MULTIFACILITY LOCATION WITH IMPRECISE DATA

A THESIS

SUBMITTED TO THE DEPARTMENT OF INDUSTRIAL ENGINEERING

AND THE INSTITUTE OF ENGINEERING AND SCIENCES

OF BILKENT UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

MASTER OF SCIENCE

By

Muhittin Hakan Demir

September, 1994

MULTIFACILITY LOCATION WITH IMPRECISE DATA

A THESIS

SUBMITTED TO THE DEPARTMENT OF INDUSTRIAL ENGINEERING

AND THE INSTITUTE OF ENGINEERING AND SCIENCES

OF BILKENT UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

MASTER OF SCIENCE

By

Muhittin Hakan Demir

September, 1994

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

Barbaros Ç. Tansel

Assoc. Prof. Barbaros Ç. Tansel(Principal Advisor)

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

Assoc. Prof. Osman Oğuz

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

Assoc. Prof. Mustafa Akgül

Approved for the Institute of Engineering and Sciences:

M. Baray

Prof. Mehmet Baray
Director of Institute of Engineering and Sciences

ABSTRACT

MULTIFACILITY LOCATION WITH IMPRECISE DATA

Muhittin Hakan Demir
M.S. in Industrial Engineering
Supervisor: Assoc. Prof. Barbaros Ç. Tansel
September, 1994

Locational decisions often suffer from lack of precise data. In this study, we consider a class of multifacility location problems where the demands of existing and new facilities are unknown, with a known set of possible realizations. The set may be finite or infinite. In the latter case, the data is assumed to be of interval type. We use various criteria to evaluate candidate solutions to these problems and build a framework for decision making.

Key words: Facility location, multifacility minisum problem, imprecise optimization

ÖZET

BELİRSİZ VERİLERLE ÇOKTESİSLİ YERSEÇİMİ

Muhittin Hakan Demir
Endüstri Mühendisliği Bölümü Yüksek Lisans
Tez Yöneticisi: Doç. Dr. Barbaros Ç. Tansel
Eylül, 1994

Tesis Yerseçimi problemleri genellikle eldeki verinin belirsiz olması sorunuyla karşı karşıyadır. Bu çalışmada, taleplerin belirsiz olduğu, ancak bu taleplerin olası değerlerini kapsayan bir kaynak kümenin bilindiği durumlarda çoktesisli yerseçimi problemleri incelenmektedir. Kaynak küme sonlu yada sonsuz olabilir. İkinci durumda, veriler aralıklar tarafından tanımlanmaktadır. Bu tür problemlere aday çözümlerin değerlendirilmesi ve karar verme süreci için model ve ölçütler sunulmaktadır.

Anahtar sözcükler: Çoktesisli yerseçimi, iletişimli medyan problemi, belirsiz verilerle optimizasyon

ACKNOWLEDGEMENT

I would like to dedicate this thesis to my father İbrahim H. Demir, my mother Şükriye Demir and to my brothers Ahmet U. and Hüseyin C. Demir for their love and all the sacrifices that they have made for me.

I am very grateful to Dr. Barbaros Ç. Tansel for teaching me what being an academician meant and of course for his unique guidance and encouragements.

I am indebted to Dr. Osman Oğuz and Dr. Mustafa Akgül for showing keen interest to this study, accepting to read and review this thesis and for their invaluable comments.

I would like to take this opportunity to express my special thanks to my friends Bahar Kara, Dilek Yılmaz, Erhan Kutanoğlu, Sibel Salman and Selçuk Avcı who have always supported and encouraged me and sincerely gave all the help they could.

I also thank Fulya Çakırlar and Bilge Aydın for all their help, especially at times I felt in trouble.

And at last but surely not the least, I want to express my heartily thanks to my fiancée Meltem Karasu for her love, patience, and for giving me a new spirit of life.

Contents

Chapter 1

INTRODUCTION

In this thesis, we investigate the m-Median with Mutual Communication (MMC) Problem, where the data is inexact. The MMC problem is a facility location problem where the objective is to choose the locations of the new facilities so as to satisfy the demands of the new and existing facilities with the minimum total transportation cost. Each type of new facility provides a specific type of service. Each new facility provides service to existing facilities as well as to other new facilities. The level (degree) of interaction between a pair of facilities is expressed by a weight (demand), and the transportation cost is measured as the weighted sum of distances between pairs of facilities. The weights may be interpreted as the number of units exchanged per time period between pairs of facilities. Other interpretations such as frequencies of trips, traffic flow etc. are possible.

We consider MMC problems with inexact weights. The inexactness is modeled by a source set that contains the possible values of weights. The set may be finite or infinite. In the latter case, we assume that the data is expressed in terms of intervals, specified by the lowest and highest values that the weights can take. We further assume that the probability distribution for the elements of the source set is not known, so we do not rely on probabilities.

For such problems, we build a framework to aid in decision making and in evaluating alternative solutions to the problem.

1.1 Related Literature

The deterministic MMC problem is known to be *NP-Hard* when the problem is posed on general networks (Kolen [22]). The node optimality property holds for the problem on general networks (Tansel et.al. [31]). That is, there exists an optimal solution to the problem such that each new facility location coincides with a vertex of the network.

Xu, Francis and Lowe [35] solve the problem on the blocking graph of the network and provide localization results.

Erkut, Francis and Lowe, Francis and Lowe [14],[17] study a version of the problem with upper bound constraints on the distances between pairs of new facilities. They transform the problem to a linear programming problem with factorial constraints and solve the problem on spanning trees of the network to obtain lower bounds on the optimal objective function.

Chajjed and Lowe [8, 7], Fernandez-Baca [16] study special cases of the problem. They present polynomial order algorithms by making particular assumptions on the structure of interaction between new facilities. Their study reveals the importance of the effect of the interaction between new facilities on the problem structure.

Cabot, Francis and Stary [6] have shown that the problem in the plane with rectilinear distances decomposes into two problems, each on the line, and have proposed a maximum cost network flow procedure for solving the line problems.

Picard and Ratliff [29], Trubin [34] and Cheung [10] have developed polynomial time algorithms when the location space is a line. These algorithms have extended to the problem on the plane with rectilinear distances by using the observation that the rectilinear distance problem decomposes into two problems on the line.

When the problem is defined on a tree network, the convexity properties of the objective function and the tractability (convexity) property of the tree structure (Dearing, Francis, Lowe [12]) have been utilized to obtain polynomial time algorithms. Dearing and Langford [13] have proposed an algorithm that constructs the embedding of the tree in R^p for some p, and uses the decomposition idea to solve the problem as p problems, each on the line.

Picard and Ratliff [29] and Kolen [22] propose algorithms for the tree problem that are based on solving successive minimum cut problems on an auxiliary network. Each minimum cut problem is defined with respect to an edge of the tree. The solution of the minimum cut problem determines a subset of new facilities that will be located on an endpoint of the edge. When the problem is restricted to a particular edge of the tree, it is the relative locations of the new facilities with respect to the endpoints of the edge rather than the exact locations that is important. This observation induces the minimum cut structure for the edge problem, and leads to polynomial solvability on tree networks. The time bound for these minimum-cut based algorithms is $O(np^3)$. Tamir [30] improves the time bound for solving the tree problem by using a recursive approach. His algorithm has an order of $O((p^3 + n)logn + np)$.

Chajjed and Lowe [9] study the tree problem where the interactions between new facilities has a special structure and obtain an improved bound.

Facility location problems with uncertain data have been studied by many researchers. The uncertainty in the data is generally modeled by random edge lengths and/or weights, assuming particular probabilities for the possible realizations of these random parameters. (Mirchandani et. al. [26], Mirchandani and Odoni [25], Berman [1], Frank [19, 20], Oudjit [28]). Using the probability information, the candidate solutions are evaluated with respect to some given criterion. The criterion is generally that of minimizing the expected transportation cost. However, other criteria such as minimizing the probability of having a cost larger than a predetermined value, maximizing the probability of being within a predetermined level of the optimal cost have also been used

(Frank [19, 20]).

Other than probability based approaches, we see the parametric approaches of Brandeau and Chiu [5, 3, 2] , Erkut and Tansel [15] , Labbe et. al. [24]. These studies use a sensitivity analysis approach for location problems by parameterizing the weights or objective function components. They generate the trajectory of the optimal solutions as the parameters change (usually as a function of time).

Tansel and Scheuenstuhl [33] investigate the 1-median problem where the weights are inexact, with a given interval of realizable values for each weight. They propose new criteria for location problems with inexact data and construct solutions with respect to their criteria. This study has given the motivation for the criteria that we have used in this thesis.

The thesis is organized as follows : In the next section, we briefly review the network location problems, with particular emphasis on network location problems with inexact data. Then we present a discussion of the MMC problem, the type of inexactness that we deal with, and observations on the problem structure. We conclude this chapter with the definitions of our proposed criteria. The second chapter is devoted to the analysis of the problem where the source set is infinite (continuous case). The third chapter discusses the case when the source set is finite (discrete case). Finally, we give a conclusion and state some directions for future research.

1.2 Network Location

Network location problems deal with choosing locations of a set of new facilities on a network in order to meet (optimize) some criterion. The criterion is generally that of satisfying customers' (clients') demands with respect to the objective of optimizing some function of the (travel) distances. In some cases, the new facilities themselves are also assumed to act as customers, with demands for service from other new facilities. New facilities may provide homogeneous (the same kind of) service, or each new facility may be designed to give a particular type of service. In the latter case, we refer to the new facilities as 'distinguishable facilities'.

Capacity constraints may be imposed on the service that any new facility may provide, or alternatively the problem may be 'uncapacitated'. New facilities may also be restricted to a subset of the network, such as the vertices (vertex restricted problem), or any point on the network may be eligible for locating the new facilities.

The customers (referred to also as existing facilities, clients) may be represented by a finite number of discrete points (that can be defined to be the vertices of the network), and alternatively, each point of the network, vertex or not, may be assumed to generate demand.

The objective is generally taken as minimizing some function of the travel distances plus a function of the locations of the new facilities.

There are mainly two types of criteria to evaluate the 'travel distances' portion of the objective function, 'minisum' and 'minimax'.

Minisum type objectives try to minimize the weighted sum of distances between the new facilities and the customers. Customers may include the new facilities. The weights can reflect relative importances, demands, or cost of travel per unit distance. This type of objective is suitable especially when the transportation costs account for a large portion of distribution costs.

Minimax type of objectives try to minimize the maximum of the weighted distances of each customer to the new facilities. In situations where it would be desired to 'cover' each customer within a reasonable distance (time), minimax type of objectives are more appropriate. These are cases where the cost increases so quickly with distance that the maximum distance becomes the only determining factor. Such costs are generally induced by non-transportation related criteria. Examples would be locating fire stations or ambulances, where the 'cost' is related with human life.

In some cases, it is possible to assign weights to the minisum and minimax objectives to obtain hybrid models.

The objective may also have a component that is a function of the locations of the new facilities only. Most common examples are the existence of a fixed cost for establishing a new facility.

For a review of location problems, one may see the references : Tansel et. al. [31, 32], Brandeau and Chiu [4], Labbe, Peeters and Thisse [23].

1.2.1 Inexactness (Uncertainty) in Locational Decisions

As in many decision making situations, locational decisions often suffer from lack of precise data.

To begin with, the network under consideration is generally an abstraction and a simplified version of the real location space. The network distances (or travel times), for example, are just estimates of the true distances. There are many cases where the real system may not even exist and the network is a representation of just a hypothetical model (like highway construction projects etc.). This adds one more level of uncertainty to the model. Apart from the discussion of accurately representing the real situation, the system parameters are unlikely to be static. There are usually expected and unexpected changes in the system under consideration.

The distances (travel times) on a traffic network, for example, are subject to variation from time to time. Morning and rush hours are expected to have high traffic intensity, thus inducing longer travel times. Weather conditions, governmental policies that change the routes, unexpected events like accidents, maintenance are all factors that cause variations in travel times.

The weights (demands, cost of travel per unit distance) are also subject to changes. If weights represent demands, some sources of fluctuations in demands may be caused by changes in customers' preferences, external factors like competitors' policies, general state of the economy, internal factors like price discounts, quantity discounts, previous service performance. When weights represent the cost of travel per unit distance, changes in prices of inputs for providing the service, deterioration of equipment used to transport the product may be some sources of inexactness.

Considering the nature of location decision making then, one needs a framework for making locational decisions with inexact data.

The traditional approach in the literature is the use of probability tools. The most commonly used 'expected cost' approach compares alternative choices of new facility locations on the basis of the expected value of the cost they induce with respect to the given objective function.

However, such approaches have drawbacks. First, probabilistic approaches require a good deal of information on the probability distributions of possible outcomes (values) of distances and weights. These probabilities are generally very difficult to estimate, especially in the presence of external factors. Apart from this, the probabilities estimated by one individual will remain to be subjective, and this will bring additional difficulties when there are more than one decision makers.

Unless properties like independence, no correlation are assumed, it will be computationwise very difficult (time consuming) to calculate expected values (or other probability related measures) for alternative choices of new facility locations. It is usually the case that demands are correlated and are affected from each other .

Leaving the computational difficulties aside and assuming that the decision makers agree on the same probability figures or distributions, we observe that the expected cost criterion cares only about the long run performance of the system and misses the likely behavior in the short run, possibly a predictable transient state. The short run performance of the system may affect subsequent decisions and operating policies. A bad initial performance may also cause loss of good will and may drive the system out of operation. Besides their immediate implications, these factors show one other source of correlation for the system parameters.

One other possibility is to represent the inexactness in weights or distances by identifying a set of possible (realizable) values, without relying on a particular probability distribution.

For example, for each individual weight or distance, we may identify an interval $[l, u]$ from which the variable (weight) will take values. Such an interval can be specified easily in many cases by considering the 'worst case' and the 'best case' effects of the factors. In the presence of multiple decision makers, it is unlikely that the decision makers agree on point values of uncertain variables, but it would be easier to convince them on an interval of possible values. Even if there is a single decision maker, he would feel more comfortable to represent his estimates as an interval, covering the effects of many possible outcomes, than restricting his estimate to one single value.

We may alternately specify the set of realizable instances with a finite set such that each element of the set refers to one possible state of the system parameters. Such a representation is adequate in cases where there are previously predictable states of the system whereby we can represent each possible state by an element of the set. Also, if information about the values of a set of parameters is more or less sufficient to estimate the values of the other parameters, this approach may be suitable.

Nevertheless, the number of realizable instances may become quite large, a continuum number in the case of interval data, which is likely to cause computational problems. Moreover, this framework tends to provide less direct information (basis) for decisions as the number of realizable instances increase. We will propose ways to resolve this problem by designing our criteria so as to facilitate their use in conjunction with other tools (like expert judgment, maximizing probability on a filtered set of qualified candidate locations, trying to observe structural similarity on 'qualified' elements, etc.).

Thus, representing data by a set of realizable values is easier for multiple decision makers to agree on and makes the decision maker feel more comfortable if there is a single decision maker. In addition, it can be used in conjunction with other decision making frameworks.

Throughout this study, we assume that we are given a source set of realizable values of the parameters under consideration. We do not know *a priori* which value the nature will choose, but certainly a value within the given set. We further assume that the probability distribution for the elements of the set is not known, or very difficult to assess, so we do not rely on probabilities.

1.3 Multifacility Problem (MMC)

The problem that we consider is the m-Median with Mutual Communication (MMC) problem. The objective is to choose the locations of m distinguishable new facilities on a network so as to minimize the sum of weighted distances between new and existing facilities and between pairs of new facilities. The existing facilities are located at the vertices of the network. Each new facility can provide any quantity of the particular service; there are no capacity limitations. We assume that new facilities have no locational limitations. Each point of the network is eligible for locating any number of new facilities.

We want to emphasize that the interaction between pairs of new facilities have an important role in the structure of the problem. Without these interactions, the problem would be equivalent to m independent single facility location problems, one for each new facility.

The mathematical statement of the problem can be given as:

$$minimize_{x_1,x_2,...,x_m \in G} \sum_{i=1}^{n}\sum_{j=1}^{m} w_{ij}d(v_i,x_j) + \sum_{j=1}^{m}\sum_{k=j+1}^{m} v_{jk}d(x_j,x_k)$$

where G is the network under consideration,

x_j = Location of the $j\,th$ new facility, $j = 1,...,m$

w_{ij} = Demand of $i\,th$ existing facility for service j

$$i = 1,...,n\ ,\ \ j = 1,...,m$$

v_{jk} = Demand of $j\,th$ new facility for service k

$$j = 1,...,n\ ,\ \ k = (j+1),...,m$$

and $d(.,.)$ is the distance function

Let's use the shorthand notation W and V to show the vectors whose components are w_{ij}'s and v_{jk}'s respectively.

1.3.1 *MMC* with Inexact Weights

We deal with MMC problems where the vectors W and V are not known *a priori*, but we are given a (source) set from which these vectors will take values.

This source set can be specified as a cartesian product of intervals $E_{ij} = [\,\underline{w}_{ij}, \bar{w}_{ij}]$ for each w_{ij} and $N_{jk} = [\underline{v}_{jk}, \bar{v}_{jk}]$ for each v_{jk} (continuous case). We assume that at least one interval is nondegenerate to distinguish this case from the deterministic problem. In this situation, clearly, we have a continuum number of possibilities -realizable instances- of the problem.

Let us define $D = \{(\tilde{W},\tilde{V}) : \tilde{w}_{ij} \in E_{ij}, \tilde{v}_{jk} \in N_{jk}\}$ as the hyperrectangle containing all realizable weight vectors. That is, every $d \in D$ specifies a possible problem instance.

If the weights w_{ij} and v_{jk} have discrete (finitely many) realizations, the source set can be specified as a finite set with elements (W_i, V_i), $i = 1, ..., s$. We shall call these elements 'scenarios' where each scenario refers to a complete specification of W and V vectors.

In either case, we have a well defined set of realizable instances, we are sure that one element of this set will be realized, but we do not *a priori* know which element of this set will actually determine the problem.

Throughout, we assume that the number of new facilities m , the location space, and the number and locations of the existing facilities are fixed, whereby an instance of the problem will be specified by W and V vectors. To emphasize this dependence, we will refer to an instance of MMC as $MMC(W, V)$.

To be able to deal with this kind of inexactness, we naturally first try to identify whether we can find one feasible solution vector that will 'cover' all possible problem instances. That is, we look for a solution that will be optimal whichever problem instance is actually realized. If such a solution exists, one can place all the new facilities according to that solution without any fear of being suboptimal.

However, we may not be able to, and in many cases we are not able to find such solutions. Clearly , we are less likely to find such a 'permanently optimal' solution as the cardinality of the source set becomes larger. In such cases, then, we are interested in identifying a set of 'qualified' candidate solutions. The word 'qualified' is not operational itself, so we should clarify what we would require from a qualified candidate solution. One approach could be to try to identify solutions that are optimal for more likely instances of the problem. Or we may want to find solutions which have the highest probability of being optimal.

However, such approaches may require a good deal of information on the individual probabilities of occurrence of elements of the source set. Also, such probability related (probability maximization, expected cost minimization etc.) approaches have drawbacks like considering only the long run (expected) or most likely behavior of the nature and missing the rest of the picture such as the worst case situations, etc.

To be more general, then we try to identify all solutions which have a chance of being optimal. That is, we look for solutions that are optimal for some realizable instance (element of the source set). We note, however that the number of such solutions can be quite a few, and increases with the cardinality of the source set (in fact, nondecreasing). Still, the set of such solutions forms a Finite Dominating Set for the node restricted problem with inexact data. We know that candidate solutions outside of this set have no chance of being optimal, and each element of this set is optimal for some realizable problem instance. This information is quite valuable in some cases like when the set has a small cardinality or when the elements of this set have a kind of robustness property such as similarity to each other, accumulation of new facility locations to a region of the network etc. , or when the decision maker is provided with some additional kind of measure to choose among these solutions.

In some cases, we may not be able to find a permanently optimal solution or the size of the set of possibly optimal solutions may be too large. We may still want to 'cover' all the realizable problem instances, but do this more efficiently than using the whole set of possibly optimal solutions. By efficiency, we would like to identify a small (as small as possible) set of candidate solutions to cover all the realizable instances.

Towards this objective, we are trying to find a set of candidate solutions that will supply an optimal solution for every realizable problem instance. That is, we search for a set of solution vectors, such that the elements are 'unionwise permanent'. We know that such a set always exists; in the worst case, it is the whole set of possibly optimal solutions, as for every realizable problem instance the set of possibly optimal solutions contains an optimal solution. There may also be alternative sets of candidate solutions, each supplying an optimal solution for every realizable problem instance, in which case we try to apply selection rules to choose among such sets. One obvious rule could be to favour the set that has the lowest cardinality. We later develop other criteria like minimizing the dissimilarities in the elements of the set etc.

We also show a way of 'efficiently' constructing a set that supplies an optimal solution for every possible problem instance. This computational efficiency increases the significance of unionwise permanent solutions, especially when we consider the significance of forming a quick framework for the decision maker.

1.3.2 Observations on the MMC problem

We define an instance of the Multifacility Mutual Communication (MMC) problem by the locations of n existing facilities: $(v_1, v_2, ..., v_n)$, a positive integer m denoting the number of new facilities to be located, the weight(demand) vectors: $W = \{w_{ij} : 1 \leq i \leq n, 1 \leq j \leq m\}$ for the existing facilities and $V = \{v_{jk} : 1 \leq j < k \leq m\}$ for the new facilities , which are nonnegative reals, and finally the location space G with the type of distance measure $d(.,.)$ (metric).

Our objective is to choose locations of m new facilities so as to minimize the weighted sum of distances between new and existing facilities and between pairs of new facilities.

The combinatorial statement of the problem can be given as :

(MMC)

$$minimize_{x_1,...x_m \in G} \sum_{i=1}^{n}\sum_{j=1}^{m} w_{ij}\, d(v_i, x_j) + \sum_{j=1}^{m}\sum_{k=j+1}^{m} v_{jk}\, d(x_j, x_k)$$

Any m-vector $X = (x_1, ..., x_m)$ such that $x_i \in G$, $\forall\, i = 1, ..., m$ is called a *Candidate Solution* (feasible solution) for the problem.

We now state the vertex optimality property for the MMC problem :

Theorem 1.1 *(Tansel et.al. [31]) For the MMC problem on general networks, there exists an optimal solution $X = (x_1, ..., x_m)$ such that each new facility location x_i coincides with a vertex of the network.*

The vertex optimality property is important in the sense that the continuous set of alternative candidate solutions reduce to a Finite Dominating Set of discrete alternatives. Based on Theorem 1.1, we shall restrict our attention hereafter to vertex solutions of the MMC problem.

PRELIMINARIES AND NOTATION :

Before going into the technical discussion, we give the following definitions :

An undirected network $G = (V, A)$ consists of a set $V = \{v_1, ..., v_n\}$ of nodes and a set $A = \{(v_i, v_j)\}$ of unordered pairs of distinct nodes in G. We take G to be the continuum set of all points of an embedded network (see Dearing, Francis and Lowe [12] for embedding). Thus, if $[v_i, v_j]$ is the set of all points of the embedded arc defined by the vertices v_i and v_j, then $G = \cup\{[v_i, v_j] : (v_i, v_j) \in A\}$

A cycle C of G is an ordered set of nodes $(v^1, ..., v^k)$ such that $(v^i, v^{i+1}) \in A$, $i = 1, ..., k-1$, and $(v^k, v^1) \in A$. A path P is an ordered set of nodes $(v^1, ..., v^k)$ such that $(v^i, v^{i+1}) \in A$, $i = 1, ..., k-1$.

A tree is a connected network with no cycles. A subtree is a connected subset of a tree. A subtree rooted at vertex v_i is a maximal subtree not containing v_i. Whenever the network of interest is a tree we write T instead of G.

For notational convenience, we define the following expressions :

$$F(X, W, V) = WD_1(X) + VD_2(X) = \sum_{i=1}^{n}\sum_{j=1}^{m} w_{ij}\, d(v_i, x_j) + \sum_{j=1}^{m}\sum_{k=j+1}^{m} v_{jk}\, d(x_j, x_k)$$

Here, W is the 1 by mn vector whose $i + (j-1)n$ th component is w_{ij},

$D_1(X)$ is the mn by 1 vector with components $d(v_i, x_j)$'s (with the same ordering as W).

V is the 1 by $m(m-1)/2$ vector with components v_{jk}'s (we assume that components of V are ordered such that v_{jk} comes before v_{st} iff $j < s$ or $j = s$

and $k < t$)
and D_2 is the column vector with components $d(x_j, x_k)$'s (with the same ordering as V).

Whenever we use v_{jk} for $j > k$, we will mean v_{kj}

We use $J = \{1, ..., m\}$ to denote the index set of the new facilities.

In addition, for a subset Q of T, and subsets S, S_1, S_2 of J, we define :

$$W_S(Q) = \sum_{v_i \in Q} \sum_{j \in S} w_{ij}, \text{ and}$$

$$V_{S_1}(S_2) = \sum_{j \in S_1} \sum_{k \in S_2} v_{jk}$$

Let $\delta(v_i)$ be the degree of v_i. Given a vertex v_i, with adjacent vertices $v_i^1, v_i^2, v_i^3, ..., v_i^{\delta(v_i)}$, we define T_i^j = the subtree rooted at v_i, containing the adjacent vertex v_i^j.

Given a candidate solution $X = (x_1, x_2, x_3, ..., x_m)$ on vertices of the network, we denote the vertex on which new facility j is located alternatively by x_j and v_{a_j}, and we let $F_j = \{k \in J : x_k = x_j\}$ to show the index set of new facilities located on the same vertex as new facility j.

In the rest of the thesis, we sometimes abuse notation and write :

For $f_1, f_2 \subseteq J$, $Q_1, Q_2 \subseteq T$, we write $W_{f_1}(f_2)$ to mean $W_{f_1}(\{x_k : k \in f_2\})$ and write $V_{Q_1}(Q_2)$ to mean $V_{(\{k: x_k \in Q_1\})}(\{j : x_j \in Q_2\})$. In fact, we may further abuse notation and write, say, $V_{f_1}(Q_2 \cup f_2)$ to mean $V_{f_1}(\{k : x_k \in Q_2\} \cup f_2)$.

We now restate a theorem that gives the conditions of optimality of a given candidate solution for the deterministic problem when the location space is a tree network.

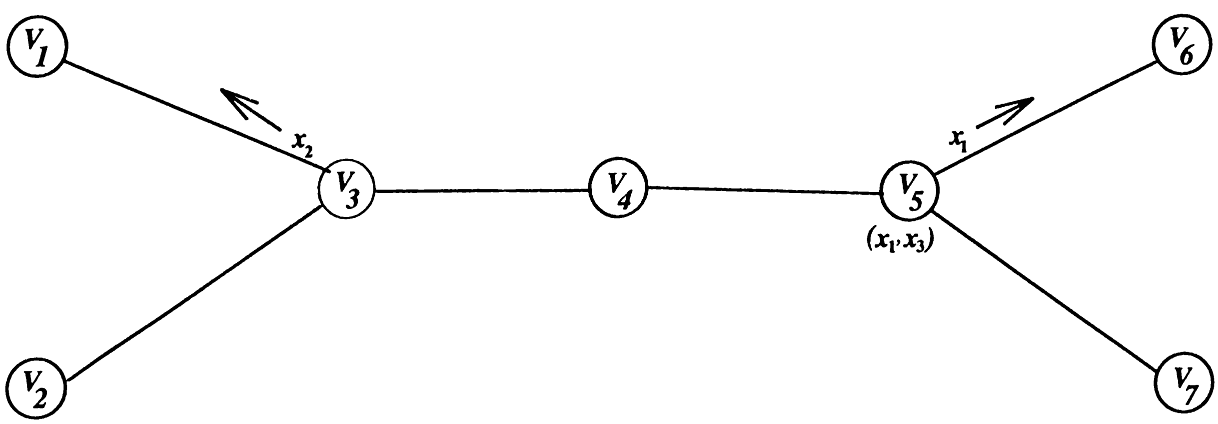
Theorem 1.2 *(Kolen [22]) : For MMC on a tree network, a given candidate solution is optimal if and only if no subset of new facilities can be moved to* *an adjacent vertex location such that the objective function improves (total cost decreases).*

This theorem actually characterizes a local optimality condition as a necessary and sufficient condition for global optimality. This property is due to the tractability of the tree structure and the convexity properties of the objective function (Dearing, Francis, Lowe [12]).

Theorem 1.2 gives us a way of testing the optimality of a given candidate solution. Based on this solution, each time we should consider moving a subset of new facilities to an adjacent location and check whether the objective function improves or not. We need not consider simultaneous movement of new facilities located at different vertices because, as we will show in the next example, such moves are accounted for by disjoint moves of its components, each component containing new facilities located on one vertex only.

The example below illustrates Theorem 1.2.

Example :(n = 7,m = 3)



Change in the objective function resulting from moving x_1 to v_6 and x_2 to v_1 =

$$(w_{11}+w_{21}+w_{31}+w_{41}+w_{51}-w_{61}+w_{71})d(v_5,v_6)+(-w_{12}+w_{22}+w_{32}+w_{42}+w_{62}+w_{72})d(v_3,v_1)$$
$$+v_{12}(d(v_6,v_1)-d(v_5,v_3)) \quad (1)$$

Change in the objective function resulting from moving x_1 to v_5 =

$$(w_{11}+w_{21}+w_{31}+w_{41}+w_{51}-w_{61}+w_{71})d(v_5,v_6)+v_{12}(d(v_6,v_3)-d(v_5,v_3)) \quad (2)$$
and

Change in the objective function resulting from moving x_2 to v_1 =
$$(-w_{12}+w_{22}+w_{32}+w_{42}+w_{62}+w_{72})d(v_3,v_1)+v_{12}(d(v_5,v_1)-d(v_5,v_3)) \quad (3)$$

(1) = (2)+(3)since $d(v_6,v_3)-d(v_5,v_3) = d(v_5,v_6)$ and $d(v_5,v_6)+d(v_5,v_1) = d(v_6,v_1)$

Based on the tree structure, when we move a subset f of new facilities located at v_i to an adjacent vertex v_i^j, the new facilities in f will now become closer to the (new and existing) facilities located in T_i^j, while they remain in the same distance to new facilities in f and are farther from all other facilities.

So, given a candidate solution, we can test its optimality by expressing a set of objective function differences (each defined by a subset of new facilities and a subtree rooted at the vertex on which they are located), and checking whether they are nonnegative or not. If all objective function differences are nonnegative, then the given candidate solution is optimal.

That is, X is optimal for $MMC(W,V)$ iff

$$W_f(T\backslash T_i^j)d(v_i,v_i^j)-W_f(T_i^j)d(v_i,v_i^j)+V_f(T\backslash(T_i^j\cup f))d(v_i,v_i^j)-V_f(T_i^j)d(v_i,v_i^j) \geq 0$$

$$\forall f \subseteq F_i$$

$$\forall\, T_i^j,\ j=1,...,\delta(x_i)$$

$$\forall\, i=1,...,m$$

By the positivity of $d(v_i,v_i^j)$ the conditions become :

$$W_f(T\backslash T_i^j)-W_f(T_i^j)+V_f(T\backslash(T_i^j\cup f))-V_f(T_i^j) \geq 0$$

$$\forall f \subseteq F_i$$

$$\forall\, T_i^j,\ j=1,...,\delta(x_i)$$

$$\forall\, i=1,...,m$$

It is important here to note that the problem becomes independent of the distances. It is rather the topology of the tree and the weight relations induced by this topology that define the problem. Each X induces its own set of inequalities in the variables W and V.

As another observation, we see that related with each distinct x_i (for each set F_i) , we have a 'block' of inequalities (call it B_i). The number of inequalities in this block is $\delta(x_i)(2^{|F_i|}-1)$ which is the product of the number of edges incident to v_{a_i} with the number of nonempty subsets of F_i.

It is notable that the number of inequalities here is proportional to the number of subsets of the F_j's, which is an exponential function of the number of co-located new facilities. This property leads to difficulties when new facility locations coincide. The analysis becomes easier for candidate solutions for which all new facility locations are distinct. We will make further observations about such structural properties and describe our ways of dealing with such difficulties in the next sections.

In the set of inequalities, the w_{ij} variables denoting the demands of the existing facilities appear only in the block related to new facility j. The v_{jk} variable denoting the demands of the new facilities appears both in the block related to new facility j and in the block related to new facility k. If new facility j and new facility k are at different locations, then there is no connection between the block related to x_j and the block related to x_k other than the variable v_{jk}; the v_{jk} variables act as 'links' between blocks.

So, we observe that the interaction between pairs of new facility locations is the determining factor for the structure of the problem. In one extreme, if there were no interactions between pairs of new facilities, the problem would decompose into m disjoint subproblems.

For a better idea of this structure, we can use an auxiliary network called the Linkage Network (LN_B). We construct LN_B as follows : We have a node N_j for each new facility j , $j = 1, ..., m$. There is an undirected arc $[N_j, N_k]$ between N_j and N_k iff $v_{jk} > 0$. For the problem with interval data, there is an undirected arc between N_j and N_k iff $\bar{v}_{jk} > 0$. We can observe that the problem decomposes into disjoint subproblems in the number of disjoint components of LN_B, since we do not have any constraint that involves variables from different components of LN_B. For that matter, we assume throughout this study that LN_B is connected. Further properties of LN_B will be discussed in the next sections.

1.4 Criteria

Now, we present formal definitions of the three main criteria that we have discussed in the previous section.

Given the source set D that defines all realizable problem instances:

Definition 1.1 *(Permanent Solution) A candidate solution X is a Permanent Solution if X solves $MMC(W,V)$ for every $d = (W,V) \in D$. The set of Permanent Solutions is called the Permanent Set.*

Definition 1.2 *(Weak Solution) A candidate solution X is a Weak Solution if X solves $MMC(W,V)$ for some $(W,V) \in D$. We call the set of Weak solutions the Weak Set.*

Definition 1.3 *(Unionwise Permanent Solution) A set $U = \{X_1, X_2X_3, ..., X_p\}$ of candidate solutions is called a Unionwise Permanent Solution if U supplies an optimal solution for every $MMC(W,V)$ such that $d = (W,V) \in D$. (That is, given any $(W,V) \in D$, X_1 solves $MMC(W,V)$, or X_2 solves $MMC(W,V), ...,$ or X_p solves $MMC(W,V)$).*

Chapter 2

CONTINUOUS CASE

In the Continuous Case, we assume that the data is represented by intervals. For each w_{ij} , $i = 1, ..., n$, $j = 1, ..., m$, we are given an interval $E_{ij} = [\underline{w}_{ij}, \bar{w}_{ij}]$, and for each v_{jk} , $j = 1, ..., m$, $k = j + 1, ..., m$, we are given an interval $N_{jk} = [\underline{v}_{jk}, \bar{v}_{jk}]$.

It follows then the source set D is defined by

$$D = \{(W, V) : w_{ij} \in E_{ij}, \forall\,(i, j)\,,\ v_{jk} \in N_{jk}, \forall\,(j, k)\}$$

which is a hyperrectangle in R^t with $t = mn + m(m-1)/2$.

2.1 Weak Solutions

Recalling that the weak set consists of all the candidate solutions that have a chance of being optimal, we characterize the weak set and generate it (implicitly or explicitly) whenever possible. We limit our discussion in this section (for the continuous case) to the problems where the location space G is a tree network and extend our results to general networks whenever possible.

Theorem 1.2 gave us a way of testing the optimality of a given candidate solution for the deterministic problem. We now utilize this theorem as a tool for testing the membership of a given candidate solution to the weak set.

Given the weight vectors (W, V), a candidate solution $X = (x_1, x_2, x_3, ..., x_m) = (v_{a_1}, v_{a_2}, v_{a_3}, ..., v_{a_m})$, X is optimal for (W, V) iff

(I_X)

$$W_f(T\backslash T_i^j) - W_f(T_i^j) + V_f(T\backslash(T_i^j \cup f)) - V_f(T_i^j) \geq 0$$

$$\forall f \subseteq F_i$$

$$\forall\, T_i^j,\ j = 1, ..., \delta(x_i)$$

$$\forall\, i = 1, ..., m$$

It follows that X is a weak solution iff the system given above is feasible for some $(W, V) \in D$. That is, X is a weak solution iff there exist a solution $(W, V) \in D$ to the following system :

$$W_f(T\backslash T_i^j) + V_f(T\backslash(T_i^j \cup f)) \geq W_f(T_i^j) + V_f(T_i^j)$$

$$\forall f \subseteq F_i$$

$$\forall\, T_i^j,\ j = 1, ..., \delta(x_i)$$

$$\forall\, i = 1, ..., m$$

$$\underline{W} \leq W \leq \bar{W}\,,\quad \underline{V} \leq V \leq \bar{V}$$

We observe that each inequality above is defined by subset sums of w_{ij} and v_{jk} variables so that this system has all zeros and ones as the coefficients of the w_{ij} and v_{jk} variables. We can then represent the system as:

$$(I_X) \qquad A_1W + B_1V \geq A_2W + B_2V$$

$$\underline{W} \leq W \leq \bar{W}\,,\quad ,\ \underline{V} \leq V \leq \bar{V}$$

where A_1, A_2, B_1, B_2 are zero/one matrices. A_1 and A_2 have as many rows as there are inequalities, and mn $(=\mid W \mid)$ columns. B_1 and B_2 have as many rows as there are inequalities, and $m(m-1)/2$ $(=\mid V \mid)$ columns.

Each row of the matrices A_1, A_2, B_1, B_2 corresponds to an inequality of (I_X). For any w_{ij} variable, there is a 1 in a given row if w_{ij} appears in that inequality. If w_{ij} appears on the lefthandside of the inequality, A_1 has a 1 only, and if w_{ij} appears on the righthandside of the inequality, A_2 has a 1 only (w_{ij} cannot appear both on the right and on the left in the same row). If w_{ij} does not appear in an inequality, then the corresponding components of A_1 and A_2 are 0. Similarly, there is a 1 in a particular row of B_1 or B_2 (but not both) for v_{jk} if v_{jk} appears in the associated inequality. Otherwise, both B_1 and B_2 have 0's for v_{jk}.

Now, we conclude that the identification of membership of a given candidate solution to the weak set turns out to be a linear feasibility problem (with lower-upper bounds on the variables) in w_{ij} and v_{jk} variables.

We now present a theorem that will help eliminate some of the variables from the above feasibility problem.

Theorem 2.1 *A given candidate solution X is a weak solution iff the following system is consistent :*

$$(I'_X) \qquad A_1\bar{W} + B_1V \geq A_2\,\underline{W} + B_2V$$

$$\underline{V} \leq V \leq \bar{V}$$

where A_1, A_2, B_1, B_2 are as defined in (I_X)

This theorem eliminates the W vector from the system (I_X) by replacing w_{ij} variables with their lower or upper bound values ($\underline{w}_{ij}$ or $\bar{w}_{ij}$)

Proof : Consider the system of (I_X) for a given $X = (x_1, x_2, ..., x_m)$:

For a variable w_{ij} the inequalities involving this variable in (I_X) are either lower-upper bound inequalities or in one of the following two forms : Either

$$W_{f_1}(T \backslash T_j^a) + V_{f_1}(T \backslash (T_j^a \cup f_1)) \geq W_{f_1}(T_j^a) + V_{f_1}(T_j^a)$$

for some $f_1 \subseteq F_j$ such that $j \in f_1$ and $v_i \in T_j^a$ (corresponding to moving a subset -containing j- of new facilities to the subtree containing v_i),
or

$$W_{f_2}(T\backslash T_j^b) + V_{f_2}(T\backslash(T_j^b \cup f_2)) \geq W_{f_2}(T_j^b) + V_{f_2}(T_j^b)$$

for some $f_2 \subseteq F_j$ such that $j \in f_2$ and $v_i \notin T_j^b$ (corresponding to moving a subset -containing j- of new facilities to a subtree other than the one containing v_i).

In the former case, w_{ij} has a coefficient of $+1$ on the right side so that the former inequality can be stated as :

$$[(W_{f_1}(T\backslash T_j^a) - (W_{(f_1\backslash j)}(T_j^a) + W_j(T_j^a\backslash v_i))] + [V_{f_1}(T\backslash(T_j^a \cup f_1)) - V_{f_1}(T_j^a)] \equiv R_{f_1} \geq w_{ij}$$

In the latter case, w_{ij} has a coefficient of $+1$ on the left side so that the inequality can be stated as :

$$w_{ij} \geq [W_{f_2}(T_j^b) - (W_{(f_2\backslash j)}(T\backslash T_j^b) + W_j(T\backslash(T_j^b \cup v_i)))] + [V_{f_2}(T_j^b) - V_{f_2}(T\backslash(T_j^b \cup f_2))] \equiv L_{f_2}^b$$

Thus, the system (I_X) can be represented in the following equivalent form :

$$(I_{(X,w_{ij})})$$

$$w_{ij} \leq R_{f_1} \ \forall f_1 \subseteq F_j \, , \, j \in f_1$$

$$w_{ij} \geq L_{f_2}^b \ \forall f_2 \subseteq F_j \, , \, j \in f_2 \text{ and } b = 1, ..., \delta(x_j), \ b \neq a,$$

$$w_{ij} \leq \bar{w}_{ij} \ , w_{ij} \geq \underline{w}_{ij}$$

where $I_{(X,w_{ij})}$ denotes that part of (I_X) that does not involve w_{ij} (those corresponding to moving subsets of J that do not contain j to subtrees)

Consistency requires that the smallest upper bound on w_{ij} be at least as large the largest lower bound on w_{ij}. It follows then the initial system is consistent iff the following system is consistent :

$$(I_{(X,w_{ij})})$$

$$\underline{w}_{ij} \leq R_{f_1} \ \forall f_1 \subseteq F_j \, , \, j \in f_1$$

$$\bar{w}_{ij} \geq L_{f_2}^b \ \forall f_2 \subseteq F_j \, , \, j \in f_2 \text{ and } b = 1, ..., \delta(x_j) \ b \neq a$$

$$L^b_{f_2} \leq R_{f_1} \;\; \forall \; f_1, f_2 \subseteq F_j \;\; , j \in f_1 \cap f_2 \;\; b = 1, ..., \delta(x_j) \;\; b \neq a$$

Note that the last set of inequalities reflects the fact that the intervals $[\underline{w}_{ij}, \bar{w}_{ij}]$ and $[max_{(f_2,b)} L^b_{f_2}, min_{f_1} R_{f_1}]$ should intersect for the system to be consistent. Note also that we simply used the Fourier-Motzkin elimination so far (see Dantzig [11]) to eliminate w_{ij} and obtain an equivalent system with one less variable.

Now, we show that this last set of inequalities are implied by inequalities in $(I_{(X,w_{ij})})$, and thus are redundant. Consider the inequality for fixed f_1, f_2, b :

$$[W_{f_2}(T^b_j) - (W_{(f_2\backslash j)}(T\backslash T^b_j) + W_j(T\backslash(T^b_j \cup v_i)))] + [V_{f_2}(T^b_j) - V_{f_2}(T\backslash(T^b_j \cup f_2))] = L^b_{f_2} \leq$$

$$R_{f_1} = [W_{f_1}(T\backslash T^a_j) - (W_{(f_1\backslash j)}(T^a_j) + W_j(T^a_j \backslash v_i))] + [V_{f_1}(T\backslash(T^a_j \cup f_1)) - V_{f_1}(T^a_j)]$$

Adding w_{ij} to both sides then rearranging the terms, we have :

$$[W_{f_2}(T^b_j) - W_{f_2}(T\backslash T^b_j)] - [W_{f_1}(T\backslash T^a_j) - W_{f_1}(T^a_j)] \leq$$

$$[V_{f_1}(T\backslash(T^a_j \cup f_1)) - V_{f_1}(T^a_j)] - [V_{f_2}(T^b_j) - V_{f_2}(T\backslash(T^b_j \cup f_2))] \quad (1)$$

Consider the inequalities in $(I_{(X,w_{ij})})$ corresponding to moving $(f_1 - f_2)$ to T^a_j and moving $(f_2 - f_1)$ to T^b_j :

$$W_{(f_1-f_2)}(T^a_j) - W_{(f_1-f_2)}(T\backslash T^a_j) \leq -V_{(f_1-f_2)}(T^a_j) + V_{(f_1-f_2)}(T\backslash(T^a_j \cup (f_1 - f_2))) \quad (2)$$

In the case when $(f_1\text{-}f_2) = \emptyset$, we take this inequality to be $0 \leq 0$.

$$W_{(f_2-f_1)}(T^b_j) - W_{(f_2-f_1)}(T\backslash T^b_j) \leq -V_{(f_2-f_1)}(T^b_j) + V_{(f_2-f_1)}(T\backslash(T^b_j \cup (f_2 - f_1))) \quad (3)$$

In the case when $(f_2\text{-}f_1) = \emptyset$, we take this inequality to be $0 \leq 0$.

We claim that Lefthandside $(2)+(3) \geq$ Lefthandside (1) and Righthandside $(2)+(3) \leq$ Righthandside (1) so that $(2)+(3)$ implies (1). Using the fact that if f', f'' are two arbitrary sets then $f' = (f'\text{-}f'') \cup (f' \cap f'')$, we have:

$$LHS(1) = [W_{(f_2-f_1)}(T^b_j) + W_{(f_1\cap f_2)}(T^b_j)] - [W_{(f_2-f_1)}(T\backslash T^b_j) + W_{(f_1\cap f_2)}(T\backslash T^b_j)] -$$

$$[W_{(f_1-f_2)}(T\backslash T^a_j) + W_{(f_1\cap f_2)}(T\backslash T^a_j)] + [W_{(f_1-f_2)}(T^a_j) + W_{(f_1\cap f_2)}(T^a_j)]$$

$$= LHS(2+3) + W_{(f_1\cap f_2)}(T_j^b) - W_{(f_1\cap f_2)}(T\backslash T_j^b) - W_{(f_1\cap f_2)}(T\backslash T_j^a) + W_{(f_1\cap f_2)}(T_j^a)$$

$$= LHS(2+3) - 2W_{(f_1\cap f_2)}(T\backslash(T_j^a \cup T_j^b)) \leq LHS(2+3)$$

So, Lefthandside $(2+3) \geq$ Lefthandside (1). Similarly :

$$RHS(1) = [V_{(f_1-f_2)}(T\backslash(T_j^a \cup (f_1-f_2))) - V_{(f_1-f_2)}(f_1\cap f_2) + V_{(f_1\cap f_2)}(T\backslash(T_j^a \cup f_1))] -$$

$$[V_{(f_1-f_2)}(T_j^a) + V_{(f_1\cap f_2)}(T_j^a)] +$$

$$[V_{(f_2-f_1)}(T\backslash(T_j^b \cup (f_2-f_1))) - V_{(f_2-f_1)}(f_1\cap f_2) + V_{(f_1\cap f_2)}(T\backslash(T_j^b \cup f_2))] -$$

$$[V_{(f_2-f_1)}(T_j^b) + V_{(f_1\cap f_2)}(T_j^b)]$$

Using $V_{S_1}(S_2) = V_{S_2}(S_1)$,

$$RHS(1) = RHS(2+3) + 2V_{(f_1\cap f_2)}(T\backslash(T_j^a \cup T_j^b \cup (f_1 \cup f_2)))$$

$$RHS(1) - RHS(2+3) = 2V_{(f_1\cap f_2)}(T\backslash(T_j^a \cup T_j^b \cup (f_1 \cup f_2))) \geq 0$$

Thus, the inequality (1) is redundant. This means that any inequality in the set of inequality $L_{f_2}^b \leq R_{f_1}$ is redundant for each choice of f_1, f_2 and b.

This means that the initial system is consistent iff the following system is consistent :

$$(I_{(X,w_{ij})})$$

$$\underline{w}_{ij} \leq R_{f_1}\ \forall f_1 \subseteq F_j\,,\ j \in f_1$$

$$\bar{w}_{ij} \geq L_{f_2}^b\ \forall f_2 \subseteq F_j\,,\ j \in f_2 \text{ and } b = 1,...,\delta(x_j)\ b \neq a$$

We observe that this new set of inequalities is the initial set of inequalities with w_{ij} replaced with its lower-upper bound values. This means that we can eliminate any component of the vector W by replacing it with its lower-upper bounds conveniently. We can continue with the other components of the W vector to eliminate the whole W from the system except that we have to show that the order we eliminate variables does not affect the procedure. This is true, since the differences $LHS(1) - LHS(2+3)$ and $RHS(2+3) - RHS(1)$ are always nonpositive, no matter which value the involved variables are set, as long as they are within their bounds. For the common terms of (1) and $(2+3)$, that do not appear in the lefthandside or righthandside differences, we see that if any variable has been set to its upper (lower) bound in (1) it will be set to its upper (lower) bound in (2 + 3). After eliminating all the components of the W vector, we can say that the initial system is consistent iff :

(I'_X)

$$\bar{W}_f(T\backslash T_i^j) - \underline{W}_f(T_i^j) + V_f(T\backslash(T_i^j \cup f)) - V_f(T_i^j) \geq 0$$

$$\forall f \subseteq F_i$$

$$\forall\, T_i^j,\ j = 1, ..., \delta(x_i)$$

$$\forall i = 1, ..., m$$

$$\underline{V} \leq V \leq \bar{V}$$

which is what we wanted to prove. □

We now give a Corollary to Theorem 2.1 :

Corollary 2.1 *Suppose given a candidate solution $X = (x_1, x_2, ..., x_m)$. For any (j,k) such that $x_j = x_k$ we can replace the variable v_{jk} in (I'_X) by $\bar{v}_{jk}$. The resulting system is consistent iff the original system is consistent.*

Proof : We observe that if $x_j = x_k$ the inequalities in (I'_X) involving the variable v_{jk} are of the form :

$$v_{jk} \geq L_f^b = [\underline{W}_f(T_j^b) - \bar{W}_f(T\backslash T_j^b)] - [V_f(T_j^b) - (V_f(T\backslash T_j^b) - v_{jk})]$$

(for all $f \subseteq F_j$ such that $j \in f$ or $k \in f$ but not both, $\forall\, b = 1, ..., \delta(x_j)$)

$$v_{jk} \geq \underline{v}_{jk}$$

$$v_{jk} \leq \bar{v}_{jk}$$

Here, the term $V_f(T\backslash T_j^b) - v_{jk} = V_{(f\backslash\{j,k\})}(T\backslash T_j^b) - V_{(f\cap\{j,k\})}(T\backslash(T_j^b \cup \{j,k\}))$

Taking all these as bounding inequalities as in the proof of Theorem 2.1, we see that the whole system is consistent iff the system

$(I_{(X,v_{jk})})$

$$\bar{v}_{jk} \geq L_{f_j}\ ,\ \text{for all possible}\ f,\ \forall\, b = 1, ..., \delta(x_j)$$

is consistent. The resulting system corresponds to replacing v_{jk} in (I'_X) by $\bar{v}_{jk}$. □

So, we have converted the problem of identifying the membership of a given candidate solution to the weak set to a linear feasibility problem that involves the V vector only. The number of inequalities in the system is $\sum_j \delta(x_j)(2^{|F_j|} - 1)$, which is polynomial in the number of inequalities which are located on distinct vertices, and exponential in the number of co-located new facilities with respect to the given solution vector X. In the next section, we propose an approximation for candidate solutions with co-located facilities, so as to reduce the number of inequalities to $O(n)$.

2.1.1 ε Perturbation

In the problem of identifying whether or not a given candidate solution X is a weak solution, we have observed that there are exponentially many inequalities if some new facilities are located on the same vertex. To overcome this computational difficulty, we propose working with a 'perturbed' candidate solution instead of the original X vector.

Given the m-vector $X = (x_1, x_2, x_3, ..., x_m)$, we construct the perturbed vector X' as follows :

If all x_i are distinct vertices, then $X' = X$.

Otherwise, for each $F_q = \{k : x_k = x_q\}$ with $x_q = v_i$,
let $i_1, ..., i_k$ be an enumeration of indices in F_q with $i_1 = q,\ k = \mid F_q \mid$ and

Create distinct dummy vertices $v_{i_2}, v_{i_3}, ..., v_{i_k}$, such that
$0 < d(v_i, v_{i_j}) = \varepsilon_j < \varepsilon,\ \forall\, j = 2, ..., k$ for some small enough positive ε.

Assign $\bar{w}_{\cdot j} = 0$ for the new vertices and

$x'_{i_1} = v_i \quad , x'_{i_j} = v_{i_j} \quad j = 2, ..., k.$

The perturbed vector X' has all new facilities located on different vertices, so testing whether X' is a weak solution requires solving a feasibility problem involving $0(n)$ inequalities (constraints).

However, we have changed the tree structure by adding extra vertices. We argue now that this change in the structure does not affect our conclusions, and discuss the validity of the approximation by using X' instead of X.

First thing to note here is that adding new vertices to the tree with zero weights is just equivalent to considering some of the interior points as vertices of the tree and this does not change the topology of the tree. So ε perturbation is equivalent to moving all but one of the co-located facilities to distinct interior points of the tree.

Next, we show that $F(X,W,V)-F(X',W,V) = O(\varepsilon)$. That is $F(X,W,V)-F(X',W,V)$ goes to zero as ε goes to zero. The implication of this observation is that, if X' is optimal for some choice of weights, we can say that X is ε -optimal (or very close to optimality) for small enough ε. So, for small enough ε , X can be considered as a weak solution if X' is a weak solution.

Assume that we have moved the new facilities $(i_2, i_3, ..., i_k)$ from v_i to $(v_{i_2}, v_{i_3}, ..., v_{i_k})$

Let T_i^r denote the subtree rooted at v_i, containing v_{i_r}, and f_r denote the set of new facilities moved to T_i^r.

$$F(X,W,V) - F(X',W,V) =$$

$$\sum_r \Big(\sum_{j:v_j \in T_i^r} \sum_{s \in f_r} w_{js} d(v_i, v_{i_s}) - \sum_{j:v_j \in T \setminus T_i^r} \sum_{s \in f_r} w_{js} d(v_i, v_{i_s}) +$$

$$\sum_{x_t \in (T_i^r - f_r)} \sum_{s \in f_r} v_{st} d(v_i, v_{i_s}) - \sum_{t \in f_r} \sum_{s \in f_r} v_{st} d(v_{i_s}, v_{i_t}) - \sum_{x_t \in T \setminus (T_i^r \cup f_r)} \sum_{s \in f_r} v_{st} d(v_i, v_{i_s}) \Big)$$

Using the fact that $d(v_i, v_{i_s}) = \varepsilon_s \leq \varepsilon$, by construction,

$$F(X,W,V) - F(X',W,V) \leq$$

$$\varepsilon \Big(\sum_r (W_{f_r}(T_i^r) - W_{f_r}(T \setminus T_i^r) + V_{f_r}(T_i^r) - V_{f_r}(f_r) - V_{T \setminus (T_i^r \cup f_r)}(f_r)) \Big)$$

Or simplifying,

$$F(X,W,V)-F(X',W,V) \leq \varepsilon\,(\sum_{r}(W_{f_r}(T_i^r)-W_{f_r}(T\backslash T_i^r)+V_{f_r}(T_i^r)-V_{f_r}(T\backslash T_i^r)))$$

Thus, $F(X,W,V) - F(X',W,V)$ goes to zero as ε goes to zero.

We note here that the perturbed vector X' is not unique. We can construct different X' vectors by choosing different relative locations and orderings of the new facilities in F_j with respect to each other. For each such X', we know that X can be considered as a weak solution if X' is a weak solution. Then we have some kind of flexibility in that, we can assume that X is a weak solution if any of the systems $(I'_{X'})$ admits a feasible solution.

2.1.2 Special Cases

1.LN_B is a tree

We recall that LN_B is an auxiliary network that represents the interaction between pairs of new facilities. Each new facility NF_j is represented by a node N_j ; there is an undirected arc between N_j and N_k if $\bar{v}_{jk} > 0$.

Now, we show that, if LN_B has a tree structure, then the problem of identifying the membership of any given candidate solution X to the Weak Set can be solved easily in a recursive way.

Using Theorem 2.1, we can say that $X = (x_1, x_2, x_3, ..., x_m)$ is a weak solution iff the system

$$(I'_X) \qquad A_1\bar{W} + B_1V \geq A_2\underline{W} + B_2V$$

$$\underline{V} \leq V \leq \bar{V}$$

is consistent. The matrices A_1, A_2, B_1, B_2 are as defined in Section 2.1

Lemma 2.1 *Let N_t be a tip vertex of LN_B, with the unique adjacent vertex N_s. Any inequality involving v_{ts} in the block related to NF_t is either redundant or corresponds to a lower or upper bound inequality on v_{ts}.*

Proof : In (I'_X), each variable v_{jk} appears in the block related to NF_j and also in the block related to NF_k.

Since N_t is a tip vertex with the unique adjacent vertex N_s, $\bar{v}_{ts} > 0$ while $\bar{v}_{ti} = 0$, $\forall\, i \neq s$.

Assuming without loss of generality that $x_t = v_t$, if $x_s = x_t$, we know by Corollary 2.1 that we can set $v_{ts} = \bar{v}_{ts}$ and eliminate v_{ts} from the system.

Assume $x_s \neq x_t$. Consider any inequality in the block related to NF_t, containing v_{ts}. Such an inequality is identified by a subset $F \subseteq F_t$ such that $t \in F$, and a subtree T_t^a to which we move F.

If $F = \{t\}$, then the inequality will be of the form :

$$V_t(T_t^a) - V_t(T \backslash T_t^a) \leq -\underline{W}_t(T_t^a) + \bar{W}_t(T \backslash T_t^a)$$

As $v_{ti} = 0$ if $i \neq s$, the lefthandside of the above inequality will be just v_{ts} if $x_s \in T_t^a$, and -v_{ts} if $x_s \notin T_t^a$. Also, we know that the righthandside is a constant.

That means, if $F = \{t\}$ then the related inequalities only give lower or upper bounds on v_{ts}.

If $x_s \in T_t^s$, we will have $v_{ts} \leq c_s$ and $v_{ts} \geq -c_i$, $i \neq s$, where $c_k = -\underline{W}_t(T_t^k) + \bar{W}_t(T \backslash T_t^k)$, $k = 1, ..., \delta(x_t)$

Now, assume that $F \supset \{t\}$. Denoting $F' = F - \{t\}$, the inequality will be of the form :

$$V_F(T_t^a) - V_F(T \backslash (T_t^a \cup F)) \leq -\underline{W}_F(T_t^a) + \bar{W}_F(T \backslash T_t^a) \quad (1)$$

which is equivalent to

$$V_{F'}(T_t^a) + V_t(T_t^a) - V_{F'}(T \backslash (T_t^a \cup F')) + V_{F'}(t) - V_t(T \backslash (T_t^a \cup F)) \leq$$

$$-\bar{W}_{F'}(T_t^a) - \underline{W}_t(T_t^a) + \bar{W}_{F_t}(T\backslash T_t^a) + \bar{W}_t(T\backslash T_t^a)$$

Collecting similar terms, we have

$$[V_{F'}(T_t^a) - V_{F'}(T\backslash(T_t^a \cup F'))] + [V_t(T_t^a) + V_{F'}(t) - V_t(T\backslash(T_t^a \cup F))] \leq$$

$$[-\underline{W}_{F'}(T_t^a) + \bar{W}_{F'}(T\backslash T_t^a)] + [-\underline{W}_t(T_t^a) + \bar{W}_t(T\backslash T_t^a)] \qquad (1')$$

Considering moving F' to T_t^a we have

$$V_{F'}(T_t^a) - V_{F'}(T\backslash(T_t^a \cup F')) \leq -\underline{W}_{F'}(T_t^a) + \bar{W}_{F'}(T\backslash T_t^a) \quad (2)$$

And moving t to T_t^a we have

$$V_t(T_t^a) - V_t(T\backslash T_t^a) \leq -\underline{W}_t(T_t^a) + \bar{W}_t(T\backslash T_t^a) \quad (3)$$

Summing (2) and (3), we get

$$V_{F'}(T_t^a) - V_{F'}(T\backslash(T_t^a \cup F')) + V_t(T_t^a) - V_t(T\backslash T_t^a) \leq$$

$$-\underline{W}_{F'}(T_t^a) + \bar{W}_{F'}(T\backslash T_t^a) - \underline{W}_t(T_t^a) + \bar{W}_t(T\backslash T_t^a) \qquad (4)$$

Now, we see that (4) equivalent to $(1')$ because righthandsides of (4) and $(1')$ are the same, and lefthandside of (4) is always equal to lefthandside of $(1')$, since

(LHS(4) - LHS(1)) =

$$[-V_t(T\backslash T_t^a)] - [V_{F'}(t) - V_t(T\backslash(T_t^a \cup F))]$$

We know that $v_{ti} = 0$ if $i \neq s$. Now if $s \in T_t^a$ (LHS(4) - LHS(1)) $= -0 - (0 - 0) = 0$. Otherwise, if $s \in T\backslash T_t^a$, then (LHS(4) - LHS(1)) $= -v_{ts} - (0 - v_{ts}) = -v_{ts} + v_{ts} = 0$

So, the inequality $(1')$ is redundant.

Any inequality involving v_{ts} in the block related to NF_t is either redundant or corresponds to a lower or upper bound inequality on v_{ts}. □

Theorem 2.2 *Let N_t be a tip vertex of LN_B with the unique adjacent vertex N_s. We can eliminate $[N_t, N_s)$ from LN_B by setting :*

$$\underline{v}_{ts} \leftarrow max\{c_m, \underline{v}_{ts}\}\ ,\ \bar{v}_{ts} \leftarrow min\{c_s, \bar{v}_{ts}\}$$

and then

$$\underline{w}_{a_t s} \leftarrow \underline{w}_{a_t s} + \underline{v}_{ts} \quad , \bar{w}_{a_t s} \leftarrow \bar{w}_{a_t s} + \bar{v}_{ts}$$

where $c_m = max\{-c_i, 1 \leq i \leq \delta(x_t), i \neq s\}$

Proof : We have seen by Lemma 2.1 that any inequality involving v_{ts} in the block related to NF_t is either redundant or corresponds to a lower or upper bound inequality on v_{ts}.

We can now take $c_m = max\{-c_i, 1 \leq i \leq \delta(x_t), i \neq s\} \leq v_{ts} \leq c_s$ that come from the inequalities involving v_{ts} in the block related to NF_t with the original lower-upper bound inequalities $\underline{v}_{ts} \leq v_{ts} \leq \bar{v}_{ts}$ to obtain

$$max\{c_m, \underline{v}_{ts}\} \leq v_{ts} \leq min\{c_s, \bar{v}_{ts}\}.$$

To stay within the format of (I'_X), we can set

$$\underline{v}_{ts} \leftarrow max\{c_m, \underline{v}_{ts}\} \;,\; \bar{v}_{ts} \leftarrow min\{c_s, \bar{v}_{ts}\}$$

and obtain revised lower and upper bounds for v_{ts}.

Now, we are left with v_{ts} appearing in only the (possibly revised) lower-upper bound inequalities and in the block related to new facility s.

Since in the block related to NF_s the variables w_{a_t} and v_{ts} always appear with the same sign, we can eliminate v_{ts} from the block related to new facility s by setting :

$$\underline{w}_{a_t s} \leftarrow \underline{w}_{a_t s} + \underline{v}_{ts} \quad , \bar{w}_{a_t s} \leftarrow \bar{w}_{a_t s} + \bar{v}_{ts}$$

Replacing v_{ts} in our system of inequalities (I'_X) with lower-upper bound values corresponds to the elimination of the tip vertex N_t and the edge $[N_t, N_s)$ from LN_B. $\square$

This theorem shows that we can always eliminate tip vertices from LN_B. In the special case when LN_B has a tree structure, this result allows us to recursively solve the feasibility problem defined by (I'_X) and the bounding inequalities.

If LN_B is a tree, each time we delete a tip vertex and the unique edge incident to it, we are again left with a tree with one less number of vertices and one less number of arcs, which correspond to a new set of inequalities that has the same structure. We can continue trimming tip vertices of LN_B until we are left with a single edge and two vertices. At this stage, we can immediately conclude whether the system is consistent or not. This is done by checking whether the last remaining v_{jk} variable turns out to have a nonempty interval of lower-upper bounds.

Now, we present the algorithmic statement of the procedure for solving the feasibility problem to determine whether a given X is a weak solution or not, when LN_B has a tree structure.

Let T_t^i denote the subtree rooted at x_t, containing x_i,

1. Pick a tip vertex N_t of LN_B. Let N_s be the unique vertex adjacent of N_t.

2. If $x_t = x_s$, set $v_{ts} = \bar{v}_{ts}$. Go to 9

Else ,

3. Delete from the system the inequalities corresponding to moving any F such that, $F \supset \{t\}$ to any adjacent vertex.

4. Compute $c_k = -\underline{W}_t(T_t^k) + \bar{W}_t(T \backslash T_t^k)$, $k = 1, ..., \delta(x_t)$

5. Compute $c_m = max\{-c_i, 1 \leq i \leq \delta(x_t), i \neq s\}$

6. Set $\underline{v}_{ts} \leftarrow max\{c_m, \underline{v}_{ts}\}$ and $\bar{v}_{ts} \leftarrow min\{\bar{v}_{ts}, c_s\}$

7. If $\underline{v}_{ts} > \bar{v}_{ts}$ Stop. The system is infeasible.

Else,

8. Set $\underline{w}_{a_t s} \leftarrow \underline{w}_{a_t s} + \underline{v}_{ts}$ and $\bar{w}_{a_t s} \leftarrow \bar{w}_{a_t s} + \bar{v}_{ts}$

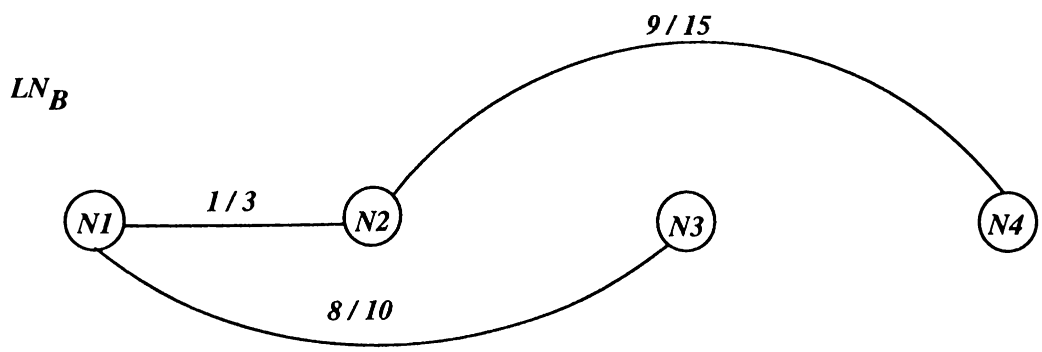
9. Delete $[N_t, N_s)$ from LN_B. If $LN_B = \{N_s\}$, the system is feasible iff the

remaining inequalities all hold. Stop.

Else $LN_B \supset \{N_s\}$. Goto 1.

The following example will help demonstrate the ideas of this section.

Example :(n = 7,m = 4)



Consider the tree structure of the example on page 16 and LN_B given as above. The other related data is as follows : (Each row represents the weight relations ([lower,upper]) of a new facility with the 7 vertices.)

$$NF1: \quad [3,3]\,,[5,5]\,,[4,4]\,,[2,6]\,,[1,2]\,,[2,3]\,,[3,5]$$

$$NF2: \quad [2,4]\,,[2,6]\,,[4,5]\,,[3,5]\,,[4,6]\,,[3,5]\,,[1,3]$$

$$NF3: \quad [6,18]\,,[1,4]\,,[2,3]\,,[0,1]\,,[1,4]\,,[1,4]\,,[0,2]$$

$$NF4: \quad [1,2]\,,[1,3]\,,[2,4]\,,[1,4]\,,[2,4]\,,[1,4]\,,[9,19]$$

Assume now that we want to identify whether $X = (v_4, v_4, v_1, v_7)$ is a Weak Solution. X is a Weak Solution iff the following system is consistent :

$$-v_{12} + v_{13} \quad \leq -\underline{w}_{11} - \underline{w}_{21} - \underline{w}_{31} + \bar{w}_{41} + \bar{w}_{51} + \bar{w}_{61} + \bar{w}_{71} = 4$$

$$-v_{12} - v_{13} \quad \leq \bar{w}_{11} + \bar{w}_{21} + \bar{w}_{31} + \bar{w}_{41} - \underline{w}_{51} - \underline{w}_{61} - \underline{w}_{71} = 12$$

$$-v_{12} \quad - v_{24} \leq -\underline{w}_{12} - \underline{w}_{22} - \underline{w}_{32} + \bar{w}_{42} + \bar{w}_{52} + \bar{w}_{62} + \bar{w}_{72} = 11$$

$$-v_{12} \quad + v_{24} \quad \leq \bar{w}_{12} + \bar{w}_{22} + \bar{w}_{32} + \bar{w}_{42} - \underline{w}_{52} - \underline{w}_{62} - \underline{w}_{72} = 12$$

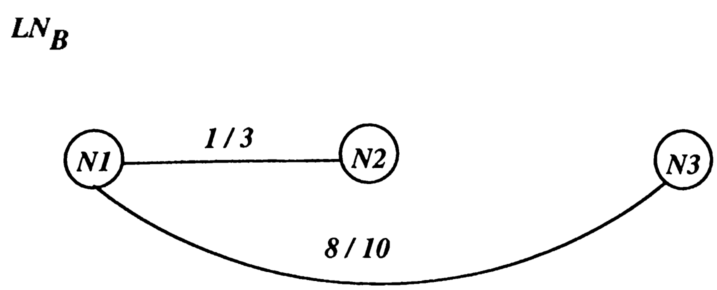
$$v_{13}\text{-}v_{24} \leq (\text{-}\underline{w}_{11}\text{-}\underline{w}_{21}\text{-}\underline{w}_{31}+\bar{w}_{41}+\bar{w}_{51}+\bar{w}_{61}+\bar{w}_{71})+(\text{-}\underline{w}_{12}\text{-}\underline{w}_{22}\text{-}\underline{w}_{32}+\bar{w}_{42}+\bar{w}_{52}+\bar{w}_{62}+\bar{w}_{72}) = 15$$

$$-v_{13}+v_{24} \leq (\bar{w}_{11}+\bar{w}_{21}+\bar{w}_{31}+\bar{w}_{41}-\underline{w}_{51}-\underline{w}_{61}-\underline{w}_{71})+(\bar{w}_{12}+\bar{w}_{22}+\bar{w}_{32}+\bar{w}_{42}-\underline{w}_{52}-\underline{w}_{62}-\underline{w}_{72}) = 24$$

$$v_{13} \leq \bar{w}_{13} - \underline{w}_{23} - \underline{w}_{33} - \underline{w}_{43} - \underline{w}_{53} - \underline{w}_{63} - \underline{w}_{73} = 13$$

$$v_{24} \leq -\underline{w}_{14} - \underline{w}_{24} - \underline{w}_{34} - \underline{w}_{44} - \underline{w}_{54} - \underline{w}_{64} + \bar{w}_{74} = 11$$

We start with the tip N_4. Taking $v_{24} \leq 11$ with $9 \leq v_{24} \leq 15$ gives $9 \leq v_{24} \leq 11$. We delete the last inequality from the system and update the inequalities related to new facility 2. The new LN_B is :



With the set of inequalities :

$$-v_{12} + v_{13} \leq 4$$

$$-v_{12} - v_{13} \leq 12$$

$$-v_{12} \leq 11 + 11 = 22$$

$$-v_{12} \leq 12 - 9 = 3$$

$$v_{13} \leq 15 + 11 = 26$$

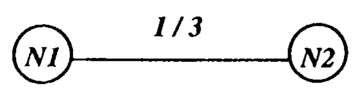
$$-v_{13} \leq 24 - 9 = 15$$

$$v_{13} \leq 13$$

Now, we take the tip N_3. Taking $-15 \leq v_{13} \leq 13$ with $8 \leq v_{13} \leq 10$ gives $8 \leq v_{13} \leq 10$. We delete the last three inequalities from the system and update the inequalities related to new facility 1.

The new LN_B is :

LN_B



With the set of inequalities :

$$-v_{12} \leq 4 - 8 = -4$$

$$-v_{12} \leq 12 + 10 = 22$$

$$-v_{12} \leq 22$$

$$-v_{12} \leq 3$$

Which give $v_{12} \geq 4$. Taking this with $1 \leq v_{12} \leq 3$, we get $4 \leq v_{12} \leq 3$, whereby we observe that the system is inconsistent because $\underline{v}_{12} = 4 > \bar{v}_{12} = 3$. So, X is not a weak solution.

2. Location Space is a Line

In this subsection, we show that, when the location space is a line, the problem of identification of a given candidate solution to the weak set can be expressed as a network flow feasibility problem.

Before further discussing this special case, we present the observations that establish the correctness of viewing the problem on the line with rectilinear distances as a special case of the problem on a tree network.

Given the line and the locations of the existing facilities, we can restrict our attention to the convex hull of the existing facilities, because for any set of weights, we know that there is an optimal solution where each new facility location is in the convex hull of new facilities (see Francis et. al. [18]) . Now, we can view this line segment as a tree consisting of a simple path only, and the tree distances are equal to the rectilinear distances. This establishes the tree representation of the line problem.

Throughout this subsection, we assume that we are given candidate solutions for which each new facility is located on a distinct vertex, or otherwise we deal with the ε - perturbed vector of the given candidate solution. That is, we consider candidate solutions of the form $X = (x_1, x_2, x_3, ..., x_m)$ where $x_i = x_j$ only if $i = j$.

We also assume that the vertices in the tree are renumbered such that one of the tip vertices gets the label v_1, and (v_i, v_{i+1}) is an edge of the tree for $i = 1, ..., n-1$

For notational convenience, we add the two vertices v_o , v_{n+1} , and the two edges (v_o, v_1) , (v_n, v_{n+1})

Given such a candidate solution $X = (v_{a_1}, v_{a_2}, ..., v_{a_m})$, we can say using Theorem 2.1 that, X is a Weak solution iff the following system is consistent :

$$\sum_{a_k<a_j} v_{jk} - \sum_{a_k>a_j} v_{jk} \leq - \sum_{i<a_k} \underline{w}_{ij} + \sum_{i\geq a_k} \bar{w}_{ij} \ , \quad j = 1, ..., m$$

$$\sum_{a_k>a_j} v_{jk} - \sum_{a_k<a_j} v_{jk} \leq - \sum_{i>a_k} \underline{w}_{ij} + \sum_{i\leq a_k} \bar{w}_{ij} \ , \quad j = 1, ..., m$$

$$\underline{v}_{jk} \leq v_{jk} \leq \bar{v}_{jk} \ , \ 1 \leq j < k \leq m$$

Since $\delta(x_j) = 2$ and $| F_j |= 1, \ \forall j \quad (F_j = \{j\})$.

The first set of inequalities correspond to moving x_j to the left, the second set corresponds to moving x_j to the right.

We observe that the lefthandside of the *r-th* inequality in the first set is the negative of the lefthandside of the *r-th* inequality in the second set. Using this, we have

$$\sum_{i>a_k} \underline{w}_{ij} - \sum_{i\leq a_k} \bar{w}_{ij} \leq \sum_{a_k<a_j} v_{jk} - \sum_{a_k>a_j} v_{jk} \leq - \sum_{i<a_k} \underline{w}_{ij} + \sum_{i\geq a_k} \bar{w}_{ij} \ , \quad j = 1, ..., m$$

$$\underline{v}_{jk} \leq v_{jk} \leq \bar{v}_{jk} \ , \ 1 \leq j < k \leq m$$

Calling the constants

$$\sum_{i>a_k} \underline{w}_{ij} - \sum_{i\leq a_k} \bar{w}_{ij} = \alpha_j^l \text{ and}$$

$$-\sum_{i<a_k} \underline{w}_{ij} + \sum_{i\geq a_k} \bar{w}_{ij} = \alpha_j^u ,$$

X is a Weak Solution iff

$$\alpha_j^l \leq \sum_{a_k<a_j} v_{jk} - \sum_{a_k>a_j} v_{jk} \leq \alpha_j^u , \; j = 1, ..., m$$

$$\underline{v}_{jk} \leq v_{jk} \leq \bar{v}_{jk} \; , \; 1 \leq j < k \leq m$$

If we define $y_j = \sum_{a_k<a_j} v_{jk} - \sum_{a_k>a_j} v_{jk}$, the system becomes

$$\sum_{a_k<a_j} v_{jk} - \sum_{a_k>a_j} v_{jk} - y_j = 0 \; , \; j = 1, ..., m$$

$$\alpha_j^l \leq y_j \leq \alpha_j^u , \; j = 1, ..., m$$

$$\underline{v}_{jk} \leq v_{jk} \leq \bar{v}_{jk} \; , \; 1 \leq j < k \leq m$$

Observe that the first set of inequalities above define the node-edge incidence matrix of a graph, with each variable v_{jk} appearing twice, once in the *j-th* inequality and once in the *k-th* inequality.

If $x_k < x_j$ then the coefficient of v_{jk} is $+1$ in the *j-th* inequality and -1 in the *k-th* inequality. Otherwise if $x_k > x_j$, v_{jk} has coefficient -1 in the *j-th* inequality and $+1$ in the *k-th* inequality.

Each y_j variable appears only once in the *j-th* inequality, with a coefficient of -1.

If we consider the y_j and v_{jk} as the flow values by taking the (α_j^l, α_j^u) and $(\underline{v}_{jk}, \bar{v}_{jk})$ as the (lower , upper) bounds on the variables y_j and v_{jk} respectively, and if we add the constraint :

$$\sum_{j=1}^{m} y_j = 0 ,$$

our system is almost a network flow feasibility problem except that some of the y_j variables can be negative if $\alpha_j^l < 0$. To convert this to a regular network flow feasibility problem with nonnegative flow variables, we do the following change of variables :

$$\hat{y}_j = y_j - \alpha_j^l, \ \ j = 1, ..., m$$

With the change of variable, the new (equivalent) system is as follows :

$$\sum_{a_k < a_j} v_{jk} - \sum_{a_k > a_j} v_{jk} - \hat{y}_j = \alpha_j^l, \ j = 1, ..., m$$

$$\sum_{j=1}^{m} \hat{y}_j = -\sum_{j=1}^{m} \alpha_j^l$$

$$0 \leq \hat{y}_j \leq (\alpha_j^u - \alpha_j^l) \ , \ j = 1, ..., m$$

$$\underline{v}_{jk} \leq v_{jk} \leq \bar{v}_{jk} \ , \ \ 1 \leq j < k \leq m$$

The network flow feasibility problem can be solved by any of the existing methods (See for example, Kennington and Helgason [21]).

We now present an example to demonstrate the construction of the flow conservation type of problem and clarify the discussion in this section.

Example : (n = 6,m = 4)

Assume that we are given the following line. We want to test whether $X = (v_2, v_1, v_6, v_4)$ is a Weak Solution.

Assume that we have used the given bound values and computed :

$$\alpha_1^l = -2, \ \alpha_1^u = 3, \ \alpha_2^l = 3, \ \alpha_2^u = 6, \ \alpha_3^l = -3, \ \alpha_3^u = 3, \ \alpha_4^l = -5, \ \alpha_4^u = -3$$

with given $2 \leq v_{12} \leq 4, \ 0 \leq v_{13} \leq 5, \ 6 \leq v_{14} \leq 8, \ 1 \leq v_{23} \leq 3, \ 4 \leq v_{24} \leq 4, \ 2 \leq v_{34} \leq 5$

Now, we know that X is a Weak Solution iff the following system is consistent :

$$v_{12} - v_{13} - v_{14} - \hat{y}_1 = -2$$

$$-v_{12} - v_{23} - v_{24} - \hat{y}_2 = 3$$

$$v_{13} + v_{23} + v_{34} - \hat{y}_3 = -3$$

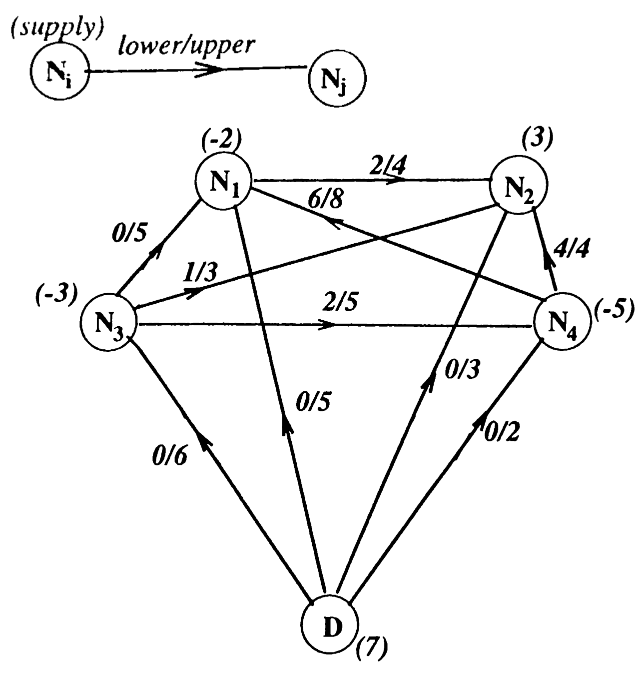
$$v_{14} + v_{24} - v_{34} - \hat{y}_4 = -5$$

$$\hat{y}_1 + \hat{y}_2 + \hat{y}_3 + \hat{y}_4 = 7$$

$$0 \leq \hat{y}_1 \leq 5,\ 0 \leq \hat{y}_2 \leq 3,\ 0 \leq \hat{y}_3 \leq 6,\ 0 \leq \hat{y}_4 \leq 2$$

$$2 \leq v_{12} \leq 4,\ 0 \leq v_{13} \leq 5,\ 6 \leq v_{14} \leq 8,\ 1 \leq v_{23} \leq 3,\ 4 \leq v_{24} \leq 4,\ 2 \leq v_{34} \leq 5$$

which defines the feasible flow problem on the following network :



2.1.3 Construction of the Weak Set for $m = 2$

For the case when there are 2 new facilities to be located, we present an $O(n^2)$ algorithm for constructing the weak set.

The algorithm makes use of the $O(n)$ 1-median tree trimming algorithm proposed by Tansel and Scheuenstuhl [33]. First we define that algorithm.

1-median Weak Solutions Algorithm :

The algorithm uses the facts that the weak set is convex for the 1-median problem and that a tip v_i of the tree is a weak solution for the 1-median problem iff

$$\bar{w}_i \geq \underline{W}(T \backslash \{v_i\})$$

So, we start with a tip vertex of the tree, and test whether it is a weak solution.

If the answer is 'No', then the tip vertex and the unique edge incident to it is deleted from the tree, by adding the bounds on the weight of the tip vertex to the corresponding bounds of the weight of the unique adjacent vertex. The procedure continues with the remaining tree and the new bounds on weights. If the answer is 'Yes', then the tip vertex is marked as 'ineligible' and the procedure continues with another tip vertex.

We stop when all the tip vertices are marked as 'ineligible'. The remaining tree is the weak set.

In our case, we have 2 new facilities to locate. We know that testing whether a given $X = (x_1, x_2)$ is a weak solution is a lower-upper bounded linear feasibility problem, involving the V vector only, which is the scalar v_{12} for $m = 2$.

Actually, testing whether a given $X = (x_1, x_2)$ is a weak solution is equivalent to testing whether a set of inequalities in v_{12} has a feasible solution within its lower-upper bounds.

Now assume that we fix the location of new facility 1 as v_i. Consider the subtrees rooted at v_i, T_i^j, $j = 1, ..., \delta(v_i)$.

Lemma 2.2 *Identifying the members of the Weak Set such that* $(x_1, x_2) = (v_i, x)$ *where* $x \in T_i^j$ *can be done by applying the 1-median Weak Solutions Algorithm with a revised set of weights.*

Proof :

For x_1 fixed (at v_i) and x_2 in the subtree T_i^j (that is (v_i, x) such that $x \in T_i^j$)

to be a Weak Solution, we require that the following system is consistent :

$$v_{12} \leq -\underline{W}_1(T_i^j) + \bar{W}_1(T \backslash T_i^j) = c_j$$

$$-v_{12} \leq -\underline{W}_1(T_i^k) + \bar{W}_1(T \backslash T_i^k) = c_k \,,\; k = 1, ..., \delta(v_i) \,,\; k \neq j$$

related with new facility 1, and

$$v_{12} \leq -\underline{W}_2(T_x^i) + \bar{W}_2(T \backslash T_x^i)$$

$$-v_{12} \leq -\underline{W}_2(T_x^k) + \bar{W}_2(T \backslash T_x^k) \,;\; k = 1, ..., \delta(x) \,,\; k \neq i$$

related with new facility 2 and the bounding inequalities

$$\underline{v}_{12} \leq v_{12} \leq \bar{v}_{12}$$

As $x_1 = v_i$ is fixed, and the subtree rooted at v_i that contains $x_2 = x$ does not change, the inequalities in the block related to new facility 1 are the same for all $x \in T_i^j$. Then, we can view these inequalities as lower-upper bound type inequalities for v_{12}.

That is, we can set

$$\underline{v}_{12} \leftarrow max\{-c_k, 1 \leq k \leq \delta(v_i) \,,\; k \neq j \;, \underline{v}_{12}\}, \bar{v}_{12} \leftarrow min\{c_j, \,\bar{v}_{12}\}$$

to get revised bounds for v_{12}.

Now, adding v_{12} on w_{i2}, that is updating

$$\underline{w}_{i2} \leftarrow \underline{w}_{i2} + \underline{v}_{12} \;,\; \bar{w}_{i2} \leftarrow \bar{w}_{i2} + \bar{v}_{12}$$

the conditions for (v_i, x) for fixed v_i and $x \in T_i^j$ being a Weak Solution become

$$0 \leq -\underline{W}_2(T_x^i) + \bar{W}_2(T \backslash T_x^i)$$

$$0 \leq -\underline{W}_2(T_x^k) + \bar{W}_2(T \backslash T_x^k) \,,\; k = 1, ..., \delta(x) \,,\; k \neq i$$

$$\underline{v}_{12} \leq v_{12} \leq \bar{v}_{12}$$

which can be expressed as :

$$0 \le -\underline{W}_2(T_x^k) + \bar{W}_2(T \backslash T_x^k)\,,\ k = 1, ..., \delta(x)$$

$$\underline{v}_{12} \le v_{12} \le \bar{v}_{12}$$

Finally, we observe that the above are just the conditions for x being a weak solution with the 1-median problem defined by the intervals $w_i \in [\underline{w}_{i2}, \bar{w}_{i2}]$ so that we can use the $O(n)$ 1-median Weak Solutions Algorithm to find $x \in T_i^j$ such that $(v_i, x) \in$ weak set. □

This Lemma tells us that we can repeat this procedure for all v_i and all subtrees of each v_i, we can construct the weak set with elements (v_i, x).

Now, we have :

Corollary 2.2 *The vertex elements of the weak set (x_1, x_2) such that $x_1 \neq x_2$ can be constructed in $O(n^2)$ time.*

Proof : We just observe that for each fixed v_i and each subtree rooted at v_i, we apply the $O(n)$ 1-median Weak Solutions Algorithm once. So, we apply the algorithm as many times as the total number of subtrees rooted at the vertices of the tree (the degrees of the vertices), which is $2(n-1)$. So, the overall order is $O(n^2)$. □

Lemma 2.3 *The vertex elements of the weak set (x_1, x_2) such that $x_1 = x_2 = v_i$ can be constructed in $O(n)$ time.*

For $(x_1, x_2) = (v_i, v_i)$, the conditions for being a weak solution are :

$$-\bar{v}_{12} \le -\underline{W}_1(T_i^k) + \bar{W}_1(T \backslash T_i^k)\ \ \forall\ k = 1, ..., \delta(v_i)$$

$$-\bar{v}_{12} \le -\underline{W}_2(T_i^k) + \bar{W}_2(T \backslash T_i^k)\ \ \forall\ k = 1, ..., \delta(v_i)$$

$$0 \le [-\underline{W}_1(T_i^k) + \bar{W}_1(T \backslash T_i^k)] + [-\underline{W}_2(T_i^k) + \bar{W}_2(T \backslash T_i^k)]\ \ \forall\ k = 1, ..., \delta(v_i)$$

To construct the elements of the weak set such that $x_1 = x_2 = v_i$ for some v_i, we will check the three sets of inequalities separately, one set of constraints at a time. (x, x) is a weak solution iff x satisfies all three sets of constraints.

For the first and second set of constraints, we use a derivative of the 1-median Weak Solutions algorithm described above. Checking the third set of constraints we see that they are the conditions for v_i being a weak solution for the 1-median problem defined by the weights $w_i \in [\underline{w}_{i1} + \underline{w}_2, \bar{w}_{i1} + \bar{w}_{i2}]$.

We now describe how we test the first or second sets of constraints by the Modified 1-median Weak Solutions Algorithm. The constraints are of the type :

$$-\bar{v}_{12} \leq -\underline{W}(T_i^k) + \bar{W}(T \backslash T_i^k) \ \forall \ k = 1, ..., \delta(v_i)$$

Modified 1-median Weak Solutions Algorithm :

We start with a tip vertex v_t where the conditions become

$$\underline{W}(T \backslash v_t) \leq \bar{w}_t + \bar{v}_{12}$$

If the answer is 'No', then the tip vertex and the unique edge incident to it is deleted from the tree, by adding the weights of the tip vertex to the weights of the unique adjacent vertex, the procedure continues with the remaining tree and the new weights. If the answer is 'Yes', then the tip vertex is marked as 'ineligible' and the procedure continues with another tip vertex.

We stop when all the tip vertices are marked as 'ineligible' or the tree is empty. The remaining tree is the set that satisfies the condition.

The algorithm requires one comparison at each call, the main step is repeated at most $(n - 1)$ times. Thus, the order is $O(n)$

Any vertex v_i that passes all the three tests identifies (v_i, v_i) as a weak solution.

So, we can identify the elements of the weak set such that $x_1 = x_2 = v_i$ by applying the $O(n)$ Modified 1-median Weak Solutions Algorithm two times, and the $O(n)$ 1-median Weak Solutions Algorithm once, with an overall order of $O(n)$. □

Before giving the statement of the procedure for constructing the weak set, we give an argument on the correctness of the Modified 1-median Weak Solutions Algorithm.

Lemma 2.4 *The Modified 1-median Weak Solutions Algorithm determines correctly the vertices that satisfy the condition :*

$$-\bar{v}_{12} \leq -\underline{W}(T_i^k) + \bar{W}(T \backslash T_i^k) \ \forall \ k = 1, \ldots, \delta(v_i)$$

Proof : Whenever the condition fails for a tip vertex v_t (of the current tree), that is, $\underline{W}(T \backslash v_t) > \bar{w}_t + \bar{v}_{12}$, we observe that, for the subtree rooted at the unique adjacent vertex v_s, containing v_t, the condition $\underline{W}(T_s^t) \leq \bar{W}(T \backslash T_s^t) + \bar{v}_{12}$ is satisfied since :

$$\underline{W}(T_s^t) = \underline{w}_t \leq \bar{w}_t + \bar{v}_{12} \text{ and } \bar{W}(T \backslash T_s^t) + \bar{v}_{12} = \underline{W}(T \backslash v_t) + \bar{v}_{12} \geq \underline{W}(T \backslash v_t)$$

So, we need not check the condition for T_s^t. Also, v_t and v_s are in the same subtree with respect to all other vertices of the current tree (and with respect to other subtrees rooted at v_s). These facts justify the deletion of $[v_t, v_s)$ from the tree and updating $[\underline{w}_s, \bar{w}_s] = [\underline{w}_s + \underline{w}_t, \bar{w}_s + \bar{w}_t]$.

Now, assuming that the tip vertices v_i, v_j satisfy the condition, that is $\underline{W}(T \backslash v_i) \leq \bar{w}_i + \bar{v}_{12}$ (1) and $\underline{W}(T \backslash v_j) \leq \bar{w}_j + \bar{v}_{12}$ (2), consider some $v_t \in P(v_i, v_j)$. The conditions for v_t are :

$$\underline{W}(T_t^i) \leq \bar{W}(T \backslash T_t^i) + \bar{v}_{12} \quad (1')$$

$$\underline{W}(T_t^j) \leq \bar{W}(T \backslash T_t^j) + \bar{v}_{12} \quad (2')$$

$$\underline{W}(T_t^k) \leq \bar{W}(T \backslash T_t^k) + \bar{v}_{12}, \ k = 1, \ldots, \delta(v_t), k \neq i, j \quad (3')$$

(1′) is implied by (2), since $\{v_j\} \subseteq T \backslash T_t^i$, (2′) is implied by (1), since $\{v_i\} \subseteq T \backslash T_t^j$ and (3′) is implied by (1) or (2), since $\{v_i, v_j\} \subseteq T \backslash T_t^k, \ k = 1, \ldots, \delta(v_t), k \neq i, j$.

This establishes the correctness of the conclusion that the vertices on the path between the tips of the remaining tree also satisfy the condition, thus the algorithm is correct. □

Now, we state the 2-median Weak Solutions Algorithm :

0. i = 0, j = 1 , Weak Set = $\emptyset$

1. For each subtree T_i^k rooted at v_i compute

$$c_k = -\underline{W}_1(T_i^k) + \bar{W}_1(T \backslash T_i^k)$$

2. Set $\underline{v}_{12} \leftarrow max\{-c_k, 1 \leq k \leq \delta(v_i)\, ,\, k \neq j\, , \underline{v}_{12}\}$, $\bar{v}_{12} \leftarrow min\{c_j,\, \bar{v}_{12}\}$

3. Set $\underline{w}_{i2} \leftarrow \underline{w}_{i2} + \underline{v}_{12}$, $\bar{w}_{i2} \leftarrow \bar{w}_{i2} + \bar{v}_{12}$

4. Apply the 1-median Weak Solutions Algorithm, with the weights :

$$w_s \in [\underline{w}_{s2}, \bar{w}_{s2}] \;\; s = 1, ..., n$$

Call the resulting set W_x. Let $W_{x_j} = \{(v_i, x) : x \in W_x \cap T_i^j\}$.

5. Set Weak Set $\leftarrow$ Weak Set $\cup\, W_{x_j}$

6. If $j < \delta(v_i)$, set $j \leftarrow j + 1$. Goto 2.

Else,

7. If $i < n$, set $i \leftarrow i + 1$. Goto 1.

Else,

8. Apply the Modified 1-median Weak Solutions Algorithm, with the weights :

$$w_s \in [\underline{w}_{s1}, \bar{w}_{s1}] \;\; s = 1, ..., n \text{ and } \bar{v}_{12}$$

Call the resulting set W_1

9. Apply the Modified 1-median Weak Solutions Algorithm with the weights :

$$w_s \in [\underline{w}_{s2}, \bar{w}_{s2}] \;\; s = 1, ..., n \text{ and } \bar{v}_{12}$$

Call the resulting set W_2

10. Apply the 1-median Weak Solutions Algorithm with the weights :

$$w_s \in [\underline{w}_{s1} + \underline{w}_{s2}, \bar{w}_{s1} + \bar{w}_{s2}] \quad s = 1, ..., n$$

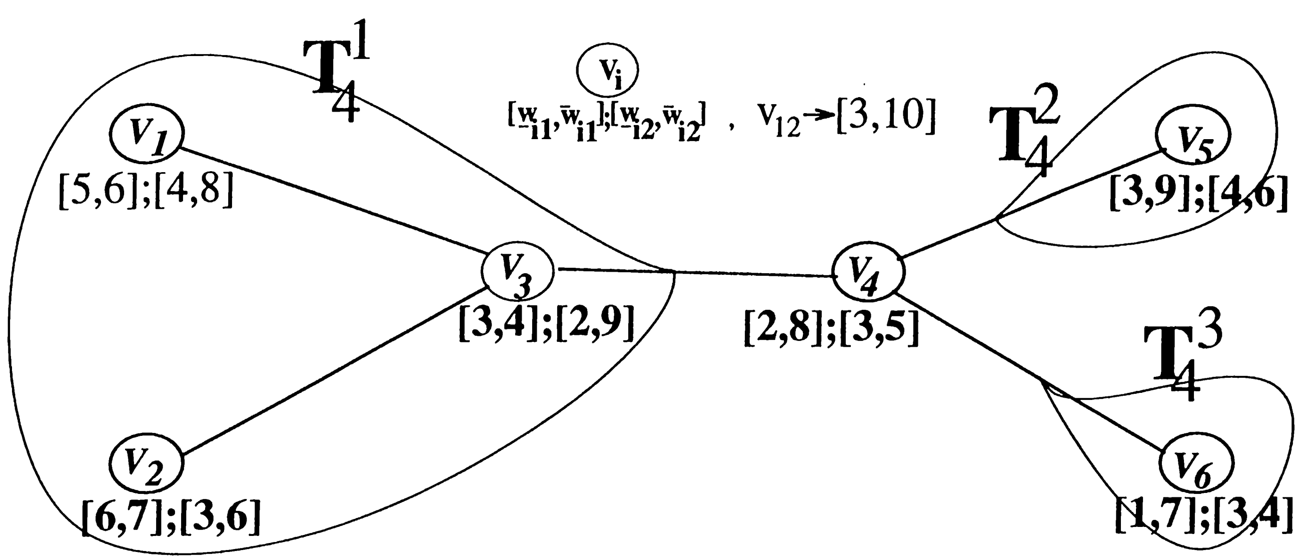
Call the resulting Weak Set W_3

11. Let $W_{int} = \{(v_s, v_s) : v_s \in \cap_{i=1}^{3} W_i\}$

12. Set Weak Set $\leftarrow$Weak Set $\cup\, W_{int}$

The following example will demonstrate the application of the 2-median Weak Solutions Algorithm.

Example : $(n = 6, m = 2)$



We first demonstrate how we identify the members of the Weak Set of the form (v_4, x), that is we fix $x_1 = v_4$

Computing $c_1 = -\underline{w}_{11} - \underline{w}_{21} - \underline{w}_{31} + \bar{w}_{41} + \bar{w}_{51} + \bar{w}_{61} = 10$ for T_4^1,

$c_2 = \bar{w}_{11} + \bar{w}_{21} + \bar{w}_{31} + \bar{w}_{41} - \underline{w}_{51} + \bar{w}_{61} = 29$ for T_4^2 and

$c_3 = \bar{w}_{11} + \bar{w}_{21} + \bar{w}_{31} + \bar{w}_{41} + \bar{w}_{51} - \underline{w}_{61} = 33$ for T_4^3

We start with T_4^1. Set $\underline{v}_{12} \leftarrow max\{-29,-33,3\} = 3$ $\bar{v}_{12} \leftarrow min\{10,10\} = 10$

Set $\underline{w}_{42} = 3+3 = 6$, $\bar{w}_{42} = 10+5 = 15$

Apply the 1-median Weak Solutions Algorithm with $w_i \in [\underline{w}_{i2}, \bar{w}_{i2}]$

Pick v_1. $\bar{w}_1 = 8 < \underline{W}(T\backslash v_1) = 18$. Delete $[v_1, v_3)$. Update $w_3 \in [4+2, 8+9]$

Pick v_2. $\bar{w}_2 = 6 < \underline{W}(T\backslash v_2) = 19$. Delete $[v_2, v_3)$. Update $w_3 \in [3+6, 6+17]$

Pick v_3. $\bar{w}_3 = 23 > \underline{W}(T\backslash v_3) = 13$. Mark v_3 ineligible.

As we have finished processing the members of T_4^1, we stop concluding that $(v_4, v_3) \in$ Weak Set

Now, we continue with T_4^2. Set $\underline{v}_{12} \leftarrow max\{-10,-33,3\} = 3$ $\bar{v}_{12} \leftarrow min\{29,10\} = 10$

Set $\underline{w}_{42} = 3+3 = 6$, $\bar{w}_{42} = 10+5 = 15$

Apply the 1-median Weak Solutions Algorithm with $w_i \in [\underline{w}_{i2}, \bar{w}_{i2}]$

Pick v_5. $\bar{w}_5 = 6 < \underline{W}(T\backslash v_5) = 18$. Delete $[v_5, v_4)$. Update $w_4 \in [4+6, 6+15]$

As we have finished processing the members of T_4^2, we stop. There is no $x \in T_4^2$ such that $(v_4, x) \in$ Weak Set.

Finally, for T_4^3, we set $\underline{v}_{12} \leftarrow max\{-10,-29,3\} = 3$ $\bar{v}_{12} \leftarrow min\{33,10\} = 10$

Set $\underline{w}_{42} = 3+3 = 6$, $\bar{w}_{42} = 10+5 = 15$

Apply the 1-median Weak Solutions Algorithm with $w_i \in [\underline{w}_{i2}, \bar{w}_{i2}]$

Pick v_6. $\bar{w}_6 = 4 < \underline{W}(T\backslash v_6) = 19$. Delete $[v_6, v_4)$. Update $w_4 \in [3+6, 4+15]$

As we have finished processing the members of T_4^3, we stop. There is no $x \in T_4^3$ such that $(v_4, x) \in$ Weak Set.

Having processed all the subtrees rooted at v_4, we conclude that the only element of the Weak Set such that $x_1 = v_4$ is (v_4, v_3).

Now we demonstrate how we determine the elements of the Weak Set such that $x_1 = x_2$.

We first apply the Modified 1-median Weak Solutions Algorithm with weights $w_i \in [\underline{w}_{i1}, \bar{w}_{i1}]$ and v_{12} :

Pick v_1. $\bar{w}_1 + \bar{v}_{12} = 6 + 10 > \underline{W}(T \backslash v_1) = 15$. Mark v_1 ineligible.

Pick v_2. $\bar{w}_2 + \bar{v}_{12} = 7 + 10 > \underline{W}(T \backslash v_2) = 14$. Mark v_2 ineligible.

Pick v_5. $w_5 + \bar{v}_{12} = 9 + 10 \geq \underline{W}(T \backslash v_5) = 17$. Mark v_5 ineligible.

Pick v_6. $\bar{w}_6 + \bar{v}_{12} = 7 + 10 < \underline{W}(T \backslash v_6) = 19$. Delete $[v_6, v_4)$. Update $w_4 \in [1 + 2, 7 + 8]$.

We stop since all tip vertices are ineligible. $W_1 = \{v_1, v_2, v_3, v_4, v_5\}$

Similarly, we apply the Modified 1-median Weak Solutions Algorithm with weights $w_i \in [\underline{w}_{i2}, \bar{w}_{i2}]$ and v_{12} :

Pick v_1. $\bar{w}_1 + \bar{v}_{12} = 8 + 10 > \underline{W}(T \backslash v_1) = 15$. Mark v_1 ineligible.

Pick v_2. $\bar{w}_2 + \bar{v}_{12} = 6 + 10 > \underline{W}(T \backslash v_2) = 16$. Mark v_2 ineligible.

Pick v_5. $\bar{w}_5 + \bar{v}_{12} = 6 + 10 > \underline{W}(T \backslash v_5) = 15$. Mark v_5 ineligible.

Pick v_6. $\bar{w}_6 + \bar{v}_{12} = 4 + 10 < \underline{W}(T \backslash v_6) = 16$. Delete $[v_6, v_4)$. Update $w_4 \in [3 + 3, 4 + 5]$

We stop since all tip vertices are ineligible. $W_2 = \{v_1, v_2, v_3, v_4, v_5\}$

Finally, we apply the 1-median Weak Solutions Algorithm with $w_i \in [\underline{w}_{i1} + \underline{w}_{i2}, \bar{w}_{i1} + \bar{w}_{i2}]$:

Pick v_1. $\bar{w}_1 = 14 < \underline{W}(T\backslash v_1) = 30$. Delete $[v_1, v_3)$. Update $w_3 \in [9+5, 14+13]$.

Pick v_2. $\bar{w}_2 = 13 < \underline{W}(T\backslash v_2) = 30$. Delete $[v_2, v_3)$. Update $w_3 \in [9+14, 13+27]$.

Pick v_3. $\bar{w}_3 = 40 > \underline{W}(T\backslash v_3) = 16$. Mark v_3 ineligible.

Pick v_5. $\bar{w}_5 = 15 < \underline{W}(T\backslash v_5) = 32$. Delete $[v_5, v_4)$. Update $w_4 \in [7+5, 15+13]$

Pick v_6. $\bar{w}_6 = 11 < \underline{W}(T\backslash v_5) = 35$. Delete $[v_6, v_4)$. Update $w_4 \in [4+12, 11+28]$

Pick v_4. $\bar{w}_4 = 39 > \underline{W}(T\backslash v_4) = 23$. Mark v_4 ineligible.

We stop since all tip vertices are ineligible. $W_3 = \{v_3, v_4\}$.Now $\cap_{i=1}^{3} W_i = \{v_3, v_4\}$. We conclude that the elements of the weak set such that $x_1 = x_2$ are $\{(v_3, v_3), (v_4, v_4)\}$

2.2 Permanent Solutions

In this section, we first present interesting results on the cardinality of the permanent set, for general networks and for tree networks, then we give an efficient method for the construction of the permanent set -or concluding that it is empty- for the tree case.

2.2.1 Cardinality of the Permanent Set

For the tree problem, we show that under reasonable assumptions, the permanent set is either empty or consists of a unique solution vector. We first state this result, the proof will follow the discussion in this section.

Theorem 2.3 *(Tree case) For each new facility $j \in J$, if there is at least one existing facility index r for which the corresponding weight interval is nondegenerate ($\bar{w}_{rj} \neq \underline{w}_{rj}$), then there is at most one permanent solution (index r need not be the same for each j).*

We first define the following notation :

$$F(X, W, V) = WD_1(X) + VD_2(X) = F(X, U) = UD(X).$$

Assume we order the elements of $U = (W, V)$ and of $D(X)$ and number them as $1, ..., t$. Recall that $t = mn + m(m-1)/2$. For a component u_s of U, we write $\bar{u}_s$ and $\underline{u}_s$ to mean the lower and upper bounds respectively on the weight w_{ij} or v_{jk} to which index s corresponds.

For two given solutions X^i and X^k, we let $\Delta(i, k) = D(X^i) - D(X^k)$ and $\delta_s = sth$ component of Δ.

We also define the index sets $I_{ik}^{+} = \{s : \delta_s > 0\}$, $I_{ik}^{-} = \{s : \delta_s < 0\}$ and $I_{ik}^{0} = \{s : \delta_s = 0\}$ (we will drop the subscripts ik when what we mean is clear from the context).

Now, we present a theorem on the cardinality of the permanent set for general networks :

Theorem 2.4 *For general networks, two solutions X^i and X^k are simultaneously permanent only if $\bar{u}_s = \underline{u}_s$ $\forall s \in I^{+} \cup I^{-}$.*

Proof : Suppose we have two permanent solutions X^i and X^k. Now, $F(X^i, U) \leq F(X^k, U)$ $\forall$ $U \in D$ since X^i is a permanent solution, and $F(X^k, U) \leq F(X^i, U)$ $\forall$ $U \in D$ since X^k is a permanent solution, which together imply that $F(X^i, U) = F(X^k, U)$ $\forall$ $U \in D$. That is :

$$\sum_{s \in I^{+}} u_s \delta_s + \sum_{s \in I^{-}} u_s \delta_s + \sum_{s \in I^{0}} u_s \delta_s = 0 \; \forall \, U \in D \text{ or,}$$

$\sum_{s\in I^+} u_s\delta_s + \sum_{s\in I^-} u_s\delta_s + \sum_{s\in I^0} u_s 0 = 0 \; \forall\, U$. Consider the three choices of $U \in D$

$$\sum_{s\in I^+} \bar{u}_s\delta_s + \sum_{s\in I^-} \underline{u}_s\delta_s = 0 \quad (1)$$

$$\sum_{s\in I^+} \underline{u}_s\delta_s + \sum_{s\in I^-} \underline{u}_s\delta_s = 0 \quad (2)$$

$$\sum_{s\in I^+} \bar{u}_s\delta_s + \sum_{s\in I^-} \bar{u}_s\delta_s = 0 \quad (3)$$

(1) and (2) imply $\sum_{s\in I^+} (\bar{u}_s - \underline{u}_s)\delta_s = 0$ which means $\bar{u}_s = \underline{u}_s \forall\, s \in I^+$

since $\delta_s > 0$ for such s.

(1) and (3) imply $\sum_{s\in I^-} (\bar{u}_s - \underline{u}_s)\delta_s = 0$ which means $\bar{u}_s = \underline{u}_s \forall\, s \in I^-$

since $\delta_s < 0$ for such s.
Thus, X^i and X^k are permanent solutions only if $\bar{u}_s = \underline{u}_s \; \forall s \in I^+ \cup I^-$. □

Before further discussion, we remark that the theorem applies to general networks. The conditions that it imposes are also the conditions for two or more solution vectors inducing the same objective function value for all choices of weights. We can also generalize this theorem to the case when the source set D is a finite set, which contains the weight vectors specified in (1),(2) and (3).

When we consider more than two permanent solutions $X^1, ..., X^p$ $(p > 2)$, we require by Theorem 2.4 that $\bar{u}_s = \underline{u}_s \; \forall s \in I^*$, where $I^* = \cup_{0\leq\alpha,\beta\leq p}(I^+_{\alpha\beta} \cup I^-_{\alpha\beta})$. Thus, we see that it is very unlikely to have more than one permanent solution for general networks, since this requires that we have many intervals that are degenerate.

Now, we continue with results on the cardinality of the permanent set for the tree problem. We first recall the observation in section 1.3 that the tree solution is independent of the distances; it is rather the topology of the tree (and the weight relations induced by this topology) that define the optimal set. Based on this observation, we can change the distances arbitrarily, without changing the topology of the tree, and obtain a new tree such that any solution is optimal for the problem defined on the original tree iff it is optimal for the problem defined on the new tree. We finally observe that the topology of the tree is preserved for all changes of edge lengths within the open interval $(0, \infty)$.

Lemma 2.5 *For the tree problem two solutions X^i and X^k are simultaneously permanent solutions only if one of the two conditions hold :*
a) All the intervals corresponding to the differing components of X^i and X^k are degenerate.
b) If X^i and X^k have differing components j and l such that $\bar{v}_{jl} \neq \underline{v}_{jl}$ for some (j,l), then $x^i_j = x^k_l$ and $x^i_l = x^k_j$.

Proof : Assume X^i and X^k are permanent solutions. We know by Theorem 2.4 that some interval s may be nondegenerate only if $s \in I^0$ (or equivalently $\delta_s = 0$). Now, there may be two cases for s being in I^0 :
Case i) s corresponds to a new facility and an existing facility, say x_j and v_r.
Case ii) s corresponds to some pair of new facilities (j,l).
Case i) : $\delta_s = d(x^i_j, v_r) = d(x^k_j, v_r) = 0$. If $x^i_j = x^k_j$, then there is nothing to prove. Assume $x^i_j \neq x^k_j$. Create $\hat{T}$ from T as follows : Pick some arc $(v_a, v_b) \in P(x^i_j, v_r) \backslash P(x^k_j, v_r)$. Increase length of (v_a, v_b) by, say ε such that $0 < \varepsilon < \min_{s \in I^+ \cup I^-} \mid \delta_s \mid$. This choice of ε guarantees that $q \in I^+$ with respect to $\hat{T}$ if $q \in I^+$ with respect to T, and $q \in I^-$ with respect to $\hat{T}$ if $q \in I^-$ with respect to T since the δ values may change less than the minimum δ.

Thus, with respect to $\hat{T}$, $\hat{I}^0 = I^0 - \{s\}$.
Observing that X^i and X^k are permanent solutions with respect to T iff they are permanent solutions with respect to $\hat{T}$, we can say using Theorem 2.4 for $\hat{T}$ that the interval corresponding to s should be degenerate.

Case ii) : $\delta_s = d(x^i_j, x^i_l) - d(x^k_j, x^k_l) = 0$

Let $P_1 = P(x^i_j, x^i_l)$ and $P_2 = P(x^k_j, x^k_l)$ be the two paths defined by locations of new facilities j and l in X^i and X^k, respectively.

If $P_1 = P_2$, then either $x^i_j = x^k_j$ and $x^i_l = x^k_l$, or $x^i_j = x^k_l$ and $x^i_l = x^k_j$. In the latter case, the interval $[\underline{v}_{jl}, \bar{v}_{jl}]$ may be nondegenerate.

If $P_1 \neq P_2$, we can create $\hat{T}$ similar to case $i)$ by picking some arc (v_a, v_b) in T that belongs to P_1 but not to P_2 and increasing the length of (v_a, v_b) by ε similarly, and we can conclude that the interval corresponding to s should be degenerate.

Thus, we conclude that either all the intervals corresponding to differing components of X^i and X^k are degenerate or there may be (j, l) such that v_{jl} has a nondegenerate interval only if $x_j^i = x_l^k$ and $x_l^i = x_j^k$. □

We observe that it is highly unlikely that two distinct solutions are simultaneously permanent. This requires that most of the intervals should be degenerate, or the two solution vectors should show structural similarities in that, either their components should be the same or 'switched' places. Next, we show that we require additional properties for the existence of more than one permanent solutions if we know that there exists at least one nondegenerate interval related with each new facility.

Corollary 2.3 *Suppose there exists at least one nondegenerate interval related with each new facility. Two solutions X^i and X^k are simultaneously permanent solutions only if for each $j \in J$ we have :*
Either i) $x_j^i = x_j^k$ or ii) there exists l such that $x_j^i = x_l^k$ and $x_l^i = x_j^k$

Proof : Suppose neither $i)$ nor $ii)$ hold. That is $x_j^i \neq x_j^k$ and there is no l such that j and l have 'switched' places in X^i and X^k. Then by Lemma 2.5, all intervals corresponding to new facility j should be degenerate. This is a contradiction since we know by assumption that there is at least one nondegenerate interval related to new facility j. □

Thus, we see that, for each $j \in J$, either $x_j^i = x_j^k$ or we have $l = p_j$ that we can 'pair' with j as they have switched places in X^i and X^k. We also recall from Lemma 2.5 that we may have only v_{jl}'s with nondegenerate intervals.

Finally, we have,

Corollary 2.4 *Given some $j \in J$, if we have an existing facility r such that $\bar{w}_{rj} \neq \underline{w}_{rj}$, then the location of j is the same in every permanent solution (whenever a permanent solution exists).*

Proof : Suppose we have two permanent solutions X^i and X^k with $\bar{w}_{rj} \neq \underline{w}_{rj}$ for some new facility r, and $x^i_j \neq x^k_j$.

Since $x^i_j \neq x^k_j$, we know by Lemma 2.5 that w_{rj} has a degenerate interval, that is $\bar{w}_{rj} = \underline{w}_{rj}$ which is a contradiction. Thus, we should have the location of new facility j fixed in every permanent solution. □

Now, we are in a position to prove Theorem 2.3 :

Theorem 2.3 *For each new facility $j \in J$, if there is an existing facility r such that $\bar{w}_{rj} \neq \underline{w}_{rj}$, then there is at most one permanent solution (index r need not be the same for each j).*

Proof : As the conclusion of Corollary 2.4 is true for each $j \in J$, the result follows. □

2.2.2 Construction of the Permanent Set

Like for the weak solutions, we use Theorem 1.2 about the optimality of a given candidate solution for the deterministic problem, as a starting point for our discussion in this section.

We recall that a given solution X is an optimal solution for $MMC(W, V)$ iff (I_X) is consistent for (W, V) , where, (I_X) is of the form :

$$A_1W + B_1V \geq A_2W + B_2V$$

From the viewpoint of permanent solutions, then, X is a permanent solution iff (I_X) is consistent for all $(W, V) \in D$. However the source set D has continuum number of elements and it is impossible to test whether X is a permanent solution by testing the optimality of X for every $d \in D$. Even using a subset of D would require a large amount of computational work and still give partial information.

We resolve this mentioned difficulty by particularly utilizing the connectedness property of the set D in the following theorem:

Theorem 2.5 *A given candidate solution X is a permanent solution iff :*

(IP_X)

$$A_1\underline{W} + B_1\underline{V} \geq A_2\bar{W} + B_2\bar{V}$$

Proof: (Necessity) Assume that there exists a permanent solution X for which (IP_X) does not hold. Then some rows of (IP_X) are violated. Let r be the index of the smallest violated row. This row will be defined by the change in the objective function resulting from moving a subset f_r of J moved to an adjacent vertex and the subtree T^a that contains the adjacent vertex. If the inequality is violated, then the objective function difference will be negative. That is :

$$\underline{W}_{f_r}(T\backslash T^a) + \underline{W}_{f_r}(T\backslash(T^a \cup f_r)) < \bar{W}_{f_r}(T^a) + V_{f_r}(T^a)$$

Now, consider the instance MMC($\tilde{W}, \tilde{V}$) defined by $(\tilde{W}, \tilde{V}) = \tilde{d} \in D$ as follows :

$$\tilde{w}_{ij} = \begin{cases} \underline{w}_{ij} & \text{if } i \notin T^a \text{ and } j \in f_r \\ \bar{w}_{ij} & \text{if } i \in T^a \text{ and } j \in f_r \\ \text{any value} \in E_{ij} & \text{otherwise} \end{cases}$$

$$\tilde{v}_{jk} = \begin{cases} \underline{v}_{jk} & \text{if } j \in f_r \text{ and } k \in T\backslash(T^a \cup f_r) \\ \bar{v}_{jk} & \text{if } j \in f_r \text{ and } k \in T^a \\ \text{any value} \in N_{jk} & \text{otherwise} \end{cases}$$

Clearly, $(\tilde{W}, \tilde{V}) \in D$ by construction. We know that (I_X) is not consistent for $(\tilde{W}, \tilde{V})$ because the *r-th* row is violated. Thus, X is not an optimal solution for $MMC(\tilde{W}, \tilde{V})$. This is a contradiction because X is a permanent solution and should be optimal for every $d \in D$.

Thus, X is a permanent solution only if (IP_X) is consistent. This establishes the necessity.

(Sufficiency)Assume that (IP_X) is consistent. Consider any $d = (\tilde{W}, \tilde{V}) \in D$.

Now,

$$A_1\tilde{W} + B_1\tilde{V} \geq A_1\underline{W} + B_1\underline{V}$$

and

$$A_2\bar{W} + B_2\bar{V} \geq A_2\tilde{W} + B_2\tilde{V} \quad (1)$$

as all components of A_1, A_2, B_1, B_2 are nonnegative and $(\underline{W}, \underline{V}) \leq (\tilde{W}, \tilde{V}) \leq (\bar{W}, \bar{V})$ for all $(\tilde{W}, \tilde{V}) \in D$

The consistency of (IP_X) gives: $A_1\underline{W} + B_1\underline{V} \geq A_2\bar{W} + B_2\bar{V}$

Together with (1) then, this means:

$A_1\tilde{W} + B_1\tilde{V} \geq A_2\tilde{W} + B_2\tilde{V}$

which is nothing but the consistency of (I_X) for $(\tilde{W}, \tilde{V})$.

So, we have shown that (I_X) is consistent for any $(\tilde{W}, \tilde{V}) \in D$, which implies that X is a permanent solution. □

This theorem gives us a very easy way of testing whether a given solution is a permanent solution or not. However, the construction of the permanent set may still require enumerating all the n^m candidate solutions and testing whether (IP_X) is consistent for each solution or not.

To overcome this difficulty, one may think of solving $MMC(W, V)$ for some $(W, V) \in D$, obtaining all solutions for that problem, and considering only those solutions as qualified candidates for the permanent set. This approach is valid simply because any solution that is not optimal for some realizable instance cannot be a permanent solution. The permanent set, then, consists of those qualified candidates for which (IP_X) is consistent - if any -

This approach may still require too much computation, especially when the number of alternate solutions for $MMC(W, V)$ is high. For an easier construction of the permanent set - or concluding that it is empty - , we propose the following algorithm :

0. $i = 1, X_{int} = \{X : X \text{ is a feasible solution}\}$

1. Solve $MMC(W, V)$ for some $(W_i, V_i) \in D$. Denote the set of all solutions by O^i

2. Set $X_{int} = X_{int} \cap O^i$

If X_{int} is small enough, Goto 3.

Else, $i = i + 1$, Goto 1

3. For every solution $X \in X_{int}$, construct (IP_X) to test whether X is a Permanent Solution.

4. The Permanent Set consists of those $X \in X_{int}$ for which (IP_X) is consistent.

2.3 Unionwise Permanent Solutions

Under the 'interval weights' scenario, the source set D is a hyperrectangle, i.e. the cartesian product of the intervals E_{ij}, N_{jk} in R^t where t is the total number of demand relations between new and existing facilities and between pairs of new facilities. In the general case, $t = mn + m(m-1)/2$.

As the 'volume' of the hyperrectangle D increases, that is, as the intervals E_{ij}, N_{jk} become larger, it becomes less likely that there exists a Permanent Solution for the problem. On the other hand, the size of the weak set is expected to increase, which means that it is both computationally more difficult to construct the whole weak set and more difficult to evaluate the elements of the weak set so as to choose from among.

It is mainly the large cardinality of the weak set that motivates the idea that we should look for some kind of criterion to be able to make a selection (elimination) among its elements. If we can 'select' a smaller subset of the weak set that possesses some desired properties, then this gives quite valuable information to the decision maker.

With this motivation, we look for a set $U = \{X_1, X_2, ..., X_p\}$ of candidate solutions such that U supplies an optimal solution for every $MMC(W, V)$ such that $(W, V) \in D$.

In a sense, we are looking for a set of candidate solutions that will 'cover' all the realizable problem instances. Investigating the similarity of finding Unionwise Permanent Solution to covering the problem instances further, we make some observations.

If a candidate solution X_i solves $MMC(W, V)$ for some realizable $d_i = (W_i, V_i)$ we say that X_i covers the point $d_i \in D$. Going one step further, we may 'expand' some components of d_i from points to subintervals such that X_i is still optimal as long as the weights remain in the subintervals. We will later see that we can perform this expansion one interval at a time, or simultaneously for a set of components. In this manner, we can think of a 'region of optimality' for some solution X_i that we know to be optimal for all d that belong to that region. That is, for every element X_i of the weak set, we can talk of a 'region of optimality' in R^t as a subset of the hyperrectangle D.

In these terms then, finding a Unionwise Permanent Solution is equivalent to identifying a subset of the weak set such that the union of the regions of optimality of the (weak) solutions in this subset is a superset of the hyperrectangle D. That is, the union of regions for optimality covers D.

Having explained one physical interpretation of what Unionwise Permanent Solutions refer to, we now point to the fact that how we can identify a Unionwise Permanent Solution is not so obvious. We require a set of candidate solutions, each having a nonempty 'region of optimality', such that the regions of optimality should be constructible, and the union of these regions should form a cover for D.

We would like to do this construction in a systematic way so that at any stage we are able to cover a larger cumulative 'volume' of D. It should be kept in mind that the decision maker will prefer the set U to have a smaller cardinality. It is natural that there may be other desirable objectives. We will propose ways of incorporating some additional objectives to our framework when possible.

With these motivations, we propose the following algorithm for finding a good Unionwise Permanent Solution:

0. $i = 1$, $U = \emptyset$

1. Pick some arbitrary $d_i \in D$, solve $MMC(D)$ to obtain an optimal solution X_i

2. Construct the region of optimality D_{X_i}, for X_i (FINDD)

3. Add X_i to U (i.e. Set $U \leftarrow U \cup \{X_i\}$)

Remove D_{X_i} from D (i.e. Set $D \leftarrow D - D_{X_i}$)

4. If $D \neq \emptyset$ then set $i \leftarrow i + 1$. Goto 1

Else Stop. U is a Unionwise Permanent Set.

Step 2. of the algorithm (FINDD) requires the computation of the region of optimality for a given candidate solution $\tilde{X}$. This is done by expanding around the components of $\tilde{d} \in D$.

That is, for each component $\tilde{w}_{ij}$ of $\tilde{d}$ we search for an interval $[\tilde{w}_{ij} - \delta^l_{ij}, \tilde{w}_{ij} + \delta^u_{ij}]$ (and similarly for each $\tilde{v}_{jk}$) such that for any choice of weights (W, V) taken from these intervals, $\tilde{X}$ will be an optimal solution. Note the similarity in that, we are trying to construct a 'source set' $D_{\tilde{X}}$ of weights, again in R^t, not necessarily a hyperrectangle, such that $\tilde{X}$ is 'Permanently Optimal' for $D_{\tilde{X}}$.

The subroutine (FINDD) utilizes this idea, by making some modifications to $(IP_{\tilde{X}})$ that was used for identifying the membership of $\tilde{X}$ to the Permanent Set.

We remark that, when we remove D_X form D, the residual D may become disconnected. In that case, D is a union of hyperrectangles which we may enumerate as $D^1, D^2, ..., D^p$, where each $D^i = \{(W,V) : \underline{W}^i \leq W \leq \bar{W}^i, \underline{V}^i \leq V \leq \bar{V}^i\}$.

SUBROUTINE (FINDD) :

0. Input ($\tilde{d} \in D, \tilde{X}$) - such that $\tilde{X}$ solves $MMC(\tilde{d})$ -

1. Construct $(IN_{\tilde{X}})$, which is $(IP_{\tilde{X}})$ with the intervals $\tilde{E}_{ij} = [\tilde{w}_{ij} - \delta^l_{ij}, \tilde{w}_{ij} + \delta^u_{ij}]$, $\tilde{N}_{jk} = [\tilde{v}_{jk} - \gamma^l_{ij}, \tilde{v}_{jk} + \gamma^u_{ij}]$

2. Let D^e be the maximal connected subset of current D such that $\tilde{d} \in D^e$. Note that D^e is a hyperrectangle (see the remark preceding the algorithm). Add the constraints :

$$0 \leq \delta^l_{ij} \leq (\tilde{w}_{ij} - \underline{w}^e_{ij})$$

$$0 \leq \delta^u_{ij} \leq (\bar{w}^e_{ij} - \tilde{w}_{ij})$$

$$0 \leq \gamma^l_{jk} \leq (\tilde{v}_{jk} - \underline{v}^e_{jk})$$

$$0 \leq \gamma^u_{jk} \leq (\bar{v}^e_{jk} - \tilde{v}_{jk})$$

to $(IN_{\tilde{X}})$.

3. Any choice of feasible solutions $(\Delta^u, \Delta^l, \Gamma^u, \Gamma^l)$ to the above system defines a region $D_{\tilde{X}} = \{(W,V) : (\tilde{W} - \Delta^l) \leq W \leq (\tilde{W} + \Delta^u), (\tilde{V} - \Gamma^l) \leq V \leq (\tilde{W} + \Gamma^u)\}$ for which $\tilde{X}$ is 'permanently optimal'.

It is seen from the statement of the subroutine that, any feasible solution to $(IN_{\tilde{X}})$ gives us an alternative for $D_{\tilde{X}}$. This may seem like a direction of flexibility on the one hand, but brings some kind of uncertainty to the procedure of finding a Unionwise Permanent Set on the other hand.

Really, we have freedom in that, it is the values of the Δ and Γ vectors that define the region which we choose, and we have many alternatives of these vectors to choose from. This means, we can reflect the preferences of the Decision Maker (or our preferences) at one more level in the process of finding a Unionwise Permanent Set. For instance, it may be important to be able to react to the changes in demand of a particular facility, or a subset of facilities. In this case, we may work towards that objective by choosing the corresponding δ_{ij} and γ_{jk} values as large as possible so as to cover as much as possible of those intervals with the same solution $\tilde{X}$. Similarly it may be desirable to react to changes in the demands of all the facilities in a balanced way. For this, we try to favor solutions where all the δ and γ variables are as close to each other as possible. Still, there are more and more possible preferences, we will return to this discussion later.

Now we make the observation that the system (IN_X) consists of a linear system of inequalities, as for given $\tilde{d}$, the quantities $\tilde{w}_{ij}, \tilde{v}_{jk}$ are constants defined by $\tilde{d} \in D$, the only variables are the Δ an d Γ vectors. This is fortunate because, if we are able to represent our objectives as linear functions of the Δ and Γ vectors, then we will be able to use linear programming as a tool for finding a 'region of optimality' with desired properties. This will guarantee us that we surely choose a nondominated D_X, because we choose an extreme point of the set of possible representations of D_X - feasible points of (IN_X), which extreme point we choose is an outcome of the objective at that step.

As an example, we may want to maximize the 'total length' of D_X in which case we have an LP of the form:

$$maximize \quad \sum_{i,j}(\delta_{ij}^u + \delta_{ij}^l) + \sum_{j,k}(\gamma_{jk}^u + \gamma_{jk}^l)$$

$$s.t. \quad (IN_X)$$

We may want to maximize the length of the 'minimum covered interval' :

$$maximize\ z$$

$$s.t. \quad z \geq \delta_{ij}^u + \delta_{ij}^l, \ \forall (i,j)$$

$$z \geq \gamma_{jk}^u + \gamma_{jk}^l, \ \forall (j,k)$$

$$(IN_X)$$

We may require to cover some intervals, say those in a given set C as much as possible :

$$maximize \sum_{(i,j)\in C} (\delta_{ij}^u + \delta_{ij}^l) + \sum_{(j,k)\in C} (\gamma_{jk}^u + \gamma_{jk}^l)$$

$$s.t. \quad (IN_X)$$

This approach completes the description of our algorithm for finding one 'good' Unionwise Permanent Solution. Summarizing, we form a Unionwise Permanent Solution such that we use each element in the set as efficiently as possible : We try to reflect preferences at each step, we choose a nondominated representation of each D_{X_i}. As D_{X_i} and D_{X_j} are disjoint (have no common area) for distinct i and j, we use each X_i efficiently by preventing same points of D to be covered by multiple candidate solutions. It may still be the case that X_i and X_j are simultaneously optimal for the same $d \in D$, but we choose D_{X_i} and D_{X_j} such that if D_{X_i} covers d, then the 'resource' X_j will be used to 'cover' points other than d. That is, the set of elements of D that are jointly covered by more than one member of U have zero measure (volume).

Note that, the algorithm is called a finite number of times, because at each call we add a candidate solution to the set U and the number of solutions added in the worst case is the total number of candidate solutions, that is n^m (in fact, if we know the weak set, then we can concentrate on the solutions in the weak set only). We also observe that, at each call of the algorithm, we solve one deterministic problem $MMC(d_i)$, we identify one point in the polyhedron D_{X_i} (FINDD). These can all be done in finite time. These facts establish the finite convergence of the algorithm.

We conclude this section with an algorithmic improvement idea for the algorithm for finding a Unionwise Permanent Solution.

If we start the algorithm with an arbitrary $d_1 \in D$ then it is possible that the removal of D_{X_1} from D will result in a residual D that is composed of two disjoint pieces. After several steps, we may have several pieces. The algorithm tries to cover each disjoint component with distinct vectors. This means, if we have more disjoint pieces, we are likely to cover them with a higher number of solution vectors, whereas we could possibly cover some of those jointly.

One idea that could be used to remedy this situation is to choose at each call of the algorithm, $d_t = (W_t, V_t)$ such that $w_{t_{ij}} = \underline{w}_{ij}$ or $\bar{w}_{ij}$, and $v_{t_{jk}} = \underline{v}_{jk}$ or $\bar{v}_{jk}$ so as to keep (residual) D connected throughout the algorithm.

Chapter 3

DISCRETE CASE

In many cases, we are faced with likely scenarios of the actual occurrences of the demand values. These scenarios may come out as a refinement of the infinite number of interval scenarios, or may result from system restrictions. Examples to such cases could be where the demanded product is produced and transferred in batches, or in the presence of quantity discounts where the client will prefer to demand at breakpoints of the quantity-price curve. Other than these, the different scenarios may be the individual estimations about the demand vectors of several decision makers, several departments etc.

When there are more than one decision makers, the alternative scenarios case is more likely to happen. Although a group of decision makers is unlikely to agree on some demand vector, it may be possible to convince them to limit attention to a set of possible demand vectors. In this set, one vector may represent the demand values for the worst case performance of the operating policy, one may represent the best case, the most likely cases, the case that takes into account some predicted trends, etc.

In the case of discrete scenarios, we assume that the source set D is given to us as a finite set of (W_i, V_i) vectors, thus, each $(W_i, V_i) \in D$ specifies a problem scenario $MMC(W_i, V_i)$.

The set D being a finite set gives us some kind of conceptual ease, because the problems that we have to deal with are finite in number. We can assume theoretically that we can explicitly solve all of them separately, and this will give us a good deal of information. Having these solutions in hand, if one can express the criteria explicitly, it is rather straightforward to make decisions. Unlike the case with interval weights, we do not have to deal with an infinite set and we do not have to develop ways of implicitly handling the problem of cardinality.

On the other hand, it is likely that computationwise we run into computational difficulties. If the location space is a tree, we have efficient polynomial time algorithms for the deterministic problem that we can use as tools for the problem with inexact data. The problem on the plane has also been studied and solution algorithms have been proposed. But the problem on general networks -as mentioned before- is proven to be *NP-Hard* [22].

Our main concern here is the introduction of the ideas and proposing a framework for decision making in the context of imprecise data

Recalling once more that our source set D is given as $D = \{(W_i, V_i)\ 1 \leq i \leq s\}$ with s being the number of possible scenarios, we move to our discussion of the Weak, Permanent and Unionwise Permanent Solutions.

3.1 Weak Solutions

By the definition of the weak set, we know that a candidate solution X is a weak solution iff X solves $MMC(W, V)$ for some $d = (W, V) \in D$.

When we are given the finite source set D of cardinality s, we can say that X is a weak solution iff X solves the problem defined by any one of the s scenarios.

For each scenario j, we define :

$$O^j = \text{The set of all optimal solutions for } MMC(W_j, V_j).$$

We note that computing O^j may not be very easy, even when the location space is a tree network, mainly because there may be too many alternative solutions for one deterministic problem, and finding all alternative solutions generally requires the testing of exponentially many inequalities.

Having computed O^j for each $j = 1, \ldots, s$, then

$$\text{Weak Set} = \cup_{j=1}^{s} O^j$$

The weak set is never empty and may have any cardinality between 1 and n^m (the number of candidate solutions), depending on the given data.

3.2 Permanent Solutions

A candidate solution is a permanent solution if it solves the MMC problem for every choice of data. For the problem with discrete scenarios, that means, X is a permanent solution if X solves $MMC(W_j, V_j) \, \forall j = 1, \ldots, s$. Again assuming that we can construct the sets O^j, $j = 1, \ldots, s$ the permanent set can be identified as :

$$\text{Permanent Set} = \cap_{j=1}^{s} O^j.$$

The permanent set may be empty. The construction (or concluding that it is empty), with this form requires a good deal of computation. This can be reduced to some extent by checking the intersection of the solution sets after computing each O^j rather than finding the intersection just at the end. That is :

0. $j = 1$. Permanent Set = Set of all candidate solutions.

1. Compute O^j

2. Permanent Set = Permanent Set $\cap O^j$

3. If Permanent Set = $\emptyset$ or $j = s$ then Goto 4.

Else $j = j + 1$. Goto 1

4. Stop. Output the Permanent Set.

3.3 Unionwise Permanent Solutions

We recall from the discussion for the continuous case that we could build a relation between the problem of identifying a Unionwise Permanent Solution and finding a cover of the source set D. We show in this section that this relation is still present in the case of discrete scenarios. Moreover the problem of finding a Unionwise Permanent Solution turns out to fit into the format of the well known 'set cover' problem (see Nemhauser and Wolsey [27]) for the discrete case.

To establish the transformation of the problem of finding a Unionwise Permanent Set to the set cover problem, we construct auxiliary vectors as follows :

Let the weak set consist of the solution vectors $\{X_1, ..., X_c\}$, where c is the cardinality of the weak set.

We construct the c by s matrix Z such that :

$$z_{ij} = \begin{cases} 1 & \text{if } X_i \in O^j \\ 0 & \text{otherwise} \end{cases}$$

Given the matrix Z, and denoting the j-th column of Z by Z^j, we define the region (SD) as :

$$(SD) = \{Y = (y_1, y_2, y_3, \ldots, y_c) : YZ^j \geq 1, \ \forall\, j = 1, \ldots, s \ , y_i \in \{0, 1\} \,\forall\, i\}$$

Now, observe that any point $(y_1, y_2, y_3, \ldots, y_c)$ in the region (SD) defines a Unionwise Permanent Set U such that $X_i \in U$ iff $y_i = 1$.

This is clear, as the constraints $YZ^j \geq 1, \forall j = 1, \ldots, s$ guarantee that at least one y_i variable is set to 1 for each scenario j (i.e. one optimal solution X_i is taken to U for each scenario j).

It is also seen that the region (SD) is the feasible region of the Set Cover problem defined by the matrix Z.

Thus, the problem of finding a Unionwise Permanent Set for the discrete case can be transformed into a set cover problem.

This idea facilitates the use of the integer programming techniques for the problem of identifying a Unionwise Permanent Set with 'desired' properties. For example, we can formulate the problem of finding a Unionwise Permanent Set of minimal cardinality as the following integer program :

$$minimize \sum_{i=1}^{c} y_i$$

$$s.t. \;\; YZ^j \geq 1 \;\; \forall j = 1, \ldots, s, \;\; y_i \in \{0, 1\} \, \forall \; i$$

We can think of adding weights (building new facilities as defined by X_i may require a cost of c_i), fixed costs, (building new facilities as defined by X_i may require a fixed cost of f_i), etc. to the model, and make use of the Integer Programming techniques for expressing these considerations.

As a final note for this section, we can say that the model may also be used with objectives like maximizing the number (weighted sum) of scenarios covered, with respect to a set of constraints such as a bound on the number solution vectors that can be used, or a predefined set of scenarios that have to be covered.

3.3.1 Minimizing Locational Variation

A Unionwise Permanent Set gives us a number of candidate solutions. If we were able to consider each particular candidate solution in the Unionwise Permanent Set, and establish facilities as induced by that candidate solution, we would have no fear of being suboptimal, whichever scenario is actually realized.

However, there is generally a cost involved with establishing one additional new facility of the same kind, or we may just assume that there is a fixed cost for establishing each new facility. In such cases, if the elements of the Unionwise Permanent Set induce all different locations for some new facility, then it would be very costly to build all of these facilities. In the other extreme, if the elements of the Unionwise Permanent Set induce the same location for a particular new facility, then we would locate the new facility on that location, and pay the related cost only once. This brings in the idea that one desired property for a Unionwise Permanent Set is to have a small locational variation of its elements.

With this motivation, then we may try to find a Unionwise Permanent Set, such that the total cost of locational variation is minimized. We assume that for each new facility, we establish as many replicas of that new facility as there are distinct locations induced for it by the elements of the Unionwise Permanent Set.

We further assume that for each particular new facility, we have a fixed cost of building (establishing) one facility.

The fixed cost for establishing one facility of type k will be considered as f_k.

Given the weak set $= \{X_k\,,\, k = 1, ..., c\}$, we assume that the matrix Z is constructed as defined in the previous section.

For each X_k we construct the n by m matrix L_k as follows :

$$l_{ij}^{k} = \begin{cases} 1 & \text{if } x_j = v_i \\ 0 & \text{otherwise} \end{cases}$$

Similarly with the previous section, assuming that $y_k = 1$ iff $X_k \in U$

Now, let R_{ij} denote the number of times the vertex v_i is induced as a location for new facility j by some element of the Unionwise Permanent Set. We can express R_{ij} as :

$$R_{ij} = \sum_{k=1}^{c} y_k l_{ij}^{k}$$

If the vertex v_i is induced as a location for new facility j by some element of the Unionwise Permanent Set, we will pay f_j to establish one facility j on v_i. To be able to express this cost, let In_{ij} be 1 if v_i is induced as a location for new facility j by some element of the Unionwise Permanent Set and 0 otherwise. This can be expressed by :

$$In_{ij} \geq (1/M)R_{ij} \ , \ In_{ij} \in \{0,1\}$$

Now, I_j, the number of facilities of type j that we have to establish is :

$$I_j = \sum_{i=1}^{n} In_{ij}$$

The integer program for finding a Unionwise Permanent Set that minimizes the total cost of locational variation can be given as :

$$minimize \quad \sum_{j=1}^{m} f_j I_j$$

$$s.t. \quad I_j = \sum_{i=1}^{n} In_{ij}, \quad j = 1, ..., m$$

$$In_{ij} \geq (1/M)R_{ij} \;\; i = 1, ..., n, \;, \;\; j = 1, ..., m$$

$$R_{ij} = \sum_{k=1}^{c} y_k l_{ij}^k, \;\; i = 1, ..., n \;, \;\; j = 1, ..., m$$

$$YZ^r \geq 1, \quad r = 1, ..., s$$

$$y_k \in \{0, 1\} \;, \;\; k = 1, ..., c$$

$$In_{ij} \in \{0, 1\}, \;\; i = 1, ..., n \;, \;\; j = 1, ..., m$$

The Unionwise Permanent Set consists of those X_k for which $y_k = 1$.

Chapter 4

CONCLUSION

In this thesis, we investigated the Multifacility Mutual Communication (MMC) problem where the demands of the new and existing facilities are inexact. We modeled this inexactness by the specification of a source set that contains all realizable values of the weights. The source set may be finite or infinite, and we do not assume any particular probability distribution on the elements of the source set. Thus, our approach is fundamentally different from the traditional probability based approaches in the literature.

We also note that although the technical developments are restricted to the MMC problem, the ideas, modeling and evaluation approaches that we try to motivate can be applied to other decision making situations.

We argue that it is easier to represent the data in terms of a source set rather than trying to assess point probabilities; and we introduce new criteria for evaluating alternative solutions to the problem. Our criteria can be used in conjunction with other criteria that the decision maker is provided with, and this brings a flexibility to the model. We also try to reflect the preferences of the decision maker at different stages of the decision making process and give examples of this approach (like in the unionwise permanent solutions).

We have two main directions for future research : To find more efficient ways for constructing particularly the weak and unionwise permanent sets, and second, to apply our approach and generalize our results to other problems that involve decision making under data uncertainty.

For more efficient construction of the weak set, we shall at the first stage try to identify structural properties like connectedness or convexity of the weak set. Such results may enable us to implicitly assess the membership of some solution vectors to the weak set without having to solve a feasibility problem. We shall also try to propose an efficient solution method for the solution of the feasibility problem to identify the membership of a given candidate solution to the weak set.

For the unionwise permanent sets, we will try to obtain a set of minimum cardinality in the continuous case by expressing this objective in an operational way.

There are also other criteria like minimax, minmax regret that we shall try to incorporate into our decision making framework.

Bibliography

[1] O. Berman. Locating a facility on a congested network with random lenghts. *Networks*, 15:275–294, 1985.

[2] M.L. Brandeau and S.S. Chiu. Establishing continuity of certain optimal parametric facility location trajectories. *Transportation Science*, 22:224–225, 1988.

[3] M.L. Brandeau and S.S. Chiu. Parametric facility location on a tree network with an l_p-norm cost function. *Transportation Science*, 22:59–69, 1988.

[4] M.L. Brandeau and S.S. Chiu. An overview of representative problems in location research. *Management Science*, 35:645–674, 1989.

[5] M.L. Brandeau and S.S. Chiu. Parametric analysis of optimal facility locations. *Networks*, 21:223–243, 1991.

[6] V. Cabot, R.L. Francis, and M.A. Stary. A network flow solution to a rectilinear distance facility location problem. *AIIE Transactions*, 2(2):132–141, 1970.

[7] D. Chajjed and T.J. Lowe. Solving structured multifacility location problems efficiently. Technical report, University of Illinois at Urbana-Champaign, October 1991.

[8] D. Chajjed and T.J. Lowe. M-median and m-center problems with mutual communication: Solvable special cases. *Operations Research*, 40:56–66, 1992.

[9] D. Chajjed and T.J. Lowe. An o(nm) algorithm for a special case of the multimedian location problem on a tree. *European Journal of Operational Research*, 63:222–230, 1992.

[10] T.Y. Cheung. Multifacility location problem with rectilinear distance by the minimum-cut approach. *ACM transactions on Math Software*, 6:549–561, 1980.

[11] G.B. Dantzig. *Linear programming and extensions.* Princeton University Press, 1963.

[12] P.M. Dearing, R.L. Francis, and T.J. Lowe. Convex location problems on tree networks. *Operations Research*, 24:628–642, 1976.

[13] P.M. Dearing and G.J. Langford. The multifacility total cost problem on a tree network. Technical report, Bell Laboratories, Holmdel, N.J., October 1975.

[14] E. Erkut, R.L. Francis, and T.J. Lowe. A multimedian problem with interdistance constraints. *Environment and Planning B: Planning and Design*, 15:181–190, 1988.

[15] E. Erkut and B.C. Tansel. On parametric medians of trees. *Transportation Science*, 26, 1992.

[16] D. Fernandez-Baca. Allocating modules to processors in a distributed system. *IEEE Trans. Software Eng.*, 15:1427–1436, 1989.

[17] R.L. Francis and T.J. Lowe. A polynomial algorithm for a tree network minisum location problem with distance constraints. Technical Report 86-2, University of Florida, April 1986.

[18] R.L. Francis, L.F. McGinnis, and J.A. White. *Facility layout and location: An analytical approach.* Prentice-Hall, 1992.

[19] H. Frank. Optimum locations on a graph with probabilistic demands. *Operations Research*, 14:409–421, 1966.

[20] H. Frank. Optimum locations on a graph with correlated normal demands. *Operations Research*, 14:552–557, 1967.

[21] J. Kennington and R.V. Helgason. *Algorithms for network programming.* Wiley, 1980.

[22] A.J.W. Kolen. *Location problems on trees and in the rectilinear plane.* PhD thesis, Stichting Mathematisch Centrum, Amsterdam, 1982.

[23] M. Labbe, D. Peeters, and J.F. Thisse. Location on networks. In *CORE Discussion Papers*, number 9340. 93.

[24] M. Labbe, J.F. Thisse, and R.E. Wendell. Sensitivity analysis in minisum facility location problems. *Operations Research*, 39:961–969, 1991.

[25] P.B. Mirchandani and A.R. Odoni. Locations of medians on stochastic networks. *Transportation Science*, 13:85–97, 1979.

[26] P.B. Mirchandani, A. Oudjit, and R.T. Wong. Multidimensional extensions and a nested dual approach for the m-medain problem. *European Journal of Operational Research*, 21:121–137, 1985.

[27] G.L. Nemhauser and L.A. Wolsey. *Integer and combinatorial optimization.* Wiley-Interscience, 1988.

[28] A. Oudjit. *Median locations on deterministic and probabilistic multidimensional networks.* PhD thesis, Rensellear Polytechnic Institute, Troy, NY, 1981.

[29] J. Picard and D. Ratliff. A cut approach to the rectilinear distance facility location problem. *Operations Research*, 26:422–433, 1978.

[30] A. Tamir. Complexity results for the p-median problem with mutual communication. Technical report, Tel Aviv University, Israel, May 1993.

[31] B.C. Tansel, R.L. Francis, and T.J. Lowe. Location on networks: A survey-part 1. the p-center and p-median problems. *Management Science*, 29:482–497, 1983a.

[32] B.C. Tansel, R.L. Francis, and T.J. Lowe. Location on networks: A survey-part 2. exploiting tree network structure. *Management Science*, 29:498–511, 1983b.

[33] B.C. Tansel and G. Scheuenstuhl. Facility location on tree networks with imprecise data. Technical Report IEOR-8819, Bilkent University, 1988.

[34] V.A. Trubin. Effective algorithm for the weber problem with a rectangular metric. *Cybernetics*, 14:874–878, 1978.

[35] Y. Xu, R.L. Francis, and T.J. Lowe. The multimedian location problem on a network: Exploiting block structure. Technical Report 88-11, University of Florida, December 1988.