MULTICOMMODITY NETWORK FLOW PROBLEM WITH SUBSTITUTION

Ekin Köker

Submitted to the Graduate School of Engineering and Natural Sciences
in partial fulfillment of the requirements for the degree of
Master of Science

Sabanci University

August, 2013

MULTICOMMODITY NETWORK FLOW PROBLEM WITH SUBSTITUTION

Approved by:

Assoc. Prof. Dr. Güvenç Şahin ....
(Thesis Supervisor)

Assoc. Prof. Dr. Tonguç Ünlüyurt ....

Assist. Prof. Dr. Murat Kaya ....

Assist. Prof. Dr. Kemal Kılıç ....

Assoc. Prof. Dr. Abdullah Daşcı ....

Date of Approval: 13/08/2013

to my family...

Acknowledgements

This thesis would not have been possible without the support of many people. The author wishes to express his gratitude towards his family, who has always been there for him, his supervisor Assoc. Prof. Dr. Güvenç Şahin, who inspired him in so many ways, his graduate friends, who shared the burden and made it easier for him and his professors at the Sabanci University, who has guided him through the journey.

İkameli Çok Ürünlü Ağ Akışı Problemi

Ekin Köker
Endüstri Mühendisliği, Yüksek Lisans Tezi, 2013
Tez Danışmanı: Güvenç Şahin

Anahtar Kelimeler: Çok Ürünlü Ağ Akışı Problemi, İkame Edilebilir Ürünler, Ünimodüler Matrisler, NP-zorluk.

Özet

Birden fazla ürünün ayrıt kapasiteleri gibi ortak kaynakları paylaştığı çok ürünlü ağ akış problemleri, tek ürünlü ağ akış problemlerinin genelleşmiş bir halidir. Tek ürünlü problemlerde ayrıtlar üzerindeki akış miktarları tam sayı olmaya zorlansa bile problem polinom zamanda çözülmesine karşın, problemin çok ürünlü ve ayrıt kapasiteli versiyonu NP-zor bir problemdir. Bu çalışmada çok ürünlü ağ akış probleminin ürünler arasında ikamenin mümkün olduğu daha da genelleşmiş bir halini tanımlıyoruz. İki veya üç ürünün yer aldığı, hem genel hem de ürüne özgü ayrıt kapasitelerin var olduğu problemlerin doğrusal tam sayılı programlama gösterimlerini matematiksel modeller olarak geliştiriyoruz. Kapasitesiz versiyonların matematiksel programlama gösterimlerindeki kısıt matrisinin tamamen ünimodüler olduğunu kanıtlıyoruz. Hipotez testi yöntemiyle rastgele yaratılan problemler üzerinden problem gösterimlerinin kapasiteli versiyonlarının deneysel hesaplama zorluğunu istatistiksel analiz yoluyla inceliyoruz. Kapasitelerin ve problem büyüklüğünün çözüm zamanına etkisini araştırıyoruz. Sonuçlarımız hem iki hem de üç ürünlü problemlerde hem genel hem de ürüne özgü kapasiteler probleme dahil edildiğinde çözüm zamanının önemli derecede arttığını gösteriyor. Problem boyutu büyüdükçe de çözüm zamanının arttığı ortaya çıkıyor. Son olarak iki ve üç ürünlü matematiksel modelleri çok ürünlü problem için genelleştiriyoruz.

MULTICOMMODITY NETWORK FLOW PROBLEM WITH SUBSTITUTION

Ekin Köker

Industrial Engineering, Master's Thesis, 2013

Thesis Supervisor: Assoc. Prof. Dr. Güvenç Şahin

Keywords: Multicommodity Network Flows, Substitutable Products, Unimodular Matrices, NP-hardness.

Abstract

Multicommodity network flow problems are generalizations of single commodity network flow problems, where a number of commodities flow through the network often sharing common resources such as arc capacities. While the single commodity problem can be solved in polynomial time even when the flow quantities are imposed as integer values only, the integer multicommodity version of the problem with arc capacities is NP-hard. We introduce a generalization of the multicommodity network flow problem where substitution is possible amongst commodities. We develop mathematical models as the linear integer programming formulations of two-commodity and three-commodity problems with both commodity-specific and overall arc capacities. We prove that constraint matrices are totally unimodular in the mathematical programming formulations for the uncapacitated versions. We investigate the empirical computational difficulty of capacitated versions of the problem formulations through a computational study with randomly generated problems and statistical analysis with hypothesis testing. In particular, we explore the effect of capacities and the problem size on solution time. Our results show that solution time significantly increases for both two-commodity and three-commodity problems when both overall and commodity-specific capacities exist. Solution time significantly increases when problem size is increased. Finally, we generalized the two and three-commodity models for the multicommodity problem.

Contents

1 Introduction **1**

2 Literature Review on Network Flow Problems **6**

3 Two-commodity Network Flow Formulation with Substitution **10**

3.1 Computational Complexity of the Capacitated Two-commodity Network Flow Problem 16

3.2 Computational Experiments 18

4 Three-commodity Network Flow Formulation with Substitution **27**

4.1 Non-transitivity in Substitution for Three-Commodity Network Flow Formulation 37

4.2 Computational Complexity of the Capacitated Three-commodity Network Flow Problem 38

4.3 Computational Experiments 41

5 Multicommodity Network Flow Formulation with Substitution **47**

6 Concluding Remarks **49**

List of Tables

3.1 Results of hypothesis testing for comparing the solution times of configuration with commodity-specific capacities and uncapacitated configuration 23
3.2 Results of hypothesis testing for comparing the solution times of configuration with overall capacities and uncapacitated configuration 23
3.3 Results of hypothesis testing for comparing the solution times of configuration with both commodity-specific and overall capacities and uncapacitated configuration 24
3.4 Results of hypothesis testing for comparing the solution times of configuration with overall capacities calculated as the sum of commodity-specific capacities and configuration with commodity-specific capacities 24
3.5 Results of hypothesis testing for comparing the solution times of levels for configuration with commodity-specific capacities 25
3.6 Results of hypothesis testing for comparing the solution times of levels for configuration with overall capacities 25
3.7 Results of hypothesis testing for comparing the solution times of problem sizes for uncapacitated configuration 26

4.1 Results of hypothesis testing for comparing the solution times of configuration with commodity-specific capacities and uncapacitated configuration 43
4.2 Results of hypothesis testing for comparing the solution times of configuration with overall capacities and uncapacitated configuration 43
4.3 Results of hypothesis testing for comparing the solution times of configuration with both commodity-specific and overall capacities and uncapacitated configuration 44
4.4 Results of hypothesis testing for comparing the solution times of configuration with overall capacities calculated as commodity-specific capacities and configuration with commodity-specific capacities 44

4.5 Results of hypothesis testing for comparing the solution times of levels for configuration with commodity-specific capacities 45
4.6 Results of hypothesis testing for comparing the solution times of levels for configuration with overall capacities 45
4.7 Results of hypothesis testing for comparing the solution times of problem sizes for uncapacitated configuration 46

Chapter 1

Introduction

Multicommodity network flow problems are generalizations of single commodity network flow problems, where a number of commodities flow through the network often sharing common resources such as arc capacities. The single commodity problem can be solved in polynomial time with the network simplex algorithm which is a specialized version of the simplex method for linear programming problems. Even in the integer version of the problem where the flow quantities may take integer values only, the solution can be obtained with the same method as the constraint matrix of the corresponding linear programming problem formulation is totally unimodular. Integrality conditions together with arc capacities make the problem NP-hard. In the well-known version of the multicommodity problem, demand and supply are independent amongst commodities yet they may share the common arc capacities of the network. In this study, we introduce a generalization of the multicommodity network flow problem in which substitution is possible amongst commodities.

In the literature, substitutable commodities are mostly considered in two problem environments: inventory planning and empty container allocation. Inventory planning context has many examples that consider substitutable commodities. Inventory planning issues and corresponding mathematical/analytical models are irrelevant within the boundaries of this study, but the current literature includes many real-life examples where substitution among commodities is possible.

Deflem and Nieuwenhuyse [8] provide an extensive and comprehensive review of the inventory planning research on substitutable commodities; we provide here a subset of their references in order to exemplify the real life examples where substitution among commodities is observed:

- Khouja et al. [14] discuss the grocery stores that sell a produce with two grades: a regular grade and a premium grade. Regular grade may substitute the premium grade since the premium grade is more expensive, it might be costly to store. Customers may prefer regular grade if they cannot find the premium grade. They also exemplify this case with fresh seafood markets; expensive seafood might be costly to keep in stock while cheap seafood might be a substitute for the expensive one. These examples may be considered as upward-substitution, since a lower-quality product is used to substitute an upper-quality product.

- An example of downward-substitution is provided by Bassok et al. [3]. In their paper, they discuss the semiconductor industry, and in particular, the integrated circuits. A circuit with higher performance characteristics (e.g. speed) can substitute a circuit with lower performance characteristics (hence, the downward substitution). Substituting the circuit with lower performance characteristics for the circuit with upper performance characteristics is not possible because if the circuit with lower performance characteristics is used to substitute the circuit with upper performance characteristics, it would not perform as required. A food substitution may not be very problematic for the customer but if the circuits do not perform as wanted, it would be a big problem for the companies or end-customers that use them in complicated computations. Bassok et al. [3] also discuss the usage of higher capacity memory chips to substitute for lower capacity memory chips. Last example by Bassok et al. [3] is in the steel industry; steel beams of a higher strength can substitute for beams of lesser strength.

- Hillier [10] discuss substitution of common products for unique products:
  - Universal power supplies can substitute regionally specific power supplies.
  - Multi-language manuals can substitute regionally-written manuals.
  - A single common microprocessor can substitute many differentiated unique microprocessors.
  - A single CD including Mac and IBM software can substitute both CDs that include Mac or IBM software.
  - An adjustable wrench can substitute various sizes of fixed wrench.

- Tibben and Bassok [20] discuss Benetton's dyeing system which can be considered as a substitution procedure. In this system, instead of dyeing the sweaters at the beginning of production, they send non-dyed fabrics to regions and then dye them there according to the final demand that depends on current fashion. By this logic, non-dyed fabrics are substitutes for the dyed fabrics.

- Bayindir et al. [4] provide the example of remanufactured products. They point out that remanufactured products can be substituted by the new ones. Remanufactured product examples include reconditioned photocopiers, retreated tires and reconditioned (upgraded) computers.

- Liu and Lee [17] focuses on power transformers. A transformer with higher capacity can be used instead of that with a lower capacity but not vice versa. This is also an example of downward substitution.

These are all inventory planning examples where substitution among commodities is possible.
Another problem environment that substitution is considered is empty container allocation:

- Crainic et al. [7] mention substitution rules for containers; for example, a 40-foot container may substitute two 20-foot containers but vice versa may not be possible if the load is longer than 20-foot.

- Ioannou et al. [12] also focus on substitutability based on size, i.e. the length of the containers, and formulate substitution problem as a transportation problem.

- Di Francesco et al. [9] consider a more general pattern where container types can substitute each other.

- Chang et al. [5] explicitly define the substitution rules for their problem context; these rules also depend on sizes of containers. Chang et al. [6] have the same problem basis with Chang et al. [5]; but, they expand their two-commodity model to a multicommodity model.

Container allocation problems are more relevant within the context of our study as empty containers are transported from one location to another which can also be casted using a network flow problem/model.
Another problem environment that does consider substitution but does not consider it within a network flow context is the energy systems planning. Substitution is mostly considered within a market-share framework in the energy sector. Some examples are:

- Kamimura et al. [13] consider natural gas as a substitute for traditional natural energy resources in Brazil.
- Aguliera and Ripple [1] analyse the market shares of gases, liquids and solids which can be substituted for each other as energy resources in the Asia Pacific.
- Xingang and Pingkuo [22] consider the biomass energy as a substitute for fossil fuel in China.
- Iniyan et al. [11] analyse the substitution of renewable energy sources for non-renewable energy sources in India.

A significant share of energy consumption is electricity production. Weidlich [21] note that real-world electricity markets have a wide heterogeneity in terms of the sources and resources used for production. This heterogeneity can be represented within a network flow context using different types of commodities. And although the characteristics of the sources differ, they can be substituted for each other since these characteristics do not matter for the end consumer. To exemplify the heterogeneity in electricity markets, Weidlich [21] points out that generator agents differ in size and spatial position, they own and operate different generating assets (e.g. fossil fuel fired, nuclear or renewable power plants) with different marginal costs and technical attributes, or they have different strategic characteristics (e.g. vertically integrated or not). All of these characteristics can be represented using a multicommodity model. Network flow models are already being used within the context of electricity transmission planning. Some examples are

- Kumar and Chebiyam [15] uses a generalized network flow model to analyse a thermal power system.
- Kumar and Radhakrishna [16] uses a network flow model to project GHG Emission caused by thermal power generation for India.
- Quelhas et al. [18] uses a multiperiod generalized network flow model for the U.S. integrated energy system. Quelhas and McCalley [19] presents the simulation results of the model by Quelhas et al. [18]

If the production and the transmission of the electricity are integrated within a network flow context, using substitutable commodities within the mathematical model would be a wise approach and a possible research area.

We focus on a general modelling approach that can be used in all kinds of network problems that substitution may occur instead of focusing on logistics of a particular product or commodity group. We introduce mathematical models for two and three-commodity network flow problems with substitution. We prove that in uncapacitated versions of our models from a computational difficulty point of view, constraint matrices are totally unimodular. Furthermore, we explore the computational properties of the capacitated versions of our models through random problem generation and hypothesis testing. Specifically, we explore the effect of capacities and problem size on solution time. Our results show that solution time significantly increases on both two and three commodity problems when in the problem both overall and commodity-specific capacities are introduced. Also, when problem size is increased, solution time significantly increases.
To sum up, we contribute to the literature by

- introducing a new network flow problem and develop mathematical models regarding that problem
- submit a theorem related to the computational complexity of the problem and prove the theorem
- investigate empirical computational complexity of the problem and present the results.

The remainder of this work is organized as follows. In Chapter 2, we review the literature on network flow problems. In Chapter 3, we introduce the two-commodity network flow problem with substitution, present mathematical models and computational experiments regarding the models. In Chapter 4, we introduce the three-commodity network flow problem with substitution, present mathematical models and computational experiments regarding the models. In Chapter 5, we introduce n-commodity network flow problem with substitution and present mathematical models. Finally, we close with concluding remarks in Chapter 6.

Chapter 2

Literature Review on Network Flow Problems

In order to introduce our multicommodity network flow problem with substitution, we must first introduce network flow problems and multicommodity flows. These problems are covered very well by Ahuja, Magnanti and Orlin [2] so we will summarize their work to provide a general view on network flows. They refer the minimum cost flow model as the most fundamental of all network flow problems. Since our model is a generalization of multicommodity network flow problem, which is a generalization of the minimum cost flow problem, we would like to state the base problem by their words: a least cost shipment of a commodity through a network has to be determined in order to satisfy demands at certain nodes from available supplies at other nodes. There are many applications of this model, such as the distribution of a product from manufacturing plants to warehouses, or from warehouses to retailers; the flow of raw material and intermediate goods through the various machining stations in a production line; the routing of automobiles through an urban street network; and the routing of calls through the telephone system.
The minimum cost flow problem is described by Ahuja, Magnanti and Orlin [2] as follows: $G(N, A)$ is a directed network defined by a set N of n nodes and a set A of m directed arcs. Each arc $(i, j) \in A$ has

- an associated cost c_{ij} that denotes the cost per unit flow on that arc.
- a capacity u_{ij} that denotes the maximum amount that can flow on that arc.
- a lower bound l_{ij} that denotes the minimum amount that must flow on that arc.

Each node $i \in N$ is associated with an integer number $b(i)$ representing its supply/demand. If

- $b(i) > 0$, node i is a supply node.
- $b(i) < 0$, node i is a demand node.
- $b(i) = 0$, node i is a transshipment node.

The decision variables of minimum cost flow problem are arc flows and they are represented by x_{ij}. Then, the minimum cost flow problem is formulated as an optimization problem as follows:

$$\text{Minimize} \sum_{(i,j)\in A} c_{ij}x_{ij} \tag{2.1}$$

$$\text{subject to} \sum_{j:(i,j)\in A} x_{ij} - \sum_{j:(j,i)\in A} x_{ji} = b(i) \qquad \forall i \in N \tag{2.2}$$

$$l_{ij} \le x_{ij} \le u_{ij} \qquad \forall (i,j) \in A \tag{2.3}$$

There are many special and generalizing version of the minimum cost flow problem, which we will summarize as follows:

- *Shortest path problem:* In this problem, there is only one supply node (also known as source node) and one demand node (also known as sink node) and the supply and demand is equal to 1. The problem is then to find the shortest path from source to sink.
- *Maximum flow problem:* This problem is somehow complementary to the shortest path problem. We try to maximize the flow from source to sink through the capacitated network.
- *Assignment problem:* In this problem, we try to pair each object in one set to another object in a second set at minimum possible cost. Sets are equally sized. Objects in first set are represented by supply nodes, which have a supply of 1 and objects in second set are represented by demand nodes, which have a demand of 1. Possible pairs are represented by arcs with a capacity of 1.
- *Transportation problem:* Transportation problem is somehow similar to assignment problem, however, sets may not be equally sized and supply and demand of nodes and capacities are not necessarily 1.

- *Circulation problem:* This problem is a minimum cost flow problem with only transshipment nodes. We try to find a feasible flow that honours the lower and upper bounds and circulates around the network.

- *Convex cost flow problems:* It is assumed that the cost is linear in the minimum cost flow problem. If the cost is a convex function of the amount of flow, then the problem becomes a convex cost flow problem.

- *Generalized flow problems:* Arcs conserve flows in the minimum cost flow problem. Arcs may consume or generate flow in generalized flow problems.

- *Multicommodity flow problems:* Minimum cost flow problem considers only one type of commodity. Multicommodity flow problems arise when several commodities use the same network. Commodities may differentiate by their characteristics or their origin-destination pairs. Our problem is a generalization of the multicommodity flow problems where substitution is possible among commodities.

Since our problem is a generalized version of the multicommodity flow problem, we would like to introduce multicommodity flow model as follows: $G(N, A)$ is a directed network defined by a set N of n nodes and a set A of m directed arcs. Let k be the set of commodity types. Each arc $(i, j) \in A$ has

- an associated cost c_{ij}^k that denotes the cost per unit flow of commodity k on that arc.

- an overall capacity u_{ij} that denotes the maximum amount that can flow on that arc.

- commodity-specific capacities u_{ij}^k that denotes the maximum amount of commodity k that can flow on that arc.

Each node $i \in N$ is associated with an integer number $b(i)^k$ representing its supply/demand of commodity k. If

- $b(i)^k > 0$, node i is a supply node for commodity k.

- $b(i)^k < 0$, node i is a demand node for commodity k.

- $b(i)^k = 0$, node i is a transshipment node for commodity k.

The decision variables of minimum cost flow problem are arc flows of commodities and they are represented by x_{ij}^k. Then, the multicommodity flow problem is formulated as an optimization problem as follows:

$$
\begin{aligned}
&\text{Minimize} && \sum_{(i,j)\in A}\sum_{k\in K} c_{ij}^k x_{ij}^k && && (2.4)\\
&\text{subject to} && \sum_{j:(i,j)\in A} x_{ij}^k - \sum_{j:(j,i)\in A} x_{ji}^k = \ b(i)^k && \forall i \in N, \forall k \in K && (2.5)\\
& && l_{ij} \leq \sum_{k\in K} x_{ij}^k \leq u_{ij} && \forall (i,j) \in A && (2.6)\\
& && i_{ij}^k \leq x_{ij}^k \leq u_{ij}^k && \forall (i,j) \in A, \forall k \in K && (2.7)
\end{aligned}
$$

Regarding the complexity of our problem, we first need to address that integer version of multicommodity flow problems are NP-hard. However, Ahuja, Magnanti and Orlin [2] note that uncapacitated multicommodity flow problem have totally unimodular constraint matrices, which provide integral optimal solutions to LP-relaxations of IP models. While investigating our problem, we solely focus on mathematical programming formulation and the computational complexity of the problem rather than possible problem contexts.

Chapter 3

Two-commodity Network Flow Formulation with Substitution

We consider two types of commodities, namely, $\mathbf{A}$ and $\mathbf{B}$. $\mathbf{A}$ is a superior commodity that can also satisfy the demand for commodity $\mathbf{B}$, whereas $\mathbf{B}$ is the inferior commodity that can not satisfy the demand for $\mathbf{A}$. Therefore, commodity $\mathbf{A}$ substitutes for commodity $\mathbf{B}$. We consider a minimum cost-flow problem on a network $G(N, A)$ where N is the set of nodes and A is the set of arcs. K denotes the set of commodity types, which include commodities $\mathbf{A}$ and $\mathbf{B}$. Supply or demand of node $i \in N$ of commodity type $k \in K$ is represented by $b^k(i)$. If $b^k(i) > 0$, then node i is a supply node for commodity k; if $b^k(i) < 0$, it is a demand node and if $b^k(i) = 0$, it is a transshipment node. c_{ij} is the unit cost of flow on arc (i, j).

For feasibility, total supply and total demand must be balanced, however, by the nature of the problem, commodity $\mathbf{A}$ has abundant supply while commodity $\mathbf{B}$ has a shortage in supply. Therefore, total supply of $\mathbf{A}$ is greater than total demand of $\mathbf{A}$, whereas total supply of $\mathbf{B}$ is smaller than total demand of $\mathbf{B}$. Surplus of $\mathbf{A}$ and deficit of $\mathbf{B}$ is equal. These assumptions are necessary only for our model. They could be unnecessary for another type of modelling approach. If a node has supply of $\mathbf{A}$ and demand of $\mathbf{B}$, then, within that node, the demand of $\mathbf{B}$ can be satisfied by the supply of $\mathbf{A}$. This is called within-node supply. In order to reflect the possibility for within-node supply in the network flow model, $G(N, A)$ is transformed to $G(N', A')$. If there is a node $i \in N$ for which $b^A(i) > 0, b^B(i) < 0$, then:

- a new node is created, namely, node $i^{\mathbf{B}-}$;
- the demand for commodity $\mathbf{B}$ of node i is transferred to this new node: $b^{\mathbf{B}}(i^{\mathbf{B}-}) =$

$b^{\mathbf{B}}(i),\ b^{\mathbf{A}}(i^{\mathbf{B}-}) = 0;$

- $b^{\mathbf{B}}(i) = 0,\ b^{\mathbf{A}}(i) = b^{\mathbf{A}}(i);$
- a new arc $(i, i^{\mathbf{B}-})$ is created and $c_{i,i^{\mathbf{B}-}} = 0$.

As a result, $N^{'}$ includes additional nodes and $A^{'}$ includes additional arcs with zero cost. In order to differentiate the commodity flows, we define three types of flow: AA, AB and BB. AA represents the flow of commodity $\mathbf{A}$ to satisfy the demand of commodity $\mathbf{A}$. AB represents the flow of commodity $\mathbf{A}$ to satisfy the demand of commodity $\mathbf{B}$. Therefore, flow type AB is required to represent the substitution of commodity $\mathbf{A}$ for commodity $\mathbf{B}$. BB represents the flow of commodity $\mathbf{B}$ to satisfy the demand of commodity $\mathbf{B}$. We denote the set of flow types as F and a flow type with $f \in F$. The subset of flow types that can be satisfied using commodity k is defined by F^k. Thus, F^A includes AA and AB whereas F^B includes BB.

In the mathematical programming formulation of the problem, x_{ij}^f represents the flow of type f on arc $(i,j) \in A$ and $u^f(i)$ represents the net in-flow/out-flow of type f on node $i \in N$. Auxiliary variable $u^f(i)$ is used for the sake of clarity in presenting the mathematical model. The linear programming formulation of the two-commodity network flow problem with substitution is

$$\text{Minimize} \quad \sum_{(i,j)\in A'} \sum_{f \in F} (c_{ij} x_{ij}^f) \tag{3.1}$$

$$\text{subject to} \quad \sum_{j:(i,j)\in A'} x_{ij}^{AA} - \sum_{j:(j,i)\in A'} x_{ji}^{AA} = u^{AA}(i) \qquad \forall i \in N^{'} : b^{\mathbf{A}}(i) > 0 \tag{3.2}$$

$$\sum_{j:(i,j)\in A'} x_{ij}^{AA} - \sum_{j:(j,i)\in A'} x_{ji}^{AA} = -u^{AA}(i) \qquad \forall i \in N^{'} : b^{\mathbf{A}}(i) < 0 \tag{3.3}$$

$$\sum_{j:(i,j)\in A'} x_{ij}^{AA} - \sum_{j:(j,i)\in A'} x_{ji}^{AA} = 0 \qquad \forall i \in N^{'} : b^{\mathbf{A}}(i) = 0 \tag{3.4}$$

$$\sum_{j:(i,j)\in A'} x_{ij}^{BB} - \sum_{j:(j,i)\in A'} x_{ji}^{BB} = u^{BB}(i) \qquad \forall i \in N^{'} : b^{\mathbf{B}}(i) > 0 \tag{3.5}$$

$$\sum_{j:(i,j)\in A'} x_{ij}^{BB} - \sum_{j:(j,i)\in A'} x_{ji}^{BB} = -u^{BB}(i) \qquad \forall i \in N^{'} : b^{\mathbf{B}}(i) < 0 \tag{3.6}$$

$$\sum_{j:(i,j)\in A'} x_{ij}^{BB} - \sum_{j:(j,i)\in A'} x_{ji}^{BB} = 0 \qquad \forall i \in N' : b^{\mathbf{B}}(i) = 0 \tag{3.7}$$

$$\sum_{j:(i,j)\in A'} x_{ij}^{AB} - \sum_{j:(j,i)\in A'} x_{ji}^{AB} = u^{AB}(i) \qquad \forall i \in N' : b^{\mathbf{A}}(i) > 0 \tag{3.8}$$

$$\sum_{j:(i,j)\in A'} x_{ij}^{AB} - \sum_{j:(j,i)\in A'} x_{ji}^{AB} = -u^{AB}(i) \qquad \forall i \in N' : b^{\mathbf{B}}(i) < 0 \tag{3.9}$$

$$\sum_{j:(i,j)\in A'} x_{ij}^{AB} - \sum_{j:(j,i)\in A'} x_{ji}^{AB} = 0 \qquad \forall i \in N' : b^{\mathbf{A}}(i) \leq 0, b^{\mathbf{B}}(i) \geq 0 \tag{3.10}$$

$$u^{AA}(i) + u^{AB}(i) = b^{\mathbf{A}}(i) \qquad \forall i \in N' : b^{\mathbf{A}}(i) > 0 \tag{3.11}$$

$$-u^{AA}(i) = b^{\mathbf{A}}(i) \qquad \forall i \in N' : b^{\mathbf{A}}(i) < 0 \tag{3.12}$$

$$u^{BB}(i) = b^{\mathbf{B}}(i) \qquad \forall i \in N' : b^{\mathbf{B}}(i) > 0 \tag{3.13}$$

$$-u^{AB}(i) - u^{BB}(i) = b^{\mathbf{B}}(i) \qquad \forall i \in N' : b^{\mathbf{B}}(i) < 0 \tag{3.14}$$

$$x_{ij}^{f} \geq 0 \qquad \forall (i,j) \in A', \forall f \in F \tag{3.15}$$

$$u^{f}(i) \geq 0 \qquad \forall f \in F, \forall i \in N' \tag{3.16}$$

The objective function (3.1) minimizes the total cost of flow on all arcs. For commodity $\mathbf{A}$, constraint set (3.2) and constraint set (3.3) calculate the net out-flow on node i as $u^{AA}(i)$, respectively for supply nodes and demand nodes of commodity $\mathbf{A}$. Since out-flow is greater than in-flow for supply nodes and vice-versa for demand nodes, u^{AA} is multiplied by -1 on the right hand side of constraint (3.3). For transshipment nodes of commodity $\mathbf{A}$, constraint set (3.4) ensures that difference between out-flow and in-flow of flow type AA is zero. Constraint sets (3.5)-(3.7) do the same for commodity $\mathbf{B}$ and flow type BB as constraints (3.2)-(3.4) for commodity $\mathbf{A}$.

Constraint set (3.8) and constraint set (3.9) calculate the net out-flow on node i as $u^{AB}(i)$, respectively for supply nodes of commodity $\mathbf{A}$ and demand nodes of commodity $\mathbf{B}$. For the nodes that are not supply nodes of commodity $\mathbf{A}$ or demand nodes of commodity $\mathbf{B}$, constraint set (3.10) ensures that difference between out-flow and in-flow of flow type AB is zero. For supply nodes of commodity $\mathbf{A}$, constraint set (3.11) calculates the sum of net out-flow of flow type AA and net out-flow of flow type AB as supply of commodity $\mathbf{A}$. For demand nodes of commodity $\mathbf{A}$, constraint set (3.12) calculates net out-flow of flow type AA as demand of commodity $\mathbf{A}$. For supply nodes of commodity $\mathbf{B}$ constraint set (3.13) calculates the net out-flow of flow type BB as supply of commodity $\mathbf{B}$. For demand nodes of commodity $\mathbf{B}$, constraint set (3.14) calculates the sum of net out-flow of flow type AB and net out-flow of flow type BB as demand of commodity $\mathbf{B}$. Constraint

sets (3.15) - (3.16) define variable domains, which are all non-negative.
In order to develop a matrix notation for the formulation of the problem, we define the vector form of the parameters and variables as follows:

- **c** represents the cost vector, which consists of elements c_{ij}.
- x^f denotes the vector of x variables and u^f represent the vector of u variables.
- $b^{\mathbf{A}}$ denotes supply/demand vector of commodity $\mathbf{A}$
- $b^{\mathbf{B}}$ denotes supply/demand vector of commodity $\mathbf{B}$.

Then, we can rewrite the problem formulation (3.1)-(3.16) in the matrix notation as follows:

$$\text{Minimize} \sum_{f \in F} cx^f \tag{3.17}$$

$$\text{subject to} \begin{pmatrix} Q & 0 & 0 & \tilde{I}^{\mathbf{A}} & 0 & 0 \\ 0 & Q & 0 & 0 & \tilde{I}^{\mathbf{B}} & 0 \\ 0 & 0 & Q & 0 & 0 & \tilde{I}^{AB} \\ 0 & 0 & 0 & -\tilde{I}^{\mathbf{A}} & 0 & -\tilde{I}^{\mathbf{A}+} \\ 0 & 0 & 0 & 0 & -\tilde{I}^{\mathbf{B}} & -\tilde{I}^{\mathbf{B}-} \end{pmatrix} \begin{pmatrix} x^{AA} \\ x^{BB} \\ x^{AB} \\ u^{AA} \\ u^{BB} \\ u^{AB} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ b^A \\ b^B \end{pmatrix} \tag{3.18}$$

$$x \geq 0 \tag{3.19}$$

$$u \geq 0 \tag{3.20}$$

where Q is the node-arc incidence matrix of the network $G(N', A')$ and $\tilde{I}^{\mathbf{A}}$, $\tilde{I}^{\mathbf{B}}$, $\tilde{I}^{AB}$, $-\tilde{I}^{\mathbf{A}}$, $-\tilde{I}^{\mathbf{B}}$, $-\tilde{I}^{\mathbf{A}+}$, $-\tilde{I}^{\mathbf{B}-}$ are all (n x n) matrices, which are variants of identity matrix such that

$$[\tilde{I}^{\mathbf{A}}]_{ii} = \begin{cases} -1 & \forall i \in N^{'} : b^{A}(i) > 0 \\ 0 & \forall i \in N^{'} : b^{A}(i) = 0 \\ 1 & \forall i \in N^{'} : b^{A}(i) < 0 \end{cases}$$

$$[\tilde{I}^{\mathbf{B}}]_{ii} = \begin{cases} -1 & \forall i \in N^{'} : b^{B}(i) > 0 \\ 0 & \forall i \in N^{'} : b^{B}(i) = 0 \\ 1 & \forall i \in N^{'} : b^{B}(i) < 0 \end{cases}$$

$$[\tilde{I}^{AB}]_{ii} = \begin{cases} -1 & \forall i \in N^{'} : b^{A}(i) > 0 \\ 0 & \forall i \in N^{'} : b^{A}(i) \leq 0, b^{B}(i) \geq 0 \\ 1 & \forall i \in N^{'} : b^{B}(i) < 0 \end{cases}$$

$$[-\tilde{I}^{\mathbf{A}}]_{ii} = \begin{cases} 1 & \forall i \in N^{'} : b^{A}(i) > 0 \\ 0 & \forall i \in N^{'} : b^{A}(i) = 0 \\ -1 & \forall i \in N^{'} : b^{A}(i) < 0 \end{cases}$$

$$[-\tilde{I}^{\mathbf{B}}]_{ii} = \begin{cases} 1 & \forall i \in N^{'} : b^{B}(i) > 0 \\ 0 & \forall i \in N^{'} : b^{B}(i) = 0 \\ -1 & \forall i \in N^{'} : b^{B}(i) < 0 \end{cases}$$

$$[-\tilde{I}^{\mathbf{A}+}]_{ii} = \begin{cases} 1 & \forall i \in N^{'} : b^{A}(i) > 0 \\ 0 & \forall i \in N^{'} : b^{A}(i) = 0 \\ 0 & \forall i \in N^{'} : b^{A}(i) < 0 \end{cases}$$

$$[-\tilde{I}^{\mathbf{B}-}]_{ii} = \begin{cases} 0 & \forall i \in N^{'} : b^{B}(i) > 0 \\ 0 & \forall i \in N^{'} : b^{B}(i) = 0 \\ -1 & \forall i \in N^{'} : b^{B}(i) < 0 \end{cases}$$

According to this matrix notation of the problem formulation, we investigate the unimodularity property of the constraint matrix. We exploit the matrix unimodularity through this property: if there exists at most one +1 and one -1 on a (1,0,-1) matrix, the matrix is totally unimodular. Therefore, we conduct a columnwise analysis of the constraint matrix as follows:

Property 1 *Q is a node-arc incidence matrix. Therefore, each column in Q contains only one +1 and only one -1.*

Property 2 *In the first three blocks of m columns, the only non-zero entries are the entries of Q matrices. Therefore, in each column of these three blocks, there exists only one +1 and only one -1.*

Property 3 *For the fourth block of n columns including column $3m+1$ through $3m+n$, the non-zero entries belong to $\tilde{I}^{\mathbf{A}}$ and $-\tilde{I}^{\mathbf{A}}$. In each column of $\tilde{I}^{\mathbf{A}}$, there exists at most either one +1 or one -1. In each column of $-\tilde{I}^{\mathbf{A}}$, the entry has the opposite sign of the entry in the same column of $\tilde{I}^{\mathbf{A}}$. Therefore, in each column there exists at most one +1 and one -1.*

Property 4 *For the fifth block of n columns including column $3m+n+1$ through $3m+2n$, the non-zero entries belong to $\tilde{I}^{\mathbf{B}}$ and $-\tilde{I}^{\mathbf{B}}$. In each column of $\tilde{I}^{\mathbf{B}}$, there exists at most either one +1 or one -1. In each column of $-\tilde{I}^{\mathbf{B}}$, the entry has the opposite sign of the entry in the same column of $\tilde{I}^{\mathbf{B}}$. Therefore, in each column there exists at most one +1 and one -1.*

Property 5 *In $G(N', A')$, there exists no node $i \in N$ where $b^{\mathbf{A}}(i) > 0$ and $b^{\mathbf{B}}(i) < 0$. Thus, if $b^{\mathbf{A}}(i) > 0$, then $b^{\mathbf{B}}(i) \geq 0$.*

Property 6 *In the sixth block of columns including column $3m+2n+1$ through $3m+3n$, the non-zero entries belong to diagonal elements of $\tilde{I}^{AB}$, $-\tilde{I}^{\mathbf{A}+}$ and $-\tilde{I}^{\mathbf{B}-}$. Based on the entries of $\tilde{I}^{AB}$, we observe the following cases:*

> *Case 1. If there exists a -1 in column $3m+2n+i$, then $b^A(i) > 0$. In the same column in $-\tilde{I}^{\mathbf{A}+}$, the entry is +1 since $b^A(i) > 0$. In the same column in $-\tilde{I}^{\mathbf{B}-}$, the nonzero entry is 0 as $b^B(i) \nleq 0$ due to Property 5.*
> *Case 2. If there exists a +1 in column $3m+2n+i$, then $b^B(i) < 0$. In the same column in $-\tilde{I}^{\mathbf{A}+}$, the nonzero entry is 0 since $b^A(i) \ngeq 0$ due to Property 5. In the same column in $-\tilde{I}^{\mathbf{B}-}$, the entry is -1 since $b^B(i) < 0$.*
> *Case 3. If there exists a 0 in column $3m+2n+i$, then $b^A(i) \leq 0$ and $b^B(i) \geq 0$. In the same column in $-\tilde{I}^{\mathbf{A}+}$, the nonzero entry is 0. In the same column in $-\tilde{I}^{\mathbf{B}-}$, the nonzero entry is 0.*

Therefore, in each column, there exists at most one +1 and at most one -1. As a result of Property 2, Property 3, Property 4 and Property 6, we prove the next theorem.

Theorem 1 *The constraint matrix of the two-commodity network flow formulation with substitution is totally unimodular.*

In (3.1)-(3.16) and (3.17)-(3.18), we formulate the uncapacitated version of the two-commodity network flow problem with substitution. If w_{ij}^k denotes the commodity-specific capacity of arc $(i, j) \in A^{'}$, then the commodity-specific capacity constraint is:

$$\sum_{f \in F^k} x_{ij}^f \leq \; w_{ij}^k \;, \forall (i,j) \in A^{'}, \; \forall k \in K \tag{3.21}$$

If v_{ij} denotes the overall capacity of arc $(i, j) \in A^{'}$, then the overall capacity constraint is

$$\sum_{f \in F} x_{ij}^f \leq \; v_{ij} \;, \forall (i,j) \in A^{'} \tag{3.22}$$

As a result, the capacitated version of the problem is formulated as (3.1)-(3.16) along with (3.21) and (3.22).

3.1 Computational Complexity of the Capacitated Two-commodity Network Flow Problem

We discuss the computational complexity of the capacitated two-commodity network flow problem with substitution through its relation with the common two-commodity network flow problem as it is well-known that the common problem is NP-hard. The two-commodity network flow problem with substitution is a generalized version of the common two-commodity network flow problem. In other words, the common 2-commodity network flow problem is a special case of 2-commodity network flow problem with substitution. In order to show this relationship formally, we work with mathematical model (3.1)-(3.16), (3.21),(4.2) of the problem with substitution. We need to show that our mathematical model is the same as the network flow model of the common problem when substitution is not allowed. In order to avoid substitution, we first eliminate the flow variable AB since it represents substitution of commodity $\mathbf{A}$ for commodity $\mathbf{B}$. Then, constraint sets (3.8)-(3.10) drop completely from the model. Moreover, constraint sets

(3.11) and (3.14) become:

$$
\begin{aligned}
u^{AA}(i) =& \ b^{\mathbf{A}}(i) && \forall i \in N^{'} : b^{\mathbf{A}}(i) > 0 && (3.23)\\
-u^{BB}(i) =& \ b^{\mathbf{B}}(i) && \forall i \in N^{'} : b^{\mathbf{B}}(i) < 0 && (3.24)
\end{aligned}
$$

With this arrangement, we now do not need u variables any more and we can eliminate constraint sets (3.23), (3.12), (3.13), (3.24) and (3.16) by replacing $u^{AA}(i)$ in constraint set (3.2) with $b^{\mathbf{A}}(i)$ and $-u^{BB}(i)$ in constraint set (3.6) with $b^{\mathbf{B}}(i)$, eliminating constraint sets (3.23) and (3.24), respectively. Moreover, we replace $-u^{AA}(i)$ in constraint set (3.3) with $b^{\mathbf{A}}(i)$ and $u^{BB}(i)$ in constraint set (3.5) with $b^{\mathbf{B}}(i)$, eliminating constraint sets (3.12) and (3.13), respectively. Finally, we replace the zeros in constraint sets (3.4) and (3.7) with $b^{\mathbf{A}}(i)$ and $b^{\mathbf{B}}(i)$ respectively because $b^{\mathbf{A}}(i) = 0$ and $b^{\mathbf{B}}(i) = 0$ for (3.4) and (3.7), respectively. As a result, constraint sets (3.2)-(3.4) become

$$
\begin{aligned}
\sum_{j:(i,j)\in A'} x_{ij}^{AA} - \sum_{j:(j,i)\in A'} x_{ji}^{AA} =& \ b^{\mathbf{A}}(i), && \forall i \in N^{'}, b^{\mathbf{A}}(i) > 0 && (3.25)\\
\sum_{j:(i,j)\in A'} x_{ij}^{AA} - \sum_{j:(j,i)\in A'} x_{ji}^{AA} =& \ b^{\mathbf{A}}(i), && \forall i \in N^{'}, b^{\mathbf{A}}(i) < 0 && (3.26)\\
\sum_{j:(i,j)\in A'} x_{ij}^{AA} - \sum_{j:(j,i)\in A'} x_{ji}^{AA} =& \ b^{\mathbf{A}}(i), && \forall i \in N^{'}, b^{\mathbf{A}}(i) = 0. && (3.27)
\end{aligned}
$$

(3.25)-(3.27) can be rewritten as

$$
\sum_{j:(i,j)\in A'} x_{ij}^{AA} - \sum_{j:(j,i)\in A'} x_{ji}^{AA} = \ b^{\mathbf{A}}(i) \qquad \forall i \in N^{'} \qquad (3.28)
$$

Constraint sets (3.5)-(3.7) become

$$
\begin{aligned}
\sum_{j:(i,j)\in A'} x_{ij}^{BB} - \sum_{j:(j,i)\in A'} x_{ji}^{BB} =& \ b^{\mathbf{B}}(i), && \forall i \in N^{'}, b^{\mathbf{B}}(i) > 0 && (3.29)\\
\sum_{j:(i,j)\in A'} x_{ij}^{BB} - \sum_{j:(j,i)\in A'} x_{ji}^{BB} =& \ b^{\mathbf{B}}(i), && \forall i \in N^{'}, b^{\mathbf{B}}(i) < 0 && (3.30)\\
\sum_{j:(i,j)\in A'} x_{ij}^{BB} - \sum_{j:(j,i)\in A'} x_{ji}^{BB} =& \ b^{\mathbf{B}}(i), && \forall i \in N^{'}, b^{\mathbf{B}}(i) = 0 && (3.31)
\end{aligned}
$$

Similarly, (3.29)-(3.31) can be rewritten as

$$\sum_{j:(i,j)\in A'} x_{ij}^{BB} - \sum_{j:(j,i)\in A'} x_{ji}^{BB} = b^{\mathbf{B}}(i) \qquad \forall i \in N' \tag{3.32}$$

Capacity constraints do not change

$$\sum_{f\in F^k} x_{ij}^f \leq w_{ij}^k \text{ }, \forall (i,j) \in A' \text{ }, \forall k \in K \tag{3.21}$$

$$\sum_{f\in F} x_{ij}^f \leq v_{ij} \text{ }, \forall (i,j) \in A' \tag{3.22}$$

Then, the mathematical model becomes

$$\begin{aligned}
\text{Minimize} \quad & \sum_{(i,j)\in A'} (c_{ij}x_{ij}^{AA} + c_{ij}x_{ij}^{BB}) && && (3.33)\\
\text{subject to} \quad & \sum_{j:(i,j)\in A'} x_{ij}^{AA} - \sum_{j:(j,i)\in A'} x_{ji}^{AA} = b^{\mathbf{A}}(i) && \forall i \in N' && (3.28)\\
& \sum_{j:(i,j)\in A'} x_{ij}^{BB} - \sum_{j:(j,i)\in A'} x_{ji}^{BB} = b^{\mathbf{B}}(i) && \forall i \in N' && (3.32)\\
& \sum_{f\in F^k} x_{ij}^f \leq w_{ij}^k && \forall (i,j) \in A' \text{ }, \forall k \in K && (3.21)\\
& \sum_{f\in F} x_{ij}^f \leq v_{ij} && \forall (i,j) \in A' && (3.22)\\
& x_{ij}^f \geq 0 && \forall (i,j) \in A', \forall f \in F && (3.34)
\end{aligned}$$

The mathematical model (3.33), (3.28), (3.32), (3.21), (3.22), (3.34) is equivalent to the network flow model for the two-commodity problem. Therefore, we have shown that our problem is a generalization of the common two-commodity network flow problem and it is at least as hard as the common problem. Since the integer version of the common problem is NP-hard as pointed out by Ahuja, Magnanti and Orlin [2], our problem is also NP-hard.

3.2 Computational Experiments

In order to investigate how the size of the problem along with the capacity tightness affect the model, we conduct computational experiments that explore the change in the

solution time of the problem with respect to the size and the arc capacities. For this purpose, we build the mathematical model on CPLEX Studio 12.4 environment. We first generate random instances of the capacitated 2-commodity network flow problem with substitution and solve both the linear programming relaxation and integer programming problem. Computational experiments are conducted on a 8-core computer with a Intel Core i7 CPU @3.20 gHz and 24.0 GB of RAM. CPU times and objective function values are reported after the instances are solved.
While generating the problem instances, we pay particular attention to the distribution of the parameter values along a controllable range. In order to achieve this, for each parameter, we first generate an upper bound and a lower bound, and then we generate a parameter value within this range randomly. These input parameters are;

- arc density of the network;
- minimum total supply, maximum total supply;
- minimum arc cost, maximum arc cost;
- minimum commodity-specific arc capacity, maximum commodity-specific arc capacity;
- minimum overall arc capacity, maximum overall arc capacity;
- overall capacity density;
- capacity densities for commodities.

We generate problem instances of six different sizes with respect to the number of nodes: 10, 50, 100, 200, 300, 500. For each node size, 10 different generation seeds are used. Arc capacities are calculated depending on total supply and number of nodes. In order to examine the effect of arc capacities, different levels of arc capacities are developed. To develop these capacities, two coefficients are introduced, one for commodity-specific and one for overall capacities. Total supply divided by number of nodes is multiplied by these capacity coefficients. Thus, we developed different capacity levels, making some instances tighter on capacity and some instances looser on capacity. For further examination, different capacity settings are also developed:

- only commodity-specific capacities are introduced
- only overall capacities are introduced

- both capacities are introduced
- both capacities are introduced but overall capacities are calculated as the sum of commodity-specific capacities
- uncapacitated setting.

The pseudo-code for the algorithm that generates the problem instances is given in Algorithm (1).

Algorithm 1 Algorithm that generates the problem instances for two-commodity problem

```
Begin
Read generation parameters
Initialize a node-to-node adjacency matrix P of size nxn of 0's
for each P_ij (i ≠ j) do
    if random number < arc density then
        P_ij := 1
    end if
end for
Calculate row and column sums
for each P_ij (i ≠ j) do
    if P_ij := 1 then
        Generate the corresponding arc
        Generate the random cost of corresponding arc
        if random number < overall capacity density then
            Generate random overall capacity for the arc
        end if
        for Each commodity do
            if random number < commodity capacity density then
                Generate random capacity for that commodity for the arc
            end if
        end for
    end if
end for
```

for Each commodity **do**
Generate total supply
total demand = total supply
end for
for Superior commodity **do**
Increase total supply by a random ratio of total supply
end for
for Inferior commodity **do**
Increase total demand as much as the increase of supply of superior commodity
end for
Using row and column sums, determine pure supply nodes, pure demand nodes and transshipment nodes
for Each commodity **do**
Assign random transshipment nodes as supply nodes, demand nodes and pure transshipment nodes
Distribute total supply among supply nodes randomly around average
Distribute total demand among demand nodes randomly around average
end for
End

The main reason for designing different capacity configurations is to create problems whose nature is different with respect to tightness of the capacities. Eventually, we want to observe which type of capacity configuration is more effective on the difficulty of the problem. We use following parameters while generating the random problems:

- minimum supply is 100, 500, 1000, 2000, 3000, 5000 for node sizes 10, 50, 100, 200, 300, 500
- maximum supply is 500, 2500, 5000, 10000, 15000, 25000 for node sizes 10, 50, 100, 200, 300, 500
- minimum cost is 10
- maximum cost is 50
- commodity specific capacity coefficient has 5 levels; 0.05, 0.075, 0.1, 0.25, 0.5
- minimum commodity specific capacity equals to commodity specific capacity coefficient times minimum supply over number of nodes
- maximum commodity specific capacity equals to commodity specific capacity coefficient times maximum supply over number of nodes

- overall capacity has 3 levels; 0.1, 0.25, 0.625
- minimum overall capacity equals to overall capacity coefficient times minimum supply over number of nodes
- maximum overall capacity equals to overall capacity coefficient times maximum supply over number of nodes
- arc density is 0.8
- overall capacity density is 0.5
- bound density of commodity $\mathbf{A}$ is 0.6
- bound density of commodity $\mathbf{B}$ is 0.4.

As a result, we generate 50 problems for each size of the network. For each size, we use 5 different capacity configurations.
While investigating the effect of capacities and the network size, we use a hypothesis testing framework to determine whether our inferences are statistically significant or not. In this manner, we would like to answer the following questions:

- How much of an effect do commodity-specific capacity constraints have on solution time?
- How much of an effect do overall capacity constraints have on solution time?
- How much of an effect do both capacity constraints together have on solution time?
- How much of an effect does inclusion of overall capacity constraints over commodity-specific capacity constraints have on solution time? (i.e. sum of commodity-specific capacities are introduced as overall capacities.)
- How much of an effect does the tightening of capacity constraints have on solution time?
- How much of an effect does problem size have on solution time?

Two-sample two-tailed t-tests are used for all hypotheses because we would like to compare means of two samples (solution times of different configurations). The null hypotheses assume that their means are equal. StatTools software is used to perform the tests. The results are presented in Table (3.1) through Table (3.7).

Table 3.1: Results of hypothesis testing for comparing the solution times of configuration with commodity-specific capacities and uncapacitated configuration

	Commodity-Specific	**Uncapacitated**
Sample Summaries	**Data Set #1**	**Data Set #1**
Sample Size	59	60
Sample Mean	8461.61	6565.00
Sample Std Dev	10086.47	7518.91
Hypothesis Test (Difference of Means)	**Equal Variances**	**Unequal Variances**
Hypothesized Mean Difference	0	0
Alternative Hypothesis	$\neq 0$	$\neq 0$
Sample Mean Difference	1896.61	1896.61
Standard Error of Difference	1629.02116	1632.969151
Degrees of Freedom	117	107
t-Test Statistic	1.1643	1.1614
p-Value	0.2467	0.2480

Table 3.2: Results of hypothesis testing for comparing the solution times of configuration with overall capacities and uncapacitated configuration

	overall	**Uncapacitated**
Sample Summaries	**Data Set #1**	**Data Set #1**
Sample Size	60	60
Sample Mean	7954.98	6565.00
Sample Std Dev	9444.79	7518.91
Hypothesis Test (Difference of Means)	**Equal Variances**	**Unequal Variances**
Hypothesized Mean Difference	0	0
Alternative Hypothesis	$\neq 0$	$\neq 0$
Sample Mean Difference	1389.98	1389.98
Standard Error of Difference	1558.515273	1558.515273
Degrees of Freedom	118	112
t-Test Statistic	0.8919	0.8919
p-Value	0.3743	0.3744

Table 3.3: Results of hypothesis testing for comparing the solution times of configuration with both commodity-specific and overall capacities and uncapacitated configuration

	Full Capacity	**Uncapacitated**
Sample Summaries	**Data Set #1**	**Data Set #1**
Sample Size	57	60
Sample Mean	9542.56	6565.00
Sample Std Dev	11317.35	7518.91
Hypothesis Test (Difference of Means)	**Equal Variances**	**Unequal Variances**
Hypothesized Mean Difference	0	0
Alternative Hypothesis	$\neq 0$	$\neq 0$
Sample Mean Difference	2977.56	2977.56
Standard Error of Difference	1768.045868	1785.859363
Degrees of Freedom	115	96
t-Test Statistic	1.6841	1.6673
p-Value	0.0949	0.0987

Table 3.4: Results of hypothesis testing for comparing the solution times of configuration with overall capacities calculated as the sum of commodity-specific capacities and configuration with commodity-specific capacities

	Commodity Specific	**Sum of Commodity Specific**
Sample Summaries	**Data Set #1**	**Data Set #1**
Sample Size	59	59
Sample Mean	8461.61	8851.94
Sample Std Dev	10086.47	10576.17
Hypothesis Test (Difference of Means)	**Equal Variances**	**Unequal Variances**
Hypothesized Mean Difference	0	0
Alternative Hypothesis	$\neq 0$	$\neq 0$
Sample Mean Difference	-390.33	-390.33
Standard Error of Difference	1902.684289	1902.684289
Degrees of Freedom	116	115
t-Test Statistic	-0.2051	-0.2051
p-Value	0.8378	0.8378

Table 3.5: Results of hypothesis testing for comparing the solution times of levels for configuration with commodity-specific capacities

	Level 1	**Level 5**
Sample Summaries	**Data Set #2**	**Data Set #2**
Sample Size	59	59
Sample Mean	8564.76	7949.71
Sample Std Dev	10215.94	9345.70
Hypothesis Test (Difference of Means)	**Equal Variances**	**Unequal Variances**
Hypothesized Mean Difference	0	0
Alternative Hypothesis	$\neq 0$	$\neq 0$
Sample Mean Difference	615.05	615.05
Standard Error of Difference	1802.576231	1802.576231
Degrees of Freedom	116	115
t-Test Statistic	0.3412	0.3412
p-Value	0.7336	0.7336

Table 3.6: Results of hypothesis testing for comparing the solution times of levels for configuration with overall capacities

	Level 1	**Level 3**
Sample Summaries	**Data Set #3**	**Data Set #3**
Sample Size	60	60
Sample Mean	8565.37	7703.93
Sample Std Dev	10337.55	9054.22
Hypothesis Test (Difference of Means)	**Equal Variances**	**Unequal Variances**
Hypothesized Mean Difference	0	0
Alternative Hypothesis	$\neq 0$	$\neq 0$
Sample Mean Difference	861.43	861.43
Standard Error of Difference	1774.091263	1774.091263
Degrees of Freedom	118	115
t-Test Statistic	0.4856	0.4856
p-Value	0.6282	0.6282

Table 3.7: Results of hypothesis testing for comparing the solution times of problem sizes for uncapacitated configuration

	10	**50**
Sample Summaries	**Data Set #4**	**Data Set #4**
Sample Size	10	10
Sample Mean	1218.70	1402.00
Sample Std Dev	21.35	28.02
Hypothesis Test (Difference of Means)	**Equal Variances**	**Unequal Variances**
Hypothesized Mean Difference	0	0
Alternative Hypothesis	$\neq 0$	$\neq 0$
Sample Mean Difference	-183.30	-183.30
Standard Error of Difference	11.13956911	11.13956911
Degrees of Freedom	18	16
t-Test Statistic	-16.4549	-16.4549
p-Value	< 0.0001	< 0.0001

The only significant results are achieved when the capacity configuration with both commodity specific and overall capacities are compared against uncapacitated configuration (See Table 3.3) and different problem sizes are compared against each other within the uncapacitated configuration (See Table 3.7). For the configuration with both capacities, null hypothesis is rejected at 10% significance level since its p-value is around 0.09 (See Table 3.3). For problem sizes, p-value is significantly small so that null hypothesis is rejected at any significance level (See Table 3.7). Therefore, we conclude that when both capacities are active, solution time significantly increases. Moreover, as problem size increases, solution time significantly increases. Thus, we imply that if both capacities are active in the problem or if the problem size is increased, the problem becomes more difficult to solve.

Chapter 4

Three-commodity Network Flow Formulation with Substitution

So far, we discussed a problem where two commodities flow through the network. More commodities can be included to the problem. In this chapter, we consider three types of commodities, namely, $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$. Through substitution, $\mathbf{A}$ can satisfy the demand for both $\mathbf{B}$ and $\mathbf{C}$, and $\mathbf{B}$ can satisfy the demand of $\mathbf{C}$. Therefore, commodity $\mathbf{A}$ substitutes for commodities $\mathbf{B}$ and $\mathbf{C}$, and commodity $\mathbf{B}$ substitutes for commodity $\mathbf{C}$. We consider a minimum-cost flow problem on a network $G(N, A)$ where N is the set of nodes and $\mathbf{A}$ is the set of arcs. Let K denotes the set of commodity types, which include commodities $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$. Supply or demand of node $i \in N$ of commodity type $k \in K$ is represented by $b^k(i)$. If $b^k(i) > 0$, then node i is a supply node for commodity k; if $b^k(i) < 0$, it is a demand node and if $b^k(i) = 0$, it is a transshipment node. c_{ij} is the unit cost of flow on arc (i, j).

As discussed for the two-commodity problem, total supply and total demand must be balanced, however, by the nature of the problem, commodity $\mathbf{A}$ has abundant supply while commodity $\mathbf{C}$ has a shortage in supply. Commodity $\mathbf{B}$ can have abundant or shortage in supply depending on the supply of $\mathbf{A}$ and $\mathbf{C}$. Therefore, total supply of $\mathbf{A}$ is greater than total demand of $\mathbf{A}$, whereas total supply of $\mathbf{C}$ is smaller than total demand of $\mathbf{C}$. Within-node supply is also considered.

In order to reflect the possibility for within-node supply, in the network flow model, $G(N, A)$ is transformed to $G(N', A')$. If there is a node $i \in N$ for which $b^A(i) > 0, b^B(i) < 0$ or $i \in N$: $b^A(i) > 0, b^C(i) < 0$, or $i \in N$: $b^B(i) > 0, b^C(i) < 0$, then

- a new node is created, namely, node $i^{\mathbf{B}-}$, or node $i^{\mathbf{C}-}$;

- the demand for commodity $\mathbf{B}$ or $\mathbf{C}$ of node i is transferred to this new node: $b^{\mathbf{B}}(i^{\mathbf{B}-}) = b^{\mathbf{B}}(i),\ b^{\mathbf{A}}(i^{\mathbf{B}-}) = 0$ or $b^{\mathbf{C}}(i^{\mathbf{C}-}) = b^{\mathbf{C}}(i),\ b^{\mathbf{A}}(i^{\mathbf{C}-}) = 0$ or $b^{\mathbf{C}}(i^{\mathbf{C}-}) = b^{\mathbf{C}}(i),\ b^{\mathbf{B}}(i^{\mathbf{C}-}) = 0$;

- $b^{\mathbf{B}}(i) = 0,\ b^{\mathbf{A}}(i) = b^{\mathbf{A}}(i)$ or $b^{\mathbf{C}}(i) = 0,\ b^{\mathbf{A}}(i) = b^{\mathbf{A}}(i)$ or $b^{\mathbf{C}}(i) = 0,\ b^{\mathbf{B}}(i) = b^{\mathbf{B}}(i)$;

- a new arc $(i, i^{\mathbf{B}-})$ or $(i, i^{\mathbf{C}-})$ is created and $c_{i,i^{\mathbf{B}-}} = 0$ or $c_{i,i^{\mathbf{C}-}} = 0$.

As a result, N' includes additional nodes and A' includes additional arcs with zero cost. In order to differentiate the commodity flows, we define six types of flow: AA, AB, BB, AC, BC and CC. AA represents the flow of commodity $\mathbf{A}$ to satisfy the demand of commodity $\mathbf{A}$. AB represents the flow of commodity $\mathbf{A}$ to satisfy the demand of commodity $\mathbf{B}$. Therefore, flow type AB is required to represent substitution of commodity $\mathbf{A}$ for commodity $\mathbf{B}$. BB represents the flow of commodity $\mathbf{B}$ to satisfy the demand of commodity $\mathbf{B}$. AC represents the flow of commodity $\mathbf{A}$ to satisfy the demand of commodity $\mathbf{C}$. Therefore, flow type AC is required to represent substitution of commodity $\mathbf{A}$ for commodity $\mathbf{C}$. BC represents the flow of commodity $\mathbf{B}$ to satisfy the demand of commodity $\mathbf{C}$. Therefore, flow type BC is required to represent substitution of commodity $\mathbf{B}$ for commodity $\mathbf{C}$. CC represents the flow of commodity $\mathbf{C}$ to satisfy the demand of commodity $\mathbf{C}$. We denote the set of flow types as F and a flow type with $f \in F$. The subset of flow types that can be satisfied using commodity k is defined by F^k. Thus, F^A includes AA, AB, and AC whereas F^B includes BB and BC and F^C includes CC.

In the mathematical programming formulation of the problem, x_{ij}^f represents the flow of type f on arc $(i, j) \in A$ and $u^f(i)$ represents the net in-flow/out-flow of type f on node $i \in N$. Auxiliary variable $u^f(i)$ is used for the sake of clarity in presenting the mathematical model. The linear programming formulation of the three-commodity network flow problem with substitution is

$$\text{Minimize} \sum_{(i,j)\in A'} \sum_{f\in F} (c_{ij} x_{ij}^f) \tag{4.1}$$

$$\text{subject to} \sum_{j:(i,j)\in A'} x_{ij}^{AA} - \sum_{j:(j,i)\in A'} x_{ji}^{AA} = u^{AA}(i) \qquad \forall i \in N' : b^{\mathbf{A}}(i) > 0 \tag{4.2}$$

$$\sum_{j:(i,j)\in A'} x_{ij}^{AA} - \sum_{j:(j,i)\in A'} x_{ji}^{AA} = -u^{AA}(i) \qquad \forall i \in N' : b^{\mathbf{A}}(i) < 0 \tag{4.3}$$

$$\sum_{j:(i,j)\in A'} x_{ij}^{AA} - \sum_{j:(j,i)\in A'} x_{ji}^{AA} = 0 \qquad \forall i \in N' : b^{\mathbf{A}}(i) = 0 \tag{4.4}$$

$$\sum_{j:(i,j)\in A'} x_{ij}^{BB} - \sum_{j:(j,i)\in A'} x_{ji}^{BB} = u^{BB}(i) \qquad \forall i \in N' : b^{\mathbf{B}}(i) > 0 \tag{4.5}$$

$$\sum_{j:(i,j)\in A'} x_{ij}^{BB} - \sum_{j:(j,i)\in A'} x_{ji}^{BB} = -u^{BB}(i) \qquad \forall i \in N' : b^{\mathbf{B}}(i) < 0 \tag{4.6}$$

$$\sum_{j:(i,j)\in A'} x_{ij}^{BB} - \sum_{j:(j,i)\in A'} x_{ji}^{BB} = 0 \qquad \forall i \in N' : b^{\mathbf{B}}(i) = 0 \tag{4.7}$$

$$\sum_{j:(i,j)\in A'} x_{ij}^{CC} - \sum_{j:(j,i)\in A'} x_{ji}^{CC} = u^{CC}(i) \qquad \forall i \in N' : b^{\mathbf{C}}(i) > 0 \tag{4.8}$$

$$\sum_{j:(i,j)\in A'} x_{ij}^{CC} - \sum_{j:(j,i)\in A'} x_{ji}^{CC} = -u^{CC}(i) \qquad \forall i \in N' : b^{\mathbf{C}}(i) < 0 \tag{4.9}$$

$$\sum_{j:(i,j)\in A'} x_{ij}^{CC} - \sum_{j:(j,i)\in A'} x_{ji}^{CC} = 0 \qquad \forall i \in N' : b^{\mathbf{C}}(i) = 0 \tag{4.10}$$

$$\sum_{j:(i,j)\in A'} x_{ij}^{AB} - \sum_{j:(j,i)\in A'} x_{ji}^{AB} = u^{AB}(i) \qquad \forall i \in N' : b^{\mathbf{A}}(i) > 0 \tag{4.11}$$

$$\sum_{j:(i,j)\in A'} x_{ij}^{AB} - \sum_{j:(j,i)\in A'} x_{ji}^{AB} = -u^{AB}(i) \qquad \forall i \in N' : b^{\mathbf{B}}(i) < 0 \tag{4.12}$$

$$\sum_{j:(i,j)\in A'} x_{ij}^{AB} - \sum_{j:(j,i)\in A'} x_{ji}^{AB} = 0 \qquad \forall i \in N' : b^{\mathbf{A}}(i) \leq 0, b^{\mathbf{B}}(i) \geq 0 \tag{4.13}$$

$$\sum_{j:(i,j)\in A'} x_{ij}^{AC} - \sum_{j:(j,i)\in A'} x_{ji}^{AC} = u^{AC}(i) \qquad \forall i \in N^{'} : b^{\mathbf{A}}(i) > 0 \tag{4.14}$$

$$\sum_{j:(i,j)\in A'} x_{ij}^{AC} - \sum_{j:(j,i)\in A'} x_{ji}^{AC} = -u^{AC}(i) \qquad \forall i \in N^{'} : b^{\mathbf{C}}(i) < 0 \tag{4.15}$$

$$\sum_{j:(i,j)\in A'} x_{ij}^{AC} - \sum_{j:(j,i)\in A'} x_{ji}^{AC} = 0 \qquad \forall i \in N^{'} : b^{\mathbf{A}}(i) \leq 0, b^{\mathbf{C}}(i) \geq 0 \tag{4.16}$$

$$\sum_{j:(i,j)\in A'} x_{ij}^{BC} - \sum_{j:(j,i)\in A'} x_{ji}^{BC} = u^{BC}(i) \qquad \forall i \in N^{'} : b^{\mathbf{B}}(i) > 0 \tag{4.17}$$

$$\sum_{j:(i,j)\in A'} x_{ij}^{BC} - \sum_{j:(j,i)\in A'} x_{ji}^{BC} = -u^{BC}(i) \qquad \forall i \in N^{'} : b^{\mathbf{C}}(i) < 0 \tag{4.18}$$

$$\sum_{j:(i,j)\in A'} x_{ij}^{BC} - \sum_{j:(j,i)\in A'} x_{ji}^{BC} = 0 \qquad \forall i \in N^{'} : b^{\mathbf{B}}(i) \leq 0, b^{\mathbf{C}}(i) \geq 0 \tag{4.19}$$

$$u^{AA}(i) + u^{AB}(i) + u^{AC}(i) = b^{\mathbf{A}}(i) \qquad \forall i \in N^{'} : b^{\mathbf{A}}(i) > 0 \tag{4.20}$$

$$-u^{AA}(i) = b^{\mathbf{A}}(i) \qquad \forall i \in N^{'} : b^{A}(i) < 0 \tag{4.21}$$

$$u^{BB}(i) + u^{BC}(i) = b^{\mathbf{B}}(i) \qquad \forall i \in N^{'} : b^{\mathbf{B}}(i) > 0 \tag{4.22}$$

$$-u^{AB}(i) - u^{BB}(i) = b^{\mathbf{B}}(i) \qquad \forall i \in N^{'} : b^{\mathbf{B}}(i) < 0 \tag{4.23}$$

$$u^{CC}(i) = b^{\mathbf{C}}(i) \qquad \forall i \in N^{'} : b^{\mathbf{C}}(i) > 0 \tag{4.24}$$

$$-u^{AC}(i) - u^{BC}(i) - u^{CC}(i) = b^{\mathbf{C}}(i) \qquad \forall i \in N^{'} : b^{\mathbf{C}}(i) < 0 \tag{4.25}$$

$$x_{ij}^{f} \geq 0 \qquad \forall (i,j) \in A^{'}, \forall f \in F \tag{4.26}$$

$$u^{f}(i) \geq 0 \qquad \forall f \in F, \forall i \in N^{'} \tag{4.27}$$

The objective function (4.1) minimizes the total cost of flow on all arcs. For commodity $\mathbf{A}$, constraint set (4.2) and constraint set (4.3) calculate the net out-flow on node i as $u^{AA}(i)$, respectively for supply nodes and demand nodes of commodity $\mathbf{A}$. Since out-flow is greater than in-flow for supply nodes and vice-versa for demand nodes, u^{AA} is multiplied by -1 on the right hand side of constraint (4.3). For transshipment nodes of commodity $\mathbf{A}$, constraint set (4.4) ensures that difference between out-flow and in-flow of flow type AA is zero. Constraint sets (4.5)-(4.7) and (4.8)-(4.10) do the same for commodity $\mathbf{B}$ and flow type BB and commodity $\mathbf{C}$ and flow type CC respectively as constraints (4.2)-(4.4) do for commodity $\mathbf{A}$.

Constraint set (4.11) and constraint set (4.12) calculate the net out-flow on node i as $u^{AB}(i)$, respectively for supply nodes of commodity $\mathbf{A}$ and demand nodes of commodity $\mathbf{B}$. For the nodes that are not supply nodes of commodity $\mathbf{A}$ or demand nodes of commodity $\mathbf{B}$, constraint set (4.13) ensures that difference between out-flow and in-flow of flow type AB is zero. Constraint sets (4.14)-(4.16) and (4.17)-(4.19) do the same for flow types AC and BC respectively as (4.11)-(4.13)does for flow type AB. For supply nodes of commodity $\mathbf{A}$, constraint set (4.20) calculates the sum of net out-flow of flow type AA, net out-flow of flow type AB and net out-flow of flow type AC as supply of commodity type A. For demand nodes of commodity $\mathbf{A}$, constraint set (4.21) calculates net out-flow of flow type AA as demand of commodity $\mathbf{A}$. For supply nodes of commodity $\mathbf{B}$ constraint set (4.22) calculates the sum of net out-flow of flow type BB and net out-flow of flow type BC as supply of commodity $\mathbf{B}$. For demand nodes of commodity $\mathbf{B}$, constraint set (4.23) calculates the sum of net out-flow of flow type AB and net out-flow of flow type BB as demand of commodity $\mathbf{B}$. For supply nodes of commodity $\mathbf{C}$, constraint set (4.24) calculates the net out-flow of flow type CC as supply of commodity $\mathbf{C}$. For demand nodes of commodity $\mathbf{C}$, constraint set (4.25) calculates the sum of net out-flow of flow type AC, net out-flow of flow type BC and net out-flow of flow type CC as demand of commodity $\mathbf{C}$. Constraint sets (4.26) - (4.27) define variable domains, which are all non-negative.

In order to develop a matrix notation for the formulation of the problem, we define the vector form of the parameters and variables as follows:

- **c** denotes the cost vector, which consists of elements c_{ij}.
- x^f denotes the vector of x variables and u^f represent the vector of u variables.
- $b^{\mathbf{A}}$ denotes supply/demand vector of commodity $\mathbf{A}$.
- $b^{\mathbf{B}}$ denotes supply/demand vector of commodity $\mathbf{B}$.
- $b^{\mathbf{C}}$ denotes supply/demand vector of commodity $\mathbf{C}$.

Then, we can rewrite the problem formulation for (4.1)-(4.27) in the matrix notation as follows:

$$\text{Minimize} \sum_{f \in F} c x^f \tag{4.28}$$

subject to

$$\begin{pmatrix}
Q & 0 & 0 & 0 & 0 & 0 & \tilde{I}^{\mathbf{A}} & 0 & 0 & 0 & 0 & 0 \\
0 & Q & 0 & 0 & 0 & 0 & 0 & \tilde{I}^{\mathbf{B}} & 0 & 0 & 0 & 0 \\
0 & 0 & Q & 0 & 0 & 0 & 0 & 0 & \tilde{I}^{\mathbf{C}} & 0 & 0 & 0 \\
0 & 0 & 0 & Q & 0 & 0 & 0 & 0 & 0 & \tilde{I}^{AB} & 0 & 0 \\
0 & 0 & 0 & 0 & Q & 0 & 0 & 0 & 0 & 0 & \tilde{I}^{AC} & 0 \\
0 & 0 & 0 & 0 & 0 & Q & 0 & 0 & 0 & 0 & 0 & \tilde{I}^{BC} \\
0 & 0 & 0 & 0 & 0 & 0 & -\tilde{I}^{\mathbf{A}} & 0 & 0 & -\tilde{I}^{\mathbf{A}+} & -\tilde{I}^{\mathbf{A}+} & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & -\tilde{I}^{\mathbf{B}} & 0 & -\tilde{I}^{\mathbf{B}-} & 0 & -\tilde{I}^{\mathbf{B}+} \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\tilde{I}^{\mathbf{C}} & 0 & -\tilde{I}^{\mathbf{C}-} & -\tilde{I}^{\mathbf{C}-}
\end{pmatrix}
\begin{pmatrix} x^{AA} \\ x^{BB} \\ x^{CC} \\ x^{AB} \\ x^{AC} \\ x^{BC} \\ u^{AA} \\ u^{BB} \\ u^{CC} \\ u^{AB} \\ u^{AC} \\ u^{BC} \end{pmatrix}
=
\begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ b^A \\ b^B \\ b^C \end{pmatrix} \tag{4.29}$$

$$x \geq 0 \tag{4.30}$$

$$u \geq 0 \tag{4.31}$$

where Q is the node-arc incidence matrix of the network $G(N', A')$ and $\tilde{I}^{\mathbf{A}}$, $\tilde{I}^{\mathbf{B}}$, $\tilde{I}^{\mathbf{C}}$, $\tilde{I}^{AB}$, $\tilde{I}^{AC}$, $\tilde{I}^{BC}$, $-\tilde{I}^{\mathbf{A}}$, $-\tilde{I}^{\mathbf{B}}$, $-\tilde{I}^{\mathbf{C}}$, $-\tilde{I}^{\mathbf{A}+}$, $-\tilde{I}^{\mathbf{B}-}$, $-\tilde{I}^{\mathbf{B}+}$, $-\tilde{I}^{\mathbf{C}-}$ are all (n x n) matrices, which are variants of identity matrix such that

$$[\tilde{I}^{\mathbf{A}}]_{ii} = \begin{cases} -1 & \forall i \in N' : b^A(i) > 0 \\ 0 & \forall i \in N' : b^A(i) = 0 \\ 1 & \forall i \in N' : b^A(i) < 0 \end{cases}$$

$$[\tilde{I}^{\mathbf{B}}]_{ii} = \begin{cases} -1 & \forall i \in N' : b^B(i) > 0 \\ 0 & \forall i \in N' : b^B(i) = 0 \\ 1 & \forall i \in N' : b^B(i) < 0 \end{cases}$$

$$[\tilde{I}^{\mathbf{C}}]_{ii} = \begin{cases} -1 & \forall i \in N' : b^C(i) > 0 \\ 0 & \forall i \in N' : b^C(i) = 0 \\ 1 & \forall i \in N' : b^C(i) < 0 \end{cases}$$

$$[\tilde{I}^{AB}]_{ii} = \begin{cases} -1 & \forall i \in N' : b^A(i) > 0 \\ 0 & \forall i \in N' : b^A(i) \leq 0, b^B(i) \geq 0 \\ 1 & \forall i \in N' : b^B(i) < 0 \end{cases}$$

$$[\tilde{I}^{AC}]_{ii} = \begin{cases} -1 & \forall i \in N' : b^A(i) > 0 \\ 0 & \forall i \in N' : b^A(i) \leq 0, b^C(i) \geq 0 \\ 1 & \forall i \in N' : b^C(i) < 0 \end{cases}$$

$$[\tilde{I}^{BC}]_{ii} = \begin{cases} -1 & \forall i \in N' : b^B(i) > 0 \\ 0 & \forall i \in N' : b^B(i) \leq 0, b^C(i) \geq 0 \\ 1 & \forall i \in N' : b^C(i) < 0 \end{cases}$$

$$[-\tilde{I}^{\mathbf{A}}]_{ii} = \begin{cases} 1 & \forall i \in N' : b^A(i) > 0 \\ 0 & \forall i \in N' : b^A(i) = 0 \\ -1 & \forall i \in N' : b^A(i) < 0 \end{cases}$$

$$[-\tilde{I}^{\mathbf{B}}]_{ii} = \begin{cases} 1 & \forall i \in N' : b^B(i) > 0 \\ 0 & \forall i \in N' : b^B(i) = 0 \\ -1 & \forall i \in N' : b^B(i) < 0 \end{cases}$$

$$
[-\tilde{I}^{\mathbf{C}}]_{ii} = \begin{cases} 1 & \forall i \in N' : b^{C}(i) > 0 \\ 0 & \forall i \in N' : b^{C}(i) = 0 \\ -1 & \forall i \in N' : b^{C}(i) < 0 \end{cases}
$$

$$
[-\tilde{I}^{\mathbf{A}+}]_{ii} = \begin{cases} 1 & \forall i \in N' : b^{A}(i) > 0 \\ 0 & \forall i \in N' : b^{A}(i) = 0 \\ 0 & \forall i \in N' : b^{A}(i) < 0 \end{cases}
$$

$$
[-\tilde{I}^{\mathbf{B}-}]_{ii} = \begin{cases} 0 & \forall i \in N' : b^{B}(i) > 0 \\ 0 & \forall i \in N' : b^{B}(i) = 0 \\ -1 & \forall i \in N' : b^{B}(i) < 0 \end{cases}
$$

$$
[-\tilde{I}^{\mathbf{B}+}]_{ii} = \begin{cases} 1 & \forall i \in N' : b^{B}(i) > 0 \\ 0 & \forall i \in N' : b^{B}(i) = 0 \\ 0 & \forall i \in N' : b^{B}(i) < 0 \end{cases}
$$

$$
[-\tilde{I}^{\mathbf{C}-}]_{ii} = \begin{cases} 0 & \forall i \in N' : b^{C}(i) > 0 \\ 0 & \forall i \in N' : b^{C}(i) = 0 \\ -1 & \forall i \in N' : b^{C}(i) < 0 \end{cases}
$$

According to this matrix notation of the problem formulation, we investigate the unimodularity property of the constraint matrix. We exploit the matrix unimodularity through this property: if there exists at most one +1 and one -1 on a (1,0,-1) matrix, the matrix is totally unimodular. Therefore, we conduct a columnwise analysis of the constraint matrix as follows:

Property 7 *Q is a node-arc incidence matrix. Therefore, each column in Q contains only one +1 and only one -1.*

Property 8 *In the first six blocks of m columns, the only non-zero entries are the entries of Q matrices. Therefore, in each column of these six blocks, there exists only one +1 and only one -1.*

Property 9 *For the seventh block of n columns including column 6m+1 through 6m+n, the non-zero entries belong to $\tilde{I}^{\mathbf{A}}$ and $-\tilde{I}^{\mathbf{A}}$. In each column of $\tilde{I}^{\mathbf{A}}$, there exists at most either one +1 or one -1. In each column of $-\tilde{I}^{\mathbf{A}}$, the entry has the opposite sign of the entry in the same column of $\tilde{I}^{\mathbf{A}}$. Therefore, in each column there exists at most one +1 and one -1.*

Property 10 *For the eighth block of n columns including column 6m+n+1 through 6m+2n, the non-zero entries belong to $\tilde{I}^{\mathbf{B}}$ and $-\tilde{I}^{\mathbf{B}}$. In each column of $\tilde{I}^{\mathbf{B}}$, there exists at most* *either one +1 or one -1. In each column of $-\tilde{I}^{\mathbf{B}}$, the entry has the opposite sign of the entry in the same column of $\tilde{I}^{\mathbf{B}}$. Therefore, in each column there exists at most one +1 and one -1.*

Property 11 *For the ninth block of n columns including column 6m+2n+1 through 6m+3n, the non-zero entries belong to $\tilde{I}^{\mathbf{C}}$ and $-\tilde{I}^{\mathbf{C}}$. In each column of $\tilde{I}^{\mathbf{C}}$, there exists at most either one +1 or one -1. In each column of $-\tilde{I}^{\mathbf{C}}$, the entry has the opposite sign of the entry in the same column of $\tilde{I}^{\mathbf{C}}$. Therefore, in each column there exists at most one +1 and one -1.*

Property 12 *In $G(N', A')$, there exists no node $i \in N$ where $b^{\mathbf{A}}(i) > 0$ and $b^{\mathbf{B}}(i) < 0$. Thus, if $b^{\mathbf{A}}(i) > 0$, then $b^{\mathbf{B}}(i) \geq 0$.*

Property 13 *In $G(N', A')$, there exists no node $i \in N$ where $b^{\mathbf{A}}(i) > 0$ and $b^{\mathbf{C}}(i) < 0$. Thus, if $b^{\mathbf{A}}(i) > 0$, then $b^{\mathbf{C}}(i) \geq 0$.*

Property 14 *In $G(N', A')$, there exists no node $i \in N$ where $b^{\mathbf{B}}(i) > 0$ and $b^{\mathbf{C}}(i) < 0$. Thus, if $b^{\mathbf{B}}(i) > 0$, then $b^{\mathbf{C}}(i) \geq 0$.*

Property 15 *In the tenth block of columns including column 6m+3n+1 through 6m+4n, the non-zero entries belong to diagonal elements of $\tilde{I}^{AB}$, $-\tilde{I}^{\mathbf{A}+}$ and $-\tilde{I}^{\mathbf{B}-}$. Based on the entries of $\tilde{I}^{AB}$, we observe the following cases:*

> *Case 1. If there exists a -1 in column 6m+3n+i, then $b^{A}(i) > 0$. In the same column in $-\tilde{I}^{\mathbf{A}+}$, the entry is +1 since $b^{A}(i) > 0$. In the same column in $-\tilde{I}^{\mathbf{B}-}$, the nonzero entry is 0 as $b^{B}(i) \not< 0$ due to Property 12.*
> *Case 2. If there exists a +1 in column 6m+3n+i, then $b^{B}(i) < 0$. In the same column in $-\tilde{I}^{\mathbf{A}+}$, the nonzero entry is 0 since $b^{A}(i) \not> 0$ due to Property 12. In the same column in $-\tilde{I}^{\mathbf{B}-}$, the entry is -1 since $b^{B}(i) < 0$.*
> *Case 3. If there exists a 0 in column 6m+3n+i, then $b^{A}(i) \leq 0$ and $b^{B}(i) \geq 0$. In the same column in $-\tilde{I}^{\mathbf{A}+}$, the nonzero entry is 0. In the same column in $-\tilde{I}^{\mathbf{B}-}$, the nonzero entry is 0.*

Property 16 *In the eleventh block of columns including column 6m+4n+1 through 6m+5n, the non-zero entries belong to diagonal elements of $\tilde{I}^{AC}$, $-\tilde{I}^{\mathbf{A}+}$ and $-\tilde{I}^{\mathbf{C}-}$. Based on the entries of $\tilde{I}^{AC}$, we observe the following cases:*

Case 1. If there exists a -1 in column 6m+4n+i, then $b^A(i) > 0$. In the same column in $-\tilde{I}^{\mathbf{A}+}$, the entry is +1 since $b^A(i) > 0$. In the same column in $-\tilde{I}^{\mathbf{C}-}$, the nonzero entry is 0 as $b^C(i) \nleq 0$ due to Property 13.
Case 2. If there exists a +1 in column 6m+4n+i, then $b^C(i) < 0$. In the same column in $-\tilde{I}^{\mathbf{A}+}$, the nonzero entry is 0 since $b^A(i) \ngeq 0$ due to Property 13. In the same column in $-\tilde{I}^{\mathbf{C}-}$, the entry is -1 since $b^C(i) < 0$.
Case 3. If there exists a 0 in column 6m+4n+i, then $b^A(i) \leq 0$ and $b^C(i) \geq 0$. In the same column in $-\tilde{I}^{\mathbf{A}+}$, the nonzero entry is 0. In the same column in $-\tilde{I}^{\mathbf{C}-}$, the nonzero entry is 0.

Property 17 *In the twelfth block of columns including column 6m+5n+1 through 6m+6n, the non-zero entries belong to diagonal elements of $\tilde{I}^{BC}$, $-\tilde{I}^{\mathbf{B}+}$ and $-\tilde{I}^{\mathbf{C}-}$. Based on the entries of $\tilde{I}^{BC}$, we observe the following cases:*

Case 1. If there exists a -1 in column 6m+5n+i, then $b^B(i) > 0$. In the same column in $-\tilde{I}^{\mathbf{B}+}$, the entry is +1 since $b^B(i) > 0$. In the same column in $-\tilde{I}^{\mathbf{C}-}$, the nonzero entry is 0 as $b^C(i) \nleq 0$ due to Property 14.
Case 2. If there exists a +1 in column 6m+5n+i, then $b^C(i) < 0$. In the same column in $-\tilde{I}^{\mathbf{B}+}$, the nonzero entry is 0 since $b^B(i) \ngeq 0$ due to Property 14. In the same column in $-\tilde{I}^{\mathbf{C}-}$, the entry is -1 since $b^C(i) < 0$.
Case 3. If there exists a 0 in column 6m+5n+i, then $b^B(i) \leq 0$ and $b^C(i) \geq 0$. In the same column in $-\tilde{I}^{\mathbf{B}+}$, the nonzero entry is 0. In the same column in $-\tilde{I}^{\mathbf{C}-}$, the nonzero entry is 0.

Therefore, in each column, there exists at most one +1 and at most one -1.
As a result of Property 8, 9, 10, 11, 15, 16 and 17, we prove the next theorem.

Theorem 2 *The constraint matrix of the three-commodity network flow formulation with substitution is totally unimodular.*

In (4.1)-(4.27) and (4.28)-(4.29), we formulate the uncapacitated version of the 3-commodity network flow model with substitution. If w_{ij}^k denotes the commodity-specific capacity of arc $(i, j) \in A'$, then the commodity-specific capacity constraint is

$$\sum_{f \in F^k} x_{ij}^f \leq \; w_{ij}^k \;, \forall (i,j) \in A' \;, \forall k \in K \tag{4.32}$$

If v_{ij} denotes the overall capacity of arc $(i,j) \in A^{'}$, then the overall capacity constraint is

$$\sum_{f \in F} x_{ij}^{f} \leq \; v_{ij} \; , \forall (i,j) \in A^{'} \tag{4.33}$$

As a result, the capacitated version of the problem is formulated as (4.1)-(4.27) along with (4.32) and (4.33).

4.1 Non-transitivity in Substitution for Three-Commodity Network Flow Formulation

Until now, it is assumed that substitution is transitive in three-commodity network flow model. Transitivity imposes that if $\mathbf{A}$ can substitute $\mathbf{B}$, and $\mathbf{B}$ can substitute $\mathbf{C}$; $\mathbf{A}$ can substitute $\mathbf{C}$, hence the flow type AC. If the substitution is non-transitive, the flow type AC is eliminated from the mathematical model. Then, the constraint sets (4.14)-(4.16) should be completely withdrawn while constraint sets (4.20) and (4.25) change as below, respectively;

$$u^{AA}(i) + u^{AB}(i) = \; b^{\mathbf{A}}(i) \qquad \forall i \in N^{'} : b^{\mathbf{A}}(i) > 0 \tag{4.34}$$

$$- \, u^{BC}(i) - u^{CC}(i) = \; b^{\mathbf{C}}(i) \qquad \forall i \in N^{'} : b^{\mathbf{C}}(i) < 0 \tag{4.35}$$

Eliminating the flow variable AC and modifying the model as described above does not affect unimodularity. For the non-transitive problem, the mathematical model with the matrix notation is

$$\text{Minimize} \sum_{f \in F} c x^{f} \tag{4.36}$$

subject to

$$
\begin{pmatrix}
Q & 0 & 0 & 0 & 0 & \tilde{I}^{\mathbf{A}} & 0 & 0 & 0 & 0 \\
0 & Q & 0 & 0 & 0 & 0 & \tilde{I}^{\mathbf{B}} & 0 & 0 & 0 \\
0 & 0 & Q & 0 & 0 & 0 & 0 & \tilde{I}^{\mathbf{C}} & 0 & 0 \\
0 & 0 & 0 & Q & 0 & 0 & 0 & 0 & \tilde{I}^{AB} & 0 \\
0 & 0 & 0 & 0 & Q & 0 & 0 & 0 & 0 & \tilde{I}^{BC} \\
0 & 0 & 0 & 0 & 0 & -\tilde{I}^{\mathbf{A}} & 0 & 0 & -\tilde{I}^{\mathbf{A}+} & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & -\tilde{I}^{\mathbf{B}} & 0 & -\tilde{I}^{\mathbf{B}-} & -\tilde{I}^{\mathbf{B}+} \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & -\tilde{I}^{\mathbf{C}} & 0 & -\tilde{I}^{\mathbf{C}-}
\end{pmatrix}
\begin{pmatrix}
x^{AA} \\ x^{BB} \\ x^{CC} \\ x^{AB} \\ x^{BC} \\ u^{AA} \\ u^{BB} \\ u^{CC} \\ u^{AB} \\ u^{BC}
\end{pmatrix}
=
\begin{pmatrix}
0 \\ 0 \\ 0 \\ 0 \\ 0 \\ b^{A} \\ b^{B} \\ b^{C}
\end{pmatrix}
\tag{4.37}
$$

$$x \geq 0 \tag{4.38}$$

$$u \geq 0 \tag{4.39}$$

As in (4.37), two columns and one row is eliminated from (4.29). This would not affect the unimodularity of the model.

4.2 Computational Complexity of the Capacitated Three-commodity Network Flow Problem

We discuss the computational complexity of the capacitated three-commodity network flow problem with substitution through its relation with the common three-commodity network flow problem as it is well known that the common problem is NP-hard. The three-commodity network flow problem with substitution is a generalized version of the common three-commodity network flow problem. In other words, the common 3-commodity network flow problem is a special case of 3-commodity network flow problem with substitution. In order to show this relationship formally, we work with the mathematical model (4.1)-(4.27), (4.32), (4.33) of the problem with substitution. We need to show that our mathematical model is the same as the network flow model of the common problem when substitution is not allowed. In order to avoid substitution, we first eliminate the flow variables AB, AC and BC since they represent substitution. Then, constraint sets (4.11)-(4.19) drop completely from the model. Moreover, constraint sets (4.20), (4.22), (4.23) and (4.25) become:

$$\begin{aligned}
u^{AA}(i) &= b^{\mathbf{A}}(i) && \forall i \in N^{'} : b^{\mathbf{A}}(i) > 0 && (4.40)\\
u^{BB}(i) &= b^{\mathbf{B}}(i) && \forall i \in N^{'} : b^{\mathbf{B}}(i) > 0 && (4.41)\\
-u^{BB}(i) &= b^{\mathbf{B}}(i) && \forall i \in N^{'} : b^{\mathbf{B}}(i) < 0 && (4.42)\\
-u^{CC}(i) &= b^{\mathbf{C}}(i) && \forall i \in N^{'} : b^{\mathbf{C}}(i) < 0 && (4.43)
\end{aligned}$$

With this arrangement, we now do not need u variables any more and we can eliminate constraint sets (4.40), (4.21), (4.41), (4.42), (4.24) , (4.43) and (4.27) by replacing $u^{AA}(i)$ in constraint set (4.2) with $b^{\mathbf{A}}(i)$ and $u^{BB}(i)$ in constraint set (4.5) with $b^{\mathbf{B}}(i)$, eliminating constraint sets (4.40) and (4.41), respectively. Moreover, we replace $-u^{AA}(i)$ in constraint set (4.3) with $b^{\mathbf{A}}(i)$ and $-u^{BB}(i)$ in constraint set (4.6) with $b^{\mathbf{B}}(i)$, eliminating constraint sets (4.21) and (4.42), respectively. Also, we replace $u^{CC}(i)$ in constraint set (4.8) with $b^{\mathbf{C}}(i)$ and $-u^{CC}(i)$ in constraint set (4.9) with $b^{\mathbf{C}}(i)$, eliminating constraint sets (4.24) and (4.43), respectively. Finally, we replace zeros in constraint sets (4.4), (4.7) and (4.10) with $b^{\mathbf{A}}(i)$, $b^{\mathbf{B}}(i)$ and $b^{\mathbf{C}}(i)$ respectively because $b^{\mathbf{A}}(i) = 0$, $b^{\mathbf{B}}(i) =$ and $b^{\mathbf{C}}(i) = 0$ for (4.4), (4.7) and (4.10).

As a result, constraint sets (4.2)-(4.4) become

$$\begin{aligned}
\sum_{j:(i,j)\in A'} x_{ij}^{AA} - \sum_{j:(j,i)\in A'} x_{ji}^{AA} &= b^{\mathbf{A}}(i), && \forall i \in N', b^{\mathbf{A}}(i) > 0, && (4.44)\\
\sum_{j:(i,j)\in A'} x_{ij}^{AA} - \sum_{j:(j,i)\in A'} x_{ji}^{AA} &= b^{\mathbf{A}}(i)), && \forall i \in N', b^{\mathbf{A}}(i) < 0, && (4.45)\\
\sum_{j:(i,j)\in A'} x_{ij}^{AA} - \sum_{j:(j,i)\in A'} x_{ji}^{AA} &= b^{\mathbf{A}}(i), && \forall i \in N', b^{\mathbf{A}}(i) = 0. && (4.46)
\end{aligned}$$

(4.44)-(4.46) can be rewritten as

$$\sum_{j:(i,j)\in A'} x_{ij}^{AA} - \sum_{j:(j,i)\in A'} x_{ji}^{AA} = b^{\mathbf{A}}(i) \qquad \forall i \in N^{'} \qquad (4.47)$$

Constraint sets (4.5)-(4.7) become

$$\sum_{j:(i,j)\in A'} x_{ij}^{BB} - \sum_{j:(j,i)\in A'} x_{ji}^{BB} = b^{\mathbf{B}}(i), \qquad \forall i \in N^{'}, b^{\mathbf{B}}(i) > 0, \tag{4.48}$$

$$\sum_{j:(i,j)\in A'} x_{ij}^{BB} - \sum_{j:(j,i)\in A'} x_{ji}^{BB} = b^{\mathbf{B}}(i), \qquad \forall i \in N^{'}, b^{\mathbf{B}}(i) < 0, \tag{4.49}$$

$$\sum_{j:(i,j)\in A'} x_{ij}^{BB} - \sum_{j:(j,i)\in A'} x_{ji}^{BB} = b^{\mathbf{B}}(i), \qquad \forall i \in N^{'}, b^{\mathbf{B}}(i) = 0. \tag{4.50}$$

Similarly, (4.48)-(4.50) can be rewritten as

$$\sum_{j:(i,j)\in A'} x_{ij}^{BB} - \sum_{j:(j,i)\in A'} x_{ji}^{BB} = b^{\mathbf{B}}(i) \qquad \forall i \in N^{'} \tag{4.51}$$

Also, constraint sets (4.8)-(4.10) become

$$\sum_{j:(i,j)\in A'} x_{ij}^{CC} - \sum_{j:(j,i)\in A'} x_{ji}^{CC} = b^{\mathbf{C}}(i), \qquad \forall i \in N^{'}, b^{\mathbf{C}}(i) > 0, \tag{4.52}$$

$$\sum_{j:(i,j)\in A'} x_{ij}^{CC} - \sum_{j:(j,i)\in A'} x_{ji}^{CC} = b^{\mathbf{C}}(i), \qquad \forall i \in N^{'}, b^{\mathbf{C}}(i) < 0, \tag{4.53}$$

$$\sum_{j:(i,j)\in A'} x_{ij}^{CC} - \sum_{j:(j,i)\in A'} x_{ji}^{CC} = b^{\mathbf{C}}(i), \qquad \forall i \in N^{'}, b^{\mathbf{C}}(i) = 0. \tag{4.54}$$

Similarly, (4.52)-(4.54) can be rewritten as

$$\sum_{j:(i,j)\in A'} x_{ij}^{CC} - \sum_{j:(j,i)\in A'} x_{ji}^{CC} = b^{\mathbf{C}}(i) \qquad \forall i \in N^{'} \tag{4.55}$$

Capacity constraints do not change

$$\sum_{f\in F^k} x_{ij}^{f} \leq w_{ij}^{k} \,, \forall (i,j) \in A^{'} \,, \forall k \in K \tag{4.32}$$

$$\sum_{f\in F} x_{ij}^{f} \leq v_{ij} \,, \forall (i,j) \in A^{'} \tag{4.33}$$

Then, the mathematical model becomes

$$\text{Minimize} \sum_{(i,j)\in A'} c_{ij}x_{ij}^{AA} + c_{ij}x_{ij}^{BB} + c_{ij}x_{ij}^{CC} \tag{4.56}$$

$$\text{subject to} \sum_{j:(i,j)\in A'} x_{ij}^{AA} - \sum_{j:(j,i)\in A'} x_{ji}^{AA} = b^{\mathbf{A}}(i) \qquad \forall i \in N' \tag{4.47}$$

$$\sum_{j:(i,j)\in A'} x_{ij}^{BB} - \sum_{j:(j,i)\in A'} x_{ji}^{BB} = b^{\mathbf{B}}(i) \qquad \forall i \in N' \tag{4.51}$$

$$\sum_{j:(i,j)\in A'} x_{ij}^{CC} - \sum_{j:(j,i)\in A'} x_{ji}^{CC} = b^{\mathbf{C}}(i) \qquad \forall i \in N' \tag{4.55}$$

$$\sum_{f\in F^k} x_{ij}^{f} \le w_{ij}^{k} \qquad \forall (i,j) \in A', \forall k \in K \tag{4.32}$$

$$\sum_{f\in F} x_{ij}^{f} \le v_{ij} \qquad \forall (i,j) \in A' \tag{4.33}$$

$$x_{ij}^{f} \ge 0 \qquad \forall (i,j) \in A', \forall f \in F \tag{4.57}$$

The mathematical model (4.56), (4.47), (4.51), (4.55), (4.57), (4.32), (4.33) is equivalent to the network flow model of the three-commodity problem. Therefore, we have shown that our problem is a generalization of the common three-commodity network flow problem and it is at least as hard as the common problem. Since the integer version of the common problem is NP-hard as pointed out by Ahuja, Magnanti and Orlin [2], our problem is also NP-hard.

4.3 Computational Experiments

In order to investigate how the size of the problem along with the capacity tightness affect the model, we conduct computational experiments that explore the change in the solution time of the problem with respect to the size and the arc capacities. For this purpose, we build the mathematical model on CPLEX Studio 12.4 environment. We first generate random instances of the capacitated 3-commodity network flow problem with substitution and solve both the linear programming relaxation and integer programming problem. Computational experiments are conducted on a 8-core computer with a Intel Core i7 CPU @3.20 gHz and 24.0 GB of RAM. CPU times and objective function values are reported after the instances are solved.

Input parameters for three-commodity network flow problem instances are the same as the input parameters in two-commodity network flow problem instances with the addition of a distribution parameter of the abundant supply of commodity $\mathbf{A}$ between the shortages of supply in commodities $\mathbf{B}$ and $\mathbf{C}$, which is assigned as 0.5. Also, we had to determine a commodity-specific capacity density for commodity $\mathbf{C}$, which is also determined as 0.5. Other than that, computational environment is totally similar to two-commodity network flow problem instances.

While investigating the effect of capacities and network size, we use a hypothesis testing framework to determine whether our inferences are statistically significant or not. In this manner, we would like to answer the following questions:

- How much of an effect do commodity-specific capacity constraints have on solution time?
- How much of an effect do overall capacity constraints have on solution time?
- How much of an effect do both capacity constraints together have on solution time?
- How much of an effect does inclusion of overall capacity constraints over commodity-specific capacity constraints have on solution time? (i.e. sum of commodity-specific capacities are introduced as overall capacities.)
- How much of an effect does the tightening of capacity constraints have on solution time?
- How much of an effect does problem size have on solution time?

Two-sample two-tailed t-tests are used for all hypotheses because we would like to compare means of two samples (solution times of different configurations). The null hypotheses assume that their means are equal. StatTools software is used to perform the tests. The results are presented in Table (4.1) through Table (4.7).

Table 4.1: Results of hypothesis testing for comparing the solution times of configuration with commodity-specific capacities and uncapacitated configuration

	Uncapacitated	**Commodity-specific**
Sample Summaries	**Data Set #1**	**Data Set #1**
Sample Size	60	59
Sample Mean	10948.77	14574.60
Sample Std Dev	14197.66	18916.64
Hypothesis Test (Difference of Means)	**Equal Variances**	**Unequal Variances**
Hypothesized Mean Difference	0	0
Alternative Hypothesis	$\neq 0$	$\neq 0$
Sample Mean Difference	-3625.83	-3625.83
Standard Error of Difference	3062.696483	3069.956007
Degrees of Freedom	117	107
t-Test Statistic	-1.1839	-1.1811
p-Value	0.2389	0.2402

Table 4.2: Results of hypothesis testing for comparing the solution times of configuration with overall capacities and uncapacitated configuration

	Uncapacitated	**Overall**
Sample Summaries	**Data Set #1**	**Data Set #1**
Sample Size	60	60
Sample Mean	10948.77	12305.04
Sample Std Dev	14197.66	16078.59
Hypothesis Test (Difference of Means)	**Equal Variances**	**Unequal Variances**
Hypothesized Mean Difference	0	0
Alternative Hypothesis	$\neq 0$	$\neq 0$
Sample Mean Difference	-1356.28	-1356.28
Standard Error of Difference	2769.159512	2769.159512
Degrees of Freedom	118	116
t-Test Statistic	-0.4898	-0.4898
p-Value	0.6252	0.6252

Table 4.3: Results of hypothesis testing for comparing the solution times of configuration with both commodity-specific and overall capacities and uncapacitated configuration

	Uncapacitated	**Full Capacity**
Sample Summaries	**Data Set #1**	**Data Set #1**
Sample Size	60	55
Sample Mean	10948.77	16808.06
Sample Std Dev	14197.66	20765.94
Hypothesis Test (Difference of Means)	**Equal Variances**	**Unequal Variances**
Hypothesized Mean Difference	0	0
Alternative Hypothesis	$\neq 0$	$\neq 0$
Sample Mean Difference	-5859.29	-5859.29
Standard Error of Difference	3293.776501	3346.639877
Degrees of Freedom	113	94
t-Test Statistic	-1.7789	-1.7508
p-Value	0.0779	0.0832

Table 4.4: Results of hypothesis testing for comparing the solution times of configuration with overall capacities calculated as commodity-specific capacities and configuration with commodity-specific capacities

	Commodity-specific	**Sum of Commodity-specific**
Sample Summaries	**Data Set #1**	**Data Set #1**
Sample Size	59	59
Sample Mean	14574.60	15331.96
Sample Std Dev	18916.64	19960.64
Hypothesis Test (Difference of Means)	**Equal Variances**	**Unequal Variances**
Hypothesized Mean Difference	0	0
Alternative Hypothesis	$\neq 0$	$\neq 0$
Sample Mean Difference	-757.36	-757.36
Standard Error of Difference	3580.233373	3580.233373
Degrees of Freedom	116	115
t-Test Statistic	-0.2115	-0.2115
p-Value	0.8328	0.8328

Table 4.5: Results of hypothesis testing for comparing the solution times of levels for configuration with commodity-specific capacities

	Level 1	**Level 5**
Sample Summaries	**Data Set #2**	**Data Set #2**
Sample Size	59	59
Sample Mean	14540.83	14596.93
Sample Std Dev	18525.37	19352.75
Hypothesis Test (Difference of Means)	**Equal Variances**	**Unequal Variances**
Hypothesized Mean Difference	0	0
Alternative Hypothesis	$\neq 0$	$\neq 0$
Sample Mean Difference	-56.10	-56.10
Standard Error of Difference	3487.794979	3487.794979
Degrees of Freedom	116	115
t-Test Statistic	-0.0161	-0.0161
p-Value	0.9872	0.9872

Table 4.6: Results of hypothesis testing for comparing the solution times of levels for configuration with overall capacities

	Level 1	**Level 3**
Sample Summaries	**Data Set #3**	**Data Set #3**
Sample Size	60	60
Sample Mean	12727.00	12051.88
Sample Std Dev	16424.77	15671.32
Hypothesis Test (Difference of Means)	**Equal Variances**	**Unequal Variances**
Hypothesized Mean Difference	0	0
Alternative Hypothesis	$\neq 0$	$\neq 0$
Sample Mean Difference	675.12	675.12
Standard Error of Difference	2930.766442	2930.766442
Degrees of Freedom	118	117
t-Test Statistic	0.2304	0.2304
p-Value	0.8182	0.8182

Table 4.7: Results of hypothesis testing for comparing the solution times of problem sizes for uncapacitated configuration

	10	**50**
Sample Summaries	**Data Set #4**	**Data Set #4**
Sample Size	10	10
Sample Mean	1210.70	1456.60
Sample Std Dev	36.78	22.16
Hypothesis Test (Difference of Means)	**Equal Variances**	**Unequal Variances**
Hypothesized Mean Difference	0	0
Alternative Hypothesis	$\neq 0$	$\neq 0$
Sample Mean Difference	-245.90	-245.90
Standard Error of Difference	13.58041891	13.58041891
Degrees of Freedom	18	14
t-Test Statistic	-18.1070	-18.1070
p-Value	< 0.0001	< 0.0001

The only significant results are achieved when the capacity configuration with both commodity - specific and overall capacities are compared against uncapacitated configuration (See Table (4.3) and different problem sizes are compared against each other within the uncapacitated configuration (See Table (4.7)). For the configuration with both capacities, null hypothesis is rejected at 10% significance level since its p-value is around 0.09 (See Table (4.3). For problem sizes, p-value is significantly small (See Table (4.7)) so that null hypothesis is rejected at any significance level. Therefore, we conclude that when both capacities are active, solution time significantly increases. Moreover, as problem size increases, solution time significantly increases. Thus, we imply that if both capacities are active in the problem or if the problem size is increased, the problem becomes more difficult to solve.

Chapter 5

Multicommodity Network Flow Formulation with Substitution

So far, we have covered two and three-commodity network flow problems with substitution. In this chapter, we present the generalized mathematical model of our problem. Within the boundaries of this study, we only present the model and do not perform any computational complexity analysis. It is a research area that we hope to pursue in the future.

We must first introduce a generalization structure for our model. First of all, the flow type $f \in F$ is composed of two indices in our previous models: one index for the supplying commodity and one index for the demanding commodity. To exemplify, the flow type AB represents the flow of commodity $\mathbf{A}$ for commodity $\mathbf{B}$. Thus, in flow type AB, $\mathbf{A}$ is the supplying commodity and $\mathbf{B}$ is the demanding commodity. In order to generalize the flow types, we define $f \in F$ with its contents: (s, d) pairs in F. $s \in K$ is the index for the supplying commodity and $d \in K$ is the index for the demanding commodity. Secondly, we define two new commodity subsets within K related with the new definition of the flow types: $D^s \in K$ and $S^d \in K$. D^s stands for the demanding commodities of commodity s and S^d stands for the supplying commodities of commodity d. To exemplify, $D^{\mathbf{A}}$ includes commodities $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$ in the three-commodity problem because the demand of these commodities can be satisfied using the commodity $\mathbf{A}$. Furthermore, $S^{\mathbf{B}}$ includes commodities $\mathbf{A}$ and $\mathbf{B}$ in two or three-commodity problem because the demand of commodity $\mathbf{B}$ can be satisfied using these commodities. Assuming the network is transformed to reflect the within-node supply possibility and using other notation as before, the linear programming formulation of the n-commodity network flow problem with substitution is

$$\text{Minimize} \sum_{(i,j)\in A'} \sum_{(s,d)\in F} c_{ij} x_{ij}^{sd} \tag{5.1}$$

$$\text{subject to} \sum_{j:(i,j)\in A'} x_{ij}^{sd} - \sum_{j:(j,i)\in A'} x_{ji}^{sd} = u^{sd}(i) \quad \forall (s,d) \in F, \forall i \in N^{'} : b^s(i) > 0 \tag{5.2}$$

$$\sum_{j:(i,j)\in A'} x_{ij}^{sd} - \sum_{j:(j,i)\in A'} x_{ji}^{sd} = -u^{sd}(i) \quad \forall (s,d) \in F, \forall i \in N^{'} : b^d(i) < 0 \tag{5.3}$$

$$\sum_{j:(i,j)\in A'} x_{ij}^{sd} - \sum_{j:(j,i)\in A'} x_{ji}^{sd} = 0 \quad \forall (s,d) \in F, \forall i \in N^{'} : b^s(i) \leq 0, b^d(i) \geq 0 \tag{5.4}$$

$$\sum_{d\in D^s} u^{sd}(i) = b^s(i) \quad \forall s \in K, \forall i \in N : b^s(i) > 0 \tag{5.5}$$

$$-\sum_{d\in S^d} u^{sd}(i) = b^d(i) \quad \forall d \in K, \forall i \in N : b^d(i) < 0 \tag{5.6}$$

$$\sum_{(s,d)\in F^k} x_{ij}^{sd} \leq w_{ij}^s \quad \forall (i,j) \in A^{'}, \forall s \in K \tag{5.7}$$

$$\sum_{(s,d)\in F} x_{ij}^{sd} \leq v_{ij} \quad \forall (i,j) \in A^{'} \tag{5.8}$$

$$x_{ij}^{sd} \geq 0 \quad \forall (i,j) \in A^{'}, \forall (s,d) \in F \tag{5.9}$$

$$u^{sd}(i) \geq 0 \quad \forall (s,d) \in F, \forall i \in N^{'} \tag{5.10}$$

The objective function (5.1) minimizes the total cost of flow on all arcs. For supplying commodity $\mathbf{s}$ and demanding commodity $\mathbf{d}$, constraint sets (5.2) and (5.3) calculates the net out-flow on node i as $u^{sd}(i)$, respectively for supply nodes of commodity $\mathbf{s}$ and demand nodes of commodity $\mathbf{d}$. Since out-flow is greater than in-flow for supply nodes and vice-versa for demand nodes, u^{sd} is multiplied by -1 on the right hand side of constraint (5.3). For other nodes, constraint set (5.4) ensures that difference between out-flow and in-flow of flow type sd is zero. For supply nodes of commodity $\mathbf{s}$, constraint set (5.5) calculates the sum of net out-flow of flow types sd as supply of commodity $\mathbf{s}$. For demand nodes of commodity $\mathbf{d}$, constraint set (5.6) calculates net out-flow of flow types sd as demand of commodity $\mathbf{d}$. Constraint sets (5.7) and (5.8) are capacity constraints. Constraint sets (5.9) - (5.10) define variable domains, which are all non-negative.

Chapter 6

Concluding Remarks

In this study, we present a generalizing variant of the multicommodity network flow problem where substitution is possible among commodities. We develop mathematical models for two-commodity and three-commodity network flow problems with substitution. To the best of our knowledge, substitutable commodities are considered in two problem environments in the literature: inventory planning and empty container allocation. However, a network flow model with substitution that does not depend on problem environment was not previously developed earlier. New concepts such as flow types and within-node supply are introduced to reflect the nature of the problem for the very first time. Theorems regarding complexity of the uncapacitated problem are proposed. Specifically, constraint matrix of the uncapacitated problem is proven to be totally unimodular. Moreover, empirical computational difficulty of capacitated versions of the problem formulations is investigated through a computational study with randomly generated problems and statistical analysis with hypothesis testing. In particular, the effect of capacities and the problem size on solution time are explored.

Our results show that solution time significantly increases for both two-commodity and three-commodity problems when both overall and commodity-specific capacities exist. Solution time significantly increases when problem size is increased. Unfortunately, our computational experiments are not conclusive, specifically because of the fact that LP-relaxations of even the capacitated models give integral solutions. We believe that other problem generation methods should be implemented. In addition, more instances should be generated and more experiments should be conducted while implementing other methods.

We conclude by pointing out that there is a grand research potential in network flow problems with substitution. To exemplify,

- In our model, it is assumed that cost structure is linear. Convex cost flows can be introduced to multicommodity network flow with substitution, therefore generalizing the model further.
- Different substitution structures can be included in the model. For example, if a superior commodity can substitute two inferior commodities, there occurs a new substitution structure that must be represented within the model.
- We assumed that substitution is costless. Substitution cost can also be considered.
- As mentioned before, the computational analysis of the n-commodity model is also a future research area.
- Last but not least, real-life problems can be solved using the models presented in this study.

Above points are the avenue of research that we hope to pursue in the future.

Bibliography

[1] Roberto F. Aguilera and Ronald D. Ripple. Modeling primary energy substitution in the asia pacic. *Applied Energy*, 111:219–224, 2013.

[2] Ravindra K. Ahuja, Thomas L. Magnanti, and James B. Orlin. *Network flows: theory, algorithms and applications*. Prentice–Hall, 1993.

[3] Yehuda Bassok, Ravi Anupindi, and Ram Akella. Single-period multiproduct inventory models with substitution. *Operations Research*, 47(4):632–642, 1999.

[4] Z. Pelin Bayindir, Nesim Erkip, and Refik Gullu. Assessing the benets of remanufacturing option under one-way substitution. *Journal of the Operational Research Society*, 56:286–296, 2005.

[5] Hwan Chang, Hossein Jula, Anastasios Chassiakos, and Petros Ioannou. Empty container reuse in the los angeles/long beach port area. In *Proceedings of the National Urban Freight Conference*, 2006.

[6] Hwan Chang, Hossein Jula, Anastasios Chassiakos, and Petros Ioannou. A heuristic solution for the empty container substitution problem. *Transportation Research*, 44:203–216, 2008.

[7] Theodor Gabriel Crainic, Michel Gendreau, and Pierre Dejax. Dynamic and stochastic models for the allocation of empty containers. *Operations Research*, 41(1):102–126, 1993.

[8] Yannick Deflem and Inneke Van Nieuwenhuyse. Managing inventories with one-way substitution: A newsvendor analysis. *European Journal of Operational Research*, 228(3):484–493, 2013.

[9] Massimo Di Francesco, Antonio Manca, Alessandro Olivo, and Paola Zuddas. Optimal management of heterogenous fleets of empty containers. In *Proceedings of In-*

ternational Conference on Information Systems, Logistics and Supply Chain, pages 922–931, 2006.

[10] Mark S. Hillier. Using commonality as backup safety stock. *European Journal of Operational Research*, 136:353–365, 2002.

[11] S. Iniyan, L. Suganthi, and Anand A. Samuel. Energy models for commercial energy prediction and substitution of renewable energy sources. *Energy Policy*, 34:2640–2653, 2006.

[12] Petros Ioannou, Anastasios Chassiakos, Hossein Jula, Hwan Chang, and Gil Valencia. Development of methods for handling empty containers with applications in the los angeles/long beach port area. Technical report, METRANS Transportation Center, 2006.

[13] Arlindo Kamimura, Sinclair Mallet Guy Guerra, and Ildo Luis Sauer. On the substitution of energy sources: Prospective of the natural gas market share in the brazilian urban transportation and dwelling sectors. *Energy Policy*, 34:3583–3590, 2006.

[14] Moutaz Khouja, Abraham Mehrez, and Gad Rabinowitz. A two-item newsboy problem with substitutability. *International Journal of Production Economics*, 44:267–275, 1996.

[15] John Arun Kumar and Radhakrishna Chebiyam. Thermal power system analysis using a generalized network flow model. *International Journal of Energy And Environment*, 3(3):447–460, 2013.

[16] John Arun Kumar and C. Radhakrishna. Thermal power generation and ghg emission projections for india using a network flow model. *The IUP Journal of Electrical and Electronics Engineering*, 5(1):56–82, 2012.

[17] Jun Liu and Chi-Guhn Lee. Evaluation of inventory policies with unidirectional substitutions. *European Journal of Operational Research*, 182:145–163, 2007.

[18] Ana Quelhas, Esteban Gil, James D. McCalley, and Sarah M. Ryan. A multiperiod generalized network flow model of the u.s. integrated energy system: Part i-model description. *IEEE Transactions on Power Systems*, 22(2):829–836, 2007.

[19] Ana Quelhas and James D. McCalley. A multiperiod generalized network flow model of the u.s. integrated energy system: Part ii-simulation results. *IEEE Transactions on Power Systems*, 22(2):837–844, 2007.

[20] Ronald S. Tibben-Lembke and Yehuda Bassok. An inventory model for delayed customization: A hybrid approach. *European Journal of Operational Research*, 165:748–764, 2005.

[21] Anke Weidlich. *Engineering Interrelated Electricity Markets: An Agent-Based Computational Approach*. Physica-Verlag, 2008.

[22] Zhao Xingang and Liu Pingkuo. Substitution among energy sources: An empirical analysis on biomass energy for fossil fuel of china. *Renewable and Sustainable Energy Reviews*, 18:194–202, 2013.