To prove the claim, suppose there is a node i which is not adjacent to any $h \in H$. Then, the distance of node i to a closest member of H is at least 2. Since $p < n - 1$ there is at least one other node $j \notin H, j \neq i$, so that $c_{i\,a(i)} + \alpha c_{a(i)\,a(j)} + c_{a(j)\,j} \geq 2 + 0 + 1 = 3$ contradicting that (H, a) is a feasible solution to the created instance of the p-hub center problem. Note also that $|H| \leq p = k$.

Hence, the dominating set problem has a YES answer if and only if the corresponding instance of the p-hub center problem has a YES answer. □

Since the recognition form of the p-hub center problem is NP-Complete, we might say that the optimization form for $p < n - 1$ is NP-Hard.

Since the recognition form of the hub covering problem is the same as the recognition form of the p-hub center problem when the upperbound on $|H|$ is p, the following corollary follows.

Corollary 8.1: The recognition form of the hub covering problem is NP-Complete. So, we might say that the optimization form is NP-Hard.

We now provide basic formulations and new models of the p-hub center and hub covering problems.

5.2 The p-hub center problem

5.2.1 Basic Model and Linearizations

In this section we first give the original integer programming formulation of Campbell(1994a). Using the variable definitions of Section 3.2, the basic formulation of the p-hub center problem is:

$$\text{(p-HC1)} \quad \min \max_{i,j,k,m} \quad X_{ik}X_{jm}(c_{ik} + \alpha c_{km} + c_{jm})$$
$$\text{s.t} \quad (5) - (8)$$

We now give the linearization of (p-HC1) proposed by Campbell (1994a). Let X_{ijkm} be a binary variable which takes on the value 1 if the path from origin i to destination j is via hubs k and m ($i \to k \to m \to j$) and 0 otherwise. The linearization proposed by Campbell is :

$$
\begin{array}{llll}
\text{(LIN1)} & \min & Z & \\
& \text{s.t} & & \\
& & Z \geq X_{ijkm}(c_{ik} + \alpha c_{km} + c_{jm}) & \forall\, i,j,k,m \quad (32)\\
& & \sum_k \sum_m X_{ijkm} = 1 & \forall\, i,j \quad (33)\\
& & \sum_j \sum_m (w_{ij}X_{ijkm} + w_{ji}X_{jimk}) = \sum_j (w_{ij} + w_{ji})X_{ik} & \forall\, i,k \quad (34)\\
& & X_{ijkm} \in \{0,1\} & \forall\, i,j,k,m \quad (35)\\
& & \text{and constraints } (5) - (8) &
\end{array}
$$

Constraints (33) and (35) ensure that there is exactly one pair of hubs (k, m) on the path from origin i to destination j ($k = m$ is possible). Constraint (34) is the constraint that correctly relates the path variables X_{ijkm} to the allocation variables X_{ik}. The right hand side of (34) is the total flow originating and ending at node i provided that i is allocated to a hub at node k.When $X_{ik} = 1$, the left side of (34) achieves the same total flow by summing all the incoming and outgoing flows on all paths each of which includes a shortest path between i and k as a subpath. Note also that when $X_{ik} = 0$, such path variables are forced to take on the value zero. We refer to the above formulation as (LIN1).

In linearizing the problem, it is desired that $X_{ijkm} = 1$ if and only if $X_{ik} =$

$X_{jm} = 1$. This is accomplished by constraint (34) in the above linearization. The same thing can be achieved by using the constraints (9) and (10) given in Section 3.3.1. as was done previously by Skorin-Kapov et al. (1996) for the p-hub median problem. Imposing the constraints (9) and (10) together with the zero/one requirement on the variables X_{ijkm} makes constraints (33) and (34) redundant. We refer to the linearization obtained from (LIN1) by replacing (33) and (34) with (9) and (10) as (LIN2).

We now propose a third linearization, called (LIN3), which we obtain from (LIN2) by replacing (9) and (10) with constraint (36) below and by replacing the zero/one requirement on the variables X_{ijkm} by $X_{ijkm} \geq 0 \, \forall i, j, k, m$.

$$X_{ijkm} \geq X_{ik} + X_{jm} - 1 \quad \forall i, j, k, m \quad (36)$$

Note that integrality on X_{ijkm} variables is not necessary in (LIN3), because the objective function and constraints (32) and (36) force X_{ijkm} variables to take on their lowest possible values which is either one or zero.

Then, the three linearizations are :

$$\text{(LIN1)} \quad \min \; Z \quad \text{s.t.} \; (32) - (35), (5) - (8)$$

$$\text{(LIN2)} \quad \min \; Z \quad \text{s.t.} \; (32), (9), (10), (35), (5) - (8)$$

$$\text{(LIN3)} \quad \min \; Z \quad \text{s.t.} \; (32), (36), (5) - (8)$$

We test these linearizations with the CAB1 data using CPLEX 5.0. Table 5.1. provides the solved instances within the time bound. We put a time bound of 15 hours.

(LIN1) has a poor computational performance as it has not been able to solve any of the 60 instances within the 15 hour limit. (LIN2) has limited success as it has been able to solve, within the 15 hour limit, only 10 of the

Model	(n, p) Combination Solved		Max. CPU
(LIN1)	None		-
(LIN2)	n = 10	p = 2, 3	14.5 hr
(LIN3)	n = 10	p = 2, 3, 4	1.1 hr
	n = 15	p = 2	13.6 hr

Table 5.1: Computational Performance of the Linearizations of (p-HC1)

60 instances corresponding to all values of α for $n = 10$ and $p = 2, 3$. The maximum CPU time of (LIN2) for the solved 10 instances is 14.45 hours. (LIN3) has a better performance. It has been able to solve the 10 instances that have also been solved by (LIN2) within a maximum time of 40.3 minutes, thus achieving about a 20-fold reduction in CPU time. In addition, it has been able to solve the 5 instances corresponding to $n = 10$ and $p = 4$ within 1.1 hour. The largest problem size that can be solved by (LIN3) is $n = 15$ for $p = 2$ (the cases p = 3, 4 are not solved within the 15 hour limit). All the 5 instances corresponding to $(n, p) = (15, 2)$ has been solved by (LIN3) within the 15 hour limit where the maximum CPU time is 13.6 hours.

As can be seen from the reported results, (LIN3) has the best performance among the three linearizations but with limited success. The largest problem size it can handle is $n = 15$ with $p = 2$ while none of the instances with larger n can be solved by (LIN3) regardless of p. In the next section we reformulate the p-hub center problem from a different perspective. The resulting model solves, for example, the $(n, p) = (15, 2)$ combination in the order of a few minutes while (LIN3) spends almost 13.5 hours to solve the same combination. Substantial improvement has also been obtained from the new model for larger sized problems.

5.2.2 New Model for the p-hub Center Problem

Define now a real variable R_{ij} which stands for the travel time from node i to node j via the two hubs to which i and j are assigned. Let $R_{ij} = L_{ir} + c_{rj}$ where L_{ir} is another real variable which stands for the travel time from origin

i to node r under the assumption that node j is assigned to a hub at node r. In order to ensure that the real variables R_{ij}'s and L_{ir}'s take on the correct values we impose the constraints

$$L_{ir} = \sum_k (c_{ik} + \alpha c_{rk}) X_{ik} \quad (37)$$

$$R_{ij} = \sum_r (L_{ir} + c_{rj}) X_{jr} \quad (38)$$

With the single assignment constraint (11), there is exactly one k for which $X_{ik} = 1$ and exactly one r for which $X_{jr} = 1$ so that (37) and (38) supply the correct values for L_{ir} and R_{ij}.

The new model, which we call p-HC2′, is as follows:

$$\begin{array}{llrcll} & \min & Z & & & \\ (\text{p-HC2}') & \text{s.t.} & Z & \geq & R_{ij}\ \forall i,j & (39) \\ & & (5) & - & (8),\ (37),\ (38). & \end{array}$$

(p-HC2′) is a nonlinear mixed integer program with $2n^2 + 1$ real variables and n^2 binary variables. The nonlinearity is due to constraint (38).

We may eliminate the real variables R_{ij} and L_{ir} from (p-HC2′) to obtain a simplified model which retains the binary variables and the real variable Z. Observe that, because of the single assignment constraint, the summation operator in (38) can be replaced by the maximum operator. With this and using the right side of (37) for L_{ir}, we have:

$$R_{ij} = \max_r \left[c_{rj} + \sum_k (c_{ik} + \alpha c_{kr}) X_{ik} \right] X_{jr} \quad (40)$$

Using (40), it is direct to replace (39) by

$$Z \geq \left[c_{rj} + \sum_k (c_{ik} + \alpha c_{kr}) X_{ik} \right] X_{jr}\ \forall r \text{ and } \forall i,j \quad (41)$$

The simplified model which we refer to as (p-HC2) is:

$$\begin{array}{lll} (\text{p-HC2}) & \min & Z \\ & \text{s.t.} & (41),\ (5) - (8) \end{array}$$

(p-HC2) is a nonlinear mixed integer program with one real and n^2 binary variables. The number of constraints is $n^3 + n^2 + n + 1$. The nonlinearity is due to constraint (41).

Observation 2: $Z \geq \sum_k (c_{ik}+\alpha c_{kr})X_{ik}+c_{jr}X_{jr} \ \forall i,j,r$ (42) correctly linearizes the constraint (41).

Proof: There are 2 cases to consider depending on the value of X_{jr}. Let s be the index for which $X_{is} = 1$. Then $\sum_k (c_{ik} + \alpha c_{kr})X_{ik} = c_{is} + \alpha c_{sr}$ both in (41) and (42).

- Case 1: $X_{jr} = 1$: Then $Z \geq c_{is} + \alpha c_{sr} + c_{jr}$ which is the time of journey between nodes i and j when i is assigned to a hub at node s and j is assigned to a hub at node r. Hence, the right sides of (41) and (42) are identical for the pair i, j in this case. $\surd$
- Case 2: $X_{jr} = 0$: In this case (42) yields $Z \geq c_{is} + \alpha c_{sr}$ while (41) yields $Z \geq 0$. If $X_{rr} = 1$, then for i, r, r (41) and (42) both yield $Z \geq c_{is} + \alpha c_{sr} + c_{rr}$. Hence $Z \geq c_{is} + \alpha c_{sr}$ is an implied constraint. If $X_{rr} = 0$, then due to constraint (5), there exists an index k such that $X_{rk} = 1$. For i, r, k (41) and (42) both yield $Z \geq c_{is} + \alpha c_{sk} + c_{kr}$. Since $0 \leq \alpha \leq 1$ and triangle inequality is assumed, $Z \geq c_{is} + \alpha c_{sr}$ is implied. $\surd$ $\square$.

The linearized version of (p-HC2), referred to as (LinNew), is as follows:

$$\text{(LinNew)} \quad \min \ Z$$
$$\text{s.t.} \quad (42),(5)\text{-}(8)$$

Note that the linearization does not change the number of variables and constraints of (p-HC2). Thus, (LinNew) requires n^2 zero/one variables and $n^3 + n^2 + n + 1$ constraints. Table 5.2 provides the number of variables and constraints for all the models of the p-hub center problem.

As can be seen from Table 5.2. the linearization of our new model, (LinNew), is best in terms of core storage requirements. It requires only n^2 binary variables whereas the other linearizations require at least n^4 binary variables. Substantial improvement is also pronounced for the CPU time usage of the linearizations. We test the computational performance of (LinNew) using 80 instances generated from the CAB2 Data and we present the CPU times

Model	Variables 0/1	Variables Real	Constraints
(LIN1)	$n^4 + n^2$	1	$n^4 + 3n^2 + 1$
(LIN2)	$n^4 + n^2$	1	$n^4 + 2n^3 + n^2 + n + 1$
(LIN3)	n^4	$n^2 + 1$	$2n^4 + n^2 + n + 1$
(LinNew)	n^2	1	$n^3 + n^2 + n + 1$

Table 5.2: Summarized Information of the Linearizations for p-hub Center

reported by CPLEX 5.0 for each of the 80 instances in Table 5.3. The last two columns of the table provide the averages and maxima over p for each setting of n. The reported CPU times are in seconds for $n = 10$ and 15, in minutes for $n = 20$, and in hours for $n = 25$.

As can be seen from Table 5.3, in comparison to (LIN3) which solves $(n, p) = (15, 2)$ in a maximum CPU time of 13.6 hours, (LinNew) solves the same combination in a maximum CPU time of 3.5 minutes. This shows that the computational performance of the new model is significantly better than all three linearizations of the basic model.

This significant improvement is also detected in the larger problem sizes. For example, while the linearizations of the basic model cannot solve the problems with $n = 15, p \geq 3$ within the 15 hour limit, the linearization of the new model solves these instances in a matter of about 5 minutes. Additionally, the 15 hour limit has not been encountered by the new model for the large problem instances $n = 20$ and 25. For $n = 20$, the maximum CPU time of the linearization of the new model is a little over 1 hour while the average time is about half an hour. For $n = 25$, the average and maximum times go up to 5.4 and 11.3 hours, respectively. This shows that the exponential behavior of the solution time becomes pronounced after $n \geq 20$.

Thus, the performance of the new model is significantly better than all the linearizations of the basic model in terms of both CPU time usage and core storage requirements. In the next section we analyze the hub covering problem which is closely related with the p-hub center problem as observed in Section 5.1.

n	α	p=2	3	4	5	Avg.	Max.
10	0.2	8.0	8.1	6.1	7.4		
	0.4	6.4	4.0	2.6	2.9		
	0.6	4.5	5.9	2.5	3.7	in secs.	
	0.8	2.4	5.5	4.0	1.2		
	1.0	1.8	4.4	3.3	1.4	4.3	8.1
15	0.2	211.8	313.2	311.8	238.2		
	0.4	124.3	180.6	137.3	62.2		
	0.6	16.3	25.9	77.4	77.4	in secs.	
	0.8	20.0	20.5	17.8	35.1		
	1.0	23.7	15.3	13.4	6 .3	96.4	313.2
20	0.2	43.4	62.2	69.2	45.4		
	0.4	35.5	55.6	56.6	34.4		
	0.6	23.3	36.1	21.8	15.4	in mins.	
	0.8	13.0	21.4	11.6	27.6		
	1.0	2.4	0.9	6.8	1.9	29.2	69.2
25	0.2	3.8	8.1	10.2	7.1		
	0.4	4.0	7.5	11.3	8.5		
	0.6	3.0	8.2	7.9	8.2	in hrs.	
	0.8	1.9	4.2	4.4	5.4		
	1.0	0.8	0.5	1.9	1.8	5.4	11.3

Table 5.3: CPU Times for (LinNew)

5.3 The Hub Covering Problem

5.3.1 Basic Model and Linearizations

Using the variable definitions of Section 3.2, for the hub covering problem, Campbell (1994a) defines

$$V_{ijkm} = \begin{cases} 1 & \text{if the time of travel from node } i \text{ to node } j \text{ via } k, m \text{ (in that order)} \\ & \text{is no more than the time bound(i.e. } c_{ik} + \alpha c_{km} + c_{jm} \leq \beta) \\ 0 & \text{otherwise} \end{cases}$$

Then, the formulation of the hub covering problem proposed by Campbell

(1994a) is:

$$\begin{array}{lll} \text{(HC1)} & \min & \sum_{k} X_{kk} \\ & \text{s.t} & \\ & \sum_{k,m} V_{ijkm} X_{ik} X_{jm} \geq 1 \quad \forall i,j & (43) \\ & (5),(7),(8) & \end{array}$$

Constraint (43) ensures that every origin destination pair is covered by some k and m. The parameter V_{ijkm} taking on the value of 0 and 1 determines if the pair (i,j) can be covered by (k,m). Campbell linearizes (HC1) in the same way as he linearizes (p-HC1). The linearization of (HC1) proposed by Campbell (1994a) is:

$$\begin{array}{lll} \text{(C-LIN1)} & \min & \sum_{k} X_{kk} \\ & \text{s.t} & \\ & & V_{ijkm} X_{ijkm} \geq 1 \quad \forall i,j,k,m \quad (44) \\ & & (34),(5),(7),(8), X_{ijkm} \geq 0 \end{array}$$

where (34) is the constraint that Campbell used for correctly relating the X_{ijkm} variables with the X_{ik} variables. We refer to the above formulation as (C-LIN1). We also develop different linearizations for the model (HC1), just as we did for the p-hub center case. We again end up with 3 different linearizations:

$$\begin{array}{lll} \text{(C-LIN1)} & \min & \sum_{k} X_{kk} \\ & \text{s.t.} & (34),(44),(5),(7),(8), X_{ijkm} \geq 0 \end{array}$$

$$\begin{array}{lll} \text{(C-LIN2)} & \min & \sum_{k} X_{kk} \\ & \text{s.t.} & (15),(16),(44),(5),(7),(8), X_{ijkm} \geq 0 \end{array}$$

$$\begin{array}{lll} \text{(C-LIN3)} & \min & \sum_{k} X_{kk} \\ & \text{s.t.} & (36),(44),(5),(7),(8), X_{ijkm} \geq 0 \end{array}$$

In all the linearizations, there are n^2 binary, n^4 real variables, while there are $n^4 + 2n^2 + n$ constraints in (C-LIN1), $n^4 + 2n^3 + n^2 + n$ constraints in (C-LIN2), and $2n^4 + n^2 + n$ constraints in (C-LIN3).

We test these linearizations with the CAB2 data using CPLEX 5.0. In the hub covering problem, p is a decision variable instead of a parameter. Additionally, we have an upperbound, β, defining the cover for each origin/destination pair. The choice of β determines which of the coefficients V_{ijkm} are 1 and which of them are zero. From the computational study that we have conducted for the p-hub center problem with the CAB2 Data set, we have the optimum costs for 80 instances. We use these costs as the β values. We take the objective value Z^* of the p-hub center solution as the β value for the related (n, α) combination. Since we have tested 4 different p values, we have 4 different Z^* values for each (n, α) combination. The optimal p-hub center objective values Z^* are given in Table 5.4 below.

	$n = 10$					$n = 15$				
α	0.2	0.4	0.6	0.8	1.0	0.2	0.4	0.6	0.8	1.0
	1425	1627	1671	1744	1839	2004	2019	2103	2424	2611
	1117	1185	1387	1589	1791	1638	1741	1844	2165	2610
	811	970	1148	1457	1770	1324	1435	1756	2100	2605
	736	863	1079	1413	1766	1149	1287	1560	2080	2600
	$n = 20$					$n = 25$				
	1851	2067	2255	2493	2611	2114	2401	2557	2713	2826
	1549	1744	1996	2264	2605	1913	2087	2336	2552	2762
	1356	1473	1835	2154	2601	1617	1881	2184	2457	2726
	1162	1386	1663	2118	2600	1319	1597	2002	2307	2725

Table 5.4: Bound Values Used for Test Problems

We solve the models (C-LIN1), (C-LIN2), and (C-LIN3) with the explained CAB2 Data set using CPLEX 5.0. Table 5.5 provides the summary information of the computational performance of the three linearizations. An upper limit of 8.5 hours is imposed on the CPU time.

Within the 8.5 hour time bound, all instances corresponding to $n = 10$ are solved to optimality with all the three linearizations. The average (maximum) CPU times encountered are 4.3 (11.6), 1.45 (3.7), and 0.99 (2.2) minutes for (C-LIN1), (C-LIN2), and (C-LIN3), respectively. For $n = 15$ all the 20 instances are solved to optimality by (C-LIN2) and (C-LIN3) with an average of 1.5

Model	Combination Solved	Avg. CPU	Max. CPU
(C-LIN1)	n = 10 ,all	4.3sec.	11.6sec
	n = 15 , 3 missed	2.5 hrs.	6.5 hrs
(C-LIN2)	n = 10 , all	1.45 min	3.7 min
	n = 15 , all	1.5 hr	4.5 hr
(C-LIN3)	n = 10 , all	0.99 min	2.2 min
	n = 15 , all	1.3 hr	3.4 hr
	n = 20 , p = 2	7.5 hr	8.5 hr

Table 5.5: Computational Performance of the Linearizations of (HC-1)

and 1.3 hours, respectively. With (C-LIN1) 17 out of 20 instances are solved to optimality within the time limit (the remaining 3 instances required 10-12 hours). For $n = 20$, 5 out of 20 instances, all of which resulted in a p^* of 2, are solved to optimality with (C-LIN3) in the time limit. The remaining 15 instances for $n = 20$ and all instances of $n = 25$ required more time than the limit. With (C-LIN1) and (C-LIN2), none of the instances of $n = 20$ and 25 are solved within the time limit.

In the next section, we propose a new model for the hub covering problem whose computational success is orders of magnitude better than any of the linearizations of the basic models that we considered.

5.3.2 New Model for the Hub Covering Problem

In this new model, we do not define the parameter V_{ijkm}, and instead, we put new covering constraints into the model. The new model, which we call (HC2), is as follows:

$$
\begin{array}{llll}
\text{(HC2)} & \min & \sum_{k} X_{kk} & \\
& \text{s.t.} & [(c_{ir} + \alpha c_{rk})X_{ir} + c_{jk}]X_{jk} \leq \beta \; \forall i,j,k,r & (45) \\
& & (5),(7),(8) &
\end{array}
$$

HC2 is a nonlinear mixed integer program with n^2 binary variables and $n^4 + n^2 + n$ constraints.

Observation 3: $(c_{ir} + \alpha c_{rk})X_{ir} + c_{jk}X_{jk} \leq \beta \forall i, j, k, r$ (46)
correctly linearizes the constraint (45).

Proof : There are 4 cases to consider depending on the values of X_{ir} and X_{jk}.

- Case 1: $X_{ir} = 1, X_{jk} = 1$: Then (45) and (46) yield the same lefthand-sides. $\surd$.
- Case 2: $X_{ir} = 1, X_{jk} = 0$: Then (46) yields
$$c_{ir} + \alpha c_{rk} \leq \beta \quad (47)$$
while (45) yields $0 \leq \beta$. If $X_{kk} = 1$, then for i, k, r, k (45) and (46) both yield $c_{ir} + \alpha c_{rk} + c_{kk} \leq \beta$. Hence, (47) is an implied constraint. $\surd$.
If $X_{kk} = 0$, then due to constraint (5), there exists and index l such that $X_{kl} = 1$. For i, k, r, l (45) and (46) both yield $c_{ir} + \alpha c_{rl} + c_{lk} \leq \beta$. Since $0 \leq \alpha \leq 1$, and triangle inequality is assumed, $c_{ir} + \alpha c_{rk} \leq \beta$ is, again, implied. $\surd$
- Case 3: $X_{ir} = 0, X_{jk} = 1$: This case gives $c_{jk} \leq \beta$. Since $X_{ir} = 0$, due to constraint (5), there exists an index $l \neq r$ such that $X_{il} = 1$ and for i, j, l, k case 1 will be encountered: $\beta \geq c_{il} + \alpha c_{lk} + c_{jk} \geq c_{jk}$ and the constraint is ineffective. $\surd$
- Case 4: $X_{ir} = 0, X_{jk} = 0$: In this case both (45) and (46) yield $0 \leq \beta$ $\surd\square$.

The linearized version of (HC2), referred to as (C-LinNew), is as follows:

$$\text{(C-LinNew)} \quad \min \quad \sum_k X_{kk}$$
$$\text{s.t.} \quad (46),(5),(7),(8)$$

Note that the linearization does not effect the number of variables and constraints.

We test the computational performance of (C-LinNew) by using 80 instances generated from the CAB2 Data set corresponding to the same combinations of (n, α, Z) described in Section 5.3.1. In Table 5.6, we present the

CPU times reported by CPLEX 5.0 for each of the 80 instances. All the values are in seconds.

	$n = 10$			$n = 15$			$n = 20$			$n = 25$		
α	Value	p^*	CPU	Value	p^*	CPU	Value	p^*	CPU	Value	p^*	CPU
	1425	2	4.5	2004	2	26.1	1851	2	307.0	2114	2	370.1
0.2	1117	3	4.5	1638	3	250.6	1549	3	1102.1	1913	3	5140.5
	811	4	0.4	1324	4	83.0	1356	4	84.2	1617	4	2814.8
	736	5	0.3	1149	5	4.2	1162	5	50.5	1319	5	263.3
	1627	2	3.2	2019	2	11.1	2067	2	91.9	2401	2	999.3
0.4	1185	3	0.7	1741	3	2.3	1744	3	68.4	2087	3	221.4
	970	4	0.3	1435	5	2.2	1473	4	42.0	1881	4	568.2
	863	5	0.2	1287	5	1.4	1386	5	16.0	1597	5	50.3
	1671	2	1.9	2103	2	12.6	2255	2	123.6	2557	2	181.7
0.6	1387	3	0.5	1844	3	1.8	1996	3	82.3	2336	3	284.5
	1148	4	0.2	1756	4	1.6	1835	4	78.6	2184	4	368.0
	1079	5	0.2	1560	5	1.5	1663	5	8.3	2002	5	165.8
	1744	2	0.6	2424	2	7.0	2493	2	13.0	2713	2	113.4
0.8	1589	3	0.5	2165	3	3.9	2264	3	25.2	2552	3	69.4
	1457	4	0.2	2100	4	1.6	2154	4	8.0	2457	4	486.0
	1413	5	0.2	2080	4	1.4	2118	5	5.7	2307	5	31.8
	1839	1	0.4	2611	1	2.6	2611	1	11.2	2826	2	31.8
1.0	1791	3	0.4	2610	3	6.5	2605	3	13.3	2762	3	36.2
	1770	4	0.5	2605	3	3.5	2601	3	12.0	2726	4	33.4
	1760	4	0.3	2600	3	4.5	2600	3	11.6	2725	5	27.3
	Avg.		1.0			21.5			107.7			612.9
	Max.		4.5			250.6			1102.1			5140.5

Table 5.6: CPU times for (C-LinNew)

Note that with the linearizations of the basic model, (C-LIN1), (C-LIN2), and (C-LIN3), the largest size that can be solved within the 8.5 hr time limit is $n = 15$ (and 5 of the 20 instances of $n = 20$). Using the linearization of our new model, (C-LinNew), we solved all the 80 instances corresponding to all n combinations in a maximum time of 1.5 hours. As can be seen from Table 5.6, the instances with $n = 10$ are solved within a few seconds whereas the best linearization of the basic model requires 2.2 minutes for this size. The improvement is more pronounced with increasing n. For $n = 15$, (C-LIN3) requires 3.4 hours whereas, with (C-LinNew) the same size can be solved within

4.5 minutes. For $n > 20$ the linearizations of the basic model do not report any solved instance within the 8.5 hr time limit whereas with the new model, the instances corresponding to $n = 20$ are solved within 18.5 minutes and the ones corresponding to $n = 25$ are solved within 1.5 hour. This shows that the computational performance of the new model is far more superior than the linearizations of the basic model.

5.3.3 Generalization to Distance Constraints

In the hub covering problem, we defined the notion of covering by having a hub restricted travel time less than a predetermined upper bound, β, for every origin destination pair. However, it might be the case that every origin destination pair has different upper bounds on the journey times, namely, β_{ij}. This time, the hub covering problem will be

$$\begin{array}{lll} \text{(DC1)} & \min & \sum_k X_{kk} \\ & \text{s.t.} & [(c_{ir} + \alpha c_{rk})X_{ir} + c_{jk}]X_{jk} \leq \beta_{ij} \ \forall i, j, k, r \quad (48) \\ & & (5), (7), (8) \end{array}$$

Observe here that we cannot directly change β to β_{ij} in the linear constraint (46) of the hub covering problem since during the proof of Observation 3 (that we can replace nonlinear constraint (45) with (46)) we use the fact that the upperbound on the constraints is the same, β, for each pair. In model (DC1) each pair has a different upper bound and so we cannot utilize Observation 3.

Observation 4: $(c_{ir} + \alpha c_{rk})X_{ir} + c_{jk}X_{jk} - \alpha(1 - X_{jk})c_{jk} \leq \beta_{ij} \ \forall i, j, r, k$ (49) correctly linearizes the constraint (48).

Proof: There are 4 cases to consider depending on the values of X_{ir} and X_{jk}.

- Case 1: $X_{ir} = 1, X_{jk} = 1$: Then $c_{ir} + \alpha c_{rk} + c_{jk}$ is the time of journey between nodes i and j when i is assigned to a hub at node r and j is assigned to a hub at node k. Hence, the left sides of (48) and (49) are identical for the pair i, j in this case. $\surd$

- Case 2: $X_{ir} = 1, X_{jk} = 0$: This case gives $c_{ir} + \alpha c_{rk} - \alpha c_{jk} \leq \beta_{ij}$ in (49) while it gives $0 \leq \beta_{ij}$ in (48). If $X_{kk} = 1$, then for i, r, k, k (48) and (49) both yield $c_{ir} + \alpha c_{rk} \leq \beta_{ij}$. Thus, $c_{ir} + \alpha c_{rk} - \alpha c_{jk} \leq \beta_{ij}$ is an implied constraint. $\surd$
  If $X_{kk} = 0$, then constraint (5)implies that there exists and index l such that $X_{kl} = 1$. For i, j, r, l both (48) and (49) yield $c_{ir} + \alpha c_{rl} + c_{lj} \leq \beta_{ij}$. Note that

  $$\beta_{ij} \geq c_{ir} + \alpha c_{rl} + c_{jl} \geq c_{ir} + \alpha c_{rl} + \alpha c_{jl} + \alpha c_{jk} - \alpha c_{jk} \geq c_{ir} + \alpha c_{rk} - \alpha c_{jk}$$

  and so the constraint $c_{ir} + \alpha c_{rk} - \alpha c_{jk} \leq \beta_{ij}$ is implied . $\surd$

- Case 3: $X_{ir} = 0, X_{jk} = 1$: This case gives $c_{jk} \leq \beta_{ij}$ in (49) and $0 \leq \beta_{ij}$ in (48). Since $X_{ir} = 0, \exists\, l \neq r$ s.t. $X_{il} = 1$ and for i, j, l, k case 1 will be encountered: $\beta_{ij} \geq c_{il} + \alpha c_{lk} + c_{jk} \geq c_{jk}$ and hence, $c_{jk} \leq \beta_{ij}$ is ineffective. $\surd$

- Case 4: $X_{ir} = 0, X_{jk} = 0$: This case gives $-\alpha c_{jk} \leq \beta_{ij}$ in (49) and $0 \leq \beta_{ij}$ in (48). Since $-\alpha c_{jk} \leq 0$, the constraint is ineffective. $\surd\square$.

The linearized version of DC1, referred to as (DC-Lin), is as follows:

$$\begin{array}{lll} \text{(DC-Lin)} & \min & \sum_k X_{kk} \\ & \text{s.t.} & (49), (5), (7), (8) \end{array}$$

Note that the linearization does not effect the number of variables and constraints.

The above analysis shows that even if there are different time bounds for the delivery times between different origin-destination pairs, the model (DC-Lin) can be used.

Chapter 6

Latest Arrival Hub Location Problem

In this chapter, we analyze the problem structure of the 'time sensitive' cargo delivery systems which are explained in Section 2.2. Recall that, for cargo delivery firms, the objective is to minimize the arrival time of the latest arrival at any destination. This is a minimax type objective but is different from the objective of the p-hub center problem, which is also minimax, since the travel time in the p-hub center problem is not the actual arrival time. In the real world, the actual delivery time from an origin i to a destination j consists of two components: flight times and the transient times spent at hubs between flights. In the models of the p-hub center problem, the transient times are not taken into account. This chapter proposes a new model which correctly computes the delivery times. We refer to the resulting model as the *latest arrival hub location* problem. Depending on the structure of the objective function, minimax, covering, and minisum versions for the latest arrival hub location problem can be distinguished. Our primary focus is on the minimax version. We study various aspects of this problem including model development, linearization, computational aspects, and sensitivity analysis of the model output.

6.1 Model Development

Using the variable and parameter definitions of Section 3.2, for a specified location, assignment pair (H, a), denote by $T_{ij}(H, a)$ the *total time spent during delivery* from i to j via the hubs $a(i), a(j) \in H$. Thus, the arrival time of the units originating at i destined to go to j is $r_i + T_{ij}(H, a)$ where r_i is the ready time of the outgoing cargo from city i. We note here that, although the packages at a given city i are collected at different times during the day, they can all be assigned a common ready time, r_i, which is the flight departure time from city i. We assume that there is a positive flow, $w_{ij} > 0$, from every origin i to every destination j. We term this assumption the *full cross-traffic assumption.* This seems to be a reasonable assumption for cargo delivery systems. The total delivery time, $T_{ij}(H, a)$ is the sum of the total flight time and the total transient time; that is,

$$T_{ij}(H, a) = c_{i\,a(i)} + \alpha c_{a(i)\,a(j)} + c_{a(j)\,j} + \tau_{ij}(a(i)) + \tau_{ij}(a(j)) \qquad (50)$$

where $\tau_{ij}(a(i))$ and $\tau_{ij}(a(j))$ are, respectively, the transient times at hubs $a(i)$ and $a(j)$ of the units going from i to j. An expression for computing the $\tau_{ij}(.)$ values, and hence, the $T_{ij}(H, a)$ values, in terms of the input data $r = (r_i, i \in N)$ and $C = [c_{ij}]$ will be derived subsequently. The minimax, covering, and minisum versions of the latest arrival hub location problem in *implicit* forms are as follows:

1. $\min\limits_{\substack{H \subset N, \\ |H| = p}} \min\limits_{a \in H^n} \max\limits_{\substack{i,j \in N \\ i<j}} r_i + T_{ij}(H, a)$ (51)
2. $\min\limits_{\substack{H \subset N, \\ a \in H^n}} |H|$

   s.t. $r_i + T_{ij}(H, a) \leq \beta \; \forall i, j \in N, i < j$ (52)
3. $\min\limits_{\substack{H \subset N, \\ |H| = p}} \min\limits_{a \in H^n} \sum\limits_{i,j \in N} w_{ij} T_{ij}(H, a)$ (53)

We now derive an algebraic expression for $T_{ij}(H, a)$. Denote by DT_{pq} the departure time of a flight going from node p to node q. For nonhub origins i, $DT_{i\,a(i)} = r_i$. To compute $T_{ij}(H, a)$, consider the journey from i to j via the hubs $a(i)$ and $a(j)$. All units going from i to j experience a flight time of

$c_{i\,a(i)}$ during the first segment of this journey. The transient time at $a(i)$ is the departure minus the arrival time of these units. That is,

$$\tau_{ij}(a(i)) = DT_{a(i)\,a(j)} - (r_i + c_{i\,a(i)}) \quad (54)$$

To correctly compute the departure time $DT_{a(i)\,a(j)}$, observe that the aircraft going from $a(i)$ to $a(j)$ transports not only those units that come from i but also the units that come from other nonhub origins that are also serviced from $a(i)$. Note however that the triangle inequality on $C = [c_{ij}]$ implies that this aircraft does not transport the units that come from other hubs. Accordingly, $DT_{a(i)\,a(j)}$ is the latest of the arrivals from nonhub origins to $a(i)$. Hence,

$$DT_{a(i)\,a(j)} = \max_{k:a(k)=a(i)} (r_k + c_{k\,a(i)}) \quad (55)$$

Observe from (55) that $DT_{a(i)\,a(j)}$ is, in fact, independent of $a(j)$. Hence the departure time from hub $a(i)$ is the same regardless of which hub the aircraft is flying to. This is true under the assumption of full cross-traffic. If this assumption is not satisfied, (55) must be written as:

$$DT_{a(i)\,a(j)} = \max_{k \in I_{a(i)\,a(j)}} (r_k + c_{ka(j)}) \quad (55')$$

where $I_{a(i)\,a(j)}$ is the set of origins k such that $a(k) = a(i)$ and $w_{kl} > 0$ for some l for which $a(l) = a(j)$.

The units going from i to j, together with other units that are serviced via the hub pair $(a(i), a(j))$, experience a common flight time of $\alpha c_{a(i)\,a(j)}$. The transient time at $a(j)$ is:

$$\tau_{ij}(a(j)) = DT_{a(j)\,j} - (DT_{a(i)\,a(j)} + \alpha c_{a(i)\,a(j)}) \quad (56)$$

Here, $DT_{a(j)\,j}$ is determined by the latest of the arriving units at $a(j)$ that are destined to go to j. A unit that is destined to go from an arbitrary node k to node j arrives at $a(j)$ at time $DT_{a(k)\,a(j)} + \alpha c_{a(k)\,a(j)}$. Hence,

$$DT_{a(j)\,j} = \max_{h \in H} (DT_{h\,a(j)} + \alpha c_{h\,a(j)}) \quad (57)$$

Substituting the right hand side of (55) for $DT_{h\,a(j)}$, we have

$$DT_{a(j)\,j} = \max_{h \in H} \left(\alpha c_{h\,a(j)} + \max_{k:a(k)=h} (r_k + c_{kh})\right) \quad (58)$$

Observe from (58) that $DT_{a(j)j}$ is in fact, independent of the destination j. This is again true under the assumption of full cross-traffic. $\tau_{ij}(a(j))$ in expression (56) is now computable given the values of $DT_{a(j)\,j}$ in (58) and of $DT_{a(i)\,a(j)}$ in (55). Substituting the computed forms of $\tau_{ij}(a(i))$ and $\tau_{ij}(a(j))$ in (50) and cancelling out like terms, $T_{ij}(H,a)$ reduces to

$$T_{ij}(H,a) = c_{a(j)\,j} + \max_{h \in H} \left[\alpha c_{h\,a(j)} + \max_{k:a(k)=h} (r_k + c_{kh}) \right] - r_i \quad (59)$$

Using (59) and dropping the constant term $\sum_{i \in N} w_{ij} r_i$ from the objective function in (62), the *explicit* forms of the minimax, covering, and minisum latest arrival hub location problems are as follows:

$$\min_{\substack{H \subset N, \\ |H|=p}} \min_{a \in H^n} \max_{j \in N} \left(c_{a(j)\,j} + \max_{h \in H} \left[\alpha c_{h\,a(j)} + \max_{k:a(k)=h} (r_k + c_{kh}) \right] \right) \quad (60)$$

$$\begin{aligned} \min_{\substack{H \subset N, \\ a \in H^n}} \quad & |H| \\ \text{s.t.} \quad & c_{a(j)\,j} + \max_{h \in H} \left[\alpha c_{h\,a(j)} + \max_{k:a(k)=h} (r_k + c_{kh}) \right] \le \beta \quad \forall j \end{aligned} \quad (61)$$

$$\min_{\substack{H \subset N, \\ |H|=p}} \min_{a \in H^n} \sum_{j \in N} W_j \left(c_{a(j)\,j} + \max_{h \in H} \left[\alpha c_{h\,a(j)} + \max_{k:a(k)=h} (r_k + c_{kh}) \right] \right) \quad (62)$$

where $W_j = \sum_i w_{ij}$ is the total flow into j .

Note that in the implicit form of the minimax problem defined in (51), the maximum is taken over all index pairs $i, j \in N \text{x} N, i < j$, whereas in the explicit form defined in (60) the maximum is taken on the index $j \in N$ alone. This is justified by the fact that the arrival time at node j is not dependent on the originating index i, i.e. regardless of the ready times, all units from different origins that are destined to go to node j arrive at node j at the same time. Similarly, in the explicit form of the covering problem in (61), one upper bound constraint is written for each index $j \in N$ whereas in the implicit form in (52) one constraint is written for each index pair $i, j \in N \text{x} N, i < j$. This again follows from the fact that the arrival time at node j (the left side of (61)) is not dependent on the originating index i which is true regardless of the ready times. Similarly, with the omission of the constant term $\sum_{i \in N} w_{ij} r_i$ from the objective function of the minisum problem, the summation of the explicit form in (62) is on the index j alone whereas the summation is over all index pairs in the implicit form defined by (53).

Hence, the explicit forms reduce the number of terms in the maximand, constraints, or the summation from n^2 to n. This helps to obtain greatly reduced and compact integer programming formulations for these problems. Additionally, the input requirement in (62) is reduced from an n by n flow matrix $W = [w_{ij}]$ to an n vector $(W_1, ..., W_n)$ which is much easier to obtain from the annual inflow records of the local stations than having to keep track of the cross-traffic on the entire network. The independence property from the originating indices seems to be a unique feature of the latest arrival hub location problem but is not observable in the traditionally studied hub location problems.

6.2 The Minimax Latest Arrival Hub Location Problem - Complexity

We now focus on the minimax latest arrival hub location problem, abbreviated from now on to *minmaxlatest*. We first show that it is NP-Hard.

Theorem 9: Minmaxlatest is NP-Hard.

Proof: To prove this, take $\alpha = 0$ and $r_i = 0 \, \forall i \in N$. With $\alpha = 0$ the $\alpha c_{h\,a(j)}$ term in (60) disappears and the innermost two maximizations output a value $g(H, a) = \max_{h \in H} \max_{k: a(k) = h} c_{kh}$ which depends only on the hub set H and the assignment vector a, but not on the index j. It is direct now to conclude that, for fixed H, assigning each node j to a closest hub in H is optimal. To see this, let a^* be such an assignment vector. For any other assignment $a \in H^n$, $c_{a^*(k)k} \leq c_{a(k)k} \; \forall k$. Hence, $g(H, a^*) \leq g(H, a)$ and consequently,

$$\max_{j \in N} c_{a^*(j)j} + g(H, a^*) \leq \max_{j \in N} c_{a(j)j} + g(H, a).$$

It follows that a^* is an optimal assignment. Hence (60) reduces to

$$\begin{aligned} & \min_{\substack{H \subset N, \\ |H|=p}} \max_{j \in N} c_{a^*(j)\,j} + g(H, a^*) \\ = & \min_{\substack{H \subset N, \\ |H|=p}} \max_{j \in N} 2(\min_{h \in H} c_{jh}) \end{aligned}$$

which is the node restricted p-center problem on a complete graph K_n with arc weights c_{ij}, $i, j \in N$. Hence, minmaxlatest is a special case of the p-center problem. It is well known that the p-center problem is NP-Hard (Kariv and Hakimi 1979), implying that the minmaxlatest is also NP-Hard even if $\alpha = 0$ and $r_i = 0 \, \forall i \in N$. □

Observe also that the recognition form of the minmaxlatest asks :
Given $\alpha \in [0, 1]$, an n by n matrix C with symmetric positive rational entries c_{ij} that satisfy the triangle inequality, rationals $r_i, i = 1, ..., n$, a positive integer $p \, (1 \leq p < n)$, and a positive rational β, does there exist a subset H of $\{1, ..., n\}$ with $|H| \leq p$ and an assignment vector $a \in H^n$ such that

$$\max_{j \in N} c_{a(j)\,j} + \max_{h \in H} \left[\alpha c_{h\,a(j)} + \max_{k:a(k)=h} (r_k + c_{kh}) \right] \leq \beta \quad ?$$

This is the same as the recognition form of the covering version of the latest arrival hub location problem when the upperbound on $|H|$ is p. The following corollary immediately follows.

Corollary 9.1. The recognition form of the covering version of the latest arrival hub location problem is NP-Complete. So, we might say that the optimization form of the covering version is NP-Hard.

6.3 IP Formulations

In this section we give integer programming formulations for the minimax version of the latest arrival hub location problem. Recall from (55) that the departure times from a hub h towards all other hubs are the same. Recall also from (58) that the departure times from a hub h towards all cities that are serviced from h are, again, the same. The following observation immediately follows:

Observation 5: At any hub h, there are two different departure times: the departure time for aircrafts that are destined to go to other hubs, and, the departure time for aircrafts that are destined to nonhub destinations.

Let $\hat{DT}_h$ and DT_h denote these two departure times, respectively. Using (55) and (58), we have :

$$\hat{DT}_h = \max_{k:a(k)=h}(r_k + c_{kh}) \quad (63)$$
$$DT_h = \max_{k \in H}(\hat{DT}_k + \alpha c_{kh}) \quad (64)$$

Using the variable and parameter definitions of Section 3.2., an integer programming formulation for the minmaxlatest is as follows:

$$\begin{aligned}
\text{(MML)} \qquad & \min \; Z \\
& \text{s.t} \\
Z &\geq (DT_k + c_{jk})X_{jk} \quad \forall\, k, j \quad (65) \\
\hat{DT}_k &\geq c_{jk}X_{jk} \quad \forall\, j,\, k \quad (66) \\
DT_k &\geq \hat{DT}_r + \alpha c_{rk}X_{rr} \quad \forall r,\, k \quad (67) \\
& (5) - (8)
\end{aligned}$$

Constraint (65) forces Z to take on the value of the latest arrival time. Constraints (66) and (67) ensure that $\hat{DT}_k$ and DT_k take on the intended values as defined in (63) and (64) at optimality.

(MML) is a mixed integer program with n^2 zero/one and $2n + 1$ real variables, and with $4n^2 + n + 1$ constraints. The model is nonlinear because of the constraint (65).

One way to linearize (MML) is to replace (65) with

$$Z \geq DT_k + c_{jk}X_{jk} - M(1 - X_{jk}) \quad (68)$$

where M is a large positive number. Unfortunately, the computational performance of this linearization is very poor. A less obvious but still correct linearization is to simply drop the last term in (68), i.e. write

$$Z \geq DT_k + c_{jk}X_{jk} \qquad (69)$$

in place of (65). We call this linearization (L1). The correctness of linearization (L1) is justified by the next two lemmas. In the lemmas $(Z, DT, \hat{DT}, X)$ stands for any solution where $DT, \hat{DT}$, and X are the vectors of variables $DT_k, \hat{DT}_k$, and X_{jk}, respectively.

Lemma 3: If $(Z, DT, \hat{DT}, X)$ is feasible for (L1), then it is also feasible for (MML).

Proof: The only two different constraints between (L1) and (MML) are (65) and (69). If $X_{jk} = 1$, then (65) and (69) yield the same right hand sides. If $X_{jk} = 0$, (69) yields $Z \geq DT_k$ whereas (65) yields $Z \geq 0$. Since $DT_k \geq 0$ due to (66) and (67), (65) is also satisfied. □.

Lemma 4: If $(Z, DT, \hat{DT}, X)$ is feasible for (MML), then a solution $(Z, \bar{DT}, \hat{DT}, X)$ can be constructed from the given feasible solution by replacing DT_k with $\bar{DT}_k = \max_r(\hat{DT}_r + \alpha c_{rk}X_{rr})$ $\forall k$ such that the solution so constructed is feasible for (L1).

Proof: If $X_{jk} = 1$, then (65) yields $Z \geq DT_k + c_{jk}$ and (69) yields $Z \geq \bar{DT}_k + c_{jk}$. Since $DT_k \geq \bar{DT}_k$ due to constraint (67), (69) is also satisfied.

If $X_{jk} = 0$, then (65) yields $Z \geq 0$ whereas (69) yields $Z \geq \bar{DT}_k$. We now prove that $Z \geq \bar{DT}_k$ is an implied constraint in (MML). There are two possibilities for the value of X_{kk}.

If $X_{kk} = 1$, then for $j = k$ (65) provides $Z \geq DT_k + c_{kk} \geq \bar{DT}_k$, and hence, $Z \geq \bar{DT}_k$ is implied.

If $X_{kk} = 0$, then, due to constraint (5), there exists an index l such that $X_{kl} = 1$. For the pair (k, l), (65) yields $Z \geq DT_l + c_{lk}$ (70). Let s be the index such that $\bar{DT}_k = \hat{DT}_s + \alpha c_{sk}$ (71). Using (70) and (67) we have,

$$Z \geq DT_l + c_{lk} \geq \hat{DT}_s + \alpha c_{sl} + c_{lk}.$$

The triangle inequality and (71) imply

$$Z \geq \hat{DT}_s + \alpha c_{sk} = \bar{DT}_k.$$

Hence, $Z \geq \bar{DT}_k$ is an implied constraint in (MML). Thus, $(Z, \bar{DT}, \hat{DT}, X)$ is feasible for (L1). □.

We can conclude from Lemmas 3 and 4 that any optimal solution $(Z^*, DT^*, \hat{DT}^*, X^*)$ to (L1) is also an optimal solution to (MML). The feasibility of this solution to (MML) follows from Lemma 1. If $(Z^*, DT^*, \hat{DT}^*, X^*)$ is not optimal for (MML), then there exists another solution $(Z', DT', \hat{DT}', X')$ which is feasible to (MML) where $Z' < Z^*$. Lemma 2 implies a solution $(Z', \bar{DT}', \hat{DT}', X')$ can be constructed form the $(Z', DT', \hat{DT}', X')$ solution which is feasible to (L1). This contradicts the optimality of $(Z^*, DT^*, \hat{DT}^*, X^*)$ since $Z' < Z^*$.

We now give a second linear model which is directly obtained from the combinatorial formulation by a reinterpretation. For fixed (H, a), let $\hat{T}_j(H, a)$ be the common arrival time at node j from all origins. That is

$$\hat{T}_j(H, a) = c_{a(j)\,j} + \max_{h \in H} \Big[\alpha c_{h\,a(j)} + \max_{k:a(k)=h} (r_k + c_{kh}) \Big].$$

Using the auxiliary variables $\hat{DT}_h$ and DT_h defined in (63) and (64), we also have $\hat{T}_j(H, a) = c_{a(j)\,j} + DT_{a(j)}$. It now follows that

$$\max_{j \in N} \hat{T}_j(H, a) = \max_{h \in H} (DT_h + \max_{k:a(k)=h} c_{kh}).$$

Hence, we may rewrite explicit form of the minimax latest arrival hub problem as:

$$\min_{\substack{H \subset N, \\ |H|=p}} \; \min_{a \in H^n} \; \max_{h \in H} (DT_h + \max_{k:a(k)=h} c_{kh})$$

This form of the combinatorial formulation directly leads to the following linear integer program, (L2):

$$\begin{array}{llll}
\text{(L2)} & \min \; Z & & \\
& \text{s. t.} & & \\
& Z \geq DT_h + \rho_h & \forall h & (72) \\
& \rho_h \geq t_{kh} X_{kh} & \forall k, h & (73) \\
& (66) - (67), (5) - (8) & &
\end{array}$$

Note that there is no nonlinearity in this new formulation. (L2) requires n^2 zero/one and $3n + 1$ real variables. The number of constraints is $4n^2 + 2n + 1$.

Observe that (L1) and (L2) are essentially the same linear integer programs since ρ_h is just and auxiliary variable and can be removed to convert (72) and (73) to the form $Z \geq DT_k + t_{jk}X_{jk}$ which is nothing but (69). Despite the fact that (L1) and (L2) have essentially the same mathematical structure, they are obtained out of entirely different considerations. (L1) is simply a linearization of the nonlinear model which is the natural model for hub location researchers since it focuses on the analysis of what goes on during the trip from an origin i to a destination j via the assigned hubs $a(i)$ and $a(j)$. On the other hand, (L2) is directly obtained from the combinatorial formulation by a reinterpretation that requires a switch from the traditional viewpoint. Instead of focusing on individual journeys from origins to destinations, it focuses on the analysis of what happens at the final destination.

6.4 Computational Results

We test the model (L2) with the CAB data using CPLEX 5.0. We take $r_i = 0 \, \forall i$ for every instance. In Table 6.1. we provide the CPU times reported by CPLEX 5.0 for each of the 80 instances of CAB2 Data set. The last two columns of the table provides the averages and maxima over p for each setting of n. The reported times are in seconds $n = 10$ and 15, in minutes for $n = 20$ and in hours for $n = 25$.

As can be seen from Table 6.1, all the instances of the CAB2 data set are solved to optimality within 23 hours. For $n = 10$, the 20 instances are solved within an average of 6.3 seconds whereas the maximum CPU time reported is 12.5 seconds. Increasing n from 10 to 15 resulted in average of 2 and maximum of 6.7 minutes. The 20 instances corresponding to $n = 20$ require an average of 38 minutes whereas the worst instance of this case require 2.5 hours. The exponential behavior begins to take over for $n > 20$ since the 20 instances of $n = 25$ case require an average of 4.2 hours and a maximum of 23 hours.

		CPU					
		p					
n	α	2	3	4	5	Avg.	Max
	0.2	3.7	5.5	6.9	9.0		
	0.4	3.7	5.9	7.5	10.6		
10	0.6	2.9	5.0	9.9	8.8	in secs.	
	0.8	3.6	4.7	12.5	5.7		
	1.0	2.3	7.5	6.3	4.5	6.3	12.5
	0.2	19.0	33.3	100.8	244.6		
	0.4	21.0	63.4	71.6	176.8		
15	0.6	11.0	25.7	199.7	260.1	in secs.	
	0.8	23.0	30.4	145.8	289.4		
	1.0	9.8	70.1	200.6	403.7	120.1	403.7
	0.2	1.8	5.8	21.7	54.4		
	0.4	1.7	8.7	44.2	122.4		
20	0.6	1.8	7.8	11.5	105.5	in mins.	
	0.8	1.0	8.8	37.8	150.8		
	1.0	1.3	8.3	33.5	129.5	37.9	150.8
	0.2	0.1	0.5	2.0	5.7		
	0.4	0.1	0.7	3.0	5.8		
25	0.6	0.1	0.5	1.7	17.3	in hrs.	
	0.8	0.1	0.6	4.4	22.9		
	1.0	0.1	0.7	3.7	13.5	4.2	22.9

Table 6.1: CPU Times for Minmaxlatest with CAB2

The average and maximum CPU times of the model (L2) with the customarily used CAB1 set is very different than that of CAB2. Table 6.2 provides the summary information of averages and maxima of the CPU time requirements of (L2) for the CAB1 set.

	n = 10	n = 15	n = 20	n = 25
Avg.	5.8 secs.	68.4 secs.	13.0 mins.	1.2 hrs
Max.	12.5 secs.	200.6 secs.	44.2 mins.	4.4 hrs

Table 6.2: CPU Times for Minmaxlatest with CAB1

Especially for $n \geq 20$, including $p = 5$ case (solving the model with CAB2) significantly increased the averages and maxima over n. When $n = 10$ and 25, the average CPU time with CAB2 is nearly the maximum CPU time of CAB1. With CAB1, 55 out of 60 instances are solved to optimality in one hour

In order to have an idea about the effectiveness of the latest arrival model, we compared the average and maximum CPU times with CAB2 with those of (EK) (the best model for the p-hub median) and (LinNew) (the best model for the p-hub center).

n	(EK)		(LinNew)		(L2)	
	Avg.	Max.	Avg.	Max.	Avg.	Max.
10	0.9secs	1.7secs.	4.3secs.	8.1secs.	6.3secs.	12.5secs.
15	10secs	21secs.	96.4secs.	313.2secs.	120.1secs.	403.7secs.
20	3.5mins	14.8mins.	29.2mins.	69.2mins.	37.9mins.	150.8mins.
25	28.2mins	4.8hrs.	5.4hrs.	11.8hrs.	4.2hrs.	22.9hrs.

Table 6.3: CPU Time Comparison of the Best Models

As can be seen from Table 6.3, the computational performance of the total sum model is very different than those of the minimax models. On the other hand, the performance of (LinNew) and (L2) can be considered somewhat similar. If we look at the solution times of (LinNew) from Table 5.3. and solution times of (L2) from Table 6.1., we see that, in terms of individual instances, (L2) is faster than (LinNew) in 44 out of 80 instances. Even though the latest arrival problem has additional requirements, the performance of its model is comparable with that of the p-hub center problem. Thus, we may conclude that, the computational performance of the model which also takes the transient times into account is comparable with the computational performance of the model which do not consider the transient times.

In the next section we present an analysis on departure times from hubs. This analysis also provides a framework to utilize the transient times on a need basis.

6.5 Analysis of Departure Times

For a specific solution (H,a), let $f(H,a)$ be the value of the outermost maximand in (60). That is, $f(H,a)$ is the latest arrival time induced by (H,a). Given such a solution, it is natural to ask how the departure times at hubs and nonhubs can be effected without increasing the latest arrival time, $f(H,a)$. Specifically, we may ask how much delay can be tolerated at a given hub without increasing the latest arrival time. Another related question is the following: Even if the delay at a hub is within the tolerable limit, does this delay increase the departure times of the flights at subsequent hubs, and if so, in what way? In this section, we analyze these and related what-if questions.

Let (H,a) be a given solution. Corresponding to (H,a), the values of the departure times DT_k and $\hat{DT}_k$ are determined by (63) and (64). Note also that

$$f(H,a) = \max_{k \in H}(DT_k + \max_{j:a(j)=k} c_{jk}) \quad (74).$$

Let us first focus on the question of how much delay can be tolerated on DT_k and $\hat{DT}_k$ without increasing $f(H,a)$. Let q be a specific index and let δ_q and $\hat{\delta}_q$ be the amounts of increase in DT_q and $\hat{DT}_q$, respectively. Assuming that $\hat{DT}_k, k \neq q$, do not change, we can determine the maximum tolerable delays, $\hat{\delta}_q^{max}$ and δ_q^{max}, as follows. From (74), we require that

$$f(H,a) \geq DT_q + \delta_q + \rho_q$$

where $\rho_q = \max_{j:a(j)=q} c_{jq}$. Thus, $\delta_q^{max} = f(H,a) - DT_q - \rho_q$. To derive the upper bound on $\hat{\delta}_q$, substitute the right side of (64) in place of DT_k in (74). This gives

$$f(H,a) = \max_{k \in H} [\rho_k + \max_{r \in H}(\hat{DT}_r + \alpha c_{rk})].$$

It follows that $\hat{\delta}_q^{max} = \min_{k \in H}(f(H,a) - \rho_k - \hat{DT}_q - \alpha c_{qk}) \quad (75).$

We can conclude now that as long as the departure time from hub q to other hubs is no later then $\hat{DT}_q + \hat{\delta}_q^{max}$ and, as long as the departure time from hub q to nonhub destinations is no later than $DT_q + \delta_q^{max}$, the maximum arrival time resulting from (H,a) will not be any later than $f(H,a)$. In particular, if

(H^*, a^*) is an optimal location-allocation decision, then the values of $\hat{\delta}_q^{max}$ and δ_q^{max}, computed relative to (H^*, a^*), give the maximum tolerable delays at hub q at optimality.

Suppose now, we conduct the one-at-a-time delay analysis separately for each $q \in \{1, ..., n\}$, i.e. we compute $\hat{\delta}_q^{max}, \delta_q^{max}$ separately for each q assuming that $\hat{DT}_k, k \neq q$, do not change when delay is allowed at hub q. This supplies a collection of maximum tolerable delays $\hat{\delta}_q^{max}, \delta_q^{max}, q = 1, ..., n$. It is direct to conclude that there is at least one hub k for which $\hat{\delta}_k^{max} = 0$ and at least one hub k' for which $\delta_{k'}^{max} = 0$. It is possible that $k = k'$, but in general they are different. Any delay at one of these hubs increases the latest arrival time by the amount of delay. Note also that, there is an origin $s \in \arg \max_{i:a(i)=k} c_{ik}$ and a destination d such that $a(d) = k'$ with $c_{k'd} = \rho_{k'}$, so that (s, k, k', d) forms a critical path that determines the latest arrival time by the relation $f(H, a) = r_s + c_{sk} + \alpha c_{kk'} + c_{k'd}$. If there is more than one k for which $\hat{\delta}_k^{max} = 0$ or more than one k' for which $\delta_{k'}^{max} = 0$, then each such pair (k, k') identifies a critical path. If $f(H, a)$ needs to be reduced for some reason, one way of doing this is to find a solution (H', a') for which $f(H', a') < f(H, a)$. If $f(H, a)$ is already optimal, then this way of reducing $f(H, a)$ is not possible. A less costly alternative that does not require a change in the given solution (H, a) is to focus on the critical paths induced by (H, a) and reduce their total journey times. This can be done by either setting the appropriate r_i's to earlier times or by decreasing the flight times by assigning faster aircrafts to critical path segments. Hence, the model on hand allows to perform a trade-off analysis between the cost of reducing the critical journey times and the benefits that would be obtained from the reduction of the latest arrival time. Such analysis may prove to be quite useful when it is desirable to reduce the latest arrival time without changing the current hub locations and the current allocations of nodes to hubs.

We now analyze the effects of what happens when the actual delay $\hat{\delta}_q(\delta_q)$ exceeds the maximum tolerable delay $\hat{\delta}_q^{max}(\delta_q^{max})$. We assume again that $\hat{DT}_k, k \neq q$, do not change. Under this assumption, if $\delta_q > \delta_q^{max}$, then the latest arrival time at the destinations which are serviced from hub q will be later than $f(H, a)$ and the amount of increase is $\delta_q - \delta_q^{max}$ time units. Note that the latest arrival time at the destinations which are serviced from other hubs $k \neq q$ will not change as long as $\hat{\delta}_q \leq \hat{\delta}_q^{max}$. Consider now the case with $\hat{\delta}_q > \hat{\delta}_q^{max}$. Observe that the delay $\hat{\delta}_q$ affects in general the departure times of flights from other hubs k to nonhub destinations. The new departure time at hub k, DT_k^{new}, is defined by:

$$DT_k^{new} = DT_k + \max(0, \hat{DT}_q + \hat{\delta}_q + \alpha c_{qk} - DT_k) \quad (76).$$

Note that the values of $\hat{DT}_k, k \neq q$, are not affected by $\hat{\delta}_q$, so the initial assumption is not violated. With the new departure times $DT_k^{new}, k \neq q$, the new maximum arrival time is

$$f^{new}(H, a) = \max_{j \in N}(DT_{a(j)}^{new} + c_{a(j)j}) \quad (77).$$

Whenever $\hat{\delta}_q > \hat{\delta}_q^{max}$, it is direct to conclude that $f^{new}(H, a)$ exceeds the old latest arrival time $f(H, a)$ by $\hat{\delta}_q - \hat{\delta}_q^{max}$.

Suppose now we allow delays at many different hubs. In this case, the delay at any given hub affects the tolerable limits on the delays at other hubs, and so the analysis of simultaneous delays must take these interdependencies into account. Let $\delta_k, \hat{\delta}_k$, $k \in H$, be the delays associated with hub k. From (74), we require

$$f(H, a) \geq DT_k + \delta_k + \rho_k, \quad k \in H \quad (78).$$

Substituting the right side of (64) for DT_k in (78), we have:

$$\hat{\delta}_j + \delta_k \leq b_{jk} \quad \forall j, k \in H \quad (79)$$

where $b_{jk} \equiv f(H, a) - \hat{DT}_j - \alpha c_{jk} - \rho_k$. (79) is a system of p^2 linear inequalities in the $2p$ variables $\hat{\delta}_k, \delta_k$, $k \in H$. Any feasible solution to the nonnegativity constraints $\hat{\delta}_k, \delta_k \geq 0$, $k \in H$, and (79) constitutes a collection of delays on departure times that does not increase the latest arrival time beyond its current value. If a given set of nonnegative delays is not feasible to (79), then the amount of increase in the current value of the latest arrival time is determined by the maximum violation in (79).

The feasibility of delays $\hat{\delta}_k, \delta_k$, $k \in H$, is dependent on the solution (H, a) but it can be seen from (79) that this dependence is reflected only on the right hand side values b_{jk} but not on the structure of the feasibility system. Hence, even if (H, a) changes, the feasibility of a given set of delays can easily be checked by recomputing the b_{jk} values relative to the new solution.

Using $\hat{\delta}_q^{max}$ and δ_q^{max} and the above system of inequalities we may answer such *what-if* questions as:

- what-if airport q has to be shut down for 2 hours starting at 11:45 a.m. due to thick fog?
- what if the sorting operations at hub q has to be delayed due to equipment malfunction for 2 hours, starting at 11:45?

Observe first that both the thick fog and the equipment malfunction are unexpected events but their effects on the system performance can be analyzed using the same approach. It is possible that the departure times at hub q are already delayed by δ_q and $\hat{\delta}_q$ time units. Thus, the planned departure times are $\hat{DT}_q + \hat{\delta}_q$ for hub-to-hub flights and $DT_q + \delta_q$ for hub-to-nonhub flights. Now, let us analyze how the additional two-hour shutdown will affect the departure times and the latest arrival time. We first analyze DT_q.

- If $DT_q + \delta_q <$ 11:45, i.e., if the departure time is before the unexpected event has started, then the two-hour shutdown does not additionally affect the system. That is, the additional delay resulting from the two-hour shutdown is zero and the latest arrival time is not affected as long as $\delta_q \leq \delta_q^{max}$. Observe here that if $\delta_q > \delta_q^{max}$, then the increase in the latest arrival time is $\delta_q - \delta_q^{max}$.
- If 11:45 $\leq DT_q + \delta_q <$ 13:45, then the aircrafts at hub q cannot depart until 13:45, so the actual delay will be 13:45 $-DT_q$. The latest arrival time is not affected if 13:45 $-DT_q \leq \delta_q^{max}$. If the actual delay 13:45 $-DT_q$ is greater than δ_q^{max}, then the latest arrival time at one of the

destinations that receives service from hub q increases by the amount $(13{:}45 - DT_q) - \delta_q^{max}$.

- If $DT_q + \delta_q \geq 13{:}45$, then the two-hour shutdown does not additionally affect the system. That is, if $\delta_q \leq \delta_q^{max}$, then the latest arrival time does not change, and if $\delta_q > \delta_q^{max}$, then the latest arrival time at one of the destinations that receives service from hub q increases by $\delta_q - \delta_q^{max}$.

Similar analysis can also be conducted for $\hat{DT}_q$.

- If $\hat{DT}_q + \hat{\delta}_q < 11{:}45$, then the two-hour shutdown at hub q does not additionally affect the system.
- If $11{:}45 \leq \hat{DT}_q + \hat{\delta}_q < 13{:}45$, then actual delay is $13{:}45 - \hat{DT}_q$. The new departure times $DT_k^{new}, k \neq q$, can be computed using $13{:}45 - \hat{DT}_q$ in place of $\hat{\delta}_q$ in (76). The new maximum arrival time is again computed using (77).
- If $\hat{DT}_q + \hat{\delta}_q \geq 13{:}45$, then the two-hour shutdown does not additionally affect the system performance.

Using the inequality system (79), we may answer more general what-if questions that address simultaneous delays at different hubs, e.g.

- what-if airport a has to be shut down for 2 hours, beginning at time t, due to stormy weather and what-if the sorting operations at hub b has to be delayed for 3 hours due to equipment malfunction beginning at time t'?

Suppose that the current delays at hubs a and b are $\hat{\delta}'_a, \delta'_a, \hat{\delta}'_b, \delta'_b$. With an analysis similar to the one that we have conducted for the single shutdown case, the actual delays $\hat{\delta}_a, \delta_a$ resulting from the unexpected events are:

$$
\begin{array}{llll}
\text{If} & \hat{DT}_a + \hat{\delta}'_a < t & \text{then} & \hat{\delta}_a = \hat{\delta}'_a \\
\text{If} & t \leq \hat{DT}_a + \hat{\delta}'_a < t+2 & \text{then} & \hat{\delta}_a = t + 2 - \hat{DT}_a \\
\text{If} & t+2 \leq \hat{DT}_a + \hat{\delta}'_a & \text{then} & \hat{\delta}_a = \hat{\delta}'_a \\
\text{If} & DT_a + \delta'_a < t & \text{then} & \delta_a = \delta'_a \\
\text{If} & t \leq DT_a + \delta'_a < t+2 & \text{then} & \delta_a = t + 2 - DT_a \\
\text{If} & t+2 \leq DT_a + \delta'_a & \text{then} & \delta_a = \delta'_a
\end{array}
$$

$\hat{\delta}_b$ and δ_b are defined similarly by replacing t with t' and $t+2$ with $t'+3$ in the above definitions. If the inequality system (79) yields a feasible solution with the so formed values of $\hat{\delta}_a, \delta_a, \hat{\delta}_b, \delta_b$, then the latest arrival time does not increase. The maximum violation in (79), if any, determines the increase in the latest arrival time. In addition, the union of the indices of the violating constraints in (79) give the set of hubs whose delays result in an increase in the old latest arrival time.

Hence, utilizing (79), the effects of delays resulting from unexpected events can be foreseen which is a critical issue in managerial decisions.

Chapter 7

Summary and Conclusions

In this thesis, we analyze some important issues surrounding the hub location problem which is a rather new research area. "Hubbing" is the main problem characteristic of the hub location problems. We first analyze different real world problems which constitute the application areas of the hub location problem. The three main areas of application are airline systems, cargo delivery systems, and large scale communication systems. We conclude that hubbing is definitely encountered in these areas, but we observe that the structure of the real problems lead to the customarily defined hub location problem only under certain assumptions. The clarification of the underlying assumptions and common features in different problem areas organizes the available literature in a new and more focused structure, thus forming an improved basis for future research. During the analysis on the real world problem requirements we also identify a problem which is not satisfactorily modeled by means of the customarily defined hub location problem. We analyze the problem and propose a model for it, the latest arrival hub location problem, in Chapter 6.

Nearly all of the literature is focused on developing linearizations of the basic model proposed by O'Kelly (1986a). We conduct a computational study of those linearizations together with 3 different linearizations that we propose in this thesis in Chapter 3.

In Chapter 4 we focus on the allocation problem. We prove that it is equivalent to a well known location problem from the literature: the restricted multi-median location problem with mutual communication. Using this equivalence we strengthen the complexity result of the allocation problem by proving that the problem is NP-Hard even if the trasport network is a star tree. We also concentrate on polynomial time solvable special cases of the problem. The polynomial time solvable cases are in two categories: the ones based on the structure of the flow data and the ones based on the transport network. The study on the transportation network leads to interesting decomposition results which are totally new to the hub location literature.

Another deficiency of the literature is on the analysis of the hub location problem under different performance measures. We analyze the problem under minimax and cover objectives, namely, the p-hub center and hub covering problems, in Chapter 5. The two problems have already been defined and modeled in the literature but without any analysis. We first prove that both are NP-Hard. We then provide integer programming models for both of the problems. The new models are far more superior than the original models both in terms of CPU time and core storage requirements.

As a final topic, we analyze the latest arrival hub location problem in which the transient times at hubs during delivery are also taken into account. This problem is totally new to the literature. We provide the combinatorial formulation of the problem under totalcost, minimax, and cover objectives. We prove that the minimax and covering versions are NP-Hard. For the minimax problem, we provide an integer programming formulation from two different perspectives.

We believe that the hub location problem is a fruitful area of research. The main reason for this is the fact that nearly all of the literature is devoted to only one aspect of the problem leaving many untouched or barely touched issues. We analyze some of those issues but there are still different faces of the problem which needs further consideration. One such issue is the allocation problem for the p-hub center and covering problems and also for the latest arrival hub location problem. Another issue is the analysis of the p-hub center, covering, and latest arrival hub location problems for multi-assignment cases. The allocation problems under the multi-assignment cases of the p-hub center and covering problems constitude another issue that need consideration. When we first started our analysis on the hub location problem, we got biased from the literature and initially concentrated on the total sum problem. However, as we proceeded with our analysis of the problem under different criteria and in different settings we put the problem into a different perspective which has led to many fruitful results. For example, the identification of an underlying transport network is a result of our analysis which lead to interesting decomposition theorems for the allocation problem. The identification of the latest arrival hub location problem is a result of our analysis on the application areas. The new models that we propose for the p-hub center and hub covering problems and their linearization are the results of a different way of studying to the problem. All this process of detailed analysis, understanding what really is going on in the real world, helped us identify different faces of our problem leading to an enrichment of the literature.

Bibliography

[1] K. Altinkemer and Z. Yu. Topological design of wide area communication networks. *Annals of Operations Research*, 36:365–382, 1992.

[2] T. Aykin. Lagrangean relaxation based approaches to capacitated hub-and-spoke network design problem. *European Journal of Operational Research*, 77:1–23, 1994.

[3] T. Aykin. Networking policies for hub-and-spoke systems with application to the air transportation syste. *Transportation Science*, 29:201–221, 1995a.

[4] T. Aykin. The hub location and the routing problem. *European Journal of Operational Research*, 83:200–219, 1995b.

[5] J.F. Campbell. Integer programming formulations of discrete hub location problems. *European Journal of Operational Research*, 72:387–405, 1994a.

[6] J.F. Campbell. A survey of network hub location. *Studies in Locational Analysis*, 6:31–49, 1994b.

[7] J.F. Campbell. Hub location and the p-hub median problem. *Operations Research*, 44:923–935, 1996.

[8] R. Ceha and H. Ohta. The evaluation of air transportation network based on multiple criteria. *Computers and Industrial Engineering*, 27:249–252, 1994.

[9] D. Chhajed and T.J. Lowe. m-median and m-center problems with mutual communication: Solvable special cases. *Operations Research*, Supplement 40(1):S56–S66, 1992.

[10] D. Chhajed and T.J. Lowe. Solving structured multifacility location problems efficiently. *Transportation Science*, 28(2):104–115, 1994.

[11] S. Chung, Y. Myung, and D. Tcha. Optimal design of a distributed network with a two-level hierarchical structure. *European Journal of Operational Research*, 62:105–115, 1992.

[12] E. Dalhaus, D.S. Johnson, C.H. Papadimitrou, P.D. Seymour, and M. Yanakis. The complexity of multiway cuts. *Proceedings of the 24th annual ACM Symposium on Theory of Computing*, pages 241–251, 1992.

[13] A.T. Ernst and M. Krishnamoorthy. Efficient algorithms for the uncapacitated single allocation p-hub median problem. *Location Science*, 4(3):139–154, 1996.

[14] A.T. Ernst and M. Krishnamoorthy. Exact and heuristic algorithms for the uncapacitated multiple allocation p-hub median problem. *European Journal of Operational Research*, 104:100–112, 1998a.

[15] A.T. Ernst and M. Krishnamoorthy. An exact solution approach based on shortest-paths for p-hub median problems. *INFORMS Journal on Computing*, 10(2), 1998b.

[16] A.T. Ernst and M. Krishnamoorthy. Solution algorithms for the capacitated single allocation hub location problem. to appear. *Annals of Operations Research*, 1998c.

[17] D. Fernandez-Baca. Allocating modules to processors in a distributed system. *IEEE Transactions on Software Engineering*, 15:1427–1436, 1989.

[18] R.W. Floyd. Algorithm 97, shortest path. *Communications ACM*, 5:345, 1962.

[19] M. R. Garey and D.S. Johnson. *Computers and Intractability. A Guide to the Theory of NP-Completeness.* W.H. Freeman and Company, Newyork, 1979.

[20] B. Gavish. Topological design of telecommunications networks- local access design methods. *Annals of Operations Research*, 33:17–71, 1991.

[21] B. Gavish. Topological design of computer communications networks-the overall design problem. *European Journal of Operational Research*, 58:149–172, 1992.

[22] A.A. Ghobrial and A. Kanafani. Future of airline hubbed networks: Some policy implications. *Journal of Transportation Engineering*, 121(2):124–134, 1995.

[23] R.W. Hall. Configuration of an overnight package air network. *Transportation Research A*, 23A(2):139–149, 1989.

[24] A. Kanafani and A.A. Ghobrial. Airline hubbing-some implications for airport economics. *Transportation Research, A*, 1:15–27, 1985.

[25] B. Y. Kara and B. C. Tansel. Modeling and solvability analysis of the p-hub location problem. Technical Report IE-OR 9710, Bilkent University, Dept. of Industrial Engineering, 06533, Bilkent, Ankara, Turkey, August 1997.

[26] B. Y. Kara and B. C. Tansel. On the allocation phase of the p-hub location problem. Technical Report IE-OR 9827, Bilkent University, Dept. of Industrial Engineering, 06533, Bilkent, Ankara, Turkey, September 1998.

[27] O. Kariv and S.L. Hakimi. An algorithmic approach to network location problems. 1: The p-centers. *SIAM Journal of Applied Math*, 37(3):513–538, 1979.

[28] L. Kaufman and F. Broeckx. An algorithm for the quadratic assignment problem using benders' decomposition. *European Journal of Operational Research*, 2:204–211, 1978.

[29] O. Kettani and M. Oral. Reformulating quadratic assignment problems for efficient optimization. *IIE Transactions*, 25(6):97–106, 1993.

[30] J. Kim and D. Tcha. Optimal design of a two-level hierarchical network with tree-star configuration. *Computers and Industrial Engineering*, 22(3):273–281, 1992.

[31] J. G. Klincewicz. Heuristics for the p-hub location problem. *European Journal of Operational Research*, 53:25–37, 1991.

[32] J. G. Klincewicz. Avoiding local optima in the p-hub location problem using tabu search and grasp. *Annals of Operations Research*, 40:283–302, 1992.

[33] J. G. Klincewicz. A dual algorithm for the uncapacitated hub location problem. *Location Science*, 4:137–184, 1996.

[34] J. G. Klincewicz. Hub location in backbone/tributary network design: A review. *Location Science*, 6:307–335, 1998.

[35] A.J.W. Kolen. Tree networks and planar rectilinear location theory. Technical report, Centrum voor Wiskunde en Informatica, Amsterdam, the Netherlands, 1986.

[36] M.J. Kuby and R.G. Gray. The hub network design problem with stopovers and feeders: The case of federal express. *Transportation Research*, 27A(1):1–12, 1993.

[37] C. Lee, H. Ro, and D. Tcha. Topological design of a two-level network with ring-star configuration. *Computers and Operations Research*, 20(6):625–637, 1993.

[38] R.E. Marsten and M.R. Muller. A mixed-integer programming approach to air cargo fleet planning. *Management Science*, 26(7):1096–1107, 1980.

[39] J. R. Meyer and C.V. Oster. *Deregulation and the New Airline Entrepreneurs.* The MIT Press, 1984.

[40] W. E. O'Connor. *An Introduction to Airline Economics.* Praeger, 1989.

[41] M. E. O'Kelly. The location of interacting hub facilities. *Transportation Science*, 20(2):92–105, May 1986a.

[42] M.E. O'Kelly. Activity levels at hub facilities in interacting networks. *Geographical Analysis*, 18(4):343–356, October 1986b.

[43] M.E. O'Kelly. A quadratic integer program for the location of interacting hub facilities. *European Journal of Operational Research*, 32:393–404, 1987.

[44] M.E. O'Kelly. A clustering approach to the planar hub location problem. *Annals of Operations Research*, 40:339–353, 1992a.

[45] M.E. O'Kelly. Hub facility location with fixed costs. *Papers in Regional Science*, 71(3):293–306, 1992b.

[46] M.E. O'Kelly, D. Bryan, D. Skorin-Kapov, and J. Skorin-Kapov. Hub network design with single and multiple allocation: A computational study. *Location Science*, 4:125–138, 1994.

[47] M.E. O'Kelly and H.J. Miller. Solution strategies for the single facility minimax hub location problem. *Papers in Regional Science*, 70(4):367–380, 1991.

[48] M.E. O'Kelly and H.J. Miller. The hub network design problem a review and synthesis. *Journal of Transport Geography*, 2(1):31–40, 1994.

[49] M.E. O'Kelly, D. Skorin-Kapov, and J. Skorin-Kapov. Lower bounds for the hub location problem. *Management Science*, 41:713–721, 1995.

[50] R. A. Sigafoos and R. R. Easson. *Absolutely Positively Overnight! The Unofficial Corporate History of Federal Express*. St. Lukes Press Memphis, 1988.

[51] D. Skorin-Kapov and J. Skorin-Kapov. On tabu search for the location of interacting hub facilities. *European Journal of Operations Research*, 73:502–509, 1994.

[52] D. Skorin-Kapov, J. Skorin-Kapov, and M. O'Kelly. Tight linear programming relaxations of uncapacitated p-hub median problems. *European Journal of Operations Research*, 94:582–593, 1996.

[53] J. Sohn and S. Park. A linear programming for the two hub location problem. *European Journal of Operational Research*, 100:617–622, 1997.

[54] J. Sohn and S. Park. Efficient solution procedure and reduced size formulations for p-hub location problems. *European Journal of Operational Research*, 108:118–126, 1998a.

[55] J. Sohn and S. Park. The single allocation problem in the interacting three-hub network. Technical report, Department of Industrial Engineering, Korea Advanced Institute of Science and Technology, 373-1 Gusong-dong, Yusong-gu, Taejon 305-710, Korea, March 1998b.

[56] W. Stallings. *Data and computer communications.* Macmillan Publishing Company, 1991.

[57] A. Tamir. Complexity results for the p-median problem with mutual communication. *Operations Research Letters*, 14:79–84, 1993.

[58] B.C. Tansel, R.L. Francis, and T.J. Lowe. Location on networks: A survey part ii: Exploiting tree network structure. *Management Science*, 29(4):498–511, 1983.

[59] R.S. Toh and R.G. Higgins. The impact of hub and spoke network centralization and route monopoly on domestic airline profitability. *Transportation Journal*, 24:16–27, 1985.

[60] Gang Yu. *Operations Research in the Airline Industry.* Kluwer Academic Publishers, 1998.