

To the Graduate Council:

I am submitting herewith a dissertation written by Ramaier Sriram, entitled "Locating Interactive Hub Facilities." I have examined the final copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Management Science.

Robert Garfinkel

Robert Garfinkel, Major Professor

We have read this dissertation
and recommend its acceptance:

Bruce A. Bell

Charles E. Mann

Kenneth C. Gilbert

Accepted for the Council:

Lawrence Mink

Associate Vice Chancellor and
Dean of The Graduate School

LOCATING INTERACTIVE HUB FACILITIES

**A Dissertation
Presented for the
Doctor of Philosophy
Degree
The University of Tennessee, Knoxville**

**Ramaier Sriram
May 1991**

Copyright © Ramaier Sriram, 1991
All rights reserved

ACKNOWLEDGEMENTS

In the course of dissertation research a number of scholars have assisted me generously. In particular I am indebted to Professors Robert Garfinkel, Charles Noon, Kenneth Gilbert, and Bruce Ralston, who served as my director and committee respectively. With their criticism and encouragement I was able to improve the quality of my work substantially.

I also acknowledge with thanks the input provided by Dr. Moshe Segal of the AT&T Bell Laboratories.

Finally, I thank my wife and son for their tremendous support.

ABSTRACT

We examine the problem of simultaneous determination of facility location and flow routing in the context of hub location in private-line networks. The flows in this case are the number of channels of communication demanded between the user locations, and the facilities location aspect is due to the need to analyze the location of hubs to service the need between the user locations. Hubs are special points in the network containing switching equipment which serve to connect the network links. We address questions about the topology of the network using hubs: where should the hubs be located, how should the node-hub assignments be made, which network links should be provided, and what are the capacities of these links. Three special situations of the problem are examined in detail providing mathematical formulations, complexity arguments, analysis of special characteristics and special cases of the situations, solution strategies, bounding strategies, and computational experience with randomly generated problems. A description of interesting extensions to this research are suggested.

TABLE OF CONTENTS

Chapter 1 - Introduction	
1.1 Research Theme	1
1.2 Introduction to Private-Line Networks.....	3
1.3 Hubbing in Private-Line Networks.....	7
1.4 The General Research Problem.....	9
1.5 Research Focus.....	14
1.5.1 Problem P1.....	14
1.5.2 Problem P2.....	15
1.5.3 Problem P3.....	15
1.6 Organization of Research Presentation.....	16
Chapter 2 - Literature Review	17
Chapter 3 - Problem P1: Mathematical Formulation and Algorithms	
3.1 Problem Statement.....	21
3.2 Mathematical Formulation	24
3.3 Complexity of P1	26
3.4 Extreme Point Property	29
3.5 Exact Solution Approach.....	31
3.6 Lagrangian Relaxation of P1	31
3.7 2-Partition Case of P1: A Special Structure	34
3.8 Heuristic Solution Approach for P1 with Euclidean Distance - HSP1	38
3.9 Heuristic Solution Approach for P1 with Given Distance Measure - HSP2.....	43
3.10 Variable Elimination	45
3.10 Bounding Procedures	45
3.12 Computational Experience.....	47
Chapter 4 - Problem P2: Mathematical Formulation and Algorithms	
4.1 Problem Statement.....	54
4.2 Mathematical Formulation	56
4.3 Complexity of P2	59
4.4 Solution Strategies.....	60

4.5	Heuristic Solution Approaches	61
4.5.1	Total Enumeration of Hub Locational Configurations	61
4.5.2	Pair Wise Interchange Heuristic for P2 - HSP3.....	61
4.5.2.1	Initial Hub Sites Selection.....	64
4.5.3	Solution to P2 Using Heuristic HSP2 - HSP4	65
4.6	Lower Bound Procedures for P2	67
4.7	Computational Experience.....	69
 Chapter 5 - Problem P3: Mathematical Formulation and Algorithms		
5.1	Problem Statement.....	74
5.2	Integer Linear Programming Formulation of P3	77
5.3	Complexity of P3	80
5.4	Linear Programming Relaxation of P3.....	82
5.5	Lower Bound Procedures for P3.....	86
5.6	Heuristic Solution Procedures for P3 - FS1 and FS2.....	89
5.7	Computational Experience.....	95
 Chapter 6 - General Problem: Mathematical Formulation and Algorithms		
6.1	Problem Statement.....	100
6.2	Mathematical Formulation of GP	103
6.3	Complexity of GP	107
6.4	Heuristic Solution Approaches for GP	107
6.5	Lower Bound Procedures for GP	108
 Chapter 7 - Summary, Future Directions and Planning Issues		
		111
References		118

APPENDICES

Appendix A

A1	Example of complexity proof of P1.....	123
A2	Counter Example for Non Unimodularity of LP1.....	125
A3	Example of Application of Lagrangian Approach.....	129
A4	Example of Application of Heuristic HSP1.....	136
A5	Example of Application of Heuristic HSP2.....	139
A6	Limitation of Commercial LP codes for solving P1	142

Appendix B

B1	Description of the Min-sum Multicenter Problem.....	143
B2	Example of Application of Heuristic HSP3.....	144
B3	Example of Application of Initial Hub Sites Selection Procedure.....	147
B4	Example of Application of Lower Bound Procedure LB3.....	149

Appendix C

C1	Example Illustrating Complexity Argument for P3.....	151
C2	Example of Application of Heuristic Procedures FS1 and FS2.....	153

VITA

LIST OF TABLES

3.1	Summary of results for 20 node problems	50
3.2	Summary of results for 50 node problems	51
3.3	Summary of results for 100 node problems.....	52
4.1	Summary of results for 15 node problems	72
4.2	Summary of results for 25 node problems	73
5.1	Summary of results for 10 node problems	98
5.2	Summary of results for 20 node problems	99

LIST OF FIGURES

1.1	Topology of the PLN.....	5
1.2	Routing Strategies for the General Problem.....	13
3.1	Routing Strategies for problem P1.....	22
3.2	2-partition example: illustration & network representation.....	36
3.3	3-partition example: illustration & network representation.....	39

CHAPTER 1

INTRODUCTION

1.1 Research Theme

RESEARCH Traditionally research in facilities location and flow routing have been
THEME separate. Research in flow routing assumes the location of facilities and
 attempts to determine the best flow pattern. Research in facilities location,
 on the other hand, attempts to determine the best locations for the facilities
 for a given pattern of flow. For example in flow problems like
 multicommodity flow, transportation, shortest path and vehicle routing the
 location of facilities are given and the objective is to determine the flow
 pattern satisfying some criterion, usually minimizing cost. On the other
 hand in problems like plant location, and facilities layout the pattern of flow
 between the entities is assumed and the objective is to determine the best
 locations for the facilities satisfying some criterion, usually minimizing cost.

In situations where the pattern of flow is dependent on the locations of the facilities and vice-versa, determination of the best flow pattern and facilities location must be simultaneous. There is growing interest in this interaction between flow pattern and facilities location, and it is emerging as an important research theme. (Selected references include Ahuja [1], Garfinkel and Segal [11], Garfinkel and Smith [12], Gavish [14], Geoffrion and Graves [16], Mirzaian[24], O'Kelly [25,26,27].)

This study deals with one particular manifestation of this theme of simultaneous determination of facility location and flow routing. The case of hubbing in **private-line communications networks (PLN)** is considered. The

flows in this case are the number of channels of communication demanded between two user locations. The facilities location side of the case arises from the need to analyze the location of hubs to service the flows between the user locations. The study attempts to address questions about the topology of the network using hubs: where should the hubs be located, which network links should be provided, and what are the capacities of these links, subject to the constraints of connectivity. Hubs are special points in the network containing switching equipment which serve to connect the network links.¹ A link is defined as the direct connection between two nodes.

We first describe the PLN and then develop a general research problem which deals with using hubs in PLN. It must be noted here that in the general problem hubbing in PLN is not considered with all its generalities. Three specific realizations of the general research problem, which will be part of the research focus, are defined next. In the descriptions we shall use the generic network terms - flows and nodes, to represent the number of channels of communication demanded between user locations, and the user locations themselves. This study addresses the approaches to determine the minimum cost topological design - locations of hubs, network links and link capacities etc. - of the network in each of three specific realizations, and for the general research problem. The topological design of the network assigns the links and the link capacities for connecting the nodes.

¹ A detailed description of hubs is given in section 1.3.

1.2 Private-Line Networks²

The description of the PLN is included here to provide the reader a background to the motivating problem.

PRIVATE-LINE NETWORKS PLN services are used by large business customers to interconnect their geographically dispersed locations. Private-line services are dedicated to the customer and not shared as in the public telecommunications network, thereby ensuring a through connection at all times. Every channel of communication demanded is like a physical connection between the end points of demand. These channels are used for both voice and data communication. PLN's range from large nationwide networks that service hundreds of locations to small networks in metropolitan areas that serve fewer locations. AT&T [4] lists the following PLN services: CORNET (CORporate NETwork used in the Bell system), FTS (the Federal Telecommunication Service network serving the civil organizations of the Federal Government), and AUTOVAN (the AUTOMATIC Voice Network of the US Military).

STRUCTURE OF THE PLN Two basic configurations of the PLN are switched and non-switched. In a non-switched network, user locations with communications flow between them are connected without the switching systems. In a switched network, the channels are connected using the switching systems via one or more other nodes in the network. Switching systems serve to automatically interconnect channels. Those nodes in the network which contain the switching systems

²Description of the private-line network is derived from AT&T [4].

are called switching centers, and these may be located at the user locations. In a pure switched network, all nodes will be connected only via the switching centers. A general network may consist of both non-switched and switched segments and thus the connecting path may contain one or more links of the network. The link from a customer node to the switching center is called access, and the link between two switching centers is called trunk. The switching systems are shared among different customers because most customers do not have enough traffic to justify a dedicated switching system. But the channels in the access and the trunk links, and the switch connections are dedicated to the customers. Switching centers are of two types, *local* and *tandem*. Local switching centers serve to connect users assigned to the same center, and access links to trunk links. Tandem switching centers connect trunk links to trunk links. Figure 1.1 describes the topology of the PLN. The topology of the PLN has a three level hierarchy; user nodes at level one; local switching centers at level two; and tandem switching centers at level three. The path between two users is either direct or established via the local and the tandem switching centers.

PBX - Private Branch Exchange, is a switching equipment located at the user premises and serves as an interface between the user's basic equipment (phones, computers etc.) and the network.

CHANNEL	The channels available in any link come bundled in groups. In other words,
GROUPS	each group in a link contains a fixed number of channels and this is known as the <u>group capacity</u> . Since groups carry the communications traffic they are also referred to as transmission lines.

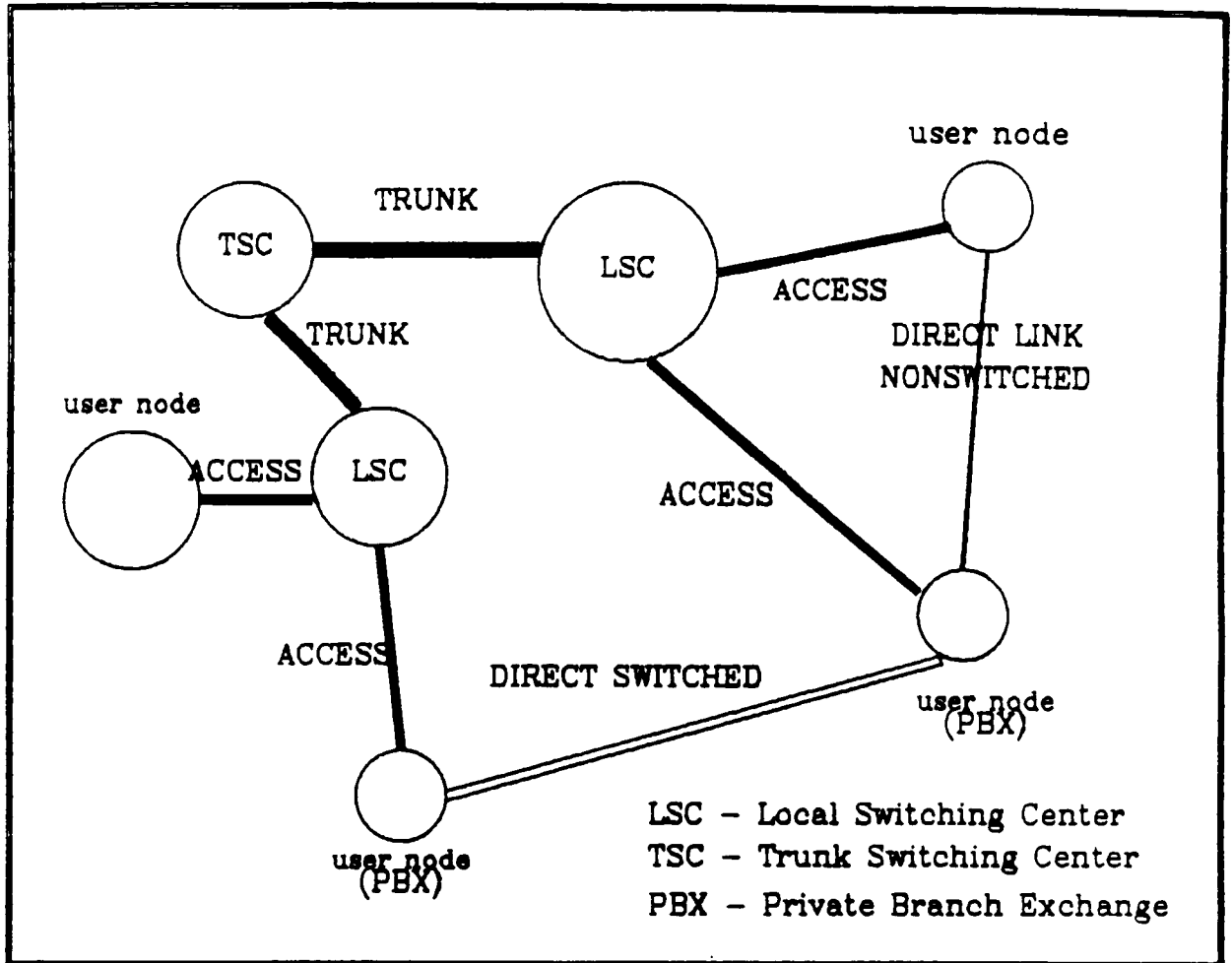


Figure 1.1: Topology of the PLN

FRAMES Switching systems interconnect channels in the adjoining links. The switching equipment are called frames, and a group attaches at each end to a frame at a port. The number of ports available per frame is defined as the frame capacity, and is predetermined. The frames located at a switching center are interconnected. The capacity of a switching center is expressed in number of frames.

DEMAND The demand for channels in a PLN is stochastic and dynamic. The length of
PATTERN time a demand exists has been found to follow a negative exponential distribution with a mean of 18 months (Garfinkel & Segal [11]). As the demands for the channels between the nodes change over time due to change in the current demand, the current topology of the network and the routing strategies adopted determine the new topology.

ROUTING The basic strategies used for routing internode flows in PLN are:

- STRATEGIES (a) direct route,
 (b) shortest available path, and
 (c) predetermined routes.

Direct route: flow between a pair of nodes is routed directly. The direct connection may be either non-switched or switched via PBX.

Shortest available path: each unit of flow is routed via the shortest available path in the existing network. In some of the links capacity may be augmented as required. For this routing strategy, all the intermediate nodes in the shortest available path will have to contain switching equipment whether or not they are demand nodes. In other words the intermediate nodes will be either local switching centers or tandem

switching centers or both.

Predetermined routes: for each flow, a set of acceptable routes in the existing network are given, and the possibility of routing the flows via one of these is investigated. If none of the paths have sufficient capacity to accommodate the flow, then additional capacity may be provided as necessary to route the flow.

BACKUP To ensure reliability, in addition to primary connections backup connections
CONNECTIONS between nodes are provided. This ensures that if the economical primary route is not available, then the backup route is used.

1.3 Hubbing in Private-Line Networks

HUBBING Hubs are selected points in the network, containing switching equipment, which aggregate and distribute flow between nodes. Since the pairwise node flows are usually less than the capacity of a transmission line, by aggregating flows from/to a node we can make efficient use of the capacity. Aggregation of flows enables the use of high capacity transmission lines. As the capacity of a transmission line increases cost per channel decreases thus leading to economies of scale. The cost of a transmission line is nonlinear in capacity (AT&T [4], page 138). (For a given capacity the cost is fixed.) In the hub structure, there is also savings in the number of links necessary to interconnect the nodes. Suppose, there are n nodes. If all the nodes were connected directly we will need a total of $\left[\frac{n(n-1)}{2} \right]$ links. In this instance the link capacities may not be utilized effectively. This is because the transmission lines have fixed capacity and the capacity in excess of the

demand will remain unutilized. If two hubs were located among these n nodes and all connections were established via the hubs then we will require only $(n-1)$ links, one from each node to its assigned hub, and one between the hubs. With hub routing the number of links required will decrease and also, due to aggregation of flows, there will be effective use of the link capacities.

In the hub network structure every node is assigned to one hub. Hub routing is a form of predetermined routing where the path between two nodes is either direct or via the respective hubs.

Other advantages associated with hub routing are: (a) all connecting paths will contain at most three links; (b) since switching is done only at designated points, there will be easier control over the connection process; (c) aggregation of flows at hubs will reduce vulnerability to forecast pairwise node flows. However, there are disadvantages associated with the hub routing. Since hub routes are indirect, the connections have to be switched and there will be a need for more switching. Also, this means that some of the routes may be less direct than with other routing strategies.

1.4 The General Problem

In this section we describe the general research problem based on using hubs in the PLN.

TERMINOLOGY In order to be consistent with standard OR terminology the following terminology will be used. Every demand point in the network will be called a node. Communications demand between nodes will also be referred to as (communications) flow. A link is a direct connection between two nodes. Groups will also be referred to as arcs. Frames and ports are as defined before.

AGGREGATE We consider a two level hierarchy for our network - nodes and hubs. Each
NODES network node in the general problem will be an aggregate node - a collection of smaller nodes. This level of aggregation in modeling does not distort the model from reality very much and has been applied previously (AT&T [4], and Garfinkel & Segal [11]). The demand at each node is therefore aggregated from smaller nodes. Each node may therefore contain frames and be a possible site for a hub.

We now define the general problem.

GENERAL There exists a collection of nodes with known coordinate locations. The flows
MODEL are given in number of channels, and are assumed to be undirected, deterministic, and static over some planning period. The matrix of direct distances between the nodes is given and is symmetric. A selected number of hubs - specified by the decision maker, are to be located at a subset of the

nodes. Each node will be assigned to one hub only. In other words, the nodes are partitioned into sets based on their assignment to hubs. This assignment means that the flow from(to) a node may be aggregated(distributed) only at(from) the hub to which it is assigned. A trunk is defined as a link between two hubs, an access is a link between a node and its hub, and direct link is a link between two nonhub nodes. Flow in any link is bundled into groups of predetermined capacity. In standard network terminology, one could equivalently think of multiple arcs each with the same fixed capacity. We will use the terms groups and arcs interchangeably. Like flow, group capacity is also expressed in number of channels, and may be different for each link. Since the volume of flow in any trunk link is likely to be more than the flow in any access link or direct link, a higher arc capacity in the trunk links will provide economies of scale. Where the arc capacities would be different, the arcs in trunk links will have, comparatively, the largest capacity, the arcs in the direct links the smallest, and the arcs in the access links in between. Each group connects to frames at each end, and occupies one port on each frame. The capacity of a node is expressed in number of frames, with each frame having a fixed number of ports. The capacities of nodes and links are assumed unlimited. Reliability considerations will be ignored in this study. Therefore, the connections between the nodes service the demand, without backup.³

³Backup network will be considered in future research and is discussed in Chapter 7.

RELEVANT	The cost associated with providing the necessary connections are:
COSTS	fixed node cost: fixed costs associated with buildings to house the switching equipment; <i>variable node cost</i>: variable costs associated with the number of frames at each node; <i>connection cost</i>: one time fixed cost associated with providing a direct connection between two nodes; <i>arc cost</i>: variable costs based on the number of arcs in a link, and the link distance; <i>port utilization cost</i>: fixed cost of connecting an arc to a port of a frame.

Calculating number of ports required at each node:

All nodes are aggregate nodes and contain frames. Since the network nodes are aggregate nodes, we do not know the number of groups extending from the small nodes to a network node. The capacity of a frame is assumed the same at all the nodes. We estimate the number of ports required at a node by the following expression:

$$\text{number of arcs connecting at the node} + \left\lceil \frac{\text{total flow from the node}}{\text{minimum arc capacity}} \right\rceil * \alpha,$$

where α is a user specified parameter ($\alpha \geq 0$) and $\lceil a \rceil$ denotes the smallest integer greater than or equal to a .⁴

Given the frame and the arc capacity, it will be reasonable to assume that the frame costs and arc costs will be linear i.e., for example frame cost at a

⁴Garfinkel and Segal [11] use a similar scheme to estimate the number of ports required at an aggregate node.

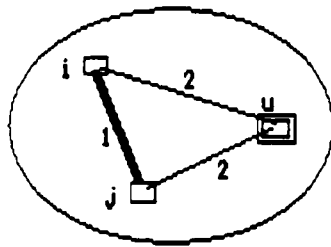
node = number of frames * cost per frame; arc costs = number of arcs * link distance * cost/arc/unit distance. Similarly, port utilization costs may be assumed to be linear: port utilization costs = number of ports utilized * cost per port.

ROUTING While there are many strategies to route the flows (connect the nodes) we
STRATEGIES shall focus attention on two specific ones. Flow between any two nodes may
be routed either,

- a. direct - nodes are connected directly, or
- b. via their assigned hubs - nodes are connected via the hubs to
which they are assigned.

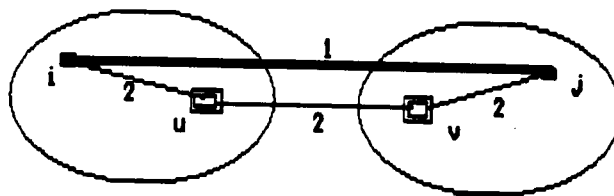
One or a combination of these two strategies may be used in routing all the flows. For example, consider the flow between nodes *i* and *j*. Suppose, both nodes are assigned to the same hub at location *u*. Then, the flow between *i* and *j* may be routed either directly *i-j* or via hub *u* *i-u-j*. If nodes *i* and *j* are assigned to hubs at locations *u* and *v*, respectively, the flow between them may be routed either directly *i-j* or via the two hubs, *i-u-v-j*. See figure 1.2 for an illustration of these strategies. According to these connection strategies there will be at most three links in any connection.

PLANNING The objective is to determine the minimum cost topology of the network
OBJECTIVES using hubs, satisfying the flow requirements. Determining the minimum cost
topology will involve: finding locations of the hubs; assigning nodes to hubs;
assigning links and link capacities to provide the necessary connections
between the nodes; and assigning frame capacities. In this design we assume
that there is no existing network.



- 1 - direct path $i-j$
- 2 - via hub path $i-u-j$

a: Routing when nodes i and j belong to the same partition



- 1 - direct path $i-j$
- 2 - via hub path $i-u-v-j$

b: Routing when nodes i and j belong to different partitions

Figure 1.2 Routing when nodes i and j belong to the same or different partitions

1.5 Research Focus

RESEARCH The research presented also considers three scenarios of the general problem
FOCUS (GP). The objective in all three problems is to develop a minimum cost
 topology using hubs. We shall refer to the three situations as P1, P2, and P3
 and these are described next. For each of the three situations only those
 aspects different from the general problem are described.

1.5.1 Problem P1

DESCRIPTION In this situation, we are given a partition of the n nodes into p nonempty
 subsets. Exactly one hub is to be located in each partition at a node. All
 flows will be routed via the hubs. Thus there will be links only between the
 nodes and their respective hubs, and between the hubs. In this network there
 will be a total of $\left[(n - p) + \frac{p(p - 1)}{2} \right]$ links.

OBJECTIVE The objective is to determine the locations for the hubs such that
 the total network cost is a minimum.

NETWORK The only relevant cost for this problem is the distance based cost for the
COSTS groups. Since the number of hubs are prespecified, the fixed hub costs are a
 constant. By the routing strategy all connections are via the hubs; hence the
 number of groups in the access and trunk links can be determined from the
 given flows and the group capacity. From this, the frame capacities at the
 nodes can be determined. Thus the only variable cost is the arc costs. The
 hub costs, frame costs, port utilization costs, and connection costs are all
 constant for a given problem. Locations of the hubs will determine the arc
 costs.

1.5.2 Problem P2

DESCRIPTION The problem is essentially the same as P1 except that there are no predefined partitions of the nodes. Given the number of hubs and the same routing strategy, the objective is to simultaneously locate the hubs and assign each node to exactly one hub. With the number of hubs given and all routes being via the hubs, the hub costs, and the connection costs are constant for the problem. The total number of links in the network will be the same as in P1.

NETWORK COSTS The variable costs for the problem are: (a) frame costs, (b) arc costs, and (c) port utilization costs.

OBJECTIVE The objective is to simultaneously partition the nodes, and locate a hub in each partition so as to minimize the total network cost.

1.5.3 Problem P3

DESCRIPTION In this case, we are given a specified number of partitions and the location of the hub in each partition. The total flow between any two nodes will be routed either directly or via their respective hub(s). The arc capacity may be different in different links, but is the same within a link. Frame capacity is assumed the same at all nodes. The total number of links in this network is at most $\left\lceil \frac{n(n-1)}{2} \right\rceil$.

OBJECTIVE The objective is to determine the flow routing so as to minimize the total cost.

RELEVANT COSTS Since the number of partitions is given, the hub costs are constant for the problem. The relevant variable costs are: (a) connection costs; (b) arc costs; (c) frame costs; and (d) port utilization costs. It will be shown in section 5.3 that cost factors a, b, and d may be combined to produce a single cost measure per arc for each link.

1.6 Organization of Research Presentation

This work is organized as follows. Chapter 1 introduced the reader to the research theme, described the motivating situation and the general research problem based on it. Then we developed three scenarios of the general problem which will be the research focus. In Chapter 2, we discuss the work on related problems, and compare and contrast this research with the related problems. In Chapters 3, 4, 5 and 6 we describe the three research problems and the general problem in detail, consider their complexity, and discuss solution methods for each. Also, we describe lower bound procedures to measure the effectiveness of the heuristic algorithms and report the computational experience with the solution procedures. Chapter 7 contains a discussion of some issues related to network planning, and extensions of this research. The presentation concludes with the listing of the references, illustrative examples, and the computer programs for the solution methods.

CHAPTER 2

LITERATURE REVIEW

Some examples of the contexts in which the problem of simultaneous location and routing manifests are:

- locating hubs in telecommunications networks,
- locating homing stations in satellite communications networks,
- locating terminal concentrators in computer networks, and
- locating sorting hubs for express package delivery.

Several researchers have studied the hierarchical terminal concentrator location problem in computer communications networks. For example, see Ahuja [1], Gavish [14], and Mirzaian [24]. The problem is how to connect several computer terminals to the hosts at a central site. The approach is to use concentrators. Given a set of potential concentrator sites, usually a subset of the terminal locations, the task is to select a subset to be the actual concentrator sites. Each terminal is connected to a concentrator, and the concentrators are connected to the hosts at the central site. Decisions involved in this problem are:

- a. number of concentrators,
- b. assignment of terminals to concentrators,
- c. layout of terminals associated with a particular concentrator,
- d. layout of concentrators,
- e. capacities of links connecting terminals to concentrators, and concentrators to hosts, and
- f. capacities of concentrators.

In cases where the concentrator capacity is not limited, each terminal will be assigned to

the nearest concentrator. Usually the layout of terminals associated with a concentrator and the layout of concentrators is either a tree or a star. Variations of the simple problem situation are obtained by limiting,

- a. the number of nodes assigned to a concentrator,
- b. the number of arcs incident at a node, and
- c. the number of direct connections to the hosts.

The terminal concentrator problem departs from ours in the following aspects.

1. Type of flow - demands are created between the nodes and the hosts, rather than between the nodes.
2. The cost function does not address the sizes of the links and concentrators.
3. The flow routing strategy - layout of terminals associated with a concentrator can be a tree.
4. Absence of fixed capacity arcs in the links.

O'Kelly [25, 26, 27] considers the situation of locating sorting hubs for express package delivery systems. The flows in this case are the volume of package traffic between the cities, and the sorting hubs are used to aggregate and distribute the flow. All flows are via the hubs. The objective is to determine the hub locations and the network links such that the total cost of flow is a minimum. This problem is similar to P2 except that it is only concerned with hub locations and the network links, and not the link capacities. This is because the link and the hub capacities in the express package delivery problem are assumed to be continuous. As in P2, all flows are routed via the respective hubs. The solution approach suggested by O'Kelly is to evaluate all the possible hub locational configurations with the additional assumption that every node will be assigned to its closest hub. Thus, the determination of network links becomes a byproduct of the locational configuration. For an n -node, p -hub problem the number of hub locational

configurations equal ${}^nC_p$. This is of order $O(n^P)$. As the problem size grows totally enumerating all locational configurations becomes computationally infeasible. O'Kelly shows that as the magnitude of interaction between two hubs decreases, the distance between the two locations may increase.

The problem of locating hubs in communications networks and locating homing stations in satellite communications networks are similar with the exception that communications between the hubs use terrestrial links in the former, and satellite links in the latter. Homing stations are similar to hubs, and they switch local and satellite connections. For example see Garfinkel and Segal [11], Garfinkel and Smith [12], Gomory and Hu [18], and Helme and Magnanti [21].

Helme and Magnanti consider a situation similar to P2 with the following differences:

- a. since the interaction between the hubs is via satellite link there is no distance based cost for satellite link;
- b. the capacity of terrestrial links and homing stations is assumed to be continuous rather than fixed as in P2; and
- c. for each node a list of possible hub sites to which the node can attach is given.

They use branch and bound and heuristic approaches on an integer linear programming formulation of the problem.

Garfinkel and Segal [11] compare the hub routing strategy with other routing strategies mentioned in Chapter 1 (page 1-7) using a simulation model. Given a backbone network they consider expanding the network to route the new flow between nodes. When expanding the network link capacities are increased and links are added to the backbone network, as necessary.

Gomory and Hu consider the design of a network to handle time varying demand. In this problem the shortest path between nodes is considered as the routing strategy. The routing strategy, continuous link capacities, absence of hubs and directed flows set this problem different from that considered by Garfinkel and Segal [11]. They use a linear programming approach for solution.

Garfinkel and Smith [11] consider a 2-partition instance of problem P3 and derive theoretical results for the optimal location and routing decisions. As in P3 the flow routing is either direct or via the respective hubs. However, in this instance intra-partition flows are taken to be zero. In other words, flows exist only across the partitions.

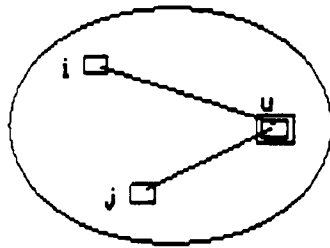
Problems P2 is in some sense a combination of the uncapacitated location allocation problem (ULAP) (Cooper [7]) and the Quadratic Assignment Problem (QAP) (Francis and White [8] Chapter 5). In the location-allocation problem a selected number of facilities are to be located at a subset of the nodes and each node is to be assigned to a facility. If the capacity of the facilities is unlimited each node will be assigned to the closest facility. There is interaction only between the nodes and their assigned facility. In the Quadratic Assignment Problem interaction between the facilities is considered. Since P2 involves both type of interactions we may consider this as a combination of the ULAP and the QAP.

NP-Completeness results for this class of problems have not been developed, and this study attempts to show relation of the research problems to instances of other difficult problem.

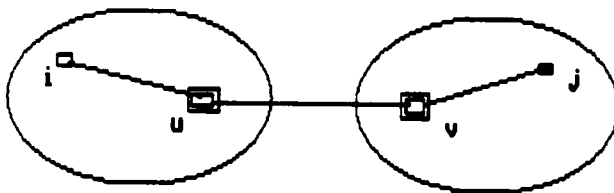
CHAPTER 3
PROBLEM P1:
MATHEMATICAL FORMULATION
AND
ALGORITHMS

3.1 Problem Statement

PROBLEM Given a collection of n nodes in a plane, partitioned into p subsets. The co-
STATEMENT ordinates of the nodes are known. The matrix of flows between the nodes, expressed in number of channels, is given and is symmetric and undirected. There is no flow from a node to itself. Thus, every node is considered as an origin and destination of flows. The matrix of direct distances between nodes is given, and is symmetric and metric. Exactly one hub is to be located in each partition at a node. The routing strategy is that all flows will be routed via the respective partition hubs. In other words, connection between any two nodes is established only via their respective partition hub(s). Suppose nodes i and j belong to the same partition and the hub for the partition is located at u . Then the flow between i and j will be routed along the path i - u - j . Suppose, i and j are located in two different partitions, and u and v are the hub locations respectively. The flow between i and j will be routed via the path i - u - v - j . Figure 3.1 illustrates the above possibilities. A network link is defined as a direct connection between any two nodes. By the routing strategy there will be links only between the nodes and their partition hubs (access links), and between the hubs (trunk links). Flow along any link is bundled into groups of fixed capacity. The capacity of a group is expressed in number of channels. Equivalently, one can consider this as multiple arcs with the same fixed capacity. Since all the flows are via the hubs, the



a: Routing via hub when nodes i and j belong to the same partition



b: Routing via hub when nodes i and j belong to different partitions

Figure 3.1 Routing via hub when nodes i and j belong to the same or different partitions

number of groups in the intra- and inter-partition links can be determined given the flows, and the link group capacities. Hence, for simplicity without loss of generality we could assume one arc capacity for the access links and another for the trunk links.

RELEVANT COSTS Since the number of groups in each link is determined from the information given, the number of groups connecting at a node is also known. Due to the routing strategy, the total number of connections will be exactly $\left[(n-p) + \frac{p(p-1)}{2} \right]$. Therefore, the hub costs, the connection costs, the frame costs, and the port utilization costs are fixed for the problem, given the number of hubs, pairwise node flows, and the group capacity. The only cost that will vary with the location of the hubs is the group distance costs based on the links. Therefore, this is the only relevant cost for the problem.

OBJECTIVE We have to determine the hub locations which minimize the total group distance costs.

An integer linear programming formulation of the problem is presented next.

3.2 Mathematical Formulation

MATHEMATICAL Integer Linear Programming Formulation of P1

FORMULATION

Notation:

Q^A - Arc capacity in channels in the access links

Q^T - Arc capacity in channels in the trunk links

P - Partition index set; $|P| = p$

S_l - Nodes in partition l

k, l - Partition indices

N - Node set; $|N| = n$

n_k - Number of nodes in partition k

i, j - Node indices

b_{ij} - Flow between nodes i and j

Z_i - Number of groups from node i to its partition hub

$$= \left\lceil \frac{\sum_{j \in N} b_{ij}}{Q^A} \right\rceil, \text{ where } \lceil a \rceil \text{ is the smallest integer greater than or equal to } a$$

W_{kl} - Number of groups between partitions k and l

$$= \left\lceil \frac{\sum_{i \in S_k} \sum_{j \in S_l} b_{ij}}{Q^T} \right\rceil, \text{ where } \lceil a \rceil \text{ is the smallest integer greater than or equal to } a$$

$X_i = 1$ if hub is located at node i

$= 0$ otherwise

$X_{ij} = 1$ if hubs are located at nodes i and j

$= 0$ otherwise

c^A - Cost per group per unit distance in the access links

c^T - Cost per group per unit distance in the trunk links

d_{ij} - Distance between nodes i and j

(ILP1)

Minimize

$$\begin{aligned} & \sum_{k \in P} \sum_{j \in S_k} \left(\sum_{i \in S_k} z_i c^A d_{ij} \right) X_j \\ & + \sum_{\substack{k, l \in P \\ k < l}} \left(W_{kl} \left(\sum_{i \in S_k} \sum_{j \in S_l} (c^T d_{ij} X_{ij}) \right) \right) \end{aligned} \quad (3.1)$$

Subject to

$$\sum_{i \in S_k} X_i = 1 \quad \forall k \in P \quad (3.2)$$

$$\sum_{j \in S_k} X_{ij} = X_i \quad \forall i \in S_l \ni k, l \in P \text{ and } k \neq l \quad (3.3)$$

$$X_i, X_{ij} \geq 0 \text{ and integer } \forall i \in S_k, j \in S_l \ni k, l \in P: k \neq l \quad (3.4)$$

In our discussion the constraints (3.2) - (3.4) will be symbolically represented by $(Ax=b, Bx=d, x \geq 0 \text{ \& integer})$. Each element of the matrix A and B is 1, 0, or -1. The dimension of matrix A is $[p, n + \sum_{\substack{k, l \in P \\ k < l}} n_k n_l]$; dimension of matrix B is $[n(p-1),$

$\sum_{\substack{k, l \in P \\ k < l}} n_k n_l]$; dimension of vector x is $[n + \sum_{\substack{k, l \in P \\ k < l}} n_k n_l]$; dimension of vector b is

$[p]$; and dimension of vector d is $[n(p-1)]$.

The first set of terms in the objective function (3.1) measure the total distance cost within each partition, assuming that the partition hub is located at node j . The second set of terms measure the inter-partition distance costs, assuming that nodes i and j are the hub locations in the respective partitions. The constraint set (3.2) is a multiple choice set which defines that there will be only one hub location per partition. The constraint set (3.3) expresses that inter-partition links from a node i in partition S_l to partition S_k can only exist if i is a hub location. The number of such inter-partition links from node i to partition S_k will be either 0 (if i is not a hub) or 1 (if i is a hub). A form of this constraint is used by Glover-Woolsey [17] to convert 0-1 polynomial programs to 0-1 linear programs. It must be recognized that (3.3) is equivalent to $X_{ij} = X_i X_j$. Finally, the variables X_i and X_{ij} are specified as integer. Note that the implicit upper bound on all variables is 1.

3.3 Complexity of P1

We establish that problem P1 belongs to the class of NP-Complete problems by transforming an instance of the 3-dimensional matching problem(3DM) to P1.

Given an instance of the 3DM problem with three sets (I, J, K) of n items each, the objective is to determine the groupings of three items, one from each set, such that the total value of the matchings is maximum. A mathematical programming statement of the 3DM problem is as follows:

$$\text{Maximize } \sum_{i \in I} \sum_{j \in J} \sum_{k \in K} C_{ijk} X_{ijk}$$

subject to

$$\sum_{j \in J} \sum_{k \in K} X_{ijk} = 1 \quad \forall i \in I$$

$$\sum_{i \in I} \sum_{k \in K} X_{ijk} = 1 \quad \forall j \in J$$

$$\sum_{i \in I} \sum_{j \in J} X_{ijk} = 1 \quad \forall k \in K$$

$$X_{ijk} \in \{0, 1\}$$

where $|I| = |J| = |K| = n$; $C_{ijk} (\geq 0)$ is the value of the feasible match between items i , j , and k ; and X_{ijk} is 1 if the match between items i , j , and k is realized, zero otherwise.

The 3DM problem is NP-Hard (Garey & Johnson[10]). We shall transform an instance of the 3DM problem to an instance of P1 and thus show that P1 is at least as hard as the 3DM problem to solve. Let the n items in the three sets (I , J , and K) be labeled as follows: $\{x_1, x_2, \dots, x_n\}$, $\{y_1, y_2, \dots, y_n\}$, $\{z_1, z_2, \dots, z_n\}$, respectively.

The instance of P1 obtained by transformation of the instance of the 3DM problem has the following properties:

- . $3n$ partitions corresponding to the $3n$ items
- . Each partition has n^2 nodes corresponding to the n^2 feasible matches for each item - for example a node in partition x_i will be labeled as $x_i: y_j z_k$ indicating that the node corresponds to a feasible match between x_i , y_j , and z_k .

. Zero intra-partition cost

. Each inter-partition arc will connect either nodes corresponding to a feasible match - for example the arc connecting the node pair $x_i: y_j, z_k$ - $y_j: x_i, z_k$ which corresponds to the match between x_i , y_j and z_k , or those not corresponding to a feasible match - for example the arc connecting the node pair $x_i: y_j, z_k$ - $x_i: y_j, z_k$. The inter-partition arc costs are defined as follows. Those arcs connecting the nodes corresponding to a feasible match, for example the arcs connecting the node pairs $x_i: y_j, z_k$ - $y_j: x_i, z_k$, $x_i: y_j, z_k$ - $z_k: x_i, y_j$, and $y_j: x_i, z_k$ - $z_k: x_i, y_j$, which correspond to a feasible match between items x_i , y_j , and z_k , will each have a cost of $(A - C_{x_i y_j z_k})/3$, where A is the smallest integer greater than maximum $C_{x_i y_j z_k}$. The inter-partition arcs connecting the nodes which do not correspond to a feasible match, for example the arcs connecting the node pairs $y_j: x_i, z_k$ - $z_i: x_i, y_j$ or $x_i: y_j, z_k$ - $x_i: y_j, z_k$ will have a cost of M (M is a large value; $M > nA$).

Let $\nu(3DM)$ be the value of the optimal solution to the 3DM problem. In the optimal solution to the instance of P1 we will have $3n$ inter-partition arcs connecting the nodes corresponding to the n feasible marriages. The total cost of these arcs will be $(nA - \nu(3DM))$. The total number of inter-partition arcs is $\frac{3n(3n-1)}{2}$. Then, the total cost of inter-partition arcs connecting nodes not corresponding to feasible matches will be

$$\left(\frac{3n(3n-1)}{2} - 3n \right) \cdot M = \frac{9n(n-1)}{2} \cdot M.$$

Thus, the value of the optimal solution to the instance of P1 derived from the 3DM problem, will be $\left(nA - \nu(3DM) + \frac{9n(n-1)}{2} \cdot M \right)$. The node pairs

corresponding to feasible matches between which inter-partition arcs exist in the optimal solution to P1 will correspond to the optimal matches in the solution to the 3DM problem. Therefore an optimal solution to the instance of P1 with objective value less than $\left(\frac{9n(n-1)}{2} \cdot M + M\right)$ will correspond to an optimal solution to the instance of the 3DM problem. Hence, solution to this instance of P1 obtained by transforming the given instance of the 3DM problem will be at least as hard as the solution to the 3DM problem.

Given the instance of the 3DM problem we have to create $3n$ partitions, $3n^3$ nodes, and assign $\frac{3n^5(3n-1)}{2}$ inter-partition arc costs based on feasible and infeasible matchings. Thus, the transformation from the 3DM problem to P1 is polynomial.

Thus, we can say that P1 also belongs to the class of NP-Complete problems.

An example illustrating the transformation is given in appendix A1.

3.4 Extreme Point Property

Here we examine the LP obtained from the ILP1. Let, the LP obtained by replacing (3.4) with

$$X_i, X_{ij} \geq 0 \quad \forall i, j \in N \text{ and } k, l \in P \ni k \neq l, i \in S_k \text{ \& } j \in S_l \quad (3.4a)$$

in the ILP1 be denoted by LP1.

Theorem

The optimal solution to ILP1 occurs at an extreme point of the polyhedron associated with its LP relaxation.

Proof

Due to (3.2), (3.3), and (3.4a) the implicit upper bound on all X variables is

1. Hence, the following statements are true.

- (a) The polyhedron contains no integer feasible solutions in its interior. The integer points are at the corners of the unit hypercube.
- (b) The polyhedron is contained within the unit hypercube because the implicit upper bound on all the X variables is 1.

Due to (a), and (b) a feasible integer solution must occur at an extreme point. This implies that the optimal integer solution is also an extreme point of the polyhedron. □

This is similar to the result for the set partitioning problem reported by Andrew, Hoffman & Krabek [3], and that for zero-one problems reported by Taha [30].

ILP1 is not totally unimodular. However, we observe that if there is symmetry with respect to the flow between nodes and, the spatial distribution of the nodes the optimal solution to the LP relaxation is not likely to be integer. Otherwise, the optimal solution to LP1 is likely to be integer. We illustrate this by examples in appendix A2.

3.5 Exact Solution Approach

In cases where the optimal LP solution is noninteger cutting plane or branch and bound procedures may be used to determine the optimal integer solutions. However, we will not pursue this further in this work.

A direct but computationally time consuming exact solution approach will be to totally enumerate all the possible combinations of hub locations. Suppose the number of nodes in the p partitions are $n_1, n_2, \dots, n_p$ respectively. Then, the total number of combinations of possible hub locations will be equal to $\prod_{k=1}^p n_k$. For a n -node p -partition problem the maximum number of such possible combinations will be $(n/p)^p$. This can be proved easily. Thus the time required for solution by total enumeration is exponential in the number of partitions. For a two partition case solution by total enumeration will involve evaluating at most $(n/2)^2$ combinations.

We next discuss a lagrangian relaxation of P1, and the special structure of the two partition case.

3.6 Lagrangian Relaxation of P1

MOTIVATION It is our observation that in the absence of symmetry in the spatial distribution of nodes and the flow between nodes, the solution to the LP relaxation will most likely be integer. In these situations a lagrangian relaxation of P1 can also be expected to yield integer solution.

A lagrangian relaxation of P1 is obtained by selecting constraint (3.2) as the active constraints and adding the product of constraint (3.3) and a vector λ to the objective function yielding:

LR-P1

$\nu(\lambda) = \text{Minimize}$

$$\sum_{k \in P} \sum_{j \in S_k} \left(\sum_{i \in S_k} z_i c_{ij} \right) X_j + \sum_{\substack{k, l \in P \\ k < l}} w_{kl} \left(\sum_{i \in S_k} \sum_{j \in S_l} (c_{ij} X_{ij}) \right) + \sum_{l \in P} \sum_{i \in S_l} \sum_{\substack{k \in P \\ k \neq l}} \lambda_{ik} \cdot \left(X_i - \sum_{j \in S_k} X_{ij} \right) \quad (3.5)$$

Subject to

$$\sum_{i \in S_k} X_i = 1 \quad \forall k \in P \quad (3.6)$$

$$\left. \begin{array}{l} X_i \geq 0 \text{ and integer} \\ X_{ij} \in \{0,1\} \end{array} \right\} \forall i \in S_k, j \in S_l \ni k, l \in P: k \neq l \quad (3.7)$$

This can be symbolically represented as:

$$\nu(\lambda) = \min Cx + \lambda (b - Ax)$$

subject to

$$Bx = d$$

$$x \in \{0,1\}$$

By removing constraint (3.3) from the active set we eliminate the implicit upper bound on the X_{ij} variables. Therefore to maintain their upper bound we add the constraint (3.7) imposing the binary requirement $X_{ij} \in \{0,1\}$.

Since the active constraints (3.6) are of the multiple choice type, given the values of the multipliers λ_{ik} LR-P1 can be easily solved by inspection. The value of the solution $\nu(\lambda)$ to the lagrange relaxation depends on the value of λ . In the solution those X_i variables with the least objective function

coefficients from each set will be fixed at 1; those X_{ij} variables with negative coefficients will be set to 1 and the rest to zero.

It is well known that $\nu(\lambda) \leq \nu(\text{ILP1})$. This is easy to show by assuming an optimal solution x^* to (ILP1) and observing that

$$\nu(\lambda) \leq Cx^* + \lambda(b - Ax^*) = Cx^* = \nu(\text{ILP1}).$$

The inequality in this relation follows from the definition of $\nu(\lambda)$ and the equality from $\nu(\text{ILP1}) = Cx^*$ and $Ax^* - b = 0$. In order to determine the best bound the best choice for λ would be an optimal solution to the dual problem

$$\nu(\lambda^*) = \max_{\lambda} \nu(\lambda) \quad (\text{D})$$

The following relationship holds between the LP relaxation of P1, lagrangian relaxation of P1, and the ILP1:

$$\nu(\text{LP1}) \leq \nu(\lambda^*) \leq \nu(\text{ILP1}).$$

The relationship between the linear programming bound and the lagrangian bound was shown by Geoffrion [15]. Since the feasible solution set of LR-P1 has the integrality property we have $\nu(\lambda^*) = \nu(\text{LP1})$ (see Parker & Rardin [28], page 211); the best bound the lagrangian relaxation can provide is the same as the LP bound.

In practice a good set of multipliers is located by the subgradient algorithm (Held et al. [20]). The subgradient method is an adaptation of the gradient method in which gradients are replaced by subgradients. Given an initial vector λ a sequence $\{\lambda^k\}$ is generated by the rule

$$\lambda^{k+1} = \lambda^k + t_k (Ax^k - b)$$

where x^k is an optimal solution to $\nu(\lambda^k)$ and t_k is a positive scalar step size.

The step size used most commonly in practice is

$$t^k = \frac{d_k (\nu_f - \nu(\lambda^k))}{\|Ax^k - b\|^2}$$

where ν_f is an upper bound on $\nu(\lambda)$, frequently obtained by applying a heuristic to IPL1, and d_k is a scalar satisfying $0 < d_k \leq 2$. The sequence d_k is determined by setting d_0 to 2 and halving d_k whenever $\nu(\lambda)$ fails to improve in the last 20 iterations. Usually $\lambda^0 = 0$ is the natural choice.

An example of the above lagrangian approach is given appendix A3.

One may also consider selecting constraint set (3.3) as the active constraints and adding the product of constraint (3.2) and a vector λ to the objective function. However the resulting problem does not provide a relatively easy structure for solution. Also, this alternative relaxation (which we will call LR2-P1) has the integrality property and hence the best bound obtained by this relaxation will be the same as the LP bound.

3.7 2-Partition Case: A Special Structure

2-PARTITION CASE For a two partition case, the problem P1 reduces to the shortest path problem. For example, consider the instance where each partition contains two nodes as shown in figure 3.2a. The formulation of this problem instance is as follows:

$$\text{Let } C'_j = \sum_{i \in S_k} Z_i C_{ij} \quad \text{and} \quad C'_{ij} = W_{kl} C_{ij}$$

Minimize

$$C'_1X_1 + C'_2X_2 + C'_3X_3 + C'_4X_4 + C'_{12}X_{13} + C'_{12}X_{14} + C'_{12}X_{23} + C'_{12}X_{24} \quad (3.8)$$

subject to

$$X_1 + X_2 = 1 \quad \text{- network terminal node 1} \quad (3.9)$$

$$X_3 + X_4 = 1 \quad \text{- network terminal node 2} \quad (3.10)$$

$$X_{13} + X_{14} - X_1 = 0 \quad \text{- network intermediate node 3} \quad (3.11)$$

$$X_{23} + X_{24} - X_2 = 0 \quad \text{- network intermediate node 4} \quad (3.12)$$

$$X_{13} + X_{23} - X_3 = 0 \quad \text{- network intermediate node 5} \quad (3.13)$$

$$X_{14} + X_{24} - X_4 = 0 \quad \text{- network intermediate node 6} \quad (3.14)$$

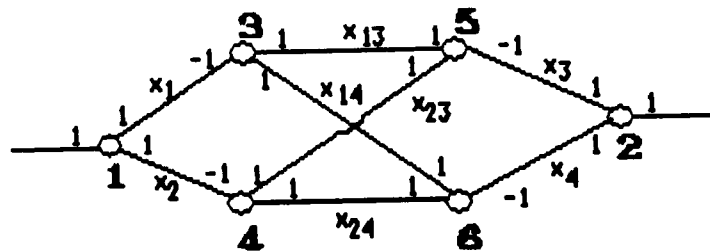
$$X_i, X_{ij} \geq 0 \text{ and integer} \quad (3.15)$$

The network representation of this is given in figure 3.2b. The X variable represents the flow along the arcs, and the objective function coefficients of the X variables represent the network arc lengths as shown in 3.2b. The solution to this problem will be the shortest path between terminal nodes 1 and 2.

In the above two partition problem the two choice constraints (3.9) and (3.10) correspond to the start/terminal nodes, and each of the other four constraints corresponds to an intermediate node of the network. The cost coefficients given in the objective function represent the network arc lengths. The optimal solution to the 2-partition hub location problem can be determined by finding the shortest distance between the two terminal nodes of the network. For a network with n -nodes, the optimal solution procedure to find the shortest path between two nodes will have a complexity of $O(n^2)$ (Lawler [23]).



a: A 2-partition example



b: Network representation of the above 2 partition case

3.2 A two partition example and its network representation

For a general two partition problem in n nodes with n_1 and n_2 being the number of nodes in each partition, there will be two terminal nodes, and n intermediate nodes. The complexity of solution will remain $O(n^2)$.

GENERAL CASE For a problem with more than two partitions the network representation does not yield a simple structure as in the 2-partition case, for an easy solution. We illustrate the above statement by a 3-partition example with two nodes in each partition. The network representation of the a 3-partiton problem with two nodes in each partition is as follows.

Minimize

$$C'_1X_1 + C'_2X_2 + C'_3X_3 + C'_4X_4 + C'_5X_5 + C'_6X_6 + C'_{12}X_{13} + C'_{12}X_{14} + C'_{13}X_{15} + C'_{13}X_{16} + C'_{12}X_{23} + C'_{12}X_{24} + C'_{13}X_{25} + C'_{13}X_{26} + C'_{23}X_{35} + C'_{23}X_{36} + C'_{23}X_{45} + C'_{23}X_{46} \quad (3.16)$$

subject to

$$X_1 + X_2 = 1 \quad - \text{ network terminal node 1} \quad (3.17)$$

$$X_3 + X_4 = 1 \quad - \text{ network terminal node 2} \quad (3.18)$$

$$X_5 + X_6 = 1 \quad - \text{ network terminal node 3} \quad (3.19)$$

$$X_{13} + X_{14} - X_1 = 0 \quad - \text{ intermediate network node 4} \quad (3.20)$$

$$X_{15} + X_{16} - X_1 = 0 \quad - \text{ intermediate network node 4} \quad (3.21)$$

$$X_{23} + X_{24} - X_2 = 0 \quad - \text{ intermediate network node 5} \quad (3.22)$$

$$X_{25} + X_{26} - X_2 = 0 \quad - \text{ intermediate network node 5} \quad (3.23)$$

$$X_{13} + X_{23} - X_3 = 0 \quad - \text{ intermediate network node 6} \quad (3.24)$$

$$X_{35} + X_{36} - X_3 = 0 \quad - \text{ intermediate network node 6} \quad (3.25)$$

$$X_{14} + X_{24} - X_4 = 0 \quad - \text{ intermediate network node 7} \quad (3.26)$$

$$X_{45} + X_{46} - X_4 = 0 \quad - \text{ intermediate network node 7} \quad (3.27)$$

$$X_{15} + X_{25} - X_5 = 0 \quad - \text{ intermediate network node 8} \quad (3.28)$$

$$X_{35} + X_{45} - X_5 = 0 \quad \text{- intermediate network node 8} \quad (3.29)$$

$$X_{16} + X_{26} - X_6 = 0 \quad \text{- intermediate network node 9} \quad (3.30)$$

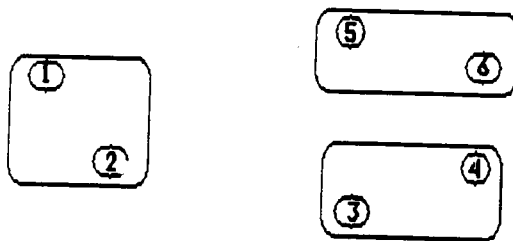
$$X_{36} + X_{46} - X_6 = 0 \quad \text{- intermediate network node 9} \quad (3.31)$$

$$X_i, X_{ij} \geq 0 \text{ and integer} \quad (3.32)$$

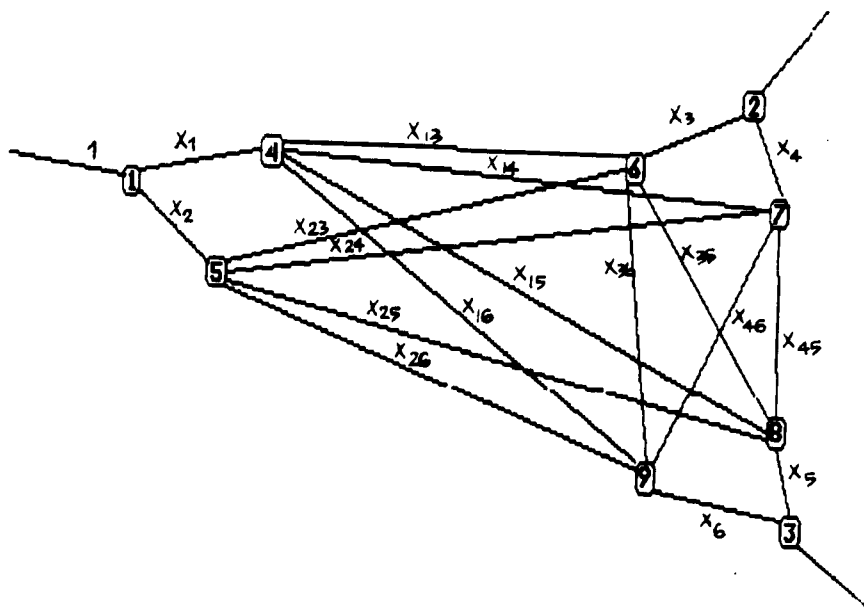
Figure 3.3 illustrates this network. The cost coefficients of the X_{ij} variables in the objective function represent the network arc lengths as indicated in figure 3.3. Network nodes 1, 4 and 5 are in one cluster; 2, 6, and 7 are in a second cluster; and 3, 8 and 9 are in a third cluster. In terms of the general network the optimal solution to the 3-partition problem will be to find one intermediate node from each cluster such that the sum of arc lengths from the intermediate nodes to their respective terminal nodes, and the lengths of arcs connecting the selected intermediate nodes is a minimum. The solution to the 3-partition may also be described as finding the simultaneous shortest path between the terminal nodes with one intermediate node in each cluster selected as its gateway node. Determining this is not as easy as finding the shortest path between two points.

3.8 Heuristic Solution Approach For P1 With Euclidean Distance

MOTIVATION P1 with euclidean distance is isomorphic to the discrete interactive multifacility location problem. The discrete property is due to the finite possible sites for the hubs. If the restriction on hub sites is relaxed, the relaxed problem will be the same as the continuous interactive multifacility location problem (Francis & White [9], Chapter 5). An approach to solving P1 will be to first solve the continuous case, and then use a scheme to



a: A 3-partition example



b: Network representation of the 3-partition example

3.3 A three partition example and its network representation

determine the discrete locations from the solution to the continuous case. The relaxed problem is solved using the Hyperboloid Approximation Procedure (HAP) (Francis & White [9], pp 227-233). We next develop a heuristic solution procedure for P1 with euclidean distance, which uses the above described scheme.

Let the problem P1 with euclidean distance be denoted by P1E, and the corresponding relaxed problem by P1E-R. Again, the relaxation is obtained by removing the restriction on possible hub sites. Here we consider the nodes in each partition as the existing facilities and the hubs as the new facilities. The existing facilities in each partition interact only with their respective new facility. The new facilities interact between themselves. Let the number of nodes in each partition be $n_1, n_2, \dots, n_p$ respectively. Let V_i denote the location of node i belonging to partition k , and U_k denote the hub location for partition k . Let $C(U_k, V_i)$ represent the cost per group in the link between node i and its partition hub at U_k , and $C(U_k, U_l)$ represent the cost per group in the link between the hubs at U_k and U_l . W_{kl} is the number of groups between the partitions k and l , and Z_i is the number of groups in the link between node i and its partition hub.

Then, the relaxed problem P1R is given by,

P1R

minimize

$$f(U_1, \dots, U_p) = \sum_{\substack{k < l \\ k, l \in P}} W_{kl} C(U_k, U_l) + \sum_{k \in P} \sum_{i \in S_k} Z_i C(V_i, U_k) \quad (3.33)$$

Since cost $C(\cdot, \cdot)$ is based on the distance between the points, we can replace the cost measure with (distance * cost per group) measure. Then the relaxed

problem with euclidean distance P1E-R may be stated as follows.

P1E-R

Minimize

$$\begin{aligned} f(U_1, U_2, \dots, U_P) = & \sum_{\substack{k < l \\ k, l \in P}} W_{kl} c_{kl} [(X_k - X_l)^2 + (Y_k - Y_l)^2]^{1/2} + \\ & \sum_{k \in P} \sum_{i \in S_k} Z_i c_{ik} [(X_k - a_i)^2 + (Y_k - b_i)^2]^{1/2} \end{aligned} \quad (3.34)$$

where (X_k, Y_k) are the coordinates of hub location for partition k , and (a_i, b_i) are the coordinates of node i belonging to partition k . The optimal solution to problem P1E-R is obtained by taking the partial derivatives of the function f with respect to X_k and Y_k , setting the partial derivatives to zero, and solving for X_k and Y_k . But, function f is not continuously differentiable. If the hub for partition k coincides with the location of node i , the partial derivatives are undefined. By approximating f by $\tilde{f}$ as described below, we have a smooth function which is continuously differentiable.

$$\begin{aligned} \tilde{f}(X_1, X_2, \dots, X_P, \epsilon) = & \sum_{\substack{k < l \\ k, l \in P}} W_{kl} c_{kl} [(X_k - X_l)^2 + (Y_k - Y_l)^2 + \epsilon]^{1/2} + \\ & \sum_{k \in P} \sum_{i \in S_k} Z_i c_{ik} [(X_k - a_i)^2 + (Y_k - b_i)^2 + \epsilon]^{1/2} \end{aligned} \quad (3.34a)$$

where $\epsilon > 0$.

The following iterative expressions are used to obtain the coordinates (X_k, Y_k) from the partial derivatives of $\tilde{f}$ (Francis & White [9] pp 227-233).

$$X_k^{(h+1)} = \frac{\sum_{i \in S_k} \frac{Z_i c_{ik} a_i}{E_i} + \sum_{\substack{k, l \in P \\ l \neq k}} \frac{W_{kl} c_{kl} X_l^{(h)}}{D_{kl}}}{\sum_{i \in S_k} \frac{Z_i c_{ik}}{E_i} + \sum_{\substack{k, l \in P \\ l \neq k}} \frac{W_{kl} c_{kl}}{D_{kl}}} \quad (3.35)$$

$$Y_k^{(h+1)} = \frac{\sum_{i \in S_k} \frac{Z_i c_{ik} b_i}{E_i} + \sum_{\substack{k,l \in P \\ l \neq k}} \frac{W_{kl} c_{kl} Y_l^{(h)}}{D_{kl}}}{\sum_{i \in S_k} \frac{Z_i c_{ik}}{E_i} + \sum_{\substack{k,l \in P \\ l \neq k}} \frac{W_{kl} c_{kl}}{D_{kl}}} \quad (3.36)$$

$$\text{where } E_i = \left[(X_k^{(h)} - a_i)^2 + (Y_k^{(h)} - b_i)^2 + \epsilon \right]^{1/2},$$

$$D_{kl} = \left[(X_k^{(h)} - X_l)^2 + (Y_k^{(h)} - Y_l)^2 + \epsilon \right]^{1/2},$$

ϵ = a positive valued constant, and

(h) = denotes the iteration number.

This iterative procedure to determine the coordinate values X_k and Y_k is called the Hyperboloid Approximation Procedure (HAP).

Reporting experience with the HAP Charalambous[5], and Francis & White [9] indicate that the procedure has never failed to converge.

We now state in algorithmic form the heuristic procedure to solve P1 with euclidean distance.

- (HSP1) 1. Compute $X_k^{(1)}, Y_k^{(1)}$ the initial coordinates for the hub locations using the (3.35) and (3.36). Initially, no hub sites are fixed; hence for each partition the starting hub coordinates are determined as follows:

$$X_k^{(0)} = \frac{\sum_{i \in S_k} Z_i \cdot a_i}{\sum_{i \in S_k} Z_i} \quad Y_k^{(0)} = \frac{\sum_{i \in S_k} Z_i \cdot b_i}{\sum_{i \in S_k} Z_i}$$

These give the coordinates of the weighted median for each partition considering only the intra-partition flows.

2. For each partition whose hub is not fixed yet, find the node closest to its approximate hub location. Let j_k denote the index of the closest node for partition k .
3. Among all such j_k 's select the closest one and fix this node as the hub location for its partition.
4. If all the hub locations are fixed, then stop. Otherwise continue.
5. Using the revised coordinates for the hub location fixed in step 3, recompute the coordinates for the hubs not fixed yet, using (3.35) and (3.36). Return to step 2.

An example of the above algorithm is given in appendix A4. Since at each iteration one hub location is fixed, this procedure is finite. The reported experience on HAP suggests that each iteration will be convergent.

One possible disadvantage of the HAP will be that the function $f(X_1, X_2, \dots, X_p, \epsilon)$ may become more and more difficult to minimize as ϵ gets smaller and smaller. Charalambous [5] characterizes that when none of the new facilities occupy the same location, but at least one of them occupy an existing facility location, choosing ϵ very small will not add difficulty. The function is nondifferentiable at the location of the existing facility but is differentiable in its neighborhood.

3.9 Heuristic Solution Approach for P1 With Given Distance Measure

MOTIVATION The hub location for a partition is determined by the location of nodes in the partition and the location of other hubs. Hence, given the location of other hubs, finding a partition's hub location is similar to finding the absolute

median (Hakimi [19]). The heuristic described below is based on this concept.

PROCEDURE Initially, disregard inter-partition interactions and determine the hub
SUMMARY locations solely on the basis of intra-partition interactions. Then iteratively
improve each hub location considering the location of the other hubs.
Change in total cost is used as criterion for improvement.

We now state in algorithmic form the heuristic procedure to solve P1 with given distance measure.

HSP2 1: Compute the number of groups in the intra-partition and the inter-partition links.
2: For each partition ignore inter-partition interactions, and determine the hub location which will minimize the group costs considering only the intra-partition interactions. This is the same as finding the absolute median for each partition.

$$\text{Initial hub site for partition } k: h_k^{(0)} = \min_{j \in S_k} \left(\sum_{i \in S_k} Z_i C_{ij} \right).$$

3: For each partition redetermine the location for its hub, which will minimize the group costs, taking into account the location of the other hubs and the intra-partition interactions.

Revised hub site for partition k:

$$h_k^{(m+1)} = \min_{j \in S_k} \left(\sum_{i \in S_k} Z_i C_{ij} + \sum_{l \in P} W_{kl} C_{jh_l^{(m)}} \right),$$

where m is the iteration number.

4: Compute the total group costs for the new hub locational configuration. If new total cost is less than the old total cost, replace the old configuration with the new one and return to step 3. If the new total cost equals the old one, check if the two locational configurations are the same. If different, replace the old one with the new configuration and return to step 3. Otherwise stop.

An example illustrating this algorithm is given in appendix A5.

3.10 Variable Elimination

In the event the LP relaxation of P1 (LP1) does not yield an integer solution, we may use the LP bound and the reduced costs of the variables to eliminate a candidate location from consideration. If a candidate location is considered the corresponding x_i variable will have a value of 1. Let $\bar{c}_i$ be the reduced cost corresponding to location i , and UB be an upper bound on ILP1 obtained by using either HSP1 or HSP2. Then if $\bar{c}_i + v(LP1) > UB$ then no solution with $x_i=1$ can be optimal. The tighter the value of UB more candidate locations may be eliminated.

3.11 Bounding Procedures

LOWER BOUND To examine the efficiency of the heuristic solution procedure HSP1 and HSP2 good lower bounds on the solution are necessary. The solution to the LP relaxation will be a lower bound. Another lower bound procedure is as follows. For each pair of partitions, say k and l , compute the minimum of

the sum of intra- and inter-partition group costs. Let, C_{kl}^* denote the minimum of the sum of inter- and intra-partition group costs for the partitions k and l . Then,

$$C_{kl}^* = \min_{i \in S_k, j \in S_l} \left(\frac{1}{(p-1)} \sum_{h \in S_k} Z_h C_{hi} + \frac{1}{(p-1)} \sum_{h \in S_l} Z_h C_{hj} + W_{kl} C_{ij} \right). \quad (3.37)$$

The lower bound (LB2) on the solution to P1 will be then equal to

$$\sum_{\substack{k < l \\ k, l \in P}} C_{kl}^*. \quad \text{We now show that LB2 is a lower bound to problem P1.}$$

Let i^* and j^* be the optimal hub locations in partitions k and l , respectively.

Then the optimal solution to P1 may be expressed as follows.

$$\sum_{(k < l) \in P} A_{kl} = \sum_{(k < l) \in P} \left\{ \frac{1}{(p-1)} \sum_{j \in S_k} Z_j C_{i^*j} + \frac{1}{(p-1)} \sum_{i \in S_l} Z_i C_{j^*i} + W_{kl} C_{i^*j^*} \right\} \quad (3.38)$$

Comparing the corresponding terms of (3.37) and (3.38), for all k and l , it is evident that A_{kl} must be at least as large as C_{kl}^* . Hence, LB2 is a lower bound on solution to P1. $\square$

In computing LB2 we take two partitions at a time. This procedure could be extended to consider a combination of three or more partitions at a time to determine a better lower bound. However, the computational difficulty of such a procedure increases as more partitions are considered at a time.

3.11 Computational Results

EXPERIMENTAL To study the performance of the solution procedures for P1 sample problems

DESIGN were generated using a random process. The flow between nodes and the coordinates of nodes were generated from a uniform distribution. For the LP method and the heuristic HSP2 the distance measure applied was euclidean, since this will facilitate comparing performance of these two with HSP1. Cost per arc in the access links is assumed the same; similarly in the trunk links. Due to economies of scale the cost per channel in the trunk group is likely to be smaller than the cost per channel in the access group. In the experimental design, we consider the ratio of cost per channel in the trunk link to cost per channel in the access link. Given the cost per channel the cost per arc will be (cost per channel * arc capacity).

DESIGN The number of nodes, the number of partitions, and the ratio of per channel
VARIABLES cost were selected as the design parameters. These were selected in order to study the effect of the number of partitions on total network cost and solution difficulty, and the effect of the cost ratio on hub locations.

Three node sizes - 20, 50, and 100, were used. For each node size five different node distributions were generated. For problems with 20 nodes the number of partitions created were 2, 4, or 8; for problems with 50 nodes the number of partitions created were 2, 4, or 10; and for problems with 100 nodes the number of partitions created were 5, 10 or 20. For each partition type four different partition arrangements were created. Each problem generated was tested using six values of the cost ratio - 0.5, 0.6, 0.7, 0.8, 0.9, and 1.0. By this design 120 problems were generated for each partition type,

360 problems for each node size, and 1080 problems overall. A graphic illustration of this design scheme is given below.

20 (5 problems)	50 (5 problems)	100 (5 problems)
2 4 8	2 4 10	5 10 20
(4 partitions per type)		
.5 .6 .7 .8 .9 1.		
(6 cost ratios)		

The three solution procedures - LP, HSP1, and HSP2, the lagrangian method, and the bounding procedure - LB2, were tested with the sample problems. For each method, the solution and the execution time (CPU time in seconds) were recorded. The execution time was the straight run time without input/output. All except the LP method were coded in Turbo Pascal (version 3.1) and implemented on an IBM PC-XT compatible with a clock speed of 8Mhz and a 8087 coprocessor. The LP method was implemented using LINDO (version 4) running on a VAX-8600, rated at 4.4 mips.

The data collected is summarized in tables 3.1-3.3. The information presented in each table is:

- a. percentage of problems for which optimal integer solution were obtained,
- b. average run time in seconds,
- c. maximum run time,
- d. minimum run time,
- e. average % deviation of solution from optimal solution,
- f. average % deviation of solution from lower bound,
- g. average % deviation of optimal solution from lower bound,
- h. maximum deviation of solution from optimal solution,

i. minimum deviation of solution from optimal solution.

The LP method yielded integer solutions for all the sample problems tested. This supports our observation that the LP solution is likely to be integer except when there is symmetry in node distribution and flow with respect to the partitions. The run time and the number of iterations to the optimal solution were observed to increase with the number of partitions, for a given node size. The lagrangian approach was applied for 20 and 50 node problems only. Memory was insufficient to test 100 node problems. The upper bound used for the lagrangian method was arbitrary. The first node read for each partition was taken to be the hub location in calculating the upper bound. A better upper bound may reduce the average run time for this method. Heuristic HSP2 may be used for this purpose. The run time was observed to increase with the number of nodes, and the number of partitions for a given number of nodes.

The experiment results indicate that the percentage of problems for which the optimal integer solution was obtained using the heuristic methods declines for HSP1 with the number of partitions, and increases for HSP2 with the number of partitions. While the execution time for HSP1 increases with the number of partitions for a given node size, for HSP2 it remained more or less at the same level. Execution times for all solution methods increased with the number of nodes. For both heuristic methods the average percentage deviation of solution from the optimal solution declined with an increase in the number of partitions. It appears that for euclidean distance problems HSP1 tends to perform better by yielding final solutions very close to the

Table 3.1: Summary of results for 20 node problems

20 Nodes	Partitions		
	2	4	8
# of Problems	120	120	120
LP Solution:			
% Integer solutions	100.00%	100.00%	100.00%
Average run time (secs)	0.04	0.19	0.94
Average # of steps	15	50	98
HSP1 Solution:			
% Problems with opt. solu.	64.17%	61.67%	50.83%
Average run time (secs)	0.88	1.56	4.17
Max. run time	2.03	3.84	7.69
Min. run time	0.60	1.05	2.81
Avg % deviation from optimal	0.10%	0.16%	0.07%
Avg % deviation from LB2	2.02%	2.63%	1.97%
Avg % gap bet. opt. & LB2	1.86%	2.40%	1.83%
Max. devi. from optimal solu.	2.18%	2.74%	0.80%
Min. devi. from optimal solu.	0.00%	0.00%	0.00%
HSP2 Solution:			
% Problems with opt. solu.	92.50%	99.17%	100.00%
Average run time (secs)	0.56	0.49	0.60
Max. run time	0.82	0.72	1.10
Min. run time	0.49	0.38	0.49
Avg % deviation from optimal	0.11%	0.01%	0.00%
Avg % deviation from LB2	0.11%	4.05%	5.33%
Avg % gap bet. opt. & LB2	0.00%	3.82%	5.03%
Max. devi. from optimal solu.	2.13%	0.67%	0.00%
Min. devi. from optimal solu.	0.00%	0.00%	0.00%
Lagrangian:			
% Problems with opt. solutions	100.00%	100.00%	100.00%
Average run time (secs.)	0.67	1.35	2.67

% deviation from optimal solution = $(\text{HSP} - \text{OPT.})/\text{OPT.}$
 % deviation from LB = $(\text{HSP} - \text{LB})/\text{LB}$
 % gap between optimal solution & LB = $(\text{OPT.} - \text{LB})/\text{OPT.}$

Table 3.2: Summary of results for 50 node problems

50 Nodes	Partitions		
	2	4	10
# of Problems	120	120	120
LP Solution:			
% Integer solutions	100.00%	100.00%	100.00%
Average run time (secs)	0.40	2.31	16.87
Average # of steps	14	42	119
HSP1 Solution:			
% Problems with opt. solution	73.33%	60.00%	53.33%
Average run time (secs)	3.61	6.47	21.73
Max. run time	7.91	15.54	64.81
Min. run time	1.36	2.69	7.03
Avg % deviation from optimal	0.21%	0.11%	0.05%
Avg % deviation from LB2	1.71%	1.56%	1.72%
Avg % gap bet. opt. & LB2	1.46%	1.42%	1.63%
Max. deviation from opt.	4.81%	2.36%	0.60%
Min. deviation from opt.	0.00%	0.00%	0.00%
HSP2 Solution:			
% Problems with opt. solution	88.33%	90.83%	93.33%
Average run time (secs)	3.17	2.45	2.24
Max. run time	4.89	3.79	3.19
Min. run time	2.85	1.97	1.86
Avg % deviation from optimal	0.09%	0.43%	0.16%
Avg % deviation from LB2	0.09%	4.75%	6.12%
Avg % gap bet. opt. & LB2	0.00%	4.10%	5.60%
Max. deviation from opt.	1.50%	9.39%	3.67%
Min. deviation from opt.	0.00%	0.00%	0.00%
Lagrangian:			
% Problems with opt. solution	100.00%	100.00%	100.00%
Average run time (secs.)	4.23	6.43	14.85

% deviation from optimal solution = (HSP - OPT.)/OPT.

% deviation from LB = (HSP - LB)/LB

% gap between optimal solution & LB = (OPT. - LB)/OPT.

Table 3.3: Summary of results for 100 node problems

100 Nodes	Partitions		
	5	10	20
# of Problems	120	120	120
LP Solution:			
% Integer solutions	100.00%	100.00%	100.00%
Average run time (secs)	0.40	2.31	16.87
Average # of steps	14	42	119
HSP1 Solution:			
% Problems with opt. solution	73.33%	60.00%	53.33%
Average run time (secs)	3.61	6.47	21.73
Max. run time	7.91	15.54	64.81
Min. run time	1.36	2.69	7.03
Avg % deviation from optimal	0.21%	0.11%	0.05%
Avg % deviation from LB2	1.71%	1.56%	1.72%
Avg % gap bet. opt. & LB2	1.46%	1.42%	1.63%
Max. deviation from opt.	4.81%	2.36%	0.60%
Min. deviation from opt.	0.00%	0.00%	0.00%
HSP2 Solution:			
% Problems with opt. solution	88.33%	90.83%	93.33%
Average run time (secs)	3.17	2.45	2.24
Max. run time	4.89	3.79	3.19
Min. run time	2.85	1.97	1.86
Avg % deviation from optimal	0.09%	0.43%	0.16%
Avg % deviation from LB2	0.09%	4.75%	6.12%
Avg % gap bet. opt. & LB2	0.00%	4.10%	5.60%
Max. deviation from opt.	1.50%	9.39%	3.67%
Min. deviation from opt.	0.00%	0.00%	0.00%

% deviation from optimal solution = $(\text{HSP} - \text{OPT.})/\text{OPT.}$
 % deviation from LB = $(\text{HSP} - \text{LB})/\text{LB}$
 % gap between optimal solution & LB = $(\text{OPT.} - \text{LB})/\text{OPT.}$

optimal than HSP2. Based on the performance of HSP2 for euclidean distance problems we could expect this method to yield good results for problems with other distance measures. More than the number of nodes, the number of partitions seem to have a profound effect on the performance of the solution methods. Comparing the execution times, HSP2 produced solutions much faster than HSP1 or the LP method. Since commercial codes available for LP have a limitation on the number of variables and the number of equations, problems involving large number of nodes and partitions cannot be solved using commercial LP codes, for example LINDO or MPSX. (See appendix A6.) Heuristic HSP2 offers an attractive alternative for such situations. Since the number of variables generated by HSP2 is smaller it will be possible to solve large problems using this method.

Also, we observe that as the per channel cost ratio increases the hub locations are drawn closer to each other. For a given number of nodes an increase in the number of partitions sometimes decreases the total network costs for a given cost ratio.

CHAPTER 4

PROBLEM P2:

MATHEMATICAL FORMULATION & ALGORITHMS

4.1 Problem Statement

PROBLEM STATEMENT Given a collection of n nodes in a plane. The matrix of flow between the nodes expressed in number of channels, is given and is symmetric. Flow between nodes is undirected. There is no flow from a node to itself. Every node is considered as an origin/destination of flows. The matrix of direct distances between the nodes is given, and is symmetric and metric. The coordinates of the nodes are known. A specified number - p : $p < n$, of hubs are to be located at a subset of the nodes. Every node will be assigned to exactly one hub and thus the n nodes will be partitioned into p subsets. The routing strategy is that all pair wise node flows will be routed via their assigned hub(s). In other words, connection between two nodes are established only via their assigned hubs. Suppose nodes i and j are assigned to a hub at u . Then the flow between i and j will be routed via the path $i-u-j$. If nodes i and j are assigned to separate hubs u and v respectively, then the flow will be routed via the path $i-u-v-j$. (See figure 3.1 for an illustration of the routing strategies.) A network link is defined as a direct connection between two nodes. By the routing strategy, there would be links only between the nodes and their assigned hubs (access links), and between the hubs (trunk links). Flow along any link is bundled into groups of fixed capacity. The capacity of a group is expressed in number of channels. In network terminology, each group would be a fixed capacity arc. There may be one or more group in any link. Capacity of groups within a link will be

assumed to be same while group capacity may be different for each link. Since the flow from a nonhub node is along the access link, given the flow and the group capacity we can determine the number of groups in the access links. However the number of groups in the trunk links will depend on the node partition. For simplicity and without loss of generality we shall assume that all access links have the same group capacity, and similarly the trunk links. Since all nodes are aggregate nodes we do not know how many arcs extend from the smaller nodes to an aggregate node. All aggregate nodes contain frames and the capacity of a frame is assumed the same at all the nodes. We estimate the number of ports required at a node as follows:

$$\text{number of arcs connecting at the node} + \left\lceil \frac{\text{total flow at the node}}{\text{minimum arc capacity}} \right\rceil * \alpha,$$

where α is a user specified parameter ($\alpha \geq 0$), and $\lceil a \rceil$ denotes the smallest integer greater than or equal to a .

- NETWORK COSTS** Since the number of hubs is given, hub cost is constant for the problem. Due to the given routing strategy, there will be exactly $\left[(n-p) + \frac{p(p-1)}{2} \right]$ links in the network, and hence the connection costs are also constant. The variable costs for the problem will be (1) arc costs, (2) port utilization costs, and (3) frame cost.
- OBJECTIVE** The objective is to simultaneously determine the hub locations, node partitions, network links, and the link capacities such that the total relevant network costs is a minimum.

Next, we present an integer nonlinear programming formulation of the problem.

4.2 Mathematical Formulation

MATHEMATICAL Integer Nonlinear Programming Formulation of P2

FORMULATION

Notation:

G^A - Group capacity in access links

G^T - Group capacity in trunk links

Q^F - Capacity of frame

p - Number of hubs

N - Node set; $|N| = n$

b_{ij} - Flow between nodes i and j

Z_i - Number of groups from node i to its assigned hub at j

$$= \left\lceil \frac{\sum_{h \in N} b_{ih}}{G^A} \right\rceil, \text{ where } \lceil a \rceil \text{ is the smallest integer greater than or equal to } a$$

W_{kl} - Number of groups between hubs at locations k and l

$X_{ii} = 1$ if node i is a hub site

$= 0$ otherwise

$X_{ik} = 1$ if node i is assigned to hub at location k

$= 0$ otherwise

c^1 - Fixed cost per group

c^{2A} - Cost per unit distance per group in an access link

c^{2T} - Cost per unit distance per group in a trunk link

c^3 - Port utilization cost

d_{ij} - Distance between nodes i and j

C_{ij} - Total cost per group in link i-j
 $= (c^1 + c^{2A} * d_{ij} + 2c^3)$, if node i is assigned to hub at j
 $= (c^1 + c^{2T} * d_{ij} + 2c^3)$, if node i and j are hubs

F_i - Number of frames at node i

C^F - Cost per frame

β_i - Estimate of extra ports required at node i

$$= \left\lceil \frac{\sum_{j \in N} b_{ij}}{\min(G^A, G^T)} \right\rceil * \alpha$$

α - User specified parameter ($\alpha \geq 0$)

(IP2)

Minimize

$$\sum_{i \in N} \sum_{\substack{j \in N \\ j \neq i}} Z_i C_{ij} X_{ij} + \sum_{\substack{k, l \in N \\ k < l}} W_{kl} C_{kl} + \sum_{i \in N} C^F F_i \quad (4.1)$$

subject to

$$\sum_{i \in N} X_{ii} = p \quad (4.2)$$

$$\sum_{j \in N} X_{ij} = 1 \quad \forall i \in N \quad (4.3)$$

$$\sum_{i \in N} X_{ij} \leq (n-p+1) X_{jj} \quad \forall j \in N \quad (4.4)$$

$$\sum_{i \in N} \sum_{\substack{j \in N \\ j \neq i}} b_{ij} X_{ik} X_{jl} \leq G^T \cdot W_{kl} \quad \forall k, l \in N \ni k < l \quad (4.5)$$

$$\sum_{\substack{i \in N \\ i \neq j}} Z_i X_{ij} + Z_j (1 - X_{jj}) + \sum_{\substack{k \in N \\ k < j}} W_{kj} + \sum_{\substack{k \in N \\ j < k}} W_{jk} + \beta_j \leq F_j Q^F \quad \forall j \in N \quad (4.6)$$

$$X_{ij} \in \{0,1\} \quad (4.7)$$

$$W_{ij}, F_i \geq 0 \text{ integer} \quad (4.8)$$

In the objective function (4.1) the first term gives the total group cost in the access links, the second term gives the total group costs in the trunk links, and the third term defines the total frame cost. Constraint (4.2) defines that a set of p nodes will be selected as sites for the hubs. The constraint set (4.3) defines that each node will be assigned to exactly one hub. The constraint (4.4) gives that nodes can be assigned to a location j only if it is a hub site, and at most $(n-p+1)$ nodes can be assigned to it. This includes the assignment of j to itself. This is a relaxed form of the constraint $X_{ij} \leq X_{jj} \forall i \in N$. The constraint set (4.5) indicates that the capacity in the trunk links must at least equal the flow via them. The flow between nodes i and j will use the link $k-l$ only if node i is assigned to hub at k and j to hub at l , or vice-versa. Constraint (4.6) gives that the number of ports available at a node must at least equal the number of groups connecting to it. Constraint sets (4.7) and (4.8) specify the integrality requirements.

This formulation is similar to that for the uncapacitated discrete Location-Allocation problem (see Chen [6], Cooper [7], and Francis & White [9]). In the location-allocation problem there is interaction only between each new facility and the destinations assigned to it. There is no interaction between the new facilities. The Z_i 's described above may be considered as flow from a facility to a destination assigned to it. The interaction between hubs is similar to the situation encountered in the Quadratic Assignment Problem (QAP).

Since this formulation is complex it is not applicable directly for algorithm development.

Next we discuss the similarity of P2 with other difficult problems, and propose solution and bounding strategies.

4.3 Complexity of P2

We establish the complexity of P2 by transforming an instance of the minsum multicenter (p-median) problem to P2.¹

Consider an instance of P2 with the following restrictions:

- a. large frame capacity so that only one frame is required at each node;
- b. large arc capacity in each potential link so that only one arc is necessary in each link;
- c. flow between each pair of nodes is positive;
- d. cost per arc per unit distance in access links = 1; and
- e. cost per arc per unit distance in trunk links = 0.

This instance of P2 can be obtained by transformation of an instance of the min-sum multicenter problem in which the flow from each node is 1 and the cost per unit of flow is proportional to the distance between the nodes. Since the cost per arc in the trunk links is zero, in this instance of P2 as in the min-sum multicenter problem all nodes will be assigned to their closest hub. Solution to this instance of P2 will also yield the solution to the instance of the minsum multicenter problem. Hence, the solution to P2 will be at least as difficult as the solution to the min-sum multicenter problem. The instance of the min-sum multicenter problem in which the flow from each

¹A description of the min-sum multicenter problem is given in appendix B1.

node is 1 is NP-Complete (Garey & Johnson [10]).

4.4 Solution Strategies

TOTAL A direct approach to finding the optimal solution to P2 is to totally
ENUMERATION enumerate the network costs for all the possible hub locational configurations and corresponding hub-node assignments. For a n-node p-partition case, the number of these possible combinations equals the Stirling number of the second kind (Roberts [29])

$$\frac{1}{p!} \sum_{i=0}^p (-1)^i \binom{p}{i} (p-i)^n.$$

Here we make an implicit assumption that no partition is empty. For a given number of partitions the number of possible combinations is of $O(p^n)$, exponential in the number of nodes. As one can see this approach is computationally infeasible except for very small problems.

Since P2 is complex heuristic solution approaches will be practical for determining good feasible solutions.

HEURISTIC The two primary decisions in P2 are:

- APPROACHES**
- a. location of hubs, and
 - b. assignment of nodes to hubs.

These two decisions are interdependent. The location of a hub will depend upon the nodes assigned to it and the locations of other hubs. Assignment of a node to a hub will depend on the location of the hub and the other nodes already assigned to it. One recognizes that this is a circular decision. An approach to developing heuristics will be to eliminate this interdependency.

This can be accomplished in one of the two following ways.

- (a) Select the hub locations by some procedure and assign each node to a hub. For example a node may be assigned to its closest hub or second closest hub etc. We will assign each node to its closest hub.
- (b) Partition the nodes by some procedure and determine the best hub locations based on the selected partitions.

Next we describe two heuristics using these approaches.

4.5 Heuristic Solution Approaches

4.5.1 Total Enumeration of Hub Locational Configurations

One approach to selecting the hub locations is to consider all the possible hub locational configurations. For example if p hubs are to be located among n nodes the number of possible hub configurations will be $\frac{n!}{p! (n-p)!}$. By approach (a) since all nodes will be assigned to their closest hub, the number of solutions to be compared equals the number of possible hub configurations. Since this is of order $O(n^p)$ except for small problems this approach will not be feasible. We will not pursue this further in this study.

4.5.2 Pair Wise Interchange Heuristic - HSP3

MOTIVATION In approach (a) by assigning nodes to their closest hub we make the node assignment a byproduct of the hub locational configuration. This is similar to the approach used for solving the P-median problem. Using a pair wise interchange procedure we can generate the hub locational configurations and

the corresponding node assignments.

The pair wise interchange procedure may be described as follows. A non-hub node is considered as a replacement for a current hub site. Based on the revised hub locational configuration the nodes are reassigned and the corresponding network cost is computed. If the replacement results in cost improvement then the revised configuration replaces the current one. This procedure is repeated until no further improvement is possible.

When a non-hub location is considered as a replacement for an existing site, the resultant effect has a number of possibilities. Suppose, node b is considered as a replacement for hub site v .

(1) Say node i is currently assigned to the hub at k .

- a. If b is closest to i , then i is reassigned to b ;
- b. if both k and b are closest, then reassign i to the hub set with which it has larger total flow;
- c. otherwise no change in assignment.

(2) Say node i is currently assigned to the hub at v .

- a. If b is closest to i , then reassign i to b ;
- b. if another hub site u is closest to i , reassign i to u ;
- c. If b and another hub at u are closest, then reassign i to the set with which it has larger total flow.

HEURISTIC Now we state the heuristic procedure in algorithmic form.

PROCEDURE 1. Select an initial set of hub sites (H^0) from among the n nodes. Assign
(HSP 3) each node to its nearest hub. Break ties for assignment by assigning node

to the hub set with which it has largest total flow. Compute the number of groups between the hubs, and the total network costs. Let the total network costs be denoted by C .

2. With respect to H^0 , define a list M of non-hub nodes: $M = \{N \setminus H^0\}$.
3. If list M is empty go to step 6. Otherwise select a non-hub node b from the list for pair wise interchange and delete it from the list.
4. Consider b to replace each hub site v in the current configuration; we would have p new configurations by this systematic replacement. For each new configuration complete the node reassignment and compute the total network costs. Let $C_1, C_2, \dots, C_p$ denote the corresponding network costs.
5. If $\min(C_1, C_2, \dots, C_p) > \text{current network cost } C$, then replace the current configuration with that corresponding to the minimum cost and let $C = \min(C_1, C_2, \dots, C_p)$. Return to step 3.
6. Relabel the current configuration as H^0 and redefine list $M = \{N \setminus H^0\}$. If the last $(n-p)$ pair wise interchanges considered failed to yield any improvement in the total network costs, terminate the procedure. Otherwise, return to step 3.

An example illustrating the application of this heuristic is given in appendix B2.

It must be noted that the nearest hub assignment rule may not necessarily produce optimal node-hub assignments. A counter example is provided to illustrate this situation. Let nodes (1,2,3) and (4,5,6) be located at the vertices of two equilateral triangles as shown below.



Suppose, node 1 has interactions only with nodes 4,5, and 6 amounting to exactly 2 groups. If nodes 2 and 4 are selected as hub sites, then, by nearest hub rule nodes 1 and 3 will be assigned to node 2, and nodes 5 and 6 will be assigned to 4. Since the distance between the nodes is metric, it is evident that by assigning node 1 to hub at node 4 network costs could be reduced.

4.5.2.1 Initial Hub Sites Selection

The choice of initial hub sites must be based on the distance between nodes or the flow between nodes or both. Since the node-hub assignments are not known a priori using the flows may be difficult. We modify a procedure suggested by Dysart and Georganas [8] based on distance between nodes, to choose the initial hub sites.

Let $|N|=n$ be the total number of node locations, and k be a design parameter that specifies the number of nearest nodes to be considered for each node location. Let Z_i be the number of groups from a node i to its assigned hub. The steps of the initial hub site selection heuristic are as follows.

1. List each of the nodes in the network and its k nearest neighbors.
2. For each node determine its frequency of occurrence in the node-neighbor list, obtained in step 1.

3. Each node will have one of the frequencies f , $f=1,\dots,F$, where F is the maximum frequency of occurrence. Partition all the nodes into lists $S(f)$, $f=1,\dots,F$, where the list $S(f)$ includes the nodes with frequency f .

4. Obtain the weighted mean plus 1, KM , as

$$KM = \left(\sum_{f=1}^F \frac{f x(f)}{n} \right) + 1$$

where $x(f)$ is the number of nodes in list $S(f)$. Note that

$$\sum_{f=1}^F x(f) = n.$$

5. Arrange the lists $S(f)$ in descending order based on the Z values associated with each node. The initial hub sites list is obtained by the node locations in the list $S(F)$. Further additions are provided by including the nodes from the lists $S(f)$, $KM < f < F$.

This procedure selects those nodes for hub sites that are close to other nodes and also have large total interaction with the other nodes. The remaining nodes are assigned to their nearest hub site. The ordering of nodes obtained by this procedure could be maintained when constructing the nonhub node list M (see page 4-10). This could improve the efficiency of the pair wise interchange procedure. An example illustrating the application of this procedure for 10 nodes with parameter k set to 3 is included in appendix B3.

4.5.3 Solution to P2 Using Heuristic HSP2

MOTIVATION To partition the nodes one may use any of the many available clustering techniques. Almost all of the cluster approaches are based on the distance between the nodes. The reader is referred to Anderberg [2] for a review of

cluster analysis techniques. An initial partition of the nodes may be generated using the procedure detailed in section 4.5.2.1. Once a partition is obtained, the hub locations may be determined using heuristic HSP2 described in Chapter 3. After the hub locations are found we can revise the partitions by reassigning each node and then reapplying HSP2 to the revised partitions. This process is repeated until no further improvement in cost is possible. We will revise the partitions by assigning each node to its closest hub. One may revise the partitions by reassigning each node to its second closest hub or third closest hub etc. However, we will not investigate these choices in our work.

HEURISTIC Now we state the procedure in algorithmic form.

PROCEDURE 1. Use procedure described in 4.5.2.1 to select an initial set of hubs and
(HSP4) assign each node to its closest hub. Evaluate the total network cost based
 on the initial hub sites. Thus, we have obtained an initial partition of
 nodes and an upper bound on the total network cost.

 2. Using HSP2 determine the hub locations for the partitions and evaluate
 the associated network cost.

 3. If the new total cost is less than the current total cost redefine new
 partitions by assigning each node to its closest hub, and return to step 2.
 If the new and the current costs are the same, then compare the partitions.
 If the partitions are identical then stop. Otherwise go to step 2. Also, if
 the new total cost is more than the current total cost stop. We have
 reached a local optimal solution.

4.6 Lower Bound Procedures

To examine the efficiency of heuristics we need to know either the optimal solution or good lower bounds on the optimal solution to the problem. For small problems it is possible to obtain the optimal solution by examining all possible node partitions and the corresponding optimal hub locations. For larger problems this is not feasible. We suggest the following lower bound procedure for large problems.

MOTIVATION From (4.1) we can see that the items making up the total cost of the network are:

- a. total access arc cost,
- b. total trunk arc cost, and
- c. total frame cost.

We can compute a lower bound on the total cost by determining a lower bound on each of these items. Let us consider an n -node p -hub problem. To get a lower bound on the access arc cost we could take the scalar product of the $(n - p)$ smallest Z_i values and the $(n - p)$ smallest cost per arc among the candidate access links. Since the arc cost is a product of the cost per arc and the number of arcs in the links, we can get a better lower bound on the access arc cost by first arranging one of the arrays in the ascending order and the other in the descending order, and then taking a product of the two. If all flows are positive then there will be at least one arc in each trunk link. We can get a lower bound on the trunk arc cost by taking the sum of the product of the $\frac{p(p-1)}{2}$ smallest distances, beyond the $(n - p)$ values considered before, and the cost per arc per unit distance in the trunk links. The number of frames at a node is determined by the number arcs connecting to it. For a

nonhub node the number of arcs connecting to it equals $(Z_i + \beta_i)$. We can get a lower bound on the frame cost at the nonhub nodes by taking the $(n-p)$ smallest $(Z_i + \beta_i)$ values. For a hub node the number of arcs connecting to it equals (the number of arcs in the access links to it + the number of arcs in the trunk links to it + the arcs extending to the nodes clustered in the aggregate node). A lower bound for the first component will be the sum of the $(n-p)$ smallest Z_i values. If we assume that there will be at least one arc in each trunk link, a lower bound on the second component will be $p(p-1)$. A lower bound on the third component will be the sum of the p smallest β_i values.

LOWER BOUND Let

- (LB3) LD - vector of the $(n-p)$ smallest distance between the nodes, arranged in an ascending order;
- LZ - vector of the $(n-p)$ smallest Z_i values, arranged in a descending order;
- LP - vector of the $\frac{p(p-1)}{2}$ smallest distance between the nodes beyond the values in LD, arranged in an ascending order;
- LM - vector of the $(n-p)$ smallest $(Z_i + \beta_i)$ values;
- LB - vector of the p smallest β_i values.

A lower bound on the solution is then equal to:

$$\sum_{i=1}^{(n-p)} \left(c^1 + c^{2A} \cdot ld_i + 2c^3 \right) lz_i + \sum_{i=1}^{\frac{p(p-1)}{2}} \left(c^1 + c^{2T} \cdot lp_i + 2c^3 \right) +$$

$$\left(\sum_{i=1}^{n-p} \left[lm_i \cdot \frac{1}{Q^F} \right] \right) \cdot C^F + \left[\left(\sum_{i=1}^{n-p} lz_i + p(p-1) + \sum_{i=1}^p lb_i \right) \cdot \frac{1}{Q^F} \right] \cdot C^F \quad (4.9)$$

A lower bound on the total arc cost in the access and the trunk links is given by the sum of the first and the second terms of (4.9). A lower bound on the number of frames required at the nonhub nodes is given by the third term of (4.9). The fourth term of (4.9) gives the lower bound on the number of frames required at the hub nodes. Hence, (4.9) gives a lower bound on the solution to P2. $\square$

An example of this procedure is given in appendix B4.

4.7 Computational Experience

EXPERIMENTAL To study the performance of the solution and lower bound procedures for P2

DESIGN sample problems were generated using a random process. The flow between nodes and the coordinates of nodes were generated from a uniform distribution. For the heuristic HSP3 and HSP4, and the lower bound LB3 the distance measure applied was euclidean. Cost per arc in the access links is assumed the same; and similarly in the trunk links. Due to economies of scale the cost per channel in the trunk link is likely to be smaller than the cost per channel in the access link. In the experimental design, we consider the ratio of cost per channel in the trunk link to cost per channel in the access link. Given the cost per channel the cost per arc will be (cost per channel * arc capacity). The frame capacity is assumed the same at all nodes.

DESIGN The number of nodes, the number of hubs(partitions), the ratio of per
 VARIABLES channel cost, and cost per frame were selected as the design parameters.
 These were selected in order to study the effect of the number of partitions
 on total network cost and solution difficulty, and the effect of the cost ratio
 on hub locations.

Two node sizes - 15, and 25 were used. For each node size four different node
 distributions were generated. For both problem sizes the number of
 partitions created were 2, 3, or 5. Each problem generated was tested using
 three values of the cost ratio - 0.6, 0.8, and 1.0. By this design 24 problems
 were generated for each partition type, 36 problems for each node size, and
 72 problems overall. A graphic illustration of this design scheme is given
 below.

15 (4 problems)	25 (4 problems)
2 3 5 (3 partitions)	2 3 5
.6 .8 1.0 (3 cost ratios)	

The two solution procedures - HSP3, and HSP4, and the bounding procedure
 - LB3, were tested with the sample problems. For each method, the solution
 and the execution time (CPU time in seconds) were recorded. The execution
 time was the straight run time without input/output. All algorithms were
 coded in Turbo Pascal (version 4) and implemented on an IBM PC-XT
 compatible with a clock speed of 8Mhz and a 8087 coprocessor.

The data collected is summarized in tables 4.1-4.2. The information presented in each table is:

- a. average run time in seconds,
- b. maximum run time,
- c. minimum run time,
- d. average % deviation of solution from lower bound,
- e. maximum deviation of solution from lower bound,
- f. minimum deviation of solution from lower bound.

It appears that with the increase in the number of nodes, and the number of partitions the performance of the heuristics deteriorate. Since HSP4 evaluates fewer hub configurations than HSP3 it executes faster with poorer performance. The lower bound we have appears to be rather weak. However generating a better lower bound may be difficult because we have no knowledge of the node-hub assignments. Specifying a list of potential hub sites, and a list of potential hubs to which a node may be assigned may improve the performance of these methods.

Table 4.1: Summary of results for 15 node problems

15 Nodes	Partitions		
	2	3	5
# of Problems	12	12	12
HSP3 Solution:			
Average run time (secs)	1.28	2.56	6.47
Max. run time	2.54	5.48	9.86
Min. run time	0.62	1.12	3.42
Avg % deviation from LB3	15.02%	23.53%	28.76%
Max. devi. from LB3	22.45%	28.01%	45.28%
Min. devi. from LB3	8.43%	17.67%	15.46%
HSP4 Solution:			
Average run time (secs)	0.76	1.32	2.89
Max. run time	1.50	2.76	4.43
Min. run time	0.45	0.78	0.49
Avg % deviation from LB3	22.45%	27.26%	38.73%
Max. devi. from LB3	26.22%	45.87%	48.23%
Min. devi. from LB3	15.33%	19.65%	23.12%

% deviation from LB

$$= (\text{HSP} - \text{LB})/\text{LB}$$

Table 4.2: Summary of results for 25 node problems

25 Nodes	Partitions		
	2	3	5
# of Problems	12	12	12
HSP3 Solution:			
Average run time (secs)	1.64	4.06	10.34
Max. run time	3.06	6.82	14.56
Min. run time	0.98	1.72	8.93
Avg % deviation from LB3	18.00%	29.00%	39.00%
Max. devi. from LB3	23.00%	41.00%	45.00%
Min. devi. from LB3	12.00%	22.00%	23.00%
HSP4 Solution:			
Average run time (secs)	0.85	1.63	3.12
Max. run time	1.53	2.98	4.51
Min. run time	0.59	1.03	1.61
Avg % deviation from LB3	24.00%	34.00%	43.00%
Max. devi. from LB3	35.00%	51.00%	49.00%
Min. devi. from LB3	18.00%	29.00%	34.00%

% deviation from LB

$$= (\text{HSP} - \text{LB})/\text{LB}$$

CHAPTER 5

PROBLEM 3:

MATHEMATICAL FORMULATION & ALGORITHMS

5.1 Problem Statement

PROBLEM Given a collection of n nodes in a plane, partitioned into a specified number
STATEMENT (say p) of subsets. The coordinates of the nodes are known. The matrix of
flows between the nodes, expressed in number of channels, is given and is
symmetric. All flows are deterministic, static, and undirected. There is no
flow from a node to itself. The matrix of direct distances between the nodes
is given, and is symmetric and metric. Direct distance between two nodes
may not be euclidean. Exactly one hub is located in each partition at a
node, and its location is known. The routing strategy is that a flow may be
routed either direct or via the respective partition hub(s). For example, if
nodes i and j belong to the same partition and the partition hub is located at
 u , then the flow between i and j may be routed directly $i-j$; or via the hub $i-u-j$.
Suppose the nodes belong to separate partitions and u and v are the hub
locations, respectively, then the flow routing may be either $i-j$ or $i-u-v-j$.
Flow between two nodes will not be split for routing. For flows between a
node and its partition hub, and between two hubs the two routing strategies
are identical.

A network link is defined as a direct connection between two nodes. By the
routing strategy there may be a link between any pair of nodes of the graph.
Flow along a link is bundled into arcs of fixed capacity. Equivalently, one
can consider this as multiple arcs with the same fixed capacity. The arc

capacity is expressed in number of channels. Arc capacity may be different for each link. For purpose of simplicity and without loss of generality, we categorize the links into three types and assign a different arc capacity for each link type.

Direct link - link between two non-hub nodes or link between a node and another partition's hub;

Access link - link between a node and its partition hub;

Trunk link - link between two hubs.

The following relationship between the three link arc capacities is assumed: capacity of a trunk arc $\geq$ capacity of an access arc $\geq$ capacity of a direct arc. Since aggregation of demand is most across partitions, next on the access links, and none in the direct link such an assumption will be reasonable.

Since all nodes are aggregate nodes we do not know how many arcs extend from the smaller nodes to an aggregate node. All aggregate nodes contain frames and the capacity of a frame is assumed the same at all the nodes. We estimate the number of ports required at a node by the following expression:

$$\text{number of arcs connecting at the node} + \left\lceil \frac{\text{total flow at the node}}{\text{minimum arc capacity}} \right\rceil = \alpha,$$

where α is a user specified parameter ($\alpha \geq 0$), and $\lceil a \rceil$ denotes the smallest integer greater than or equal to a .¹

¹Garfinkel and Segal [11] use a similar concept to estimate the additional ports required at an aggregate node.

NETWORK COSTS Since the number of hubs is given, the fixed hub costs will be constant for the problem. The node costs will be due to the number of frames required at each node. The other variable costs for the problem are: connection costs, arc costs and port utilization costs. The connection cost is incurred if there is direct connection between two nodes. Since there is no flow splitting, for direct connections the connection cost may be combined with arc costs to obtain a single cost measure per arc. This cost factor combination is shown as a part of the mathematical formulation.

OBJECTIVE The objective is to determine the flow routing, and hence the network links and the link capacities, such that the total relevant network cost is minimum.

We next present an integer linear programming formulation and then discuss the complexity of the problem.

5.2 Integer Linear Programming Formulation of Problem P3

- NOTATIONS
- b_{ij} - Flow between nodes i and j , expressed in number of channels
 - c^1 - Fixed cost per arc
 - c^{2A} - Cost per unit distance per arc in the access link
 - c^{2T} - Cost per unit distance per arc in the trunk link
 - c^{2D} - Cost per unit distance per arc in the direct link
 - c^3 - Port utilization cost per arc
 - c^4 - Cost per direct connection
 - C_{ij} - Arc cost in direct link i - j

$$= (c^1 + c^{2D} \cdot d_{ij} + 2c^3) g_{ij} + c^4 \quad \forall (i, j) \ni i, j \in \{N-P\} \text{ or } j \in P \text{ \& } i \in \{N-S_j-P\}$$
 - C_F - Cost per frame
 - d_{ij} - Distance between nodes i and j
 - F_i - Number of frames at node i
 - g_{ij} - Number of arcs in direct link i - j

$$= \left\lceil \frac{b_{ij}}{G^D} \right\rceil \quad \forall (i, j) \ni i, j \in \{N-P\} \text{ or } j \in P \text{ \& } i \in \{N-S_j-P\}$$

where $\lceil a \rceil$ is the smallest integer greater than or
equal to a
 - G^D - Arc capacity in channels in a direct link
 - G^A - Arc capacity in channels in an access link
 - G^T - Arc capacity in channels in a trunk link
 - N - Node set; $|N| = n$
 - P - Hub set; $|P| = p$
 - Q^F - Frame capacity in ports
 - S_j - Nodes in the same partition as hub node j
 - $w_{ij} = 1$ if flow between nodes i and j is routed direct
 $= 0$ otherwise (flow routed via the hubs)

X_{ij} - Number of arcs in the link i - j

$Y_{ij} = 1$ if nodes i and j are connected direct

$= 0$ otherwise

β_i - Estimate of extra ports required at node i

$$= \left\lceil \frac{\sum_{j \in N} b_{ij}}{\min(G^D, G^A, G^T)} \right\rceil * \alpha$$

α - User specified parameter ($\alpha \geq 0$)

M - Very large positive constant

(ILP3) Minimize

$$\begin{aligned} & \sum_{k \in P} \sum_{i \in S_k} (X_{ik} (c^1 + c^{2A} d_{ik} + 2c^3) + Y_{ik} c^4) \\ & + \sum_{\substack{k, l \in P \\ k < l}} (X_{kl} (c^1 + c^{2T} d_{kl} + 2c^3) + Y_{kl} c^4) + \sum_{i, j \in \{N-P\}} w_{ij} C_{ij} \\ & + \sum_{k \in P} \sum_{j \in \{N-S_k-P\}} w_{jk} C_{jk} + \sum_{i \in N} F_i C_F + \sum_{i \in N} \beta_i c^3 \end{aligned} \quad (5.1)$$

Subject to

$$G^A \cdot X_{ik} + \sum_{\substack{j \in N \\ j \neq k}} b_{ij} w_{ij} \geq \sum_{j \in N} b_{ij} \quad \forall i \in S_k, k \in P \text{ \& } i \neq k \quad (5.2)$$

$$G^T \cdot X_{kl} + \sum_{i \in S_k} \sum_{\substack{j \in S_l \\ i=k \Rightarrow j \neq l}} b_{ij} w_{ij} \geq \sum_{i \in S_k, j \in S_l} b_{ij} \quad \forall k, l \in P \ni k < l \quad (5.3)$$

$$Q^F F_k - \sum_{i \in S_k} X_{ik} - \sum_{l \in P} X_{kl} - \sum_{j \in \{N-P-S_k\}} g_{jk} \cdot w_{jk} - \beta_k \geq 0 \quad \forall k \in P \quad (5.4)$$

$$Q^F F_i - X_{ik} - \sum_{j \in \{N-k\}} g_{ij} \cdot w_{ij} - \beta_i \geq 0 \quad \forall i \in \{S_k - k\} \text{ \& } k \in P \quad (5.5)$$

$$Y_{ik} \cdot M \geq X_{ik} \quad \forall i \in S_k \& k \in P \quad (5.6)$$

$$Y_{kl} \cdot M \geq X_{kl} \quad \forall k, l \in P \& k < l \quad (5.7)$$

$$Y_{ij}, w_{ij} \in \{0,1\} \quad (5.8)$$

$$X_{ij}, F_i \geq 0, \text{ integer} \quad (5.9)$$

The first term in the objective function (5.1) measures the total cost of arcs in the access links. The second term measures the total cost of arcs in the trunk links. The costs of direct connections are measured by the third and the fourth terms. While the third term gives the total cost of direct connections between non-hub nodes, the fourth term gives the cost of direct connection between a node and a hub of another partition. The fifth term measures the total cost of frames, and the sixth the utilization cost for the additional ports required at the nodes. Constraint sets (5.2) and (5.3) define the capacity available in the access and the trunk links. The capacity available in these links must at least equal the flow along them. Constraint set (5.4) specifies that the number of ports available at a hub must at least equal the number of arcs connecting to it plus the additional ports required. Constraint set (5.5) defines the number of frames needed at a non-hub node. Constraint sets (5.6) and (5.7) check whether a direct connection exists between two nodes. Finally, the constraint sets (5.8) and (5.9) define the integrality requirements.

There will be $(n-p)$ X_{ik} variables, $\frac{p(p-1)}{2}$ X_{kl} variables, $\left[\frac{n(n-1)}{2} - (n-p) - \frac{p(p-1)}{2} \right]$ w_{ij} variables, $(n-p)$ constraints in the set (5.2),

and $\frac{p(p-1)}{2}$ in the set (5.3).

5.3 Complexity of P3

Consider an instance of 0-1 single knapsack problem.

$$\text{Maximize } \sum_{j=1}^n v_j x_j$$

subject to

$$\sum_{j=1}^n z_j x_j \leq W$$

$$x_j \in \{0, 1\}$$

z_j and v_j are the size and the value of item j , respectively, and W is the capacity of the knapsack. This problem is NP-hard (Garey & Johnson[10]).

Consider a variant of the 0-1 single knapsack problem in which exactly one variable is integer and the rest are 0-1 variables.

$$(KP1) \text{ Maximize } v_0 y + \sum_{j=1}^{n-1} v_j x_j$$

subject to

$$z_0 y + \sum_{j=1}^{n-1} z_j x_j \leq W$$

$$x_j \in \{0, 1\}$$

$$y \geq 0, \text{ integer}$$

Clearly, this variant is NP-hard since it contains the zero-one knapsack problem as a special case. Given any instance of KP1 we can transform it to an instance of problem P3 and thus show that P3 is NP-hard.

Define

$$y + y' = \left\lceil \frac{W}{z_0} \right\rceil = \alpha, \text{ and}$$

$$x_j + x'_j = 1.$$

Substitute for variables y and x in KP1 with their complements defined above. The transformed KP1-T will be as follows.

$$\begin{aligned} \text{(KP1-T)} \quad & \text{Minimize } v_0 y' + \sum_{j=1}^{n-1} v_j x'_j - \left(v_0 \alpha + \sum_{j=1}^{n-1} v_j \right) \\ & \text{subject to} \\ & z_0 y' + \sum_{j=1}^{n-1} z_j x'_j \geq \left(z_0 \alpha + \sum_{j=1}^{n-1} z_j - W \right) \\ & x'_j \in \{0, 1\} \\ & y' \geq 0, \text{ integer} \end{aligned}$$

Given an instance of the problem KP1 the instance of P3 obtained from its transformation KP1-T will have the following properties:

- . only two partitions
- . one hub in each partition
- . the number of nodes in a partition (not including the hub node) will exactly equal the number of 0-1 variables in the problem
- . demand between nodes in the same partition is zero
- . demand between nodes across the partitions is greater than or equal to zero
- . large frame capacity such that only one frame is required at each node
- . the following costs are zero:
 - cost per arc in the access links,
 - fixed cost per arc,

- port utilization cost, and
- connection cost.

Let the two hubs be labeled A and B , and the $(n-1)$ non-hub nodes be labeled $1A, 2A, \dots, (n-1)A$, and $1B, 2B, \dots, (n-1)B$, respectively.

The basic units of P3 can be constructed from units of KP1-T as follows:

demand between nodes jA and $jB = z_j$

trunk arc capacity $= z_0$

variable number of arcs in the trunk link $= y'$

direct routing variables $= x'_j$

flow between the hub nodes A and $B = (z_0 \alpha - W)$

cost per arc in the trunk link between hubs A and $B = v_0$

cost of direct connection between nodes jA and $jB = v_j$.

A numerical example illustrating this transformation is given in appendix C1.

5.4 Linear Programming Relaxation of ILP3

Next we look at the LP relaxation of P3 with a view to developing a LP based heuristic and a lower bound procedure.

Let the LP relaxation of ILP3 obtained by replacing the integrality constraints defined by (5.8) and (5.9) with

$$0 \leq Y_{ij}, w_{ij} \leq 1 \quad (5.2a)$$

$$X_{ij}, F_i \geq 0 \quad (5.3a)$$

be denoted LP3.

In ILP3 given the routing of flows the number of arcs in each link, the number of frames at each node, and the different costs can be computed. We will now show by contradiction that the values of the w_{ij} variables in the optimal LP solution will always be 0/1.

Let the values of the w_{ij} variables in the optimal LP solution be α_{ij} ($0 < \alpha_{ij} < 1$) and the corresponding optimal solution be denoted by S^* .

Given w_{ij} , it follows from 5.2 - 5.7 that the values for the X_{ij} , Y_{ij} and F_i variables will be:

$$X_{ik} = \frac{b_{ik}}{G^A} + \sum_{\substack{j \in N \\ j \neq k}} \frac{b_{ij}}{G^A} (1 - w_{ij}) \quad (5.10)$$

$$X_{kl} = \frac{b_{kl}}{G^T} + \sum_{i \in S_k} \sum_{\substack{j \in S_l \\ i=k \Rightarrow j \neq l}} \frac{b_{ij}}{G^T} (1 - w_{ij}) \quad (5.11)$$

$$F_k = \frac{1}{Q^F} \left(\sum_{i \in S_k} X_{ik} + \sum_{l \in P} X_{kl} + \sum_{j \in \{N-P-S_k\}} g_{jk} \cdot w_{jk} + \beta_k \right) \quad (5.12)$$

$$F_i = \frac{1}{Q^F} \left(X_{ik} + \sum_{j \in \{N-k\}} g_{ij} \cdot w_{ij} + \beta_i \right) \quad (5.13)$$

$$Y_{ik} = \frac{1}{M} \left(\frac{b_{ik}}{G^A} + \sum_{\substack{j \in N \\ j \neq k}} \frac{b_{ij}}{G^A} (1 - w_{ij}) \right) \quad (5.14)$$

$$Y_{kl} = \frac{1}{M} \left(\frac{b_{kl}}{G^T} + \sum_{i \in S_k} \sum_{\substack{j \in S_l \\ i=k \Rightarrow j \neq l}} \frac{b_{ij}}{G^T} (1 - w_{ij}) \right) \quad (5.15)$$

Substituting the above expressions for the X, the Y and the F variables in the objective function (5.1), and collecting the w_{ij} terms together we get the following.

For a node pair (i,j) such that $ij \in S_k$ - nodes i and j belong to the same partition, the objective function coefficient for the corresponding w_{ij} will be:

$$\left(C_{ij} + 2 \frac{C_F}{Q_F} \cdot g_{ij} \right) - \frac{b_{ij}}{G^A} \left(2c^1 + (d_{ik} + d_{jk}) c^{2A} + 4c^3 + 2 \cdot \frac{c^4}{M} + 4 \cdot \frac{C_F}{Q_F} \right) \quad (5.16)$$

For a node pair (i,j) such that $i \in S_k$ and $j \in S_l$ - nodes i and j belong to different partitions, the objective function coefficient for the corresponding w_{ij} will be as follows.

If node i is a hub and j is a non-hub the coefficient will be:

$$\left(C_{ij} + 2 \frac{C_F}{Q_F} \cdot g_{ij} \right) - \frac{b_{ij}}{G^A} \left(c^1 + d_{jl} \cdot c^{2A} + 2c^3 + \frac{c^4}{M} + 2 \cdot \frac{C_F}{Q_F} \right) - \frac{b_{ij}}{G^T} \left(c^1 + d_{il} \cdot c^{2T} + 2c^3 + \frac{c^4}{M} + 2 \frac{C_F}{Q_F} \right) \quad (5.17)$$

If node j is a hub and i is a non-hub the coefficient will be:

$$\left(C_{ij} + 2 \frac{C_F}{Q_F} \cdot g_{ij} \right) - \frac{b_{ij}}{G^A} \left(c^1 + d_{ik} \cdot c^{2A} + 2c^3 + \frac{c^4}{M} + 2 \cdot \frac{C_F}{Q_F} \right) - \frac{b_{ij}}{G^T} \left(c^1 + d_{kj} \cdot c^{2T} + 2c^3 + \frac{c^4}{M} + 2 \frac{C_F}{Q_F} \right) \quad (5.18)$$

If i and j are both non-hub nodes then the coefficient will be:

$$\left(C_{ij} + 2 \frac{C_F}{Q_F} \cdot g_{ij} \right) - \frac{b_{ij}}{G^A} \left(2c^1 + (d_{ik} + d_{jl}) \cdot c^{2A} + 4c^3 + 2 \cdot \frac{c^4}{M} + 4 \cdot \frac{C_F}{Q_F} \right) - \frac{b_{ij}}{G^T} \left(c^1 + d_{kl} \cdot c^{2T} + 2c^3 + \frac{c^4}{M} + 2 \frac{C_F}{Q_F} \right) \quad (5.19)$$

By setting those w_{ij} variables with negative coefficients to 1 and the rest to 0 we can obtain an objective function value better than S^* . Therefore the values of the w_{ij} variables in the optimal LP solution must be either zero or one. The values of the X_{ij} , Y_{ij} and F_i variables in the optimal LP solution will be fractional and can be computed using (5.10) - (5.85), respectively. $\square$

In (5.16) - (5.19) the first term in the expression is the cost of routing directly the flow between nodes i and j . The second term is the cost of routing the flow via the respective hubs. If the cost of routing directly is less than the cost of routing via the hubs then route the flow directly ($w_{ij} = 1$). Otherwise route the flow via the hubs ($w_{ij} = 0$).

(LP3) Using this scheme we can develop an $O(n^2)$ algorithm for the optimal solution to LP3. The procedure is described as follows.

step 1: Compute the cost of routing flow b_{ij} via the hub(s), and direct using (5.19) or (5.20).

step 2: If in step 1, the via hub routing cost is less than the cost of routing direct, then route the flow via the respective hubs ($w_{ij}=0$).

Otherwise route the flow direct ($w_{ij}=1$).

There are $\left\lceil \frac{n(n-1)}{2} \right\rceil$ flows, and the above two steps will be executed once for each flow. Thus, the procedure is $O(n^2)$.

5.5 Lower Bound

We can improve on the optimal solution to the LP relaxation to determine a better lower bound on the solution.

Frame costs:

For each non-hub node compute the minimum and maximum number of frames required at the node using the following expressions.

Minimum number of frames will be required if all the flows from the node are routed via the hub since by aggregating flows on access link we will need fewer arcs to carry the flow, and hence fewer number of ports. This is true since the access arc capacity will be greater than or equal to the direct arc capacity. Maximum number of frames will be required when all the flows from a non-hub node are routed direct. This is true because each flow that is routed direct will add at least one arc and hence the need for an additional port in the frame. Then the minimum and the maximum number of frames necessary will be:

$$\min F_i = \left\lceil \frac{1}{Q^F} \cdot \left(\left\lceil \frac{\sum_{j \in N} b_{ij}}{G^A} \right\rceil + \beta_i \right) \right\rceil$$
$$\max F_i = \left\lceil \frac{1}{Q^F} \cdot \left(\sum_{\substack{j \in N \\ j \neq k}} \left\lceil \frac{b_{ij}}{G^D} \right\rceil + \left\lceil \frac{b_{ik}}{G^A} \right\rceil + \beta_i \right) \right\rceil$$

If the minimum and the maximum number of frames are equal, then set the number of frames required to this value. If the number of frames in the LP solution is less than the minimum number of frames then set the number of frames to the minimum value. Otherwise, set the number of frames required

to the value in the optimal LP solution. If the number of frames in the LP solution is less than 1 then set the number of frames to 1.

For each hub node compute the maximum and the minimum number of frames required using the following expressions. Under the following conditions maximum number of frames will be required at a hub node:

- a. flows between a node in the partition and other nodes are routed via the hub;
- b. flows between the hub node and non-hub nodes in other partitions are routed direct.

When the two conditions are reversed the minimum number of frames will be required at the hub node.

$$\max F_k =$$

$$\left\lceil \frac{1}{Q^F} \left(\sum_{\substack{i \in S_k \\ i \neq k}} \left\lceil \sum_{j \in N} \frac{b_{ij}}{G^A} \right\rceil + \sum_{\substack{l \in P \\ l \neq k}} \left\lceil \frac{\sum_{\substack{i \in S_k \\ i \neq k}} \sum_{j \in S_l} b_{ij} + b_{kl}}{G^T} \right\rceil + \sum_{j \in \{N-S_kP\}} \left\lceil \frac{b_{jk}}{G^D} \right\rceil + \beta_k \right) \right\rceil$$

$$\min F_k = \left\lceil \frac{1}{Q^F} \left(\sum_{\substack{i \in S_k \\ i \neq k}} \left\lceil \frac{b_{ik}}{G^A} \right\rceil + \sum_{\substack{l \in P \\ l \neq k}} \left\lceil \sum_{j \in S_l} \frac{b_{jk}}{G^T} \right\rceil + \beta_k \right) \right\rceil$$

If the minimum number of frames required equals the maximum number required, then set the number of frames at the hub node equal to this value.

If the number of frames in the LP solution is less than the minimum number of frames then set the number of frames equal to the minimum value.

Otherwise, set the number of frames required to the value in the LP optimal

solution. If the number of frames in the LP solution is less than 1 then set the number of frames to 1.

This refinement is identical to adding to the mathematical model an upper bound constraint and a lower bound constraint on the number of frames necessary at each node.

Connection costs:

If a connection exists between a non-hub node and its hub or between two hubs consider the full connection cost only if the minimum flow between the respective nodes is positive. If the minimum flow is positive there will be at least one arc in the link. Otherwise take the connection costs as given in the LP solution. This is identical to adding a lower bound constraint on the number of arcs (number of arcs ≥ 1) in the links with positive minimum flow, and setting the connection cost full(c^4) for the respective links.

Port utilization costs:

Take the port costs as given in the LP optimal solution.

Arc costs:

Take the arc costs as given by the LP optimal solution. If in the LP solution the number of arcs in an access link or a trunk link is less than 1, then set the number of arcs in the respective link to 1 if the minimum flow in the link is positive.

The lower bound on solution is determined by the sum of the revised frame costs, connection costs, port utilization costs, and arc costs. Through this we are able to determine a better lower bound than that given by the LP optimal solution. Also, the number frames required at some of the non-hub and hub nodes may be fixed by this process.

5.6 Heuristic Solution Procedure for P3

MOTIVATION The variables which may take noninteger values in the LP optimal solution are the number of arcs in the access and trunk links, and the number of frames at each node. We can use the LP solution as a starting point and modify the number of arcs in the access and trunk links and/or the number of frames at the nodes, thereby changing the capacity in these links and at the nodes, to derive a feasible integer solution. When capacity of the links and nodes are modified it may be necessary to reroute some of the flows. We use the LP optimal solution as a starting point to develop two feasible solutions and then choose the best. One feasible solution is obtained by rounding up the number of arcs in the LP solution and rerouting via hub some of the flows that are currently routed direct. To obtain the second feasible solution the number of arcs in the LP solution are first rounded down, thus forcing some of the flows direct. Next, the solution is improved by selectively rerouting via hub some of the flows routed direct, and augmenting capacity in some of the access/trunk links. We use the smallest direct routing cost as a basis for selecting flows to reroute direct, and the largest (direct routing cost/flow) ratio as a basis for selecting flows to reroute via hub. In other words by the first approach we are segregating flows.

Hence we want to incur the least additional expenses doing this. In the second approach, we are aggregating the flows and hence wish to achieve the maximum reduction in direct routing cost. These are greedy selections and hence may not be optimal. The latter is similar to the bang-for-buck ratio used in the solution to knapsack problems. In both methods the number of frames at the nodes is treated as a continuous variable and when computing the network cost the fractional value is rounded up to an integer.

We shall refer to the feasible solution obtained by rounding up the number of arcs as FS1, and the feasible solution obtained by rounding down the number of arcs as FS2. We next describe in algorithmic form the two procedures.

- (FS1)
1. Compute the LP optimal solution using procedure (LP3) described in section 5.4.
 2. Round up the number of arcs in the access and the trunk links. For links which have integer number of arcs increase the count by 1. Compute the *available excess capacity* in each of the access/trunk link. This is equal to (link capacity - total flow via link).
 3. For all the flows routed direct compute the ratio (cost of direct routing/flow), and form a list (DL) ordering the flows in the descending order of this ratio.
 4. Select the flow with the next largest ratio from the list and check whether the *available excess capacity* in the links is sufficient to reroute it via hub. If sufficient, reroute the flow, adjust the *available excess capacity* in the affected links, and delete the flow from list (DL). Choose the next flow in the list and repeat step 4. If list is completed go to step 5.

5. If list (DL) is not empty go to step 6. Otherwise continue. Compute the number of frames required at each node. In those links where the *available excess capacity* is greater than or equal to the arc capacity reduce the number of arcs by one and adjust the available excess capacity. Compute the total network cost and stop. We have the feasible solution (FS1).
6. Select from the list (DL) the flow with the next largest ratio and determine the cost of routing the flow via hub(s). This is equal to the cost of augmenting capacity in the appropriate links. If the cost of routing flow via hub(s) is less than the cost of routing it direct, then reroute the flow via hub(s) augmenting capacity in the links appropriately, and delete it from the list. Also, adjust the *available excess capacity* in the links affected. If the list is completed then go to step 7. Otherwise, select the next flow in the list and repeat the step.
7. Compute the number of frames required at each node. In those links where the *available excess capacity* is greater than or equal to the arc capacity reduce the number of arcs by one and adjust the available excess capacity. Compute the total network cost, and set it as the current network cost. If list (DL) is not empty go to step 8. Otherwise stop. We have the feasible solution (FS1).
8. Suppose the flows currently routed direct are to be rerouted via hub(s). For each link determine the volume of additional flow due to this rerouting. If in a link (the volume of additional flow – the *available excess capacity*) is greater than half the arc capacity, then increase the number of arcs in the link by one. Compute the current *available excess capacity* in the access and the trunk links.

9. Select the flow with the largest ratio from the list (DL) and verify whether the *available excess capacity* in the links is sufficient to reroute it via the hubs. If sufficient, reroute the flow, delete it from the list, and adjust the *available excess capacity* in the affected links. If the list is not completed, select the flow with next smallest ratio and repeat the step. Otherwise continue.
10. Select from the list (DL) the flow with the next largest ratio and determine the cost of routing the flow via hub(s). This is equal to the cost of augmenting capacity in the appropriate links. If the cost of routing flow via hub(s) is less than the cost of routing it direct, then reroute the flow via hubs augmenting capacity in the links appropriately, and delete it from the list. Also, adjust the *available excess capacity* in the links affected. If the list is completed then go to step 11. Otherwise, select the next flow in the list and repeat the step.
11. In those links where the *available excess capacity* is greater than or equal to the arc capacity reduce the number of arcs by one and adjust the available excess capacity. Compute the number of frames required at each node, and the new total network cost. If the new total network cost is less than the current network cost, then revise the current cost and the configuration, and repeat steps 8-11. Each repetition of the steps 8-11 is a cycle. If there is no change in the network configuration and the network cost in two successive cycles then terminate the procedure. The current solution is the first feasible solution.

(FS2)

We next describe the procedure for the second feasible solution.

1. Compute the LP optimal solution using procedure (LP3) described in section 5.4.
2. Round down the number of arcs in the access and the trunk links, except as follows. For those links which have less than one arc, if the minimum flow in the link is positive then round up the number of arcs to one. For those links which have an integer number of arcs do not modify the number. Compute the *over capacity* for each for each access/trunk link. This equals (the current total flow via link – link capacity). A negative *over capacity* means *excess capacity* in the link.
3. Form a list (VL) ordering all flows routed via hub in the ascending order of the direct routing cost.
4. Select the flow first in the list and check whether rerouting the flow direct will decrease the *over capacity* in the via hub path links it uses. If yes, reroute the flow direct, adjust the *over capacity* in the affected links, and delete the flow from list (VL). Select the flow next in the list and repeat the step. If there is no change in the list (VL) then compute the number of frames required at each node rounding up non-integer values, and the total network cost. We have the feasible solution (FS2). Otherwise continue.
5. For each access/trunk link compute the *available excess capacity*. Form a list (DL) ordering the flows routed direct in the descending order of the ratio (cost of direct routing/flow).
6. Select the flow first in the list and check whether the *available excess capacity* in the links is sufficient to reroute it via hub. If sufficient, reroute the flow, adjust the *available excess capacity* in the affected links, and

- delete the flow from list (DL). Choose the flow next in the list and repeat step 6. If list is completed go to step 7.
7. Select flow first in the list (DL) and determine the cost of routing the flow via hub(s). This is equal to the cost of augmenting capacity in the appropriate links to reroute the flow. If the cost of routing flow via hub(s) is less than the cost of routing it direct, then route the flow via hubs augmenting capacity in the links appropriately, and delete it from the list. Also, adjust the *available excess capacity* in the links affected. If the list is not completed then, select the flow next in the list and repeat the step. Otherwise go to step 8.
8. Compute the number of frames required at each node, rounding up non-integer values. Compute the total network cost, and set it as the current network cost. If list (DL) is not empty go to step 9. Otherwise stop. We have the feasible solution (FS2).
9. Suppose the flows currently routed direct are to be rerouted via hub(s). For each link determine the volume of additional flow due to this rerouting. If in a link (the volume of additional flow – the *available excess capacity*) is greater than half the arc capacity, then increase the number of arcs in the link by one. Compute the current *available excess capacity* in the access and the trunk links.
10. Select the flow with the largest ratio from the list (DL) and verify whether the *available excess capacity* in the links is sufficient to reroute it via the hubs. If sufficient, reroute the flow, delete it from the list, and adjust the *available excess capacity* in the affected links. If the list is not completed, select the flow with next smallest ratio and repeat the step. Otherwise continue.

11. Select from the list (DL) the flow with the next smallest ratio and determine the cost of routing the flow via hub(s). This is equal to the cost of augmenting capacity in the appropriate links to route the flow. If the cost of routing flow via hub(s) is less than the cost of routing it direct, then route the flow via hubs augmenting capacity in the links appropriately, and delete it from the list. Also, adjust the *available excess capacity* in the links affected. If the list is completed then go to step 12. Otherwise, select the flow next in the list and repeat the step.
12. In those links where the *available excess capacity* is greater than or equal to the arc capacity reduce the number of arcs by one and adjust the available excess capacity. Compute the number of frames required at each node rounding up non-integer values, and the new total network cost. If the new total network cost is less than the current network cost, then revise the current cost and the configuration, and repeat steps 9-12. Each repetition of the steps 9-12 is a cycle. If there is no change in the network configuration and the network cost in two successive cycles then terminate the procedure. The current solution is the second feasible solution.

An example illustrating the feasible solution procedures FS1 and FS2 is given appendix C2.

5.7 Computational Experience

EXPERIMENTAL To study the performance of the solution procedures for P3 sample problems
DESIGN were generated using a random process. The flow between nodes and the coordinates of nodes were generated from a uniform distribution. The

distance measure generated is euclidean. Cost per arc per unit distance in the access links is assumed the same; similarly in the trunk and direct links. Due to economies of scale the cost per channel in the trunk group is likely to be smaller than the cost per channel in the access group, which in turn is likely to be smaller than the cost per channel in the direct link. In the experimental design, we consider the ratio of cost per channel in the trunk link to cost per channel in the access link to cost per channel in the direct link. Given the cost per channel the cost per arc will be (cost per channel * arc capacity).

DESIGN The number of nodes, the number of partitions, and the ratio of per channel
VARIABLES cost were selected as the design parameters. These were selected in order to
study the effect of the number of nodes and partitions on total network cost
and solution difficulty, and the effect of the cost ratio on flow routing.

Two node sizes - 10 and 20, were used. For each node size three different node distributions were generated. For both node sizes the number of partitions created were 2 or 5. Each problem generated was tested using three cost ratios (Direct/Access/Trunk) - 1/1/.5, 1/1/.8, and 1/1/1. A graphic illustration of this design scheme is given below.

10		20	
(3 problems)		(3 problems)	
2	5	2	5
(2 partitions per type)			
1/1/.5	1/1/.8	1/1/1	
(3 cost ratios)			

The three solution procedures - FS1, FS2, and LP were tested with the sample problems. For each method, the solution and the execution time (CPU time in seconds) were recorded. The execution time was the straight run time without input/output. All algorithms were coded in Turbo Pascal (version 4) and implemented on an IBM PC-XT compatible with a clock speed of 8Mhz and a 8087 coprocessor.

The results for the sample problems are summarized in tables 5.1 and 5.2.

From the experience with the sample problems it appears that the gap between the lower bound obtained from the optimal LP solution and the best solution obtained using the heuristics FS1/FS2 widens with the number of nodes and the number of partitions for a given number of nodes. The size of the gap depends on factors such as the frame cost, the ratio of per channel cost in the direct/access/trunk links, and the link distances. The average gap between the lower bound and the best heuristic solution indicate that the methods are reasonably good. When the per channel cost in the trunk links is low there is a tendency for the flows to be routed via the hubs and as the size of the cost increases the flows tend to be routed direct. This indicates that when economies of scale applies aggregating flows in the access and the trunk links will decrease the total network cost.

Table 5.1 Summary of results for 10 node problems

10 Nodes	Partitions	
	2	5
# of problems	18	18
LP Solution: Average run time (secs.)	0.34	0.45
Heuristic FS1 & FS2: Average run time (secs.)	1.27	1.75
Maximum run time	1.42	1.92
Minimum run time	0.95	1.48
Avg. % deviation from LB	5.00	7.00
Max. % deviation from LB	7.00	10.00
Min. % deviation from LB	2.00	4.00

Table 5.2 Summary of results for 20 node problems

20 Nodes	Partitions	
	2	5
# of problems	18	18
LP Solution: Average run time (secs.)	0.44	0.58
Heuristic FS1 & FS2: Average run time (secs.)	2.87	3.89
Maximum run time	3.54	5.22
Minimum run time	2.35	2.89
Avg. % deviation from LB	7.00	8.50
Max. % deviation from LB	10.00	11.00
Min. % deviation from LB	3.00	4.00

CHAPTER 6

THE GENERAL PROBLEM: FORMULATION & ALGORITHMS

6.1 Problem Statement

PROBLEM STATEMENT Given a collection of n nodes in a plane. The matrix of flows between the nodes, expressed in number of channels, is given and is symmetric. Flow between nodes is undirected, and is assumed deterministic and static over some planning period. There is no flow from a node to itself. Every node is considered as an origin/destination of flows. The matrix of direct distances between the nodes is given, and is symmetric and metric. The coordinates of the nodes are known. A specified number of p hubs ($p < n$) are to be located at a subset of the nodes. Each node will be assigned to exactly one hub, and thus the nodes will be partitioned into p subsets. The assignment of a node to a hub means that the flow from(to) a node may be aggregated(distributed) only at(from) the hub to which it is assigned. The flow routing strategy is that all pair wise flows will be routed either (a) direct, or (b) via the respective hubs. In other words, the nodes are connected either direct or via the assigned hub(s). For example, consider the flow between nodes i and j . Suppose both nodes are assigned to a hub at u . Then, the flow between i and j may be routed either direct ($i-j$) or via hub u ($i-u-j$). If nodes i and j are assigned to hubs at u and v , respectively, the flow between them may be routed either direct ($i-j$) or via the hubs ($i-u-v-j$). Flow between nodes will not be split for routing. A direct connection between two nodes is defined a *link*. The link between two non-hub nodes or that between a non-hub node and a hub other than to which it is assigned will be called a *direct link*; the link between a node and its assigned hub will

be called an *access link*; and finally the link between two hub nodes will be called a *trunk link*. Flow in any link is bundled into groups of fixed capacity. The capacity of a group is expressed in number of channels. In network terminology, each group could be considered a fixed capacity arc, and there may be multiple arcs in any link. We will use the terms groups and arcs interchangeably. We shall consider the capacity of arcs as given, and these may be different for each link. Since aggregation of flow is likely to be most in the trunk links, none in the direct links, and somewhat in the access links a higher arc capacity in the trunk links will provide economies of scale. By the same reasoning when the arc capacities would be different, the arcs in the trunk links will have, comparatively, the largest capacity; the arcs in the direct links the smallest; and the arcs in the access links in between. Each arc connects to frames at each end, and occupies one port on each frame. The capacity of a node is expressed in number of frames, with each frame having a fixed number of ports. The capacity of nodes and links are assumed unlimited. The connections between the nodes serve the primary demand only, without backup routes.

Calculating number of ports required at each node:

All nodes are aggregate nodes and contain frames. Since the network nodes are aggregate nodes, we do not know the number of groups extending from the small nodes to a network node. The capacity of a frame is assumed the same at all the nodes. We estimate the number of ports required at a node by the following expression:

$$\text{number of arcs connecting at the node} + \left\lceil \frac{\text{total flow from the node}}{\text{minimum arc capacity}} \right\rceil * \alpha,$$

where α is a user specified parameter ($\alpha \geq 0$) and $\lceil \cdot \rceil$ denotes the smallest

integer greater than or equal to a .

RELEVANT	The variable costs associated with providing the necessary connections are:
COSTS	fixed node costs: fixed costs associated with the building to house the frames; variable node costs: variable costs associated with the number of frames at each node; connection costs: one time fixed cost associated with providing a direct connection between two nodes; group costs: variable costs based on the number of arcs in the link, and the link distance; port utilization costs: fixed cost of connecting an arc to a port in a frame. Given the capacities, the variable node costs, the group costs, and the port utilization costs can be assumed to be linear functions - i.e., number of items * cost per item.
PLANNING	The objective is to determine the minimum cost topology of the network the
OBJECTIVE	hub structure satisfying the flow requirements. Determining the minimum cost topology will involve: finding the hub locations; assigning nodes to hubs; assigning network links and link capacities to provide the necessary connections between the nodes according to the routing strategies; and assigning frame capacities.

6.2 Mathematical Formulation

MATHEMATICAL Integer Nonlinear Programming Formulation of GP

FORMULATION

Notation:

G^A - Group capacity in access links

G^D - Group capacity in direct links

G^T - Group capacity in trunk links

Q^F - Capacity of frame

p - Number of hubs

N - Node set; $|N| = n$

b_{ij} - Flow between nodes i and j

Z_{ij} - Number of groups in the access link from node i to its assigned hub at j

U_{ij} - Number of groups in the direct link between node i and node j , where either i or j , or both are nonhub nodes

$= \left\lceil \frac{b_{ij}}{G^D} \right\rceil$, where $\lceil a \rceil$ is the smallest integer greater than or equal to a

W_{kl} - Number of groups in the trunk link between hubs at k and l

$X_{ii} = 1$ if node i is a hub site

$= 0$ otherwise

$X_{ik} = 1$ if node i is assigned to hub at location k

$= 0$ otherwise

$\omega_{ij} = 1$ if flow b_{ij} is routed via the respective hub(s)

$= 0$ otherwise

$\bar{\omega}_{ij} = (1 - \omega_{ij})$

$= 1$ if flow b_{ij} is routed direct

$= 0$ otherwise

Y_{ij} = 1 if there is a link between nodes i and j
 = 0 otherwise

M - Very large positive constant

c^1 - Fixed cost per group

c^{2A} - Cost per unit distance per group in an access link

c^{2T} - Cost per unit distance per group in a trunk link

c^{2D} - Cost per unit distance per group in a direct link

c^3 - Port utilization cost

c^4 - Fixed cost per direct connection

d_{ij} - Distance between nodes i and j

C_{ij} - Total cost per group in link i - j

$C_{ij}^A = (c^1 + c^{2A} * d_{ij} + 2c^3)$, if i - j is an access link

$C_{ij}^T = (c^1 + c^{2T} * d_{ij} + 2c^3)$, if i - j is a trunk link

C_{ij}^D - Total arc cost in direct link i - j

$= (c^1 + c^{2D} * d_{ij} + 2c^3) * U_{ij} + c^4$

F_i - Number of frames at node i

C^F - Cost per frame

β_i - Estimate of extra ports required at node i

$$= \left\lceil \frac{\sum_{j \in N} b_{ij}}{G^D} \right\rceil * \alpha, \text{ where } \lceil a \rceil \text{ is the smallest integer greater than or equal to } a$$

α - User specified parameter ($\alpha \geq 0$)

(GP)

Minimize

$$\sum_{i \in N} \sum_{j \in N} (Z_{ij} C_{ij}^A + Y_{ij} \cdot c^4) + \sum_{\substack{k, l \in N \\ k < l}} (W_{kl} C_{kl}^T + Y_{kl} \cdot c^4) + \sum_{i \in N} C^F F_i + \sum_{i \in N} \sum_{j \in N} C_{ij}^D \cdot \bar{\omega}_{ij} \quad (6.1)$$

subject to

$$\sum_{i \in N} X_{ii} = p \quad (6.2)$$

$$\sum_{j \in N} X_{ij} = 1 \quad \forall i \in N \quad (6.3)$$

$$\sum_{i \in N} X_{ij} \leq (n-p+1) X_{jj} \quad \forall j \in N \quad (6.4)$$

$$G^A \cdot Z_{ik} \geq b_{ik} X_{ik} + X_{ik} \sum_{\substack{j \in N \\ j \neq k}} b_{ij} \omega_{ij} \quad \forall i, k \in N \text{ \& } i \neq k \quad (6.5)$$

$$G^T \cdot W_{kl} \geq b_{kl} X_{ll} X_{kk} + \sum_{i \in N} X_{ik} \sum_{\substack{j \in N \\ i = k \Rightarrow j \neq l}} b_{ij} \omega_{ij} X_{jl} \quad \forall k, l \in N \ni k < l \quad (6.6)$$

$$\sum_{\substack{i \in N \\ i \neq k}} (Z_{ki} + Z_{ik}) + \left(\sum_{\substack{j \in N \\ j < k}} U_{jk} \bar{\omega}_{jk} + \sum_{\substack{j \in N \\ j > k}} U_{kj} \bar{\omega}_{kj} \right) (1 - X_{kj})(1 - X_{kk}) + \left(\sum_{\substack{j \in N \\ j < k}} U_{kj} \bar{\omega}_{kj} + \sum_{\substack{j \in N \\ j > k}} U_{kj} \bar{\omega}_{kj} \right) (1 - X_{jj})(1 - X_{jk}) X_{kk} +$$

$$\sum_{\substack{l \in N \\ l \neq k}} W_{kl} + \beta_k \leq Q^F F_k \quad \forall k \in N \quad (6.7)$$

$$M \cdot Y_{ik} \geq Z_{ik} \quad \forall i, k \in N \text{ \& } i \neq k \quad (6.9)$$

$$M \cdot Y_{kl} \geq W_{kl} \quad \forall k, l \in N \ni k < l \quad (6.10)$$

$$\bar{\omega}_{ij} = 1 - \omega_{ij} \quad \forall i, j \in N \quad (6.10)$$

$$X_{ij}, Y_{ij}, \omega_{ij}, \bar{\omega}_{ij} \in \{0, 1\} \quad (6.11)$$

$$W_{ij}, Z_{ij}, F_i \geq 0 \text{ integer} \quad (6.12)$$

In the objective function (6.1) the first term gives the total group cost in the access links, the second term gives the total group costs in the trunk links, the third gives the total frame costs, and the fourth the total cost of direct links. Constraint (6.2) defines that a subset of p nodes will be selected as hub sites. The constraint (6.3) defines that each node will be assigned to exactly one hub. The constraint (6.4) gives that the total number of nodes assigned to any one hub can at most be $(n - p - 1)$. This includes the assignment of a node to itself. This is a relaxed form of the constraint $X_{ij} \leq X_{jj} \forall i \in N$. The constraint (6.5) indicates that the capacity of access links must at least equal the flow via them. Constraint (6.6) indicates the same for the trunk links. Constraint (6.7) specifies the number of frames necessary at a node. The first term on the right gives the number of access arcs connecting to the node k ; the second term gives the number of direct arcs connecting to k if it were a non-hub node; the third term gives the number of direct arcs connecting to k if it were a hub node; the fourth term gives the number of trunk arcs connecting to k if it were a hub node; and the final term gives the arcs extending to the smaller nodes from the aggregate node k . Constraints (6.8) and (6.9) verify a link between two nodes. Constraint (6.10) gives the relationship between ω_{ij} and $\bar{\omega}_{ij}$. Finally, the constraints (6.11) and (6.12) specify the integrality requirements.

Since this formulation is very complex it is of little value for algorithm development purposes.

We next discuss the complexity of the general problem, and solution and bounding approaches.

6.3 Complexity of the General Problem

We have already shown that problems P1, P2, and P3 belong to the class of NP-Complete problems. By the following arguments we show that the general problem is at least as hard as problems P1, P2, and P3.

With any partition of nodes and a choice of hubs the general problem is similar to problem P3, and such instances therefore will be at least as hard as P3 to solve.

Some instances of the general problem will be identical to problem P2.

Consider an instance of GP with the following restrictions:

- a. large frame capacity so that only one frame is necessary at each node;
- b. very large arc capacity in the trunk and access links compared to the direct links, so that only one arc is required in each trunk and access links;
and,
- c. all flows are positive.

Since the access arc and the trunk arc capacities are large compared to the direct arc all the flows will be routed via the hubs, and thus this instance of the general problem will be similar to problem P2. Hence we can say that the instances of the general problem similar to P2 will be at least as hard as P2 to solve.

For any partition of nodes with the restrictions a-c above the general problem will be similar to problem P1. Hence those instances of the GP similar to P1 will have comparable computational complexity.

Therefore, we may conclude that solution to the general problem will be at least as hard as the three research problems discussed before.

6.4 Solution Approaches

MOTIVATION The decisions involved in the GP are:

- . location of hubs,
- . assignment of nodes to hubs,
- . choice of flow routes.

The network links, link capacities, and frame capacities are a byproduct of these decisions. Attempting these decisions simultaneously is difficult and hence we shall make them sequential. First we will develop the partitions, then locate the hubs, and finally choose the flow routing. However, to develop the partitions and to locate the hubs we need to assume a routing strategy. If we assume the choice of direct or via hubs we go back to the original problem which is very complex. As a measure to simplify the analysis we will assume only via hub routing for the first two decisions, and then apply the flow routing choice to improve the solution. This will mean that we can use methods developed for P2 to decide the partitions and locate the hubs, and then apply the methods developed for P3 to improve the solution.

The above solution approach for the general problem is a combination of the solution approaches for the problems P2 and P3. Since we have already presented the computational experience with these methods we will not present a computational experience for the general problem.

6.5 Lower Bounds

From (6.1) we can see that the items making up the total cost of the network are:

- a. total access arc cost,
- b. total direct arc cost,
- c. total trunk arc cost, and
- d. total frame cost.

We can compute a lower bound on total cost by determining a lower bound on each of these items or a combination of these items. Let us consider a n -node p -hub problem. When we have a routing choice the least number of arcs out of a non-hub node will be when all flows are aggregated together on the access link. If flow out of a non-hub node is only partially aggregated - some flows are routed direct and hence not aggregated, then the number of arcs incident at the node will be higher. We can find a lower bound on the combined total of (access+direct) costs by taking the product of the $(n-p)$ least Z_i values $\left(Z_i = \left\lceil \frac{\sum_{h \in N} b_{ih}}{G^A} \right\rceil \right)$ arranged in an descending order, and the $(n-p)$ least cost per arc among the candidate access and direct links, arranged in an ascending order. If all pair wise node flows are positive then at least there will be one arc in each trunk link. We can get a lower bound on the trunk arc cost by taking the sum of the $\frac{p(p-1)}{2}$ least cost per arc among the candidate trunk links. The number of frames at each node is determined by the number of arcs incident at the node. A lower bound on the number of arcs incident at a node is given by the $(Z_i + \beta_i)$ value. For nonhub nodes we can get a lower bound on the frame cost by taking the $(n-p)$ smallest $(Z_i + \beta_i)$ values. For a hub node the number of

arcs connecting to it equals (the number of arcs in the access links + number of arcs in the direct links + the number of arcs in the trunk links + the arcs extending to the nodes clustered in the aggregate node). A lower bound for the sum of first and second items will be the sum of the $(n - p)$ smallest Z_i values. If we assume that there will be at least one arc in each trunk link, a lower bound on the third item will be $p(p - 1)$. A lower bound on the fourth item will be the sum of the p smallest β_i values.

This scheme is similar to that proposed for problem P2.

CHAPTER 7

SUMMARY, FUTURE DIRECTIONS and PLANNING ISSUES

SUMMARY In this research we have presented a hub location and flow routing problem and three special situations of the general problem - problems P1, P2, and P3, discussed their complexity, and solution and bounding procedures. Since the general problem is very complex we attempt to solve it using the methods developed for problems P2 and P3. To evaluate the effectiveness of the heuristic solution procedures lower bounds are developed. The methods were coded in Turbo Pascal and implemented on an IBM PC. Experience with randomly generated sample problems are reported.

The research presented was limited by the following assumptions:

- a. the number of hubs, and hence the number of partitions, are assumed given;
- b. there are no backup routes for the flow;
- c. the routing strategy is limited to via the respective hubs or direct.

Relaxing these assumptions will extend our research and we next explore the possible extensions.

FUTURE In the general problem and its special cases the number of hubs, and hence
DIRECTIONS the number of partitions, were assumed as given. Suppose the number of hubs is set up as a variable. This situation can be handled by repeating the methods suggested for a varying number of hubs and selecting the best configuration. However, since the solution procedures suggested are not optimal this approach may not yield the best configuration always. One may

begin with two hubs (or two partitions) and add more hubs (or partitions) as long it proves worthwhile. Several criteria may be used when selecting a new hub, for e.g. minimum distance between hubs, maximum number of nodes assigned to a hub, node with the largest demand etc. For example the two nodes with maximum demand may be selected as the first two hubs. Then the node with the next highest demand which is at least at a minimum distance from the existing hubs is selected as the next hub location. The nodes may be assigned to the nearest or the second nearest or the third nearest hub and so on. For each configuration the total cost is evaluated and the procedure is terminated when either the total cost of the new configuration exceeds the current cost or each non-hub node is within the minimum distance from the existing hubs. The criterion of minimum distance between hubs may also be combined with the method suggested in section 4.5.2.1 to select an appropriate number of hubs. Another approach would be to start with the least cost two hub configuration and add another location which yields the lowest three hub configuration. The procedure is continued in this manner and terminated when no improvement in cost results by adding the new location. In all the above heuristic methods we have to use a node-hub assignment rule (e.g. assignment to nearest hub) to create the partitions. As an alternative, the mathematical formulation of the problems can be revised to reflect the number of hubs as a variable and one may attempt to solve the problems optimally. However, this may prove to be difficult in the light of their complexity.

When solving problem P2 and the general problem optimally we may get unattractive node-hub assignments. For example assigning Atlanta to a hub in

Seattle. In the heuristic solution approaches considered we avoided this issue by the nearest hub allocation rule. A scheme to avoid such unattractive assignments is to define a set of potential hub sites and also a set of candidate hubs to which a node can be assigned.

In this research and those reviewed in Chapter 2, except for the concentrator location problem, all the links of the network carry only the primary flow between the nodes and the flow between any two nodes is totally routed via one path. In the event of a link breakdown the communication between some pairs of nodes will be totally disrupted. By providing backup paths total disruption in service can be avoided. This can be accomplished as follows:

- a. split flow between any pair of nodes and route the flows via different paths; or,
- b. route the primary flow via one path, and a proportion of the primary flow which will be the backup flow via a different path.

Essentially this creates more than one path between any two nodes. When providing backup paths one must make sure that the paths do not share links. With the routing strategies considered here both direct and via hub paths will be used to provide the backup. If the routing strategies are expanded to consider routing via other hubs - hubs other than the ones to which the two nodes are attached, then we may have more choices regarding the primary and the backup connections.

The routing strategies considered in this study were limited to either via the assigned hubs or direct. By these strategies there may be direct links

between non-hub nodes and hubs other than the one to which they are assigned. Such links carry only the flow between the two nodes. We may consider aggregating flows on these links. Suppose k and l are hubs, and node i is attached to hub k and node j is attached to hub l . Also, there exists a direct link between i and l , and between j and k . Then the flow between i - j may be routed either direct or via both the hubs i - k - l - j or via one of the hubs i - k - j or i - l - j . Garfinkel and Segal [11] consider this routing strategy.

We have shown that problem P3 possess the characteristics of the multiple knapsack problem. In the multiple knapsack problem given the knapsack sizes we wish to select the items to be included in the knapsacks. The similarity between P3 and the multiple knapsack problem is:

- a. the capacity of each access/ trunk link corresponds to the size of knapsacks;
- b. each flow corresponds to an item considered for inclusion in the knapsack.

Since in P3 we attempt to determine the capacity of the links it will be similar to sizing the knapsacks. Thus we have an interesting problem, which we call "sizing multiple knapsacks", where given the cost of each type of knapsack and the penalty for not including an item in the knapsack we wish to determine the size of the knapsacks and the items to be included in each, minimizing the total cost. In the context of P3 the penalty for not including an item in the knapsacks is the cost of routing the flow directly, and the cost per knapsack is the cost per arc. This problem has not been researched so far.

In our research we have considered only deterministic and static flows. This could be extended to scenarios where the flows are single-period and stochastic, multi-period and deterministic, and multi-period and stochastic. When planning a network multi-period considerations will be very appropriate.

PLANNING Any decision maker encountering a problem similar to the GP will be faced
ISSUES with choices concerning the following:

- a. number of hubs and their locations,
- b. assignment of nodes to hubs,
- c. strategies to route flow between nodes, and
- d. selection of links and link capacities.

It is very likely that flow between nodes(demand) will not remain static over the life of the system. Existing demand may modify (either increase or decrease) and new demands may arise. Since hubs are permanent facilities it may not be possible to relocate them with the change in pattern of demand; as such the number of these facilities and their locations must be planned carefully considering demand patterns over the time period. The volume of communication between two hubs is an aggregate and it may not change very much with the change in user demands. Since a group of users would be aggregated to form a node and these aggregate nodes will be connected to the hubs, the location of hubs must be based on the current level of nodes and future additions. Based on the switch capacities available a cap on the number of nodes assigned to any hub may be placed. A factor affecting the decision to aggregate users to form an aggregate node will be the dispersion

of the users. The number of users aggregated will also be limited by the size of the aggregate node.

The choices for the routing strategy will be (a) direct, (b) via the hubs to which the nodes are assigned, (c) via one of the hubs to which the nodes are assigned, or (d) via other intermediate hubs. If the flow between two aggregate nodes is expected to be heavy one may consider linking them direct. Direct links may be unswitched. The choice of strategies *b* or *c* over strategy *a* will likely result in lower number of total links in the network. Strategy *d* may even result in lower number of total links than with the choice of either *b* or *c*. Though there are savings in the number of links the communication path may be longer if strategy *d* is employed. This may cause longer propagation/response time and attenuation of signals. Therefore repeaters may be required to maintain the signal strength, which will add to the cost. Another potential drawback of strategy *d* would be, in the event of failure of a heavily used intermediate hub node a larger number of communications paths may be rendered unavailable. Available capacity of the transmission media will also have an effect on the choice of routing strategies. The choice of the transmission media will depend on the distance over which it is to be used. If very high capacity media is considered for use (broadband) the strategy of routing all flows via hubs may be appropriate, because this strategy promotes aggregation of flows. However, this strategy would require the use of multiplexing and switching equipment at the hubs and at the aggregate nodes. In order to provide reasonable performance the maximum number of hops may be limited. To make a dedicated network

fault tolerant backup routes must be provided. This will mean providing alternate path for communications flow in the event of failure of a link or a hub node.

REFERENCES

REFERENCES

- [1] Ahuja, Vijay, "Design and Analysis of Computer Communications Networks", McGraw Hill, 1982, Chapter 4
- [2] Anderberg, Michael R., "Cluster Analysis For Applications", Academic Press, 1973
- [3] Andrew, G., T. Hoffman, and C. Krabek, "On the Generalized Set Covering Problem", Control Data Corporation, 1968
- [4] AT&T Bell Laboratories, "Engineering and Operations In The Bell System", 1984
- [5] Charalambous, Christakis, "Acceleration of the HAP Approach for the Multifacility Location Problem", Naval Research Logistics Quarterly, Vol. 32, 1985, pp. 373-389
- [6] Chen, Reuven, "Solution to Minsum and Minimax Location-Allocation Problems with Euclidean Distances", Naval Research Logistics Quarterly, Vol. 30, 1983, pp. 449-459
- [7] Cooper, Leon, "Heuristic Methods for Location Allocation Problems", Siam Review, Vol. 6, No. 1, January 1968, pp. 37-53
- [8] Dysart, Hugh G., and Nicolaos D. Georganas, "NEWCLUST: An Algorithm for the Topological Design of Two-Level, Multidrop Teleprocessing Networks", IEEE Transactions on Communications, Vol. COM-26, No. 1, January 1978, pp. 55-62
- [9] Francis, Richard L., and John A. White, "Facility Layout and Location", Prentice-Hall Inc., 1974
- [10] Garey, Michael R., and David S. Johnson, "Computers and Intractability - A Guide to the Theory of NP-Completeness", W.H. Freeman & Company 1979
- [11] Garfinkel, R.S., and M. Segal, "A Single Period Planning Model With Various Circuit Routing Strategies", AT&T Technical Memorandum 59513-850809-01TM, 1985

- [12] Garfinkel, R.S., and D.R. Smith, "Optimal Hubbing in a Communications Network - A Two Hub Model With Fixed Arc Costs", Working Paper, University of Connecticut 1987
- [13] Gavish, B., "On Obtaining the Best Multipliers For a Lagrangian Relaxation for Integer Programming", Computers and Operations Research, Vol. 5, 1978, pp 55-71
- [14] Gavish, B., "Topological Design of Centralized Computer Networks - Formulation and Algorithms", Networks, Vol. 12. 1982, pp 355-377
- [15] Geoffrion, A.M., "Lagrangian Relaxation and Its Uses in Integer Programming", Mathematical Programming Study 2, 1974, pp 82-114
- [16] Geoffrion, A.M., and G.W. Graves, "Multi Commodity Distribution System Design by Benders Decomposition", Management Science, Vol. 20, 1974, pp. 822-844
- [17] Glover, Fred S., and Eugene Woolsey, "Converting the 0-1 Polynomial Programming Problem to a 0-1 Linear Program", Operations Research, Vol. 22, 1974, pp. 180-1842
- [18] Gomory, R. E., and T. C. Hu, "Synthesis of a Communications Network", Journal of the Society of Industrial Applied Mathematics, Vol. 12, No. 2, June 1984, pp. 348-369
- [19] Hakimi, S. L., "Optimum Distribution of Switching Centers In a Communications Network and Some Related Graph Theoretic Problems", Operations Research, Vol. 13, 1965, pp. 462-475
- [20] Held, M. H., P. Wolfe and H. D. Crowder, "Validation of subgradient Optimization", Mathematical Programming, Vol. 6, 1974, pp. 62-88
- [21] Helme, Marcia P., and Thomas L. Magnanti, "Designing Satellite Communication Networks by Zero-One Quadratic Programming", Networks, Vol. 19, 1989, pp. 427-450
- [22] Kennington, Jeff L., and Richard V. Helgason, "Algorithms For Networks Programming" John Wiley & Sons, 1980, p 91

- [23] Lawler, E.L., "Combinatorial Optimization: Networks and Matroids", Holt, Rinehart and Winston, 1976
- [24] Mirzaian, Andranik, "Lagrangian Relaxation for the Star-Star Concentrator Location Problem: Approximation Algorithm and Bounds", Networks, Vol. 15, 1985, pp. 1-20
- [25] O'Kelly, Morton E., "The Location of Interacting Hub Facilities", Transportation Science, Vol. 20, No. 2, May 1986, pp. 92 - 106
- [26] O'Kelly, Morton E., "Activity Levels At Hub Facilities In Interacting Networks", Geographical Analysis, Vol. 18, No. 4, Oct. 1986, pp. 343 - 356
- [27] O'Kelly, Morton E., "A Quadratic Integer Program For The Location of Interacting Hub Facilities", European Journal of Operations Research, Vol. 32, 1987, pp. 393 - 404
- [28] Parker, Gary R., Ronald L. Rardin, "Discrete Optimization", Academic Press, 1988
- [29] Roberts, Fred S., "Applied Combinatorics", Prentice-Hall, 1984, p43
- [30] Ross, G. Terry, and Richard M. Soland, "Modelling Facility Location Problems as Generalized Assignment Problems", Management Science, Vol. 24, No. 3, November 1977, pp. 345-357
- [31] Taha, Hamdy, "Integer Programming", Academic Press, 1975, Chapter 3
- [32] Teitz, Michael B., and Polly Bart, "Heuristic Method for Estimating the Generalized Vertex Median of a Weighted Graph", Operations Research, Vol. 15, 1967, pp. 955 - 961

APPENDICES

Example illustrating transformation from a 3DM problem to P1

For illustrative purposes let us consider a 3DM problem with 2 items in each group. The groups are labeled 1, 2, and 3, and the items in each group are labeled $\{i, I\}$, $\{j, J\}$, and $\{k, K\}$, respectively. The feasible marriages between the three groups will be:

$$(ijk), (ijK), (iJk), (iJK), (Ijk), (IJk), (IJK).$$

We transform this 3DM problem to an instance of P1 with a total of six partitions corresponding to the six items $\{i, I, j, J, k, K\}$ with each partition containing four nodes. The four nodes in each partition are for example constructed as follows. Let us consider the partition labeled i . The nodes in this partition will be:

$$\{<i: j, k>; <i: j, K>; <i: J, k>; <i: J, K>\}.$$

Each node in a partition corresponds to a feasible marriage between items in the sets 1, 2, and 3. The nodes in each of the six partitions are given below.

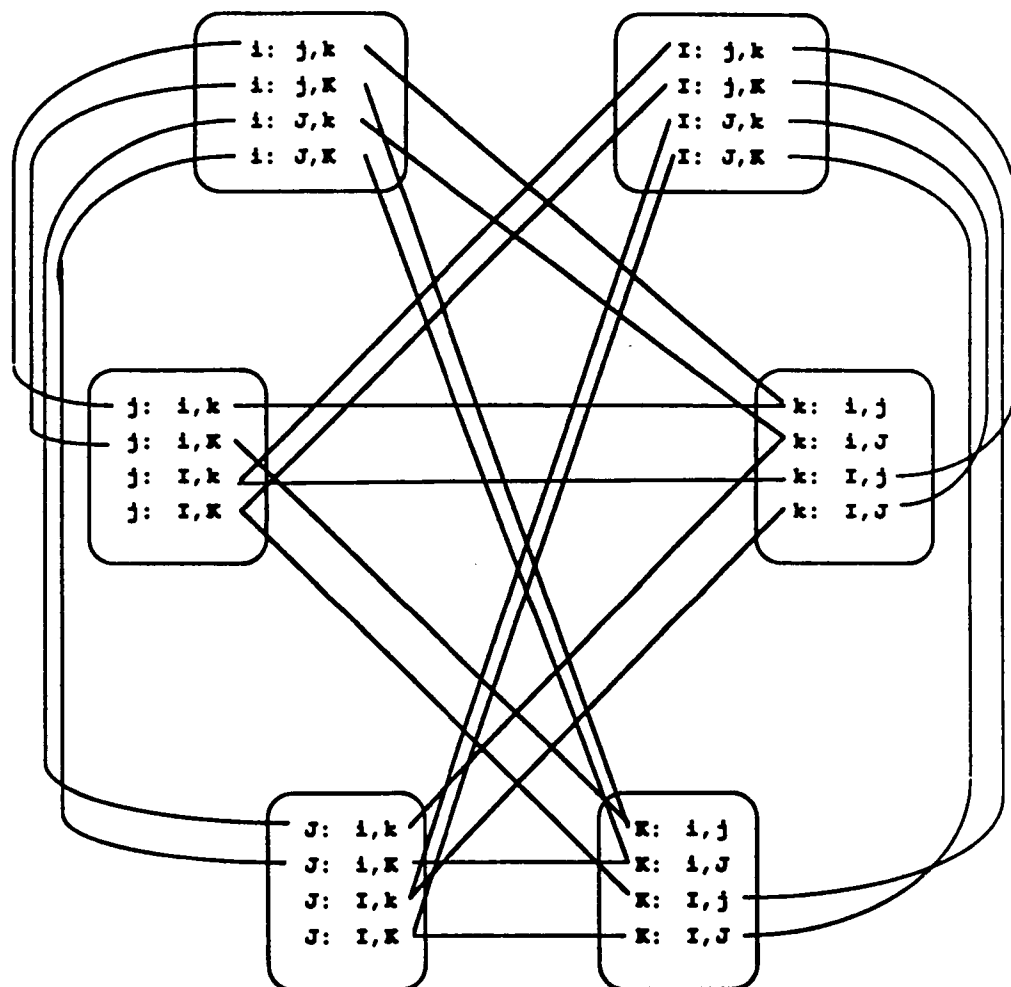
Partition label	Node list
i	$\{<i: j, k>; <i: j, K>; <i: J, k>; <i: J, K>\}$
I	$\{<I: j, k>; <I: j, K>; <I: J, k>; <I: J, K>\}$
j	$\{<j: i, k>; <j: i, K>; <j: I, k>; <j: I, K>\}$
J	$\{<J: i, k>; <J: i, K>; <J: I, k>; <J: I, K>\}$
k	$\{<k: i, j>; <k: i, J>; <k: I, j>; <k: I, J>\}$
K	$\{<K: i, j>; <K: i, J>; <K: I, j>; <K: I, J>\}$

Inter-partition arcs between nodes not corresponding to a feasible marriage have a cost of M (M very large value). For example, cost of potential inter-partition arc between nodes $k: i, j$ and $K: i, j$ is M . Inter-partition arcs between nodes corresponding to a feasible marriage have a cost of $(A - C_{ijk})/3$, where A is smallest integer greater than maximum (C_{ijk}) , and C_{ijk} is the cost of the feasible marriage. For example, cost of potential inter-partition arc

between nodes $k: i, j$ and $j: i, k$ which correspond to the feasible marriage between i, j , and k is $(A - C_{ijk})/3$.

The figure below illustrates the partitions and the nodes in each partition for the example. The inter-partition arcs shown connect nodes corresponding to feasible marriages, and have cost as defined above. All other inter-partition arcs have a cost of M .

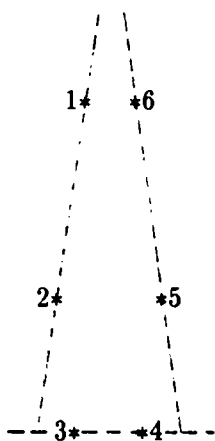
The value of the optimal solution to this instance of P1 will be $(2A - \nu(3DM) + 9M)$, and the feasible matches between which inter-partition arcs exist will be the optimal matches in the 3DM problem.



Counter example for non-unimodularity of LP1

The following is a counter example to show that the LP solution to ILP1 may not always be integer.

This is a 6 node 3 partition example where the points are located on the face of a triangle as follows.

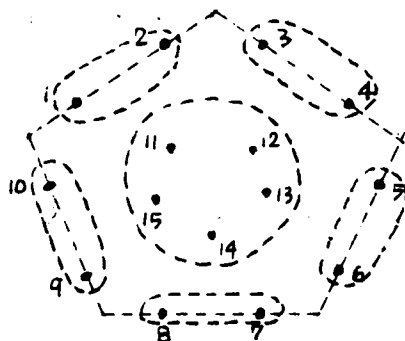


partition I: 1,2

partition II: 3,4

partition III: 5,6

The three partitions have a symmetric pattern with respect to the nodes. Symmetric patterns would be present when the nodes are located on the sides of a square, and also as given below.



Distance between nodes:

	1	2	3	4	5	6
1	-	1	1.32	1.53	1.32	0.5
2	1	-	0.5	1.32	1.53	1.32
3	1.32	0.5	-	1	1.32	1.53
4	1.53	1.32	1	-	0.5	1.32
5	1.32	1.53	1.32	0.5	-	1.
6	0.5	1.32	1.53	1.32	1.	-

Demand between points:

	1	2	3	4	5	6
1	-	1	.5	.5	.5	.5
2	1	-	.5	.5	.5	.5
3	.5	.5	-	1	.5	.5
4	.5	.5	1	-	.5	.5
5	.5	.5	.5	.5	-	1
6	.5	.5	.5	.5	1	-

Capacity of arc = 1

cost factor = 1

Formulation:

$$\begin{aligned} \text{MIN} \quad & X1 + X2 + X3 + X4 + X5 + X6 + 1.32 X13 + 1.53 X14 + 1.32 X15 \\ & + .5 X16 + .5 X23 + 1.32 X24 + 1.53 X25 + 1.32 X26 + 1.32 X35 \\ & + 1.53 X36 + .5 X45 + 1.32 X46 \end{aligned}$$

SUBJECT TO

- 2) $X1 + X2 = 1$
- 3) $X3 + X4 = 1$
- 4) $X5 + X6 = 1$
- 5) $-X1 + X13 + X14 = 0$
- 6) $-X1 + X15 + X16 = 0$
- 7) $-X2 + X23 + X24 = 0$
- 8) $-X2 + X25 + X26 = 0$
- 9) $-X3 + X13 + X23 = 0$
- 10) $-X3 + X35 + X36 = 0$
- 11) $-X4 + X14 + X24 = 0$
- 12) $-X4 + X45 + X46 = 0$
- 13) $-X5 + X15 + X25 = 0$
- 14) $-X5 + X35 + X45 = 0$
- 15) $-X6 + X16 + X26 = 0$
- 16) $-X6 + X36 + X46 = 0$

LP OPTIMUM

OBJECTIVE FUNCTION VALUE = 6.0450

VARIABLE	VALUE
X1	.500000
X2	.500000
X3	.500000
X4	.500000
X5	.500000
X6	.500000

X13	.000000
X14	.500000
X15	.000000
X16	.500000
X23	.500000
X24	.000000
X25	.500000
X26	.000000
X35	.000000
X36	.500000
X45	.500000
X46	.000000

As seen in the above example, the LP solution may not be always integer.

Example of Lagrangian Approach

Example problem data:

Number of nodes = 6

Access arc capacity = 12

Trunk arc capacity = 24

Ratio of cost per channel (access/trunk) links = 0.8

Flow between nodes:

Number of arcs in the access link:

	1	2	3	4	5	6	Node	Arcs
1	-	2	0	4	2	1	1	11
2	2	-	4	0	1	1	2	8
3	0	4	-	2	1	1	3	8
4	4	0	2	-	1	3	4	10
5	2	1	1	1	-	1	5	6
6	1	1	1	4	1	-	6	7

Distance between nodes:

	1	2	3	4	5	6
1	-	6.00	1.00	5.10	4.12	6.71
2	6.00	-	6.08	1.41	8.06	6.71
3	1.00	6.08	-	5.39	3.16	5.83
4	5.10	1.41	5.39	-	7.81	7.28
5	4.12	8.06	3.16	7.81	-	4.47
6	6.71	6.71	5.83	7.28	4.47	-

Nodes in partition 1: 1, 3

Nodes in partition 2: 5, 6

Nodes in partition 3: 2, 4

Coordinates of nodes:

	<u>X</u>	<u>Y</u>
1	7	9
2	7	3
3	6	9
4	8	4
5	3	10
6	1	6

Upperbound on solution = 43.1616

The lagragian formulation of the problem:

Minimize

$$\begin{aligned}
 & 1.2 x_1 + 1.692 x_2 + 1.2 x_3 + 1.692 x_4 + 5.364 x_5 + 5.364 x_6 + 11.52 x_{12} + 9.792 x_{14} + 7.91 \\
 & x_{15} + 12.883 x_{16} + 11.674 x_{23} + 15.475 x_{25} + 12.883 x_{26} + 10.349 x_{34} + 6.067 x_{35} + 11.194 \\
 & x_{36} + 14.995 x_{45} + 13.978 x_{46} + \\
 & \lambda_{12} (x_1 - x_{15} - x_{16}) + \lambda_{13} (x_1 - x_{12} - x_{14}) + \lambda_{32} (x_3 - x_{35} - x_{36}) + \lambda_{33} (x_3 - x_{23} - x_{34}) + \\
 & \lambda_{51} (x_5 - x_{15} - x_{35}) + \lambda_{53} (x_5 - x_{25} - x_{45}) + \lambda_{61} (x_6 - x_{16} - x_{36}) + \lambda_{63} (x_6 - x_{26} - x_{46}) + \\
 & \lambda_{21} (x_2 - x_{12} - x_{23}) + \lambda_{22} (x_2 - x_{25} - x_{26}) + \lambda_{41} (x_4 - x_{14} - x_{34}) + \lambda_{33} (x_4 - x_{45} - x_{46})
 \end{aligned}$$

subject to

$$x_1 + x_3 = 1$$

$$x_5 + x_6 = 1$$

$$x_2 + x_4 = 1$$

$$x_i, x_{ij} \in \{0,1\}$$

Let $\lambda = (\lambda_{12}, \lambda_{13}, \lambda_{32}, \lambda_{33}, \lambda_{61}, \lambda_{63}, \lambda_{21}, \lambda_{22}, \lambda_{42}, \lambda_{41})$ be the vector of lagrange multipliers.

Iteration 0:

$$\lambda^0 = 0$$

$$\text{Solution: } x_1 = 1, x_5 = 1, x_2 = 1$$

$$\text{Objective value } \nu(\lambda^0) = 1.2 + 1.692 + 5.364 = 8.256$$

$$\|Ax^0 - b\|^2 = 1+1+0+0+1+1+0+0+1+1+0+0 = 6$$

$$t^0 = \frac{2 \times (43.1616 - 8.256)}{6} = 11.6352$$

Iteration 1:

$$\lambda_{12}^1 = 0 + 11.6352 \times 1 = 11.6352$$

$$\lambda_{13}^1 = 0 + 11.6352 \times 1 = 11.6352$$

$$\lambda_{32}^1 = 0 + 11.6352 \times 0 = 0$$

$$\lambda_{33}^1 = 0 + 11.6352 \times 0 = 0$$

$$\lambda_{51}^1 = 0 + 11.6352 \times 1 = 11.6352$$

$$\lambda_{53}^1 = 0 + 11.6352 \times 1 = 11.6352$$

$$\lambda_{61}^1 = 0 + 11.6352 \times 0 = 0$$

$$\lambda_{63}^1 = 0 + 11.6352 \times 0 = 0$$

$$\lambda_{21}^1 = 0 + 11.6352 \times 1 = 11.6352$$

$$\lambda_{22}^1 = 0 + 11.6352 \times 1 = 11.6352$$

$$\lambda_{41}^1 = 0 + 11.6352 \times 0 = 0$$

$$\lambda_{42}^1 = 0 + 11.6352 \times 0 = 0$$

Variable	Coefficient of variable	Value of variable in solution
x_1	$1.2 + 11.6352 \times 2 = 24.4704$	0
x_3	1.2	1
x_5	$5.364 + 11.6352 \times 2 = 28.6344$	0
x_6	5.364	1
x_2	$1.692 + 11.6352 \times 2 = 24.9624$	0
x_4	1.692	1
x_{12}	$11.52 - 11.6352 \times 2 = -11.7504$	1
x_{14}	$9.792 - 11.6352 = -1.8432$	1
x_{15}	$7.91 - 11.6352 \times 2 = -15.3604$	1
x_{16}	$12.883 - 11.6352 = 1.2478$	0
x_{23}	$11.674 - 11.6352 = 0.0388$	0
x_{25}	$15.475 - 11.6352 \times 2 = -7.7954$	1
x_{26}	$12.883 - 11.6352 = 1.2478$	0
x_{34}	10.349	0
x_{35}	$6.0670 - 11.6352 = -5.5682$	1
x_{36}	11.194	0
x_{45}	$14.995 - 11.6352 = 3.3598$	0
x_{46}	13.978	0

Objective value $\nu(\lambda^1) = -34.0616$

$$\|Ax^1 - b\|^2 = 14$$

$$t^1 = \frac{2 \times (43.1616 + 34.0616)}{15} = 10.2964$$

Iteration 2:

$$\lambda_{12}^1 = 11.6352 + 10.2964 \times -1 = 1.3388$$

$$\lambda_{13}^1 = 11.6352 + 10.2964 \times -2 = -8.9568$$

$$\lambda_{32}^1 = 0 + 10.2964 \times 0 = 0$$

$$\lambda_{33}^1 = 0 + 10.2964 \times 1 = 10.2964$$

$$\lambda_{51}^1 = 11.6352 + 10.2964 \times -2 = -8.9568$$

$$\lambda_{53}^1 = 11.6352 + 10.2964 \times 0 = 11.6352$$

$$\lambda_{61}^1 = 0 + 10.2964 \times 1 = 10.2964$$

$$\lambda_{63}^1 = 0 + 10.2964 \times 1 = 10.2964$$

$$\lambda_{21}^1 = 11.6352 + 10.2964 \times -1 = 1.3388$$

$$\lambda_{22}^1 = 11.6352 + 10.2964 \times -1 = 1.3388$$

$$\lambda_{41}^1 = 0 + 10.2964 \times 0 = 0$$

$$\lambda_{42}^1 = 0 + 10.2964 \times 1 = 10.2964$$

Variable	Coefficient of variable	Value of variable in solution
x_1	-6.4180	1
x_3	12.4964	0
x_5	8.0424	1
x_6	10.2964	0
x_2	4.3696	1
x_4	11.9884	0
x_{12}	19.138	0
x_{14}	18.7488	0
x_{15}	15.528	0
x_{16}	1.2478	0
x_{23}	0.0388	0
x_{25}	2.501	0
x_{26}	1.2478	0
x_{34}	0.0526	0
x_{35}	15.0238	0
x_{36}	0.8976	0
x_{45}	- 6.9366	1
x_{46}	- 6.6148	1

Objective value $\nu(\lambda^2) = -7.5574$

$$\|Ax^2 - b\|^2 = 10$$

$$t^2 = \frac{2 \times (43.1616 + 7.5574)}{10} = 10.1438$$

This procedure is continued until the maximum absolute difference in the lagrange multipliers is below .0001, and the solutions for two successive iterations are identical.

Example of application of heuristic HSP1

Example problem data:

Number of nodes = 6

Access arc capacity = 12

Trunk arc capacity = 24

Ratio of cost per channel (access/trunk) links = 0.8

Flow between nodes:

Number of arcs in the access link:

	1	2	3	4	5	6	<u>Node</u>	<u>Arcs</u>
1	-	0	6	5	3	4	1	2
2	0	-	6	6	1	3	2	2
3	6	6	-	4	6	0	3	2
4	5	6	4	-	7	7	4	3
5	3	1	6	7	-	6	5	2
6	4	3	0	7	6	-	6	2

Distance between nodes:

	1	2	3	4	5	6
1	-	7.28	3.00	4.12	7.21	3.61
2	7.28	-	4.47	3.16	6.08	7.07
3	3.00	4.47	-	1.41	5.00	3.16
4	4.12	3.16	1.41	-	5.39	4.47
5	7.21	6.08	5.00	5.39	-	4.12
6	3.61	7.07	3.16	4.47	4.12	-

Nodes in partition 1: 2, 4, 5

Nodes in partition 2: 1, 3, 6

Number of arcs between partitions = 2

Coordinates of nodes:

	X	Y
1	9	6
2	2	4
3	6	6
4	5	5
5	3	10
6	7	9

$$X_1^{(0)} = \frac{2*2 + 3*5 + 2*3}{7} = 3.571 \quad Y_1^{(0)} = \frac{2*4 + 3*5 + 2*10}{7} = 6.143$$

$$X_2^{(0)} = \frac{2*9 + 2*6 + 2*7}{6} = 7.333 \quad Y_2^{(0)} = \frac{2*6 + 2*6 + 2*9}{6} = 7.000$$

$$E_1 = [(7.333 - 9.000)^2 + (7.000 - 6.000)^2 + 0.001]^{1/2} = 1.944$$

$$E_2 = [(3.571 - 2.000)^2 + (6.143 - 4.000)^2 + 0.001]^{1/2} = 2.657$$

$$E_3 = [(7.333 - 6.000)^2 + (7.000 - 6.000)^2 + 0.001]^{1/2} = 1.667$$

$$E_4 = [(3.571 - 5.000)^2 + (6.143 - 5.000)^2 + 0.001]^{1/2} = 1.830$$

$$E_5 = [(3.571 - 3.000)^2 + (6.143 - 10.000)^2 + 0.001]^{1/2} = 3.899$$

$$E_6 = [(7.333 - 7.000)^2 + (7.000 - 9.000)^2 + 0.001]^{1/2} = 2.028$$

$$D_{12} = [(3.571 - 7.333)^2 + (6.143 - 7.000)^2 + 0.001]^{1/2} = 3.859$$

$$X_1^{(1)} = \frac{(2*1*2/2.657) + (3*1*5/1.830) + (2*1*3/3.899) + (2*0.8*7.333/3.859)}{(2*1/2.657) + (3*1/1.830) + (2*1/3.899) + (2*0.8/3.859)} = 4.1253$$

$$Y_1^{(1)} = \frac{(2*1*4/2.657) + (3*1*5/1.830) + (2*1*10/3.899) + (2*0.8*7.000/3.859)}{(2*1/2.657) + (3*1/1.830) + (2*1/3.899) + (2*0.8/3.859)} = 5.6578$$

$$X_2^{(1)} = \frac{(2*1*9/1.944) + (2*1*6/1.667) + (2*1*7/2.028) + (2*0.8*3.571/3.859)}{(2*1/1.944) + (2*1/1.667) + (2*1/2.028) + (2*0.8/3.859)} = 6.8446$$

$$Y_2^{(1)} = \frac{(2*1*6/1.944) + (2*1*6/1.667) + (2*1*9/2.028) + (2*0.8*6.143/3.859)}{(2*1/1.944) + (2*1/1.667) + (2*1/2.028) + (2*0.8/3.859)} = 6.8315$$

Distance between nodes and approximate hub locations:

Partition 1: 2 - 2.6954
 4 - 1.0944
 5 - 4.4856

Partition 2: 1 - 2.3102
 3 - 1.1852
 6 - 2.4761

Among all nodes the closest to its approximate hub location is node 4. Select node 4 as the hub location for partition 1.

$$X_1 = 5.000 \quad Y_1 = 5.000$$

$$E_1 = [(6.8446 - 9.000)^2 + (6.8315 - 6.000)^2 + 0.001]^{1/2} = 2.3104$$

$$E_3 = [(6.8446 - 6.000)^2 + (6.8315 - 6.000)^2 + 0.001]^{1/2} = 1.1856$$

$$E_6 = [(6.8446 - 7.000)^2 + (6.8315 - 9.000)^2 + 0.001]^{1/2} = 2.1740$$

$$D_{12} = [(5.000 - 6.8446)^2 + (5.000 - 6.8315)^2 + 0.001]^{1/2} = 2.5996$$

$$X_2^{(1)} = \frac{(2*1*9/2.3104) + (2*1*6/1.1856) + (2*1*7/2.1740) + (2*0.8*5.000/2.5996)}{(2*1/2.3104) + (2*1/1.1856) + (2*1/2.1740) + (2*0.8/2.5996)} = 6.7097$$

$$Y_2^{(1)} = \frac{(2*1*6/2.3104) + (2*1*6/1.1856) + (2*1*9/2.1740) + (2*0.8*5.000/2.5996)}{(2*1/2.3104) + (2*1/1.1856) + (2*1/2.1740) + (2*0.8/2.5996)} = 6.5246$$

Distance between nodes and approximate hub locations:

Partition 2: 1 - 2.3497
 3 - 0.8825
 6 - 2.4924

Closest node is 3. Choose node 3 as the hub location for partition 2.

Partition 1 hub location is node 4.

Partition 2 hub location is node 3.

Example of application of heuristic HSP2

Example problem data:

Number of nodes = 6

Access arc capacity = 12

Trunk arc capacity = 24

Ratio of cost per channel (access/trunk) links = 0.8

Flow between nodes:

Number of arcs in the access link:

	1	2	3	4	5	6	<u>Node</u>	<u>Arcs</u>
1	-	0	6	5	3	4	1	2
2	0	-	6	6	1	3	2	2
3	6	6	-	4	6	0	3	2
4	5	6	4	-	7	7	4	3
5	3	1	6	7	-	6	5	2
6	4	3	0	7	6	-	6	2

Distance between nodes:

	1	2	3	4	5	6
1	-	7.28	3.00	4.12	7.21	3.61
2	7.28	-	4.47	3.16	6.08	7.07
3	3.00	4.47	-	1.41	5.00	3.16
4	4.12	3.16	1.41	-	5.39	4.47
5	7.21	6.08	5.00	5.39	-	4.12
6	3.61	7.07	3.16	4.47	4.12	-

Nodes in partition 1: 2, 4, 5

Nodes in partition 2: 1, 3, 6

Number of arcs between partitions = 2

Coordinates of nodes:

	X	Y
1	9	6
2	2	4
3	6	6
4	5	5
5	3	10
6	7	9

Initial hub locations:

Partition 1:

$$\begin{aligned}\text{node 2: } & 3*3.16 + 2*6.08 = 21.64 \\ \text{node 4: } & 2*3.16 + 2*5.36 = 17.10 \\ \text{node 5: } & 2*6.08 + 3*5.36 = 28.33\end{aligned}$$

Since node 4 gives the minimum select this as the initial hub location for partition 1.

Partition 2:

$$\begin{aligned}\text{node 1: } & 2*3.00 + 2*3.61 = 13.22 \\ \text{node 3: } & 2*3.00 + 2*3.16 = 12.32 \\ \text{node 6: } & 2*3.61 + 2*3.16 = 13.54\end{aligned}$$

Since node 3 gives the minimum select this as the initial hub location for partition 2.

Total cost for this initial hub locational configuration = 406.46.

Iterative improvement:

Redetermine the hub locations considering both intra- and inter-partition flows.

Partition 1:

$$\begin{aligned}\text{node 2: } & 3*3.16*1*12 + 2*6.08*1*12 + 2*4.47*0.8*24 = 431.33 \\ \text{node 4: } & 2*3.16*1*12 + 2*5.36*1*12 + 2*1.41*0.8*24 = 258.62 \\ \text{node 5: } & 2*6.08*1*12 + 3*5.36*1*12 + 2*5.00*0.8*24 = 530.88\end{aligned}$$

Since node 4 gives the lowest cost select this as the revised hub location for partition 1.

Partition 2:

$$\text{node 1: } 2 \times 3.00 \times 1 \times 12 + 2 \times 3.61 \times 1 \times 12 + 2 \times 4.12 \times 0.8 \times 24 = 316.85$$

$$\text{node 3: } 2 \times 3.00 \times 1 \times 12 + 2 \times 3.16 \times 1 \times 12 + 2 \times 1.41 \times 0.8 \times 24 = 201.98$$

$$\text{node 6: } 2 \times 3.61 \times 1 \times 12 + 2 \times 3.16 \times 1 \times 12 + 2 \times 4.47 \times 0.8 \times 24 = 334.13$$

Since node 3 gives the lowest cost select this as the revised hub location for partition 2.

Total cost for this revised hub locational configuration = 406.46.

Since new configuration cost = old configuration cost stop.

Partition 1 hub location is node 4.

Partition 2 hub location is node 3.

Limitation of commercial LP codes for solving P1

Any commercial LP code has a limitation in the number of variables and constraints it can handle. Typical sizes would be 60,000 variables and 20,000 equations.

For a n-node problem the maximum number of variables will be generated when the number of partitions equals $(n/2)$ i.e., each partition contains 2 or 3 nodes depending on whether n is odd or even.

For a n-node problem the maximum number of variables will be,

$$n + (n/2) \frac{((n/2) - 1)}{2},$$

and the maximum number of constraints will be,

$$(n/2) + n ((n/2) - 1).$$

Therefore,

$$\begin{aligned} n + (n/2) \frac{((n/2) - 1)}{2} &\leq 60,000 \quad \text{and} \\ (n/2) + n ((n/2) - 1) &\leq 20,000. \end{aligned}$$

The maximum number of nodes satisfying the above inequalities equals 200. Hence the commercial codes could only handle problem P1 with about 200 nodes.

Description of the min-sum multicenter (p-median) problem

Let $N = (V, A)$ be the underlying transportation network for distributing a commodity from facilities to demand points, where V is the node set and A is the arc set. Demands occur only on the node set $V = \{v_1, v_2, \dots, v_n\}$. Let $c_k(x_j, v_i)$ be the unit cost of transporting commodity k from point x_j (for example where a facility may be located) to node v_i (where a demand may occur). We have several facilities at the set of points $X_p = \{x_1, x_2, \dots, x_p\}$, and a demand v_i could be serviced by any of these facilities. When the capacities of the facilities are unlimited each demand will be completely serviced by the facility closest to the demand point. The objective is to determine the locations X_p which minimize the total transportation costs:

$$TC(X_p) = \sum_{i=1}^m w_i \min \{ c_k(x_j, v_i) : x_j \in X_p \},$$

where w_i is the demand at node i .

Example of the pair wise interchange heuristic

Group capacity = 25 circuits

$$c^1 = 0.70$$

$$c^2 = 0.80$$

$$c^3 = 2.00$$

Demand Matrix

	1	2	3	4	5	Total	Z_i
1	-	10	15	20	15	60	3
2	10	-	18	14	8	50	2
3	15	18	-	9	6	48	2
4	20	14	9	-	17	60	3
5	15	8	6	17	-	46	2

Distance Matrix

	1	2	3	4	5
1	-	36	25	36	23
2	36	-	28	32	16
3	25	28	-	12	18
4	36	32	12	-	17
5	23	16	18	17	-

FIRST ITERATION:

Initial Hub Locations = {3,5}

Node Assignments: $H_3 = \{3,4\}$

$H_5 = \{1,2,5\}$

Total cost = $(2*12 + 2*16 + 3*23 + 4*18).0.80 + 22*2 + 11*0.70 = 218.90$

First replacement: Select node 1 to replace 3 and 5 sequentially.

First replace 3 by 1.

New hub set = {1,5}

Node assignments: $H_1 = \{1\}$

$H_5 = \{2,3,4,5\}$

nodes 3,1,4 are reassigned.

Total cost = $(3*23 + 2*18 + 3*17 + 2*16).0.80 + 20*2 + 10*0.70 = 197.40$

Replace 5 by 1.

Hub set = {1,3}

Node assignments: $H_1 = \{1\}$

$H_3 = \{2,3,4,5\}$

Total cost = $(3*25 + 2*28 + 3*12 + 2*18).0.80 + 20*2 + 10*0.70 = 209.40$

Replace 3 by 1 because this configuration yields the lowest total cost.

First iteration - second replacement:

Initial hub set = {1,5}

Select 2 to replace 1 and 5, one by one.

First replace 1 by 2.

Hub set = {2,5}

Node assignments: $H_2 = \{2\}$

$H_5 = \{1,3,4,5\}$

$$\text{Total cost} = (3*23 + 2*18 + 3*17 + 2*16).0.80 + 20*2 + 10*0.70 = 197.40$$

Replace 5 by 2.

Hub set = {1,2}

Node assignments: $H_1 = \{1,3\}$

$H_2 = \{2,4,5\}$

$$\text{Total cost} = (2*25 + 4*36 + 2*16 + 2*32).0.80 + 20*2 + 10*0.70 = 279.00$$

2 is not a good location.

First iteration - third replacement:

Initial hub set = {1,5}

Select 4 to replace 1 and 5, one by one.

First replace 1 by 4.

Hub set = {4,5}

Node assignments: $H_4 = \{3,4\}$

$H_5 = \{1,2,5\}$

$$\text{Total cost} = (3*23 + 2*16 + 4*17 + 2*12).0.80 + 22*2 + 11*0.70 = 206.10$$

Replace 5 by 4.

Hub set = {1,4}

Node assignments: $H_1 = \{1\}$

$H_4 = \{2,3,4,5\}$

$$\text{Total cost} = (3*36 + 2*32 + 2*12 + 2*17).0.80 + 18*2 + 9*0.70 = 226.30$$

4 is not a good location as replacement.

SECOND ITERATION:

Initial hub set = {1,5}

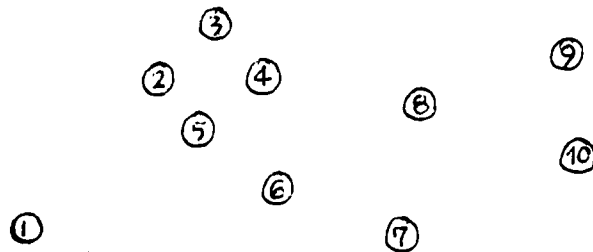
Consider 2,3, and 4 one by one for replacing 1 and 5 sequentially and repeat the procedure.

When a complete iteration is executed without change in the configuration STOP.

An example of the initial hub site selection algorithm

Number of nodes = 10

Capacity of a group $Q = 20$ circuit



Flows between nodes:

	1	2	3	4	5	6	7	8	9	10
1	-	9	16	5	6	10	5	0	12	7
2	9	-	8	0	9	1	12	8	8	6
3	16	8	-	12	8	13	2	7	7	9
4	5	0	12	-	6	9	2	1	18	5
5	6	9	8	6	-	14	9	11	7	2
6	10	1	13	9	14	-	4	10	6	13
7	5	12	2	2	9	4	-	11	3	9
8	0	8	7	1	11	10	11	-	1	12
9	12	8	7	18	7	6	3	1	-	13
10	7	6	9	5	2	13	9	12	13	-

Node	1	2	3	4	5	6	7	8	9	10
Sum of flows	70	61	82	58	72	80	57	61	75	76
No. of groups	4	4	5	3	4	4	3	4	4	4

Node location	List of the node and its K-nearest neighbors	Frequency of occurrence
1	1,2,5,6	1
2	2,3,4,5	5
3	3,2,4,5	3
4	4,3,2,5	6
5	5,2,4,6	6
6	6,5,4,7	5
7	7,6,8,10	5
8	8,7,4,6	4
9	9,8,7,10	2
10	10,7,8,9	3

S_i : Ordered list of terminal locations, where i is the frequency of occurrence

$$S_1 = (1)$$

$$S_2 = (9)$$

$$S_3 = (3,10)$$

$$S_4 = (8)$$

$$S_5 = (6,2,7)$$

$$S_6 = (5,4)$$

$$KM = \text{weighted mean} + 1 = \frac{(1+2+6+4+15=12)}{10} + 1 = 5$$

$$\text{Hub site selection list} = \{5,4,6,2,7,8,3,10,9,1\}$$

If number of hubs is 3, then $H^0 = \{5,4,6\}$, and

$$M = \{2,7,8,3,10,9,1\}$$

Example of the lower bound procedure LB3

Access arc capacity = 12 circuits

Trunk arc capacity = 24 circuits

Frame capacity = 25 ports

Number of partitions = 2

c^1 = \$0.70

c^{2T} = \$0.80

c^{2A} = \$1.00

c^3 = \$2.00

c^F = \$100

Demand Matrix

	1	2	3	4	5	Total	Z_i
1	-	10	15	20	15	60	5
2	10	-	18	14	8	50	5
3	15	18	-	9	6	48	4
4	20	14	9	-	17	60	5
5	15	8	6	17	-	46	4

Distance Matrix

	1	2	3	4	5
1	-	36	25	36	23
2	36	-	28	32	16
3	25	28	-	12	18
4	36	32	12	-	17
5	23	16	18	17	-

LD: $(n - p)$ smallest distances arranged in an ascending order

(12, 16, 17)

LZ: $(n - p)$ smallest Z_i values arranged in a descending order

(5, 4, 4)

LP: $\frac{p(p-1)}{2}$ smallest distances beyond the values in LD arranged in an ascending order

(18)

LM: $(n - p)$ smallest $(Z_i + \beta_i)$ values

(8, 8, 10)

LB: p smallest β_i values

(4, 4)

$$\begin{aligned}\text{Lower Bound} &= (.7+1*12+4) * 5 + (.7+1*16+4) * 4 + (.7+1*17+4)*4 + (.7+0.8*18+4) + \\ &\quad \left(\left\lceil \frac{8}{25} \right\rceil + \left\lceil \frac{8}{25} \right\rceil + \left\lceil \frac{10}{25} \right\rceil \right) * 100 + \left[\left((5+4+4) + 2 + (4+4) \right) \frac{1}{25} \right] * 100 \\ &= 672.20\end{aligned}$$

Example for the transformation from KP1-T to P3

Let an instance of the KP1 be described as follows:

$$\max 15 y + 17 x_1 + 6 x_2 + 11 x_3 + 8 x_4$$

such that

$$20 y + 8 x_1 + 9 x_2 + 12 x_3 + 3 x_4 \leq 45$$

$$y \geq 0, \text{ integer}$$

$$x_i \in \{0,1\}$$

Define

$$y + y' = \left\lceil \frac{45}{20} \right\rceil = 3$$

$$x_1 + x'_1 = 1$$

$$x_2 + x'_2 = 1$$

$$x_3 + x'_3 = 1$$

$$x_4 + x'_4 = 1$$

Transform the instance of KP1 by substituting for the x and y variables with their complements described above.

The transformed problem will be:

$$\min 15 y' + 17 x'_1 + 6 x'_2 + 11 x'_3 + 8 x'_4 - 87$$

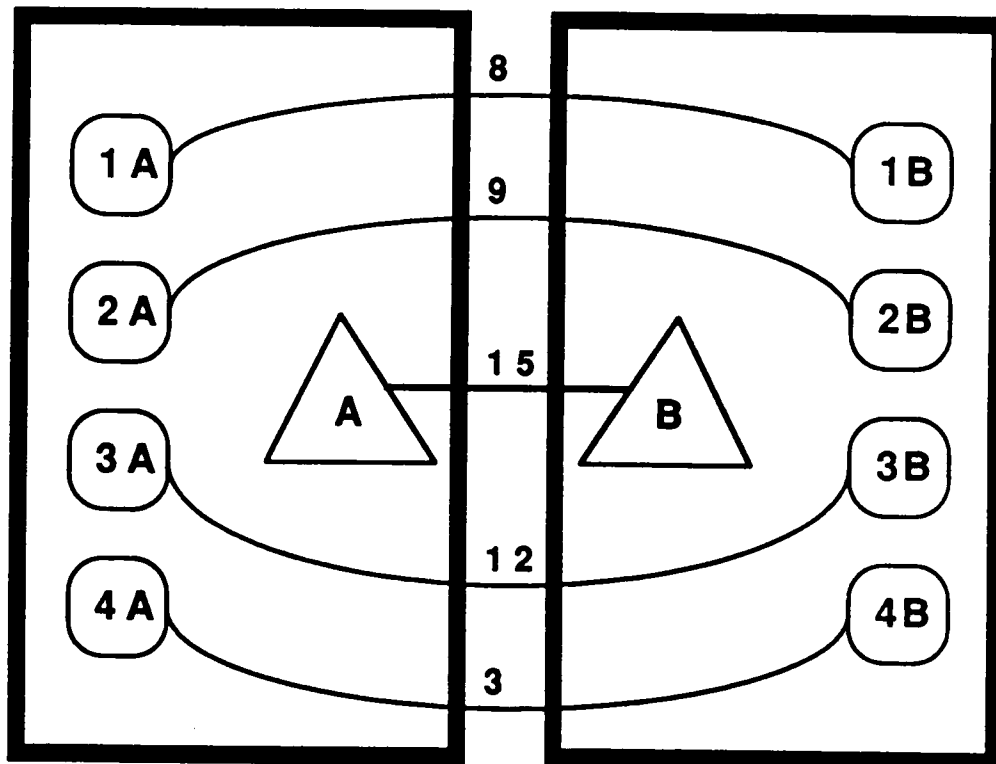
such that

$$20 y' + 8 x'_1 + 9 x'_2 + 12 x'_3 + 3 x'_4 \geq 47$$

$$y' \geq 0, \text{ integer}$$

$$x'_i \in \{0,1\}$$

This corresponds to the instance of P3 as follows:



Example of application of heuristic procedures FS1 and FS2

Example problem data:

Number of nodes = 6

Access arc capacity = 8

Direct arc capacity = 8

Trunk arc capacity = 12

Frame capacity = 20

Ratio of cost per channel (direct/access/trunk) links = 1.0/1.0/0.8

Cost per frame = 100

Flow between nodes:

	1	2	3	4	5	6
1	-	7	6	5	3	6
2	7	-	6	6	4	3
3	6	6	-	4	6	5
4	5	6	4	-	7	7
5	3	4	6	7	-	6
6	6	3	5	7	6	-

Distance between nodes:

	1	2	3	4	5	6
1	-	7.28	3.00	4.12	7.21	3.61
2	7.28	-	4.47	3.16	6.08	7.07
3	3.00	4.47	-	1.41	5.00	3.16
4	4.12	3.16	1.41	-	5.39	4.47
5	7.21	6.08	5.00	5.39	-	4.12
6	3.61	7.07	3.16	4.47	4.12	-

Nodes in partition 1: 2, 4, 5 Partition hub: 4

Nodes in partition 2: 1, 3, 6 Partition hub: 3

Coordinates of nodes:

	X	Y
1	9	6
2	2	4
3	6	6
4	5	5
5	3	10
6	7	9

The LP solution is:

(1,2), (1,6), and (5,6) are routed direct.

Link	LP solu.	Roundup	Available excess cap.
X ₃₄	2.8333	3.0000	2
X ₂₄	2.3750	3.0000	5
X ₄₅	2.5000	3.0000	4
X ₁₃	1.7500	2.0000	2
X ₃₆	1.8750	2.0000	1

$$\text{Cost of routing direct} = (c^1 + c^D * \text{distance} + 2 c^3) \cdot g + c^4 + 2 * c^F * g / \text{capacity}_f$$

Flow	Cost of direct Routing	direct cost/flow
1-2	75.94	73.94/7 = 10.849
1-6	46.58	46.58/6 = 7.763
5-6	50.66	50.66/6 = 8.443

List DL: (1-2), (5-6), (1-6).

Available excess capacity in links is not sufficient to reroute any flow via hubs.

Consider augmenting capacity in links.

$$\text{cost of augmenting capacity} = \text{distance} * \text{cost/arc} + \text{port costs} + \text{frame costs}$$

Flow	Path	Cost of augmenting capacity	Cost of direct routing	Reroute?
1-2	1-3-4-2	104.816	73.94	No
5-6	5-4-3-6	127.24	50.66	No
1-6	1-3-6	77.280	46.58	No

step 7:

current network cost = frame cost + port cost + direct arc cost + access arc cost + trunk arc cost

$$= 6*100 + 45*2 + 1*8*(7.28+4.12+3.61) + 1*8*(3+3+2+2) + 0.8*12*3*1.41$$

$$= 930.688$$

step 8:

Link	Volume of additional flow	Excess Cap. (Addl. flow - excess cap.)	New excess cap.
1-3	13	2	11
3-4	13	2	11
3-6	12	1	11
4-5	6	4	2
2-4	7	5	2

step 9:

Reroute 1-6 via hub and delete it from list.

Available excess cap.

Link Excess cap.

1-3	4
3-4	14
3-6	4
4-5	2
2-4	2

step 10:

Flow	Path	Cost of augmenting capacity	Cost of direct routing	Reroute?
1-2	1-3-4-2	80.880	73.94	No
5-6	5-4-3-6	96.400	50.66	No

No further change possible.

$$\text{New network cost} = 6*100 + 47*2 + 1*8*(7.28+4.12) + 1*8*(3+3+3+3) + 0.8*12*3*1.41$$

$$= 921.808$$

Since new network cost is lower than the current network cost accept the new configuration.

FS1 = 921.808

Procedure FS2:

Link	LP solu.	Rounddown	Over Capacity
X ₃₄	2.8333	2.0000	10
X ₂₄	2.3750	2.0000	3
X ₄₅	2.5000	2.0000	4
X ₁₃	1.7500	1.0000	6
X ₃₆	1.8750	1.0000	7

Flow	Cost of direct Routing	Flow Size	Path
1-4	50.66	5	1-3-4
1-5	75.38	3	1-3-4-5
2-3	53.46	6	2-4-3
2-5	66.34	4	2-4-5
2-6	74.26	3	2-4-3-6
3-5	57.70	6	3-4-5
4-6	53.46	7	4-3-6

Step 3:

List (VL) : (1,4), (2,3), (4,6), (3,5), (2,5), (2,6), (1,5)

Step 4:

Link	LP solu.	Rounddown	Over Capacity	Flow rerouting(over capacity)
X ₃₄	2.8333	2.0000	10	1-4(5), 2-3(-1), 4-6(-8), 3-5(-14), 1-5(-17)
X ₂₄	2.3750	2.0000	3	2-3(-3)
X ₄₅	2.5000	2.0000	4	3-5(-2) 1-5(-5)
X ₁₃	1.7500	1.0000	6	1-4(1) 1-5(-2)
X ₃₆	1.8750	1.0000	7	4-6(0)

Flows (1,4), (2,3), (4,6), (3,5), (1,5) rerouted direct.

Step 5:

Flow	Cost of direct Routing	flow size	direct cost/flow	path
1-2	75.94	7	10.849	1-3-4-2
1-4	50.66	5	10.132	1-3-4
1-6	46.58	6	7.763	1-3-6
2-3	53.46	6	8.910	2-4-3
4-6	53.46	7	7.637	4-3-6
3-5	57.70	6	9.617	3-4-5
1-5	75.38	3	2.513	1-3-4-5
5-6	50.66	6	8.443	5-4-3-6

List (DL): (1,2), (1,4), (3,5), (2,3), (5,6), (1,6), (4,6), (1,5)

From this point onwards the procedure resembles that for FS1.

VITA

Ramaier Sriram was born in Sri Lanka on April 27, 1952. He completed a Bachelor of Science degree in Mechanical Engineering at the University of Sri Lanka, Peradeniya in January 1976. After a four year service with the Ceylon Petroleum Refinery and the National Institute of Business Management, Sri Lanka he accepted in August 1980 a scholarship at the Asian Institute of Technology, Thailand and began study towards a Master's degree in Industrial Engineering and Management. This degree was awarded in April 1982. In the fall of 1982 he accepted a research assistantship at the University of Tennessee, Knoxville and began his Doctoral program. He received the Doctor of Philosophy degree in Management Science in May 1991. Mr. Sriram is at present a full time faculty at the University of Dayton, Dayton, Ohio.