FACILITY LOCATION WITH SUPPLIER SELECTION UNDER QUANTITY DISCOUNT

A Thesis

by

Görkem Emirhüseyinoğlu

Submitted to the
Graduate School of Sciences and Engineering
In Partial Fulfillment of the Requirements for
the Degree of

Master of Science

in the
Department of Industrial Engineering

Özyeğin University
July 2017

FACILITY LOCATION WITH SUPPLIER SELECTION UNDER QUANTITY DISCOUNT

Approved by:

Associate Professor Ali Ekici, Advisor
Department of Industrial Engineering
Özyeğin University

Associate Professor Erhun Kundakçıoğlu
Department of Industrial Engineering
Özyeğin University

Associate Professor Serhan Duran
Department of Industrial Engineering
Middle East Technical University

Date Approved: 06.07.2017

To My Parents

ABSTRACT

The purpose of this study is to identify the locations of distribution facilities and choose the suppliers in a two echelon system where items are purchased from suppliers under quantity discount policy and distributed to customers from the facilities so as to meet the demand. We formulate the problem on hand as a mixed integer programming problem. Moreover, to handle large instances we develop heuristic algorithms one of which is a mixed integer programming based iterative algorithm where we solve the problem in each echelon in an iterative manner. Finally, we test the proposed model and heuristic algorithms on different data sets.

Keywords: distribution facility location; supplier selection; quantity discount

ÖZETÇE

Bu tez, içerisinde dağıtım tesisleri, tedarikçi ve müşterilerin bulunduğu iki kademeli bir tedarik zinciri ağı iyileştirme çalışmasıdır. Dağıtımcının ürünü tedarikçilerden miktara bağlı indirim ile temin ettiği ve müşterilere teslim ettiği bir sistem yapısı üzerine çalışılmıştır. Amaç dağıtım tesislerinin kurulacağı yerleri tespit etmek ve doğru tedarikçileri seçerek maliyeti en aza indirgemektir. Eldeki problem karışık tamsayı doğrusal programlama problemi olarak formüle edilmiştir. Ayrıca, büyük boyuttaki problemleri çözebilmek için her kademeyi tekrarlanan şekilde çözen karışık tam sayı programlama tabanlı sezgisel çözüm yolları geliştirilmiştir. Son olarak önerilen sezgisel yöntemler farklı veri kümelerinde test edilmiştir.

Anahtar Kelimeler: dağıtım tesisi tespiti; tedarikçi seçimi; miktara bağlı indirim

ACKNOWLEDGEMENTS

Firstly, I would like to express my profound gratitude to my advisor Ali Ekici for his patience, guidance and continuous support during my M.Sc. study. I also would like to take this opportunity to record my sincere thanks to all the faculty members of the Department of Industrial Engineering at Özyeğin University for all their help to develop my skills as a researcher and future practitioner.

I would like to express my sincere appreciation to my friends and officemates, especially Cem Deniz Çaglar Bozkır, Hamed Shourabizadeh, Ömer Hikmet Sevindik, Büşra Uydaşoğlu, Tonguç Yavuz, Gizem Atasoy and Emre Çankaya, who supported me with their invaluable friendship, comments and encouragement.

Finally, I am truly grateful to my mom and dad, Suna and Melih Emirhüseyinoğlu, for their patience and unconditional love. I never would have been able to achieve my goals without their continuous support during my entire education.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION

Supply chain networks generally consists of several entities such as plants, warehouses and distributors. In a supply chain network, due to excessive effort required or economic reasons, very few entities are solely responsible from every stage. In most cases, it is necessary to get additional support, provide some services by outsourcing or settle with intermediary establishments. Every piece of this chain, from very first step, such as acquisition of raw materials to obtain intermediary and final products or delivery of items to end customers, is bound to work with perfect balance. A small problem in one of the pieces might hinder entire chain. Hence, establishing a successful supply chain network can only be achieved through successful planning. Nowadays, developed markets require multi-echelon planning in order to strengthen their operational level activities which is the main motivation behind the researchers' growing interest in such network structures. Investigating strategic supply chain decisions one at a time however, may not deliver desired results and solve supply chain related problems as expected. Integrated approaches on the other hand, deal with issues in a broader perspective and can lead us to make better decisions. [1] underlines the importance of integrated decisions by providing some industry related studies involving pharmaceutical, automotive, beverage and forestry industries. We refer the reader to [2] for a review of studies analyzing integrated models in supply chain design.

Locating facilities is a crucial strategic decision and play a pivotal role while designing supply chain networks. However, location studies generally are not considered as part of supply chain design especially in early literature [3]. [4] and [5] remarks that earlier studies had difficulty to embrace the real life features of location problems. That circumstance reduces the practicality of previous studies in real life. Thus, despite the fact that a vast amount of location research is conducted so far, location problems still attract interest in order to address real-life problems. As a result, over the last few decades, interest in integration of facility location models with other important supply chain decisions increased considerably.

[6] provided an early review about supply chain problems including studies about location analysis while [3] specifically explored integrated location and supply chain decisions. Supplier selection is one of the decisions that has to be made in conjunction with location decisions. [7] explains the importance of making correct decisions when choosing suppliers with right price and quantities and the advantages it might bring. Those decisions might be challenging since there are a lot of issues to consider especially with a possible cost burden that might come along with such decision. Supplier selection is commonly referred as a strategic decision in literature since those decisions are essential while forming supply chain networks [8, 9, 10].

[11] points out the purchasing decisions as the most strategic ones among other strategic decisions. Fully economic reasons underlie beneath this claim. Previous practices indicate that purchasing costs correspond to a huge portion of supply chain. That portion may reach up to 80% of the entire expense especially in motor and technology companies [12, 13, 14, 15]. Hence as [7] also suggested, it is no doubt that supplier selection is one of the most crucial decisions faced by purchasing departments. While purchasing department is responsible for generation of such huge portion of total expenses, it may be inefficient to exclude their activities during the planning of strategic decisions such as facility location.

In previous studies, unit purchase prices are generally considered as fixed especially by the ones analyzing location problems [4, 16]. [17] on the other hand, reports it is far from reality since the prices are open to discussion and there are even situations where the customers themselves negotiate to get quantity discount schemes. As a common sales tactic, suppliers use quantity discounts with hope of increasing their sales or gain some benefits on operating level [18]. Previous practices demonstrate that discount schemes are advantegeous economically for both suppliers and their customers [19]. [20] indicates that suppliers usually would prefer incremental discount since discounted prices will be effective only with additional items surpassing the predetermined quantities and selling $n+1$ items with less price compared to selling n items may not make much sense. On the other hand, there are other studies rejecting that suggestion and in fact, all unit schemes are generally preferred so far in the majority of quantity discount literature.

In this study, we work on a supply chain network where manufactured products by suppliers sold or sent to either distributors or warehouses so that those products could be delivered to end customers. In literature, there are real life applications with similar structures. Those applications are most commonly seen on telecommunication, health-care and spare parts industries [21, 22]. Studies such as [23, 24, 25] are some examples establishing multi-echelon networks for spare parts industry. After investigated discount policies, we notice that suppliers working with a quantity discount scheme is also a common practice. Quantity discount schemes offered by suppliers are widely studied in supply chain contracts design literature. Beverage and airline companies are known establishments working with such contracts [26]. A specific and known example of quantity discount scheme is the case of Michelin with its tire dealers [27]. According to this case, Michelin exerted several strategies in order to increase and protect their market share one of which is an all unit discount policy. With this policy, they settled more than 10 discount rates for specified price segments.

Normally it is assumed that important strategic decisions such as facility opening are not short termed decisions especially since they require a great amount of effort and money. Making those strategic decisions might be tricky in particular when you anticipate parameters such as customer demand or government regulations will change over time. According to yearly global Mckinsey survey results [28], it's quite rare for goods to appeal customers at the same level for a long time. That kind of fluctuations and uncertainty at the markets is believed to be one of the biggest issues for supply chain networks. In literature, different time-varying factors are suggested in order to validate dynamic structure of location problems. See [4] for a comprehensive overview about those factors. As a result, firms sometimes may be obliged to change their supply chain networks in order to keep up with new trends, accommodate new regulations or reach to new customer segments. Those kind of decisions however may be threatening for directors since it's really difficult to alter supply chain network at the right time and right place whilst it's economically advantageous. Dynamic models emerged to answer those concerns and make more accurate decisions. As [29] underlined, previous studies show that altering supply chain networks may lead to a significant amount of decrease in costs (up to fifteen percent).

Many different assumptions for location decisions are made by researchers while analyzing dynamic models. There are countless cases such as not allowing the closure of facilities at each time period because of high closure costs or allowing relocation of facilities due to depreciation. Here, we will mention two articles to exemplify different decision scenarios. [30] consider only opening decisions at different time periods. On the other hand, [31] discuss two different decision scenarios. First decision scenario involves both opening and closing decisions at different time periods while the second scenario only allows opening decisions for once at the beginning of planning horizon. We refer the reader to [4] for an exhaustive summary about different scenarios.

According to [32], although dynamic models are useful while dealing with time-varying elements in the problem, it comes with additional traits which might reduce practicality such as difficulty of determining appropriate time segments and planning horizon or difficulty of collecting a huge data. But most importantly, it drastically increases the complexity of the problem and reduces the possibility of solving it. As discussed above and mentioned by [33], a great amount of money and energy are required to locate facilities. And hence opening decisions are generally effective for long time spans (at least more than a couple of years). Unless you are about to study a really long planning horizon, maybe more than ten years, closing and relocating activities just does not make much sense because they are quite costly and inefficient [34]. On the other hand, planning for really long time spans may not be advantageous since long time spans bring more uncertainty and additional problems such as difficulty of data collection and effective time segmentation [32]. In our problem, for practical purposes, we assume that time spans coinciding with each time period are not that large to necessitate relocation of facilities by incurring a cost burden in short amount of time. We study a market structure in which there is an increasing demand trend. We assume that demand curve of this market follows a similar pattern as the case of introducing a new item to a new market in the Bass diffusion model [35]. As a result, in order to able to cope with increasing demand, we allow opening additional facilities at each time period however we do not permit closure of opened facilities. That means if a facility is opened at any period l, that facility must remain open during following time periods.

Authorized establishments responsible from selling and advertising products belonging to manufacturers or suppliers are generally referred as distributors. Those entities are important pillars of supply chain networks. They are not only purchasing the products from suppliers and sell them to end customers but they also advertise and store the products in distribution centers. The impact of those entities cannot be ignored as they help market share grow not by only providing logistic support but by also backing manufacturers with additional investments, product promotions and providing after sales service to final customers. Currently, many industrial practices from sectors such as automotive, electronics and healthcare survive and reach to end users with the support of distributors.

In this study, we consider a two echelon supply chain network where the distribution facilities procure the end-product from the suppliers which offer quantity discount and satisfy the customer demand. We are given a set of suppliers, a set of possible facility locations, a set of customers and we focus on selecting the suppliers and determining the location the distribution facilities in a given planning horizon to minimize purchasing, distribution and facility opening costs. We assume that each facility has its own facility opening cost and capacity while each supplier has its own price, discount levels and rates. Altering supply chain networks might be confusing since it does not end by only altering an entity from the chain such as changing the location or capacity of a facility. Entire chain is connected to each other from procurement of raw materials to distributing end product to customers, and hence a single change may harm entire chain. We construct a mixed integer programming (MIP) model and develop heuristics in which mixed integer programming based iterative approaches are used to solve the problem in each echelon sequentially.

The rest of the thesis is structured as follows. In Chapter 2, we provide a literature related to our study. Afterwards, in Chapter 3, a detailed problem description is presented. Chapter 4 includes constructed MIP model to obtain optimal solution and proposed heuristic algorithms in order to handle large instances. In Chapter 5 we present the results of the computational study conducted to evaluate the performance of the proposed heuristic algorithms. In Chapter 6 we conclude our efforts and discuss our findings.

CHAPTER II

LITERATURE REVIEW

Facility location problems have been widely studied and a great variety of location models have been proposed so far in the literature. It is not possible to cover every aspect of location studies in this thesis. After all, quite elaborated and popular surveys investigating specific features of location problems already exist [3, 33, 36]. Therefore, in this chapter, after a brief overview about multi-echelon and dynamic models, we cover only a certain portion of related literature by dividing it into two main categories. First category includes literature about supplier selection with quantity discount schemes, second category contains facility location models with economies of scale for handling larger amounts.

[37, 38] are early descriptive overviews about facility location problems. For a more recent review and detailed classification with well-known algorithms see [4, 16]. Moreover, [39] introduce an elaborated summary of previous location models according to their relevance. Multi-echelon location problems are also analyzed a lot under different circumstances from very early days. Lagrangian heuristics and branch and bound procedures are commonly preferred in early years due to difficulty of a two echelon structure. While [40, 41] are primary examples for use of branch and bound method, [42] discussed Lagrangian heuristics as a benchmark case in early literature. Beside, [22] propose a branch and bound method founded on Lagrangian relaxation for a two echelon location problem. [36] provide a comprehensive classification for multi-echelon location problems according to their features. According to review, health-care and production-distribution systems are most common examples of multi-layer structures as real life applications. [43] is also another recent survey about

multi-echelon location problems.

If a firm has hard time to locate their facilities since they feel uncertain about forthcoming periods because of possible changes they expect, that might direct them to use dynamic models. As [4] summarized with details, different classifications for dynamic models are present in literature such as discrete or continuous models according to network reconfiguration times and finite or infinite models based on planning horizon of the study. In the light of those categorizations, we can claim that we work on a deterministic system and our study belongs to discrete and finite time class.

Similar to general network structure to be used in this study, researchers investigated multi-period, multi-stage models in many occasions. Especially over the last decades, quite a number of studies analyzing such network structures have been published. In this paragraph, we mention about some of those studies. None of the referred studies in this paragraph deal with supplier selection and discount schemes. [44] build a two phase algorithm. In the first part of the study, a branch and bound method is used for each period to obtain a candidate solution set. Then in the second part, a complete solution is constructed using dynamic programming. [45] wants to locate warehouses in a two echelon multi period structure. The study allows opening new facilities on a pre-built, default network. Closure of pre-existing facilities is also allowed. As a supply chain strategy, inventory related decisions and costs are also analyzed in the study. A heuristic method is built based on relaxed Lagrangian solutions. [46] provide MIP models for a two echelon multi-period location problems. Results are interpreted based on real life data of an electronics company. [34] is a recent study dealing with two-echelon multi period facility location problem. A two stage algorithm is proposed where a clustering algorithm is used at the first stage to create customer districts by grouping them. Then at the second stage, problem is solved for those districts. The data used for testing is taken from automotive industry.

Next, we discuss the articles studying location decisions in the presence of quantity discount schemes. [47] was first linking location problem with discount schemes and explored capacitated/uncapacitated plant location problem under a concave cost function. All kinds of costs which can be encountered up until a product reach to a customer such as plant opening costs, operating costs and distribution costs are assumed to be a concave function of units handled. A branch and bound technique is introduced to solve the problem. [48] is another early study looking into a similar problem with a single difference where only operating costs are assumed concave instead. An iterative procedure is proposed to solve the problem in the article. In a more recent study, [49] consolidate all potential expenses such as procurement, transportation, labour and overheads under a single cost function and again all expenses are assumed concave. With that assumption, some properties are introduced and the problem is simplified with a heuristic by reducing solution region through those properties. Finally, branch and bound method is used on the simplified version of the problem. [50] present another location problem with quantity discount setting in a single echelon network. Based on real data, a concave cost function is defined with inclusion of all potential expenses except transportation cost. All costs except transportation follows a similar concave path as [49] but transportation cost is separated and assumed proportional with distance. Revealed real data also shows that beyond a certain threshold value, the concave cost function starts to change its shape and looks more linear since it's impossible to obtain discounts at an unlimited rate. The cost function is linearized and a Langrangian heuristic is proposed. Apart from these, it is also possible to find studies with contradictory assumptions. [51] evaluate a location model in which unit cost of each additional item increases by some amount based on a concave cost function. Linear models and branch and bound method is performed consecutively to solve single echelon location problem. All those studies we investigated use discount scheme substantially at the operational level. We, on the other hand, focus specifically on purchase discounts while exploring the location of distribution facilities. To best of our knowledge, studies performed so far with discount settings are laboured in single echelon networks where we analyse a dynamic two echelon structure.

As [12] initially brought up, supplier selection problems are usually labelled in two categories one of which is single sourcing while the other one is multiple sourcing. In single sourcing, supplier selection does not contain any limiting constraints and one supplier can fulfil all demands whereas in multiple sourcing, no supplier can meet all requirements since some restrictions such as capacity are also introduced. See also [52]. In our problem, we are not using any constraints restricting supplier selection even if we do include capacity constraints for facilities. In literature, supplier selection practices generally involve selection of right suppliers and allocating right amount of items to selected suppliers. In last decade, there has been a growing interest for supplier selection studies and different selection criterias such as price, reliability and deficiency rate have been introduced. Naturally, it's pretty hard to ignore discount schemes when you are bound to make hard decisions like supplier selection and quantity allocation. Thereby, a significant proportion of previous studies involve the integration of supplier selection and discount schemes. Studies working on supplier selection problem under quantity discount have assumed a single echelon network where the problem only includes suppliers and customers. [53] is earliest example and propose a heuristic to solve a problem involving all unit discount. [54] pursue supplier selection problem in multi-period, multi-item environment analyzing both all unit and incremental discount cases. An MIP is proposed for a setting where discounts are applied independently for each product by each supplier. [55] is also interested in different discount scenarios including incremental case. In order to deal with each scenario seperately, they develop a heuristic which can be adapted for different discount schemes. [56] work on integrated supplier selection and lot sizing problem in a multi-period environment in the presence of both all unit and incremental discounts. A genetic algorithm is proposed to handle the problem. [15] is another study analyzing a supplier selection problem with multiple products and multiple criteria under quantity discounts. A two stage algorithm has been proposed. The first stage involves weight determination for each criterion. In the second stage, an MIP model which combine all criteria on a single objective function using weights determined in previous stage is solved.

Up until now, we mentioned about location studies exercising quantity discount under single echelon structure. To our knowledge, there are a few examining two-echelon structure and quantity discount under the same category. However those studies belong to supply chain coordination literature and none of them are interested in locational analysis. [17] is a good example on this matter. The paper observes a scenario similar to our case where a two echelon structure is established under quantity discounts in order to apprehend a coordination between each segment of the chain. Additionally, [57] works on a system resembling to our network structure where the study includes retailers, distribution centers (DC) and suppliers. Briefly, suppliers provide the goods to DCs and similarly DCs distribute those goods to retailers. Objective is to locate DCs and designate inventory related decisions for both DCs and retailers. Supplier selection is not part of the study since suppliers are already known. Besides, instead of procurement discount on suppliers, an operating discount schedule is applied on DCs. An advanced cutting plane approach is proposed after an MIP for the problem is established.

CHAPTER III

PROBLEM DEFINITION

We observe a two echelon deterministic supply chain network comprising of suppliers, distribution facilities and retailers. The problem pursues a structure where distributors procure goods from suppliers under quantity discount in order to finally deliver those goods to end customers in a multi-period environment. You can refer Figure 1 for a general outlook of our network. As already discussed in Chapter 1, the planning horizon is split into relatively long time segments. Thus, our problem involves L different time period. Demand from each customer at each period is known and total demand must be met by distributors. We briefly mention in Chapter 1 that we will study a market structure in which there is an increasing demand trend similar to Bass diffusion model. Further details about periodic demand curve will be discussed in Chapter 5. Customers, suppliers and candidate distribution facilities are defined in Euclidean space, hence third dimension is ignored. The distance between each location is identified as the transportation cost of traveling between those locations. A first transportation cost occurs during the shipment of goods to facilities from suppliers and also a second transportation cost arises from the distribution of those goods from facilities to end customers. It has been assumed that each facility has its own facility opening cost and capacity while a respective periodic operating cost for opened facilities is also introduced as an administration expense. Moreover, capacity constraints are assumed fixed at each period segment. In our problem, it is neither allowed a possible capacity extension nor authorized to open an additional distribution facility at the same candidate location. Also, each supplier has its own incremental discount scheme with proper prices, discount levels and rates, thereby, procurement costs of goods are calculated according to respective discount scheme of suppliers. The problem includes D different price segment and quantity intervals for those segments for each supplier. At each period, purchase decisions are made exclusively, meaning in a given period, it is not allowed to procure more than total demand in order to cover the demand of following periods. Discounts are applied at each period separately, meaning procurement decisions from separate periods are not cumulated to acquire a bigger discount. However, it is possible to obtain a bigger discount in a given period by accumulating procurements from different distribution facilities. Besides, distribution facilities are allowed to purchase from different suppliers in a same period. Item flow decisions between suppliers, distribution facilities and customers are made periodically and assumed constant over a period. The problem is to locate distribution facilities under a set of candidate facility locations and additionally select suppliers among the candidates in order to minimize total costs (purchasing, transportation, facility opening, operating).

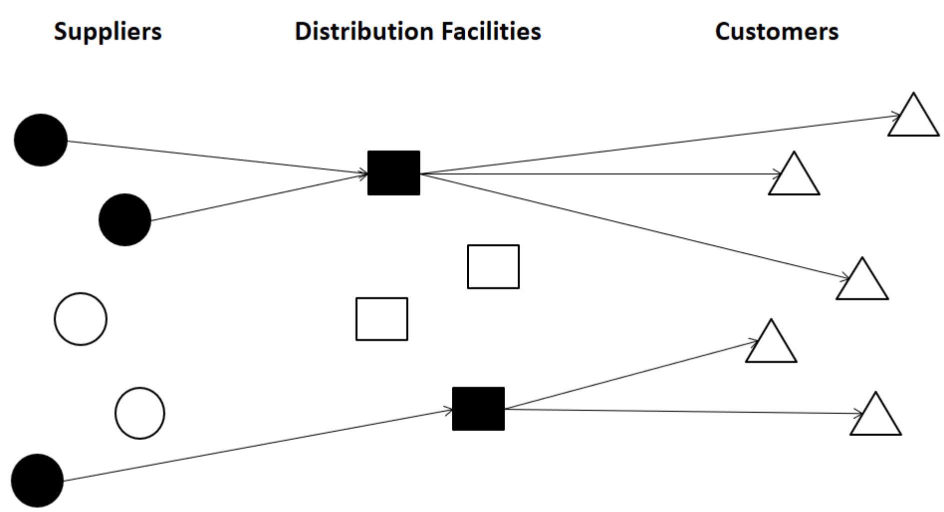

Figure 1: Example Network Structure

CHAPTER IV

METHODOLOGY

4.1. Exact Model

In this section of the study, we provide mathematical formulation of our problem built as MIP. Corresponding variables and parameters are defined as follows:

Sets

$$
\begin{array}{ll}
\mathcal{M} = \{1, ..., M\} & \text{set of customers} \\
\mathcal{I} = \{1, ..., I\} & \text{set of suppliers} \\
\mathcal{J} = \{1, ..., J\} & \text{set of facilities} \\
\mathcal{L} = \{1, ..., L\} & \text{set of periods} \\
\left.\begin{array}{l} \mathcal{D} = \{1, ..., D\} \\ \mathcal{D}_1 = \{2, ..., D\} \end{array}\right\} & \text{discount levels}
\end{array}
$$

Parameters

a_{id} = upper limit for discount level d for supplier i

p_{id} = purchasing price from supplier i at discount level d

λ_{ml} = demand level of customer m in period l

c^1_{ij} = cost of using the route between supplier i and facility j

c^2_{jm} = cost of using the route between facility j and customer m

F_j = fixed cost of establishing a facility in node j

K_{id} = ordering cost from supplier i at discount level d

o_j = operating cost for facility j

C_j = capacity of facility j

Decision Variables

x^1_{ijl} = the amount of item will be sent by using route between supplier i and facility j

x^2_{jml} = the amount of item will be sent by using route between facility j and customer m

q_{idl} = the amount of item which will be purchased by supplier i at discount level d in period l

$$t_{jl} = \begin{cases} 1, & \text{if facility } j \text{ is opened in period } l \\ 0, & \text{otherwise} \end{cases}$$

$$y_{idl} = \begin{cases} 1, & \text{if the item will be purchased from supplier } i \text{ at discount level } d \text{ at period } l \\ 0, & \text{otherwise} \end{cases}$$

Q_{il} = total amount of item which will be purchased from supplier i at period l

Network formulation is built as follows:

$$\text{Min} \quad \sum_{i\in\mathcal{I}}\sum_{d\in\mathcal{D}_1}\sum_{l\in\mathcal{L}} p_{id}y_{idl}a_{id-1} + \sum_{i\in\mathcal{I}}\sum_{d\in\mathcal{D}}\sum_{l\in\mathcal{L}} p_{id}q_{idl} + \sum_{i\in\mathcal{I}}\sum_{d\in\mathcal{D}}\sum_{l\in\mathcal{L}} K_{id}y_{idl} + \sum_{l\in\mathcal{L}}\sum_{j\in\mathcal{J}} F_j t_{jl} \quad (1)$$
$$+ \sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}}\sum_{l\in\mathcal{L}} c^1_{ij}x^1_{ijl} + \sum_{j\in\mathcal{J}}\sum_{m\in\mathcal{M}}\sum_{l\in\mathcal{L}} c^2_{jm}x^2_{jml} + \sum_{l\in\mathcal{L}}\sum_{j\in\mathcal{J}} o_j(L-l+1)t_{jl}$$

s.t.

$$Q_{il} = \sum_{d\in\mathcal{D}} q_{idl} + \sum_{d\in\mathcal{D}_1} y_{idl}a_{id-1} \qquad \forall i\in\mathcal{I}, j\in\mathcal{J} \qquad (2)$$
$$q_{i1l} \le a_{i1}y_{i1l} \qquad \forall i\in\mathcal{I}, l\in\mathcal{L} \qquad (3)$$
$$q_{idl} \le (a_{id} - a_{id-1})y_{idl} \qquad \forall i\in\mathcal{I}, l\in\mathcal{L}, d\in\mathcal{D}_1 \qquad (4)$$
$$Q_{il} = \sum_{j\in\mathcal{J}} x^1_{ijl} \qquad \forall i\in\mathcal{I}, l\in\mathcal{L} \qquad (5)$$
$$\sum_{i\in\mathcal{I}} x^1_{ijl} \le C_j \sum_{p\in l}^{l} t_{jp} \qquad \forall j\in\mathcal{J}, l\in\mathcal{L} \qquad (6)$$
$$\sum_{j\in\mathcal{J}} x^2_{jml} = \lambda_{ml} \qquad \forall m\in\mathcal{M}, l\in\mathcal{L} \qquad (7)$$
$$\sum_{i\in\mathcal{I}} x^1_{ijl} = \sum_{m\in\mathcal{M}} x^2_{jml} \qquad \forall j\in\mathcal{J}, l\in\mathcal{L} \qquad (8)$$
$$\sum_{d\in\mathcal{D}} y_{idl} \le 1 \qquad \forall i\in\mathcal{I}, l\in\mathcal{L} \qquad (9)$$
$$\sum_{l\in\mathcal{L}} t_{jl} \le 1 \qquad \forall j\in\mathcal{J} \qquad (10)$$
$$x^1_{ijl} \ge 0 \qquad \forall i\in\mathcal{I}, j\in\mathcal{J}, l\in\mathcal{L} \qquad (11)$$
$$x^2_{jml} \ge 0 \qquad \forall j\in\mathcal{J}, m\in\mathcal{M}, l\in\mathcal{L} \qquad (12)$$
$$q_{idl} \ge 0 \qquad \forall i\in\mathcal{I}, d\in\mathcal{D}, l\in\mathcal{L} \qquad (13)$$
$$y_{idl} \in \{0,1\} \qquad \forall i\in\mathcal{I}, d\in\mathcal{D}, l\in\mathcal{L} \qquad (14)$$
$$t_{jl} \in \{0,1\} \qquad \forall j\in\mathcal{J}, l\in\mathcal{L} \qquad (15)$$

Our total cost function includes purchase, distribution, facility opening and operating costs. In the mathematical formulation, objective is to minimize total cost and it is represented by Equation (1). Constraint (2) allows us to calculate the total procurement from each supplier. Constraint (3) ensures that procurement amount from each supplier do not exceed pre-determined upper discount limit at initial price segment while Constraint (4) makes sure purchase quantity from suppliers is within quantity interval for following price segments. Constraint (5) guarantees that total procurement from each supplier at each period is transshipped to distribution facilities from suppliers. Constraint (6) ensures capacity of facilities is not exceeded in any period and Constraint (7) makes sure demand requirements are met. Constraint (8) is a flow balance constraint for distribution facilities and guarantees that incoming amount is equal to outgoing. Constraint (9) compels the model to select no more than a single price segment for each period and Constraint (10) does not permit to open additional facilities at a candidate location, if a facility is already opened at the same location in previous periods. Constraint (11) to (13) are sign restrictions while Constraint (14) and (15) are integrality constraints.

4.2. Heuristic Approaches

According to our test results, using the mathematical model formulation to solve the problem was not that efficient in terms of cost and time efficiency especially while handling large instances. That's why, we develop mixed integer programming based iterative heuristic algorithms to overcome this issue. In this section of the study, we provide those algorithms with details.

In this sense, we first divide our main problem into two sub-problems. It basically means that our two-echelon supply chain network is split up into two Sub-Problems according to each echelon. Sub-Problem 1 (SP1) spans the first echelon and involves the transaction decisions between suppliers and distribution facilities. Again, Sub-Problem 2 (SP2) spans the second echelon and includes this time the transaction decisions between distribution facilities and customers. Before presenting our heuristic algorithms in detail, we start by providing the details of those two Sub-Problems.

Sub-Problem 1 (SP1)

A mathematical formulation for *SP1* is built as follows:

$$\begin{aligned} \text{Min} \quad & \sum_{i\in\mathcal{I}}\sum_{d\in\mathcal{D}_1}\sum_{l\in\mathcal{L}} p_{id}y_{idl}a_{id-1} + \sum_{i\in\mathcal{I}}\sum_{d\in\mathcal{D}}\sum_{l\in\mathcal{L}} p_{id}q_{idl} + \sum_{i\in\mathcal{I}}\sum_{d\in\mathcal{D}}\sum_{l\in\mathcal{L}} K_{id}y_{idl} \\ & + \sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}}\sum_{l\in\mathcal{L}} c^1_{ij}x^1_{ijl} \end{aligned} \tag{16}$$

s.t.

$$Q_{il} = \sum_{d\in\mathcal{D}} q_{idl} + \sum_{d\in\mathcal{D}_1} y_{idl}a_{id-1} \qquad \forall i\in\mathcal{I}, j\in\mathcal{J} \tag{17}$$

$$q_{i1l} \le a_{i1}y_{i1l} \qquad \forall i\in\mathcal{I}, l\in\mathcal{L} \tag{18}$$

$$q_{idl} \le (a_{id} - a_{id-1})y_{idl} \qquad \forall i\in\mathcal{I}, l\in\mathcal{L}, d\in\mathcal{D}_1 \tag{19}$$

$$Q_{il} = \sum_{j\in\mathcal{J}} x^1_{ijl} \qquad \forall i\in\mathcal{I}, l\in\mathcal{L} \tag{20}$$

$$\sum_{d\in\mathcal{D}} y_{idl} \le 1 \qquad \forall i\in\mathcal{I}, l\in\mathcal{L} \tag{21}$$

$$\sum_{i\in\mathcal{I}} x_{ijl} = Z_{jl} \qquad \forall j\in\mathcal{J}, l\in\mathcal{L} \tag{22}$$

$$x^1_{ijl} \ge 0 \qquad \forall i\in\mathcal{I}, j\in\mathcal{J}, l\in\mathcal{L} \tag{23}$$

$$q_{idl} \ge 0 \qquad \forall i\in\mathcal{I}, d\in\mathcal{D}, l\in\mathcal{L} \tag{24}$$

$$y_{idl} \in \{0,1\} \qquad \forall i\in\mathcal{I}, d\in\mathcal{D}, l\in\mathcal{L} \tag{25}$$

While first echelon involves supplier selection with ordering and transportation decisions between suppliers and distribution facilities, second echelon embraces facility opening and transportation decisions between distribution facilities and customers. As it is seen, the decisions from each echelon are mutually dependent and would affect the other one considerably unless they are solved simultaneously. However, solving those two echelon simultaneously increases the complexity of the problem to a large extent. Thus, while defining our two Sub-Problems, we introduce additional parameters which links those two Sub-Problems and thereby those two echelons. One of those parameters is Z_{jl} and it represents total demand transshipped to customers through facility j. Notice that equation (22) is the only new constraint introduced while building *SP1*. This constraint basically guarantees that our total demand is met through facilities. Z_{jl} calculation is provided in equation (26). Remaining constraints are already referred in Section 4.1 while we present our main model formulation.

$$Z_{jl} = \sum_{m \in \mathcal{M}} x^2_{jml} \qquad \forall j \in \mathcal{J}, l \in \mathcal{L} \tag{26}$$

Another parameter which we present is u_{jl}. This additional parameter represents average per unit cost value of *SP1* as it is shown in equations (27) to (29). It is calculated by using results obtained after solving *SP1*, for that reason, the decision variables from *SP1* turns into parameters during calculation of u_{jl}. After calculated u_{jl}, we incorporate this parameter into *SP2* as a representative of *SP1*.

u_{jl} = average per unit cost representation of using facility j in period l for *SP1*

pc_1 = unit purchasing cost of *SP1*

tc_1 = unit transportation cost of *SP1*

$$pc_1 = \begin{cases} \dfrac{\sum\limits_{i\in\mathcal{I}}\sum\limits_{d\in\mathcal{D}}\left(\left(p_{id}x^1_{ijl} + \dfrac{K_{id}x^1_{ijl}}{Q_{il}}\right)y_{id}\right)}{\sum\limits_{i\in\mathcal{I}} x^1_{ijl}}, & \text{if } \sum\limits_{i\in\mathcal{I}} x^1_{ijl} \neq 0 \\ 0, & \text{otherwise} \end{cases} \qquad \forall j \in \mathcal{J}, l \in \mathcal{L} \quad (27)$$

$$tc_1 = \begin{cases} \dfrac{\sum\limits_{i\in\mathcal{I}} c_{ij}x^1_{ijl}}{\sum\limits_{i\in\mathcal{I}} x^1_{ijl}}, & \text{if } \sum\limits_{i\in\mathcal{I}} x^1_{ijl} \neq 0 \\ 0, & \text{otherwise} \end{cases} \qquad \forall j \in \mathcal{J}, l \in \mathcal{L} \quad (28)$$

$$u_{jl} = pc_1 + tc_1 \qquad \forall j \in \mathcal{J}, l \in \mathcal{L} \quad (29)$$

Sub-Problem 2 (SP2)

Mathematical formulation for *SP2* is constructed as follows:

$$\begin{aligned} \text{Min} \quad & \sum_{l\in\mathcal{L}}\sum_{j\in\mathcal{J}} F_j t_{jl} + \sum_{j\in\mathcal{J}}\sum_{m\in\mathcal{M}}\sum_{l\in\mathcal{L}} c^2_{jm}x^2_{jml} + \sum_{l\in\mathcal{L}}\sum_{j\in\mathcal{J}} o_j(L-l+1)t_{jl} \\ & + \sum_{j\in\mathcal{J}}\sum_{m\in\mathcal{M}}\sum_{l\in\mathcal{L}} u_{jl}x^2_{jml} \end{aligned} \quad (30)$$

s.t.

$$\sum_{m\in\mathcal{M}} x^2_{jml} \leq C_j \sum_{p\in l}^{l} t_{jp} \qquad \forall j \in \mathcal{J}, l \in \mathcal{L} \quad (31)$$

$$\sum_{j\in\mathcal{J}} x^2_{jml} = \lambda_{ml} \qquad \forall m \in \mathcal{M}, l \in \mathcal{L} \quad (32)$$

$$\sum_{l\in\mathcal{L}} t_{jl} \leq 1 \qquad \forall j \in \mathcal{J} \quad (33)$$

$$x^2_{jml} \geq 0 \qquad \forall j \in \mathcal{J}, m \in \mathcal{M}, l \in \mathcal{L} \quad (34)$$

$$t_{jl} \in \{0,1\} \qquad \forall j \in \mathcal{J}, l \in \mathcal{L} \quad (35)$$

In equation (30), u_{jl}, average per unit cost parameter representing *SP1*, is included into objective cost function. Again, the details about remaining equations can be found in Section 4.1.

4.2.1. Iterative Algorithm (IA)

We develop two heuristic algorithms to solve our problem, first one is a MIP based iterative algorithm. In this section, we explain this algorithm step by step. Besides, a pseudocode is provided in Algorithm 1 while a flowchart is also illustrated in Figure 2. As a brief overview of this algorithm, by focusing on a specific candidate facility set and solving mathematical models of *SP1* and *SP2* in an iterative manner, we aim to obtain favorable solutions for our network problem.

4.2.1.1. Initialize $\boldsymbol{u_{jl}}$

In this first step, we assign a value to each Z_{jl} and u_{jl} by implementing an initial procedure. We assume that each facility is open and customer demand is met using full capacity. Thereby, Z_{jl} values are initialized in accordance with equation (36). Then, we solve *SP1* by using that assumption and assign u_{jl} values accordingly as it is decribed in equation (29). Our aim in this procedure is to estimate a rational u_{jl} value by imagining an unreal case in which all candidate facilities function with full capacity. Therefore, equation (36) is the outcome of this unreal case and solely used during initialization procedure. In following steps and remainder of the study, we continue to use equation (26) to update Z_{jl} values.

$$Z_{jl} = C_j \qquad \forall j \in \mathcal{J}, l \in \mathcal{L} \tag{36}$$

4.2.1.2. SP2 Iteration

After attained initial u_{jl} values, we move on to next step which is *SP2 Iteration*. In this step, we solve *SP2* in an iterative manner up until we reach to a stopping criteria. To further decrease the problem complexity and reduce the number of variables, a candidate facility set $cand_m(N) \in \mathcal{J}$ for each customer $m \in \mathcal{M}$ is created. According to this, each customer can only be served from facilities included to their own facility set. With this new assumption, not all $j \in \mathcal{J}$ is a customer candidate anymore since $cand_m(N) \subset \mathcal{J}$. In order to settle an upper limit to the size of candidate facility set $cand_m(N)$, a pre-selected number N is also defined. It means each customer m can be served by any of N candidate facilities which belongs to $cand_m(N)$. This N value is not necessarily fixed during *Iterative Algorithm* and its value can further decrease as shown in Algorithm 1. Initially, to generate $cand_m(N)$, we assign nearest N candidate facility to customer m. Then, we start to solve *SP2* iteratively. At the end of each iteration, unused facilities are eliminated and $cand_m(N)$ is updated from among facilities opened. To be more precise, after a *SP2* solution is obtained, we start to assign used facilities to $cand_m(N)$ by starting from the nearest one and continue to assign up until a total of N opened facility is assigned. The stopping criteria (sc) defined for this step requires two successive non-improving solution or a solution in which number of facilities opened is less than N.

4.2.1.3. Solving SP1

After it is reached to sc, we move out from *SP2 Iteration* and start to solve *SP1*. At the end of *SP2 Iteration*, we already know the facilities to be opened and total demand reaching to each facility (Z_{jl}). Hence, our single focus while solving *SP1*, is to make supplier selections with ordering quantity decisions. It means we are dealing with a simplified version of *SP1*. At the end of this step we reach to a complete solution for the entire problem and keep it in our solution set.

4.2.1.4. *Updating* $\boldsymbol{u_{jl}}$ *and Termination Criteria for IA*

After solved *SP1* and acquired a complete solution for the entire problem, we check whether we reach to Termination Criteria for *IA* or not. Termination Criteria (*tc*) for this algorithm is last s complete solution without any improvement. If *tc* is not reached yet, we update u_{jl} values as shown in equation (29), then go back to *SP2 Iteration* and start over with remaining facilities. On the other hand, if *tc* is reached, we select best complete solution from solution set as our final result.

Table 1: Quick Notation Guide for *Iterative Algorithm*

Notation	**Description**
N	a pre-selected number which limits the size of $cand_m(N)$
s	limits the number of iteration without improvement as tc
$cand_m(N)$	candidate facility set for each customer $m \in \mathcal{M}, cand_m(N) \subset \mathcal{J}$
sc	stopping criteria for intermediary step *SP2 Iteration*
tc	termination criteria which ends *Iterative Algorithm*

Algorithm 1 *Iterative Algorithm*

initialize Z_{jl}
solve *SP1* and initialize u_{jl}
while tc is not satisfied **do**
  while sc is not satisfied **do**
    solve *SP2*
    update $cand_m(N), N, Z_{jl}$
  end while
  solve *SP1* and find a complete solution
  update solution set with the new complete solution
  update u_{jl}
end while
select the best solution from solution set

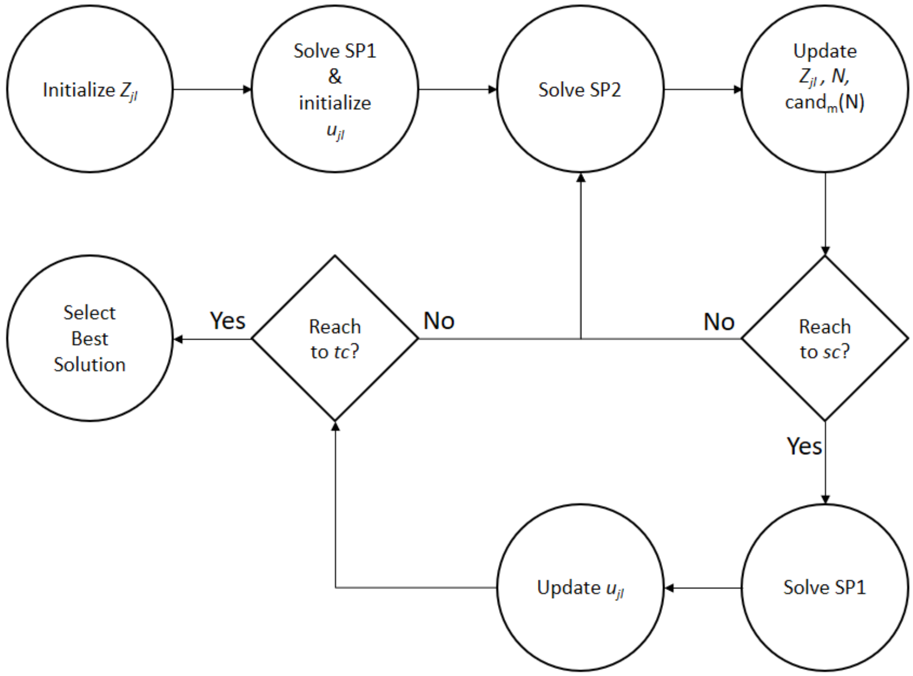

Figure 2: Flowchart of IA

4.2.2. Two-Phase Clustered Iterative Algorithm (TPCIA)

The other heuristic algorithm which is presented in this study is *Two-Phase Clustered Iterative Algorithm*. As the previous one, this heuristic is also constructed as MIP based iterative algorithm and uses a similar approach to solve our network problem. It is fair to consider this one as an extended version of previous algorithm. This extension is built in order to overcome potential shortcomings of previous algorithm. Remember that while using *IA*, in course of *SP2 Iteration* (4.2.1.2), unused facilities are eliminated and $cand_m(N)$ is updated from among facilities opened. This let us to focus on a specific facility set through each iteration and intensify our search to finally obtain a favorable solution. On the other hand, since unused facilities are eliminated for good during the remainder of the algorithm, this leds to consolidating our efforts on a limited solution region. Therefore, we focus on increasing diversity while building *TPCIA*. In this section of the study, we provide details of this new extension.

In order to increase diversity, we initially tried to apply small extensions to *IA*. Most significant one is resetting $cand_m(N)$ to initial selection and reincorporating all facilities after updated u_{jl} values. By doing so, after updated u_{jl} (4.2.1.4), all eliminated facilities are considered once again as potential candidates for $cand_m(N)$. Yet, this idea brings a couple of issues. First and most important one is calculation method of u_{jl} during its update procedure (Equation 29). Remember that, in *IA*, before going into u_{jl} update procedure, we already obtain a complete solution after solved *SP1*. Notice that, according to this update equation, if a facility is not opened in any period, its u_{jl} value will become equal to zero after the update procedure. This was not a problem for us before since we know that unused facilities are already eliminated in previous steps. However now, we try to involve eliminated facilities into our equation all over again and it means we have to assign new u_{jl} values for those facilities. That's why, in order to encourage opening unused facilities and strengthen diversification, we use a different strategy for u_{jl} update procedure. This strategy suggests a scenario in which all opened facilities continue to operate in compliance with last deduced result and meet the demand while all unused facilities are additionally assumed open to serve with full capacity. Then, we solve *SP1* with that assumption in order to adjust new u_{jl} values. Obviously, that is an unrealistic scenario where total item amount arriving to distribution facilities from suppliers exceed total customer demand. However, when we assume a case where all facilities are open and force unused facilities to serve with full capacity to meet fictional demand, we actually create a new environment where we slightly favor originally closed facilities. In this new environment, more items are now potentially transshipped from suppliers to originally closed facilities compared to already open facilities. That possibility could result lower u_{jl} costs for closed facilities after the new update procedure since increasing purchase quantity may decrease the selling price depending on quantity amount. Consequently, when we go back to solving *SP2*, we may have lower u_{jl} values which favor opening originally closed facilities. Besides, notice that it does not make any sense for a candidate facility to work with more than one supplier at a same time period. If a candidate supplier i is the best option for candidate facility j to transship an item amount of α, technically it is not possible for supplier *i+1* to be a better option to transship to facility j an additional item amount of β since suppliers do not have capacity limits. Therefore, this new update strategy may canalize us to select different suppliers with new consolidating options which can change entire network flow. In order to put into practice this new strategy, before solving *SP1* and update u_{jl} values, we combine equations (26) and (36) to update Z_{jl} values as shown in Equation (37).

$$Z_{jl} = \begin{cases} \sum\limits_{m \in \mathcal{M}} x^2_{jml}, & \text{if } \sum\limits_{i \in \mathcal{I}} x^1_{ijl} \neq 0 \\ C_j, & \text{otherwise} \end{cases} \qquad \forall j \in \mathcal{J}, l \in \mathcal{L} \tag{37}$$

When we try to directly apply this new update procedure to *IA* and try a couple of tests however, we take a notice that computational time increases excessively and final results relatively are not better than previous algorithm. At least to limit the computational time to a certain degree and in pursuit of increasing solution quality, we develop *TPCIA*. Our aim is to create a handful candidate facility clusters in *Phase 1* to reduce the amount of candidate facilities and carry out the mentioned strategy in *Phase 2*. In the remainder of this section, we explain steps of *TPCIA*.

4.2.2.1. Initialization Steps in TPCIA

Initialization steps to adjust Z_{jl} and u_{jl} values are same as *IA* as described in Section 4.2.1.1. However, additionally we solve mathematical model of *SP2* once (not iteratively as *IA*) so that we can obtain a complete initial solution before starting with

Phase 1 algorithm. Notice that we still continue to use $cand_m(N)$ sets while solving *SP2* to simplify the problem where each customer can only be served by one of the closest N candidate facilities. In *TPCIA* however, estimation of N is different and we mention about that later while explaining each phase of the algorithm.

4.2.2.2. Phase 1

Our aim in this phase is to decrease candidate facilities by creating candidate clusters. It implies that instead of considering entire facility set as candidates, we divide those facilities into clusters and select certain clusters as candidates. That means candidate facilities included to some clusters will not be considered as candidates anymore when we move out to *Phase 2*. Further details will be given later on.

After obtained a complete initial solution, we start to *Phase 1* by dividing candidate facilities into two different clusters. While first cluster involves facilities opened, second cluster includes unused facilities in initial solution. Next, we update Z_{jl} according to initial solution as shown in Equation (37) and solve *SP1* by using our new strategy so that we could update u_{jl} values.

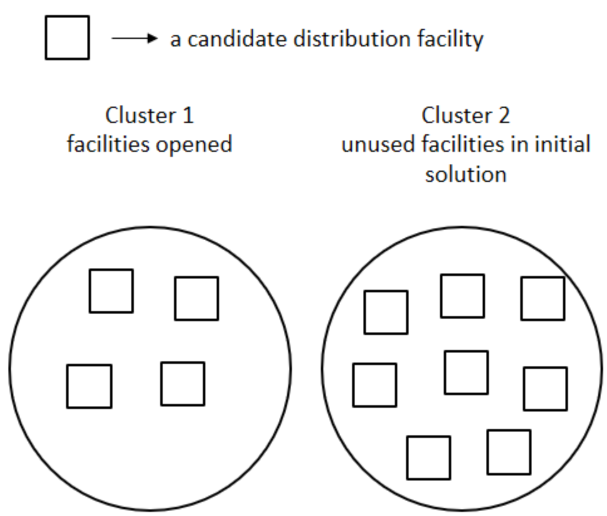

Figure 3: Initial Clustering in Phase 1

After created two initial cluster, we select n_2 candidate facility from *Cluster 2*. Then, we group those selected facilities with facilities from *Cluster 1* to create a new facility set G. Afterwards, $cand_m(N)$, facility sets for each customer, is updated to solve *S2*. n_2 is a figure predetermined before starting to algorithm. It is a representative value for selection from *Cluster 2* and symbolizes facilities snatched from *Cluster 2* in a single iteration to give them a second chance to be selected with new strategy. There is also a second predetermined number introduced in *TPCIA* which is n_1. It represents *Cluster 1*. Both n_1 and n_2 are used to estimate a rational N value as shown in Equation (38). Remember that N is the maximum size of $cand_m(N)$. Thereby, while solving *S2*, we select closest candidate facilities for customer m from another candidate set composed of Cluster 1 and n_2 facility from *Cluster 2*; $cand_m(N) \in G$. It is important to underline that Equation (38) does not necessarily mean that while updating $cand_m(N)$, n_1 facility is taken from *Cluster 1* and n_2 is taken from *Cluster 2*. Because it is possible for the number of candidate facilities in *Cluster 1* to be bigger than n_1. That's why, as it is stated, N is just a rational estimation to select closest candidate facilities for each customer. However, notice that if the number of candidate facilities in *Cluster 1* is less than predetermined n_1 value, we equate them by decreasing n_1. Same circumstances are also valid for n_2.

$$N = n_1 + n_2 \tag{38}$$

After solved *SP2*, opened facilities within *Cluster 1* remain in that cluster whereas opened facilities from *Cluster 2* also moves to *Cluster 1*. On the other hand, we generate two new cluster for unused facilities: *Cluster 3* and *Cluster 4*. If a facility is not opened after solved *SP2* and is from *Cluster 1*, we move that facility to *Cluster 3* which represents unused facilities which have been opened in at least one of previous solutions. If a candidate facility from *Cluster 2*, is not opened during this iteration, we move this facility to *Cluster 4*. Being in *Cluster 4* for a candidate facility implies that those candidate facilities are eliminated for good and they will not be candidate again neither during remainder of *Phase 1* nor during *Phase 2*.

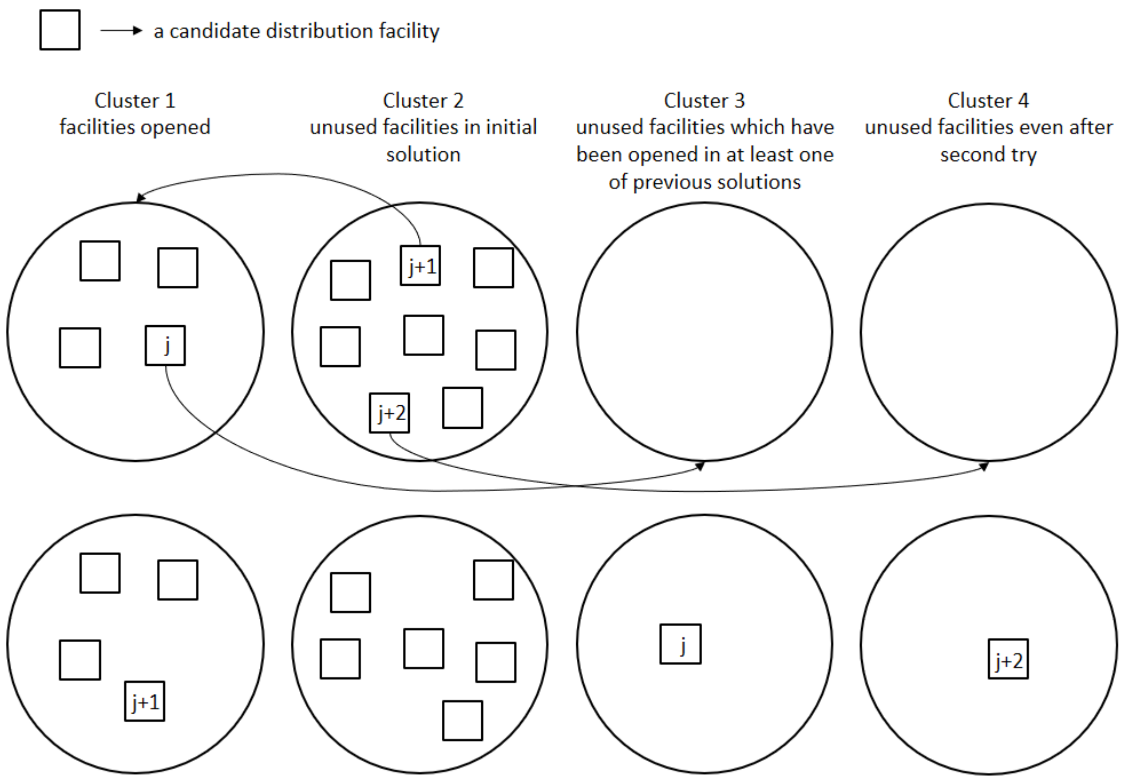

Figure 4: Clustering in Phase 1

After updated clusters, we starts all over again by going back to update Z_{jl} and solve *SP1* to update u_{jl}. Then, we advance to select a new n_2 facility from *Cluster 2* to generate a new G set. Next, $cand_m(N)$ is adjusted again from G set in order to solve *SP2*. Finally, we update clusters again and all those steps are iterated up until no candidate facility is remained in *Cluster 2* as illustrated in Figure 5. Iteration procedure for *Phase 1* is also represented in Algorithm 2.

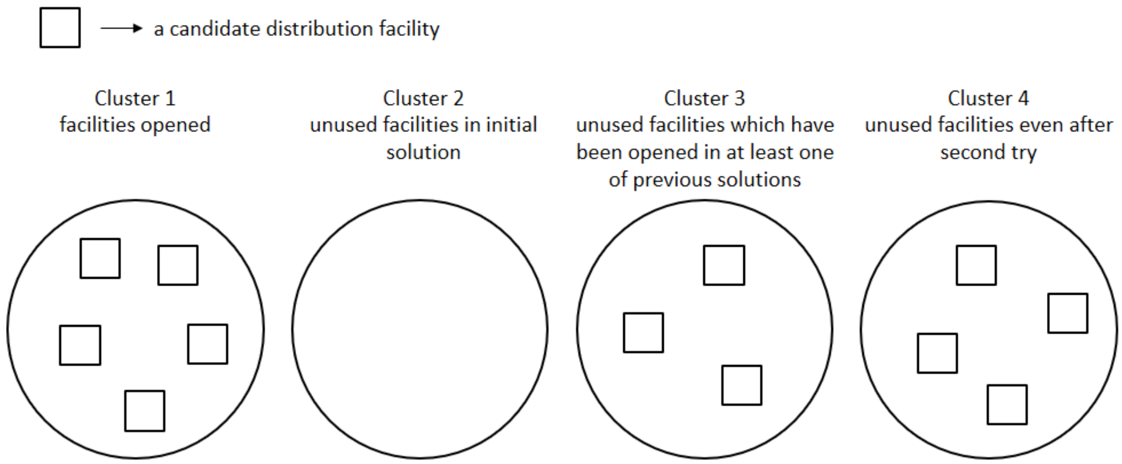

Figure 5: End of Phase 1

At the end of *Phase 1*, candidate facilities from *Cluster 4* are secluded and eliminated. Candidate facilities from *Cluster 1* and *Cluster 3* on the other hand, proceed to next phase and those candidates will further evaluated in *Phase 2*.

Algorithm 2 *Phase 1 of TPCIA*

```
initialize $Z_{jl}$
solve $SP1$ and initialize $u_{jl}$
solve $SP2$ with $cand_m(n_1) \in \mathcal{J}$ to obtain a complete initial solution
while stopping criteria for Phase 1 ($sc'$) is not reached do
    update $Z_{jl}$ using Equation (37)
    solve $SP1$ with new strategy and update $u_{jl}$
    arrange $G$ set to adjust $cand_m(N) \in G$ where $N = n_1 + n_2$
    solve $SP2$ to update all clusters and $n_1$, $n_2$ values if necessary
end while
update $Z_{jl}$ with Equation (26)
solve $SP1$ and update $u_{jl}$ to generate a complete last result from Phase 1
```

Output: *Cluster 1* and *Cluster 3*

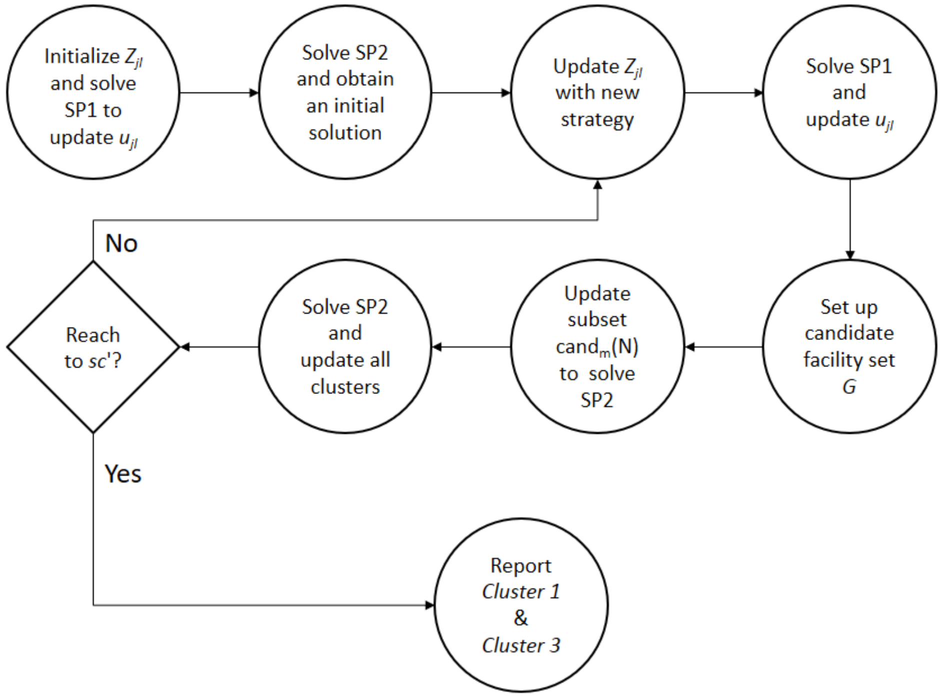

Figure 6: Flowchart of Phase 1

4.2.2.3. Phase 2

Steps of *Phase 2* is really similar to *IA*. The single variations are the new u_{jl} update strategy introduced in *TPCIA* and while applying this strategy we only consider candidate facilities from *Cluster 1 & Cluster 3* formed during *Phase 1*. So basically, in this phase we perform main *IA* procedure but in order to increase diversity, we follow those *IA* procedures with a change of strategy and reconsider all eliminated facilities once again. While we carry out this phase, instead of considering entire candidate facility set $\mathcal{J}$, we perform on a reduced facility set G' assembled from *Cluster 1&3* so that we can restrict solution time to a certain extent and substantially conserve our initial intensification purpose. In this section, we summarize those steps and provide a pseudocode for *Phase 2*.

In this phase, we start by combining *Cluster 1 & Cluster 3* to create a new candidate facilty set G'. Opened facilities will be selected from this facility set.

Following, we update Z_{jl} values using Equation (37) so that we can use new u_{jl} update strategy. Then, we solve mathematical model of *SP1* and update u_{jl} values. At this point we proceed to solving *SP2* iteratively. We select a sub-candidate $cand_m(N)$ from newly formed G' where $cand_m(N) \subset G'$. After that, we solve mathematical model of *SP2* using $cand_m(N)$ sets. According to *SP2* results, unused facilities are eliminated from G' set and Z_{jl} values are adjusted using Equation (26). Then we go back to update $cand_m(N) \in G'$ to solve *SP2* again. This iteration continues until it is reached to an intermediary stopping criteria (sc). After we reach to that criteria, we move to solve *SP1* and achieve a complete solution with aggregation of two echelons. Afterwards, we return to solve *SP1* using Z_{jl} values updated with Equation (37) as we started to *Phase 2*. We simply start *Phase 2* from beginning up until it is reached to termination criteria (tc). Both stopping criteria and termination criteria are same as *IA*.

Algorithm 3 *Phase 2 of TPCIA*

Input: *Cluster 1* and *Cluster 3* from *Phase 1*

```
generate a candidate facility set G' combining Cluster 1 & 3
while termination criteria tc is not reached do
    update Z_jl using Equation (37)
    solve SP1 with new strategy and update u_jl
    while stopping criteria (sc) is not reached do
        assign cand_m(N) ∈ G' where N = n_1 + n_2
        solve SP2 and update G', Z_jl
    end while
    solve SP1 and obtain a complete solution
end while
select best solution from solution set
```

During our computational experiments, for an elaborated analysis, at the end of proposed heuristic algorithms, we additionally solve Exact Model (4.1) with a time limit and by only including selected suppliers and facilities. By doing so, we investigate whether heuristic algorithms efficiently use selected suppliers and facilities. If not, we improve the final solution accordingly.

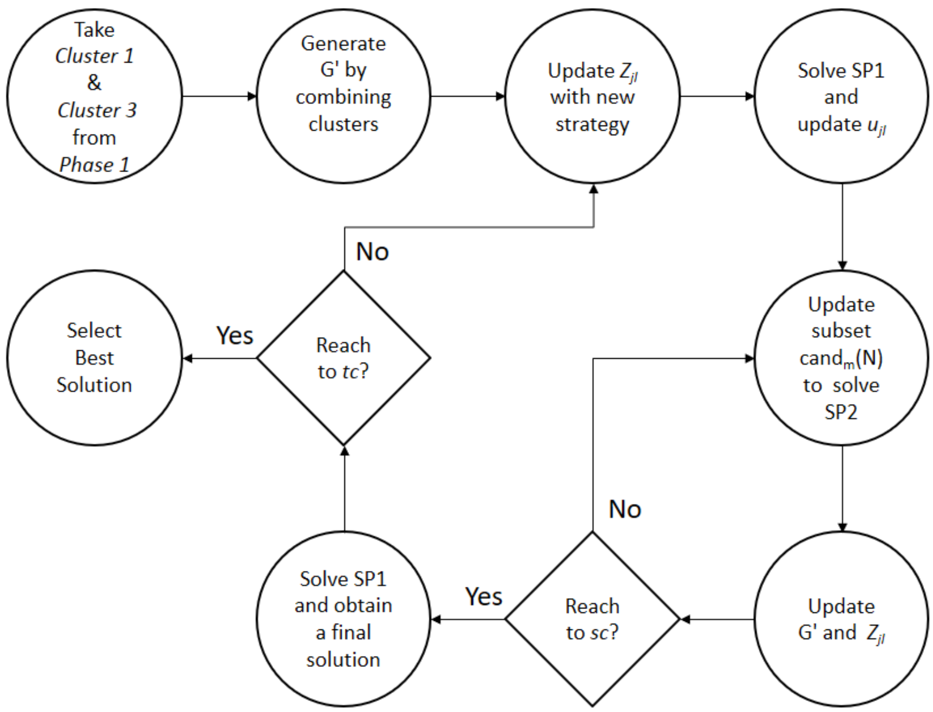

Figure 7: Flowchart of Phase 2

CHAPTER V

COMPUTATIONAL STUDY

In this chapter, computational experiments are conducted in order to evaluate the performance of presented algorithms. We implement proposed algorithms in *Java* and use ILOG CPLEX 12.6.1 solver as the optimization engine. We perform the computational experiments on a machine with Intel Core i7-6500U @ 2.50 GHz processor and 8 GB RAM.

5.1. Instance Generation

Before sharing the results of the experiments, we first explain how we generate experiment instances. Customers, suppliers and candidate distribution facility locations are uniformly distributed on 500×500 map. To calculate the distance between each entity, Euclidean distance measure is used and transportation cost is assumed equivalent to travel distance during cost calculation.

As mentioned in Chapter 1, previous practices indicate that purchasing costs correspond to a huge portion of total costs [15]. Particularly, [12] and [13] reflect that this portion is no less than 50-60% in most industries. We value this information and heed to generate experiment instances suitably. In our base instance, purchase costs are set to be around 50-60% of entire expense and double of transportation and fixed costs. Needless to say, in this study we also evaluate alternative scenarios involving different cost ratios.

Main problem data which we desire to solve on this study includes an instance set with 1000 customers, 100 candidate facility locations and 100 suppliers. It is assumed that each supplier implements an incremental discount scheme with 4 different price segments. The planning horizon is split into 8 time periods. To further analyze the performance of the algorithm, we also generate alternative instance sets one of which includes 1000 customers, 100 candidate facility locations and 20 suppliers while the other smaller instance set includes 100 customers with 25 facilities and 25 suppliers. Each instance set consists of 10 instances.

Customer demands for the first period are built uniformly from [200,600]. We assume a market case where we introduce a new item to a new market as in the Bass diffusion model [35] and we anticipate increasing periodic demand. Therefore, periodic customer demands are created randomly to be between 80% and 160% of the demand from previous period.

Initial unit selling prices of each supplier are randomly generated between [190,210]. For following price levels, it is assumed each supplier perform a randomly designated discount between 5 and 10 percent on previous price level. First discount breakpoint of each supplier is generated randomly between [6%,12%] of average cumulative demand from each period. Following discount breakpoint for the next price level is calculated randomly as to be 40 to 60 percent higher than the previous breakpoint.

Total capacity of candidate facilities is organized in a way that it will be 5 times higher than average cumulative demand from each period. Opening facility costs are correlated with facility capacity and it is calculated as $R \times \sqrt{C_j}$ for each facility j where C_j is facility capacity. R is a constant and it is regarded as 80000 during data generation. Periodic operating cost for each facility is randomly generated between $[\frac{1}{3}, \frac{1}{6}]$ of facility opening cost.

5.2. Computational Results

To analyze our data sets, we submit 7 different scenarios where each scenario involves some alterations in cost ratios. As discussed above, our data sets are initially generated to comply with base scenario in which purchase costs are set to be around

50-60% of entire expense and double of transportation and fixed costs. The alterations we present include changes such as doubling the travel costs to increase the ratio of transportation cost. Our aim here is to fairly evaluate the performance of proposed algorithms. Our inspection starts with our main instance set which includes 1000 customers, 100 suppliers and 100 candidate facility locations. Finding an accurate bound for a problem with this size was challenging. Thus, with the intention of capturing more concrete results and testing the validity of our investigations, we also examine smaller data sets. To evaluate the performance of the proposed algorithms, we implement exact model (4.1) to CPLEX solver and compare obtained results with proposed heuristics. Due to complexity of the problem on hand, we set a 12-hour time limit for the exact model. Likewise, *SP1* and *SP2* models which we use for heuristic algorithms are also implemented in CPLEX solver and to improve computational time, a time limit of 1000 seconds is designated for these models.

Remember that we use some pre-selected figures to implement our heuristic algorithms. Table 2 shows which values are assigned to those additional heuristic parameters.

Table 2: Values assigned to heuristic parameters

Notation	Assigned Figures	Description
N	25	a pre-selected number which limits the size of $cand_m(N)$
s	3	limits the number of iteration without improvement as tc
n_1	10	Both n_1 and n_2 are used in *TPCIA* where $N = n_1 + n_2$
n_2	15	

In Table 3, we share the results we obtained for base scenario with 1000 customers, 100 suppliers and 100 candidate facility locations. Table reveals that both heuristics achieve better results compared to CPLEX. Besides, only in two data instance *IA* managed to overthrow *TPCIA* yet when we compare the average results, we notice that the results of *TPCIA* are only slightly better than *IA*. Furthermore, the solution time of *IA* is much shorther compared to *TPCIA*. In order to give you a brief understanding of the cost structure in this scenario, we also want to share some additional information. According to final obtained solutions, on average 16.6 supplier is selected and 18.9 facility is opened in *IA*.

Table 3: Comparison of results obtained by heuristic algorithms and mathematical model for base scenario

Data Type	**CPLEX**		**IA**		**TPCIA**	
1000×100×100	**Time(sec)**	**Gap**	**Time(sec)**	**Gap**	**Time(sec)**	**Gap**
Data1	43200.0	20.47%	5077.9	18.42%	8747.8	17.61%
Data2	43200.0	20.88%	2308.6	17.84%	10452.5	17.21%
Data3	43200.0	20.01%	1603.1	18.34%	5635.1	18.38%
Data4	43200.0	21.13%	3021.2	18.29%	7891.2	18.21%
Data5	43200.0	20.95%	2191.0	17.96%	8538.7	18.05%
Data6	43200.0	21.45%	4441.4	17.94%	11372.0	17.72%
Data7	43200.0	20.05%	1701.5	18.56%	10246.1	17.31%
Data8	43200.0	19.97%	4015.5	18.58%	5021.2	18.32%
Data9	43200.0	20.45%	2020.1	18.33%	5681.1	17.75%
Data10	43200.0	20.34%	1895.9	18.02%	9815.4	17.85%
Average	**43200.0**	**20.57%**	**2827.6**	**18.23%**	**8340.1**	**17.84%**

In Table 4, an alternative scenario is investigated by increasing transportation cost ratio. To generate this scenario, travel costs are multiplied with 2 (doubled) on same instance set. Notice that doubling travel rates does not necessarilily double transportation cost ratio of our network. When such case occurs, the problem is inclined to open more facilities in order to elude high travel rates. Therefore, some portion of increase in expense may materialize in other cost items. From this perspective, we realize that, with this new cost structure, in *IA*, on average 20 supplier is selected for the procurement of materials and 20.6 facility is opened. In this scenario again final resuls of heuristic algorithms look significantly better. In all data instances *TPCIA* manage to beat *IA* in terms of solution quality. However, as we noticed in previous scenario, solution quality of *TPCIA* is only slightly better than *IA* and the computational time of *IA* is again much shorter. Apart from these, after increased travel rates, drop in solution time for both heuristics stands out.

In order to analyze even higher travel rates and discuss what would they result, next scenario presents a cost structure where travel costs are multiplied with 3

Table 4: Comparison of results obtained by heuristic algorithms and mathematical model for scenario where transportation cost is doubled

Data Type	**CPLEX**		**IA**		**TPCIA**	
1000×100×100	**Time(sec)**	**Gap**	**Time(sec)**	**Gap**	**Time(sec)**	**Gap**
Data1	43200.0	19.28%	2033.9	15.62%	5329.8	13.76%
Data2	43200.0	16.71%	1637.6	13.56%	6951.0	14.42%
Data3	43200.0	15.26%	1750.8	14.88%	9843.2	14.98%
Data4	43200.0	19.36%	2178.5	15.23%	6849.5	15.54%
Data5	43200.0	12.61%	1679.7	12.01%	7614.3	11.85%
Data6	43200.0	13.89%	1496.4	12.28%	8021.2	11.76%
Data7	43200.0	18.54%	1788.8	16.86%	6718.5	15.22%
Data8	43200.0	19.14%	1252.1	16.61%	5991.7	15.45%
Data9	43200.0	15.59%	2012.2	13.99%	6423.9	14.01%
Data10	43200.0	16.98%	1561.0	16.25%	7187.4	14.67%
Average	**43200.0**	**16.74%**	**1739.1**	**14.73%**	**7093.1**	**14.17%**

(tripled). We observe that as we increase transportation rates solution times for both heuristics continue to decrease. Besides, this time around, *TPCIA* find significantly better results than *IA* and heuristic results continue to outperform Exact Algorithm with increasing rates. On average 21.1 supplier is selected and 22.9 facility is opened.

Table 5: Comparison of results obtained by heuristic algorithms and mathematical model for scenario where tranportation cost is tripled

Data Type	**CPLEX**		**IA**		**TPCIA**	
1000×100×100	**Time(sec)**	**Gap**	**Time(sec)**	**Gap**	**Time(sec)**	**Gap**
Data1	43200.0	12.67%	1389.8	11.68%	5913.2	9.71%
Data2	43200.0	16.14%	1407.5	11.14%	4283.8	10.76%
Data3	43200.0	12.13%	2246.7	11.78%	4933.5	10.20%
Data4	43200.0	13.08%	1630.5	10.12%	5016.4	9.23%
Data5	43200.0	11.47%	1458.1	10.40%	5251.4	9.98%
Data6	43200.0	12.93%	2200.7	11.19%	4615.2	9.95%
Data7	43200.0	16.49%	1514	14.68%	5310.7	10.20%
Data8	43200.0	16.44%	1730.1	12.06%	4999.3	11.45%
Data9	43200.0	12.35%	1962.2	10.53%	5107.0	9.91%
Data10	43200.0	14.46%	1160.1	12.97%	4801.2	11.83%
Average	**43200.0**	**13.82%**	**1670.0**	**11.65%**	**5023.2**	**10.32%**

Next, we change fixed cost ratio of our network. Table 6 illustrates a scenario where facility opening and facility operating costs are doubled on the same data set. A drastic increase in solution times of both heuristics is noticeable. Especially computation of *TPCIA* lasts quite long. However, *TPCIA* still provide best results and overall heuristic performance looks encouraging. Again to reflect a better understanding of our network, we share that on average 14.6 supplier is selected and 16.6 distribution facility is opened. In 8 instance result out of 10 *TPCIA* outperforms *IA*.

Table 6: Comparison of results obtained by heuristic algorithms and mathematical model for scenario where fixed costs are doubled

Data Type	**CPLEX**		**IA**		**TPCIA**	
1000×100×100	**Time(sec)**	**Gap**	**Time(sec)**	**Gap**	**Time(sec)**	**Gap**
Data1	43200.0	15.45%	5259.9	13.57%	11583.5	13.31%
Data2	43200.0	16.84%	3581.6	15.37%	10209.7	14.52%
Data3	43200.0	16.14%	7785.5	15.91%	13464.2	12.49%
Data4	43200.0	16.19%	4476.9	14.42%	13827.6	15.09%
Data5	43200.0	15.93%	5562.3	13.67%	11820.0	12.62%
Data6	43200.0	16.21%	6215.8	15.39%	14626.5	14.81%
Data7	43200.0	15.52%	4978.4	13.83%	12586.8	13.97%
Data8	43200.0	16.03%	7012.3	14.89%	13425.6	13.10%
Data9	43200.0	16.94%	5264.2	16.27%	11231.1	15.48%
Data10	43200.0	16.17%	3992.7	14.01%	10917.3	13.48%
Average	**43200.0**	**16.14%**	**5413.0**	**14.73%**	**12369.2**	**13.89%**

Table 7 reveals a scenario where facility opening and operating costs are tripled on the same instance set. The results obtained looks similar to previos one. Again computational times are quite long compared to earlier scenarios. Even if in overall *TPCIA* performs slightly better than *IA* when we examine each data instance seperately, we notice that *IA* manage to beat *TPCIA* in 5 case out of 10. Lastly, in this cost structure, 13 supplier is selected on average while 16 distribution facility is opened.

Table 7: Comparison of results obtained by heuristic algorithms and mathematical model for scenario where fixed cost are tripled

Data Type	**CPLEX**		**IA**		**TPCIA**	
1000×100×100	**Time(sec)**	**Gap**	**Time(sec)**	**Gap**	**Time(sec)**	**Gap**
Data1	43200.0	13.68%	7244.5	10.13%	15454.1	10.46%
Data2	43200.0	12.77%	5318.1	11.37%	10465.0	11.17%
Data3	43200.0	12.99%	5653.1	10.49%	10656.7	10.85%
Data4	43200.0	13.25%	3467.9	11.87%	12284.2	12.00%
Data5	43200.0	13.75%	3657.5	13.02%	10687.3	11.98%
Data6	43200.0	14.02%	7402.6	11.68%	17155.0	10.52%
Data7	43200.0	13.46%	6879.2	11.63%	13532.5	11.98%
Data8	43200.0	12.30%	3980.1	11.91%	11252.6	10.86%
Data9	43200.0	13.28%	4815.0	13.05%	12713.8	12.24%
Data10	43200.0	12.71%	5146.7	11.25%	10769.4	11.33%
Average	**43200.0**	**13.22%**	**5356.5**	**11.64%**	**12497.1**	**11.34%**

For the two remaining scenario, we increase the ratio of both transportation and fixed costs. In Table 8, we double the transportation and fixed cost ratios at the same time while in Table 9 we triple both transportation and fixed cost rates. It is noticed that computational times in Table 9 are especially high. Even if *TPCIA* manage to overthrow *IA* on average with a considerable amount for both scenario, in Table 8, *IA* beats *TPCIA* in 4 cases out of 10.

Table 8: Comparison of results obtained by heuristic algorithms and mathematical model for scenario where transportation cost and fixed costs are doubled

Data Type	**CPLEX**		**IA**		**TPCIA**	
1000×100×100	**Time(sec)**	**Gap**	**Time(sec)**	**Gap**	**Time(sec)**	**Gap**
Data1	43200.0	14.27%	4278.0	13.73%	13133.8	10.97%
Data2	43200.0	17.42%	2861.4	13.60%	10614.6	13.79%
Data3	43200.0	15.85%	3569.7	13.67%	9874.5	10.23%
Data4	43200.0	16.43%	4193.4	15.23%	10481.2	13.71%
Data5	43200.0	14.98%	5061.0	13.95%	11526.8	14.03%
Data6	43200.0	16.49%	4409.8	13.82%	13025.5	13.98%
Data7	43200.0	16.45%	2407.5	14.61%	12012.3	11.30%
Data8	43200.0	15.99%	2582.1	14.59%	1017.6	13.95%
Data9	43200.0	14.18%	3739.6	12.83%	10502.7	12.58%
Data10	43200.0	15.26%	4092.3	12.68%	11567.8	13.08%
Average	**43200.0**	**15.73%**	**3719.5**	**13.87%**	**10375.7**	**12.76%**

Table 9: Comparison of results obtained by heuristic algorithms and mathematical model for scenario where transportation cost and fixed costs are tripled

Data Type	**CPLEX**		**IA**		**TPCIA**	
1000×100×100	**Time(sec)**	**Gap**	**Time(sec)**	**Gap**	**Time(sec)**	**Gap**
Data1	43200.0	14.88%	4411.5	12.69%	12185.9	10.41%
Data2	43200.0	11.95%	7553.0	9.26%	16758.2	9.24%
Data3	43200.0	12.27%	5918.3	10.59%	11460.3	10.27%
Data4	43200.0	13.45%	6025.7	12.62%	11530.9	12.49%
Data5	43200.0	14.79%	5427.3	12.43%	10826.8	10.89%
Data6	43200.0	15.01%	7125.5	13.96%	15736.2	13.08%
Data7	43200.0	11.86%	4015.2	9.45%	14092.7	9.28%
Data8	43200.0	12.42%	4381.0	10.70%	13594.1	10.74%
Data9	43200.0	13.98%	5937.4	11.28%	12729.0	9.61%
Data10	43200.0	14.01%	6982.1	12.86%	13019.4	10.47%
Average	**43200.0**	**13.46%**	**5777.7**	**11.58%**	**13193.4**	**10.65%**

Since the bounds we found is not that accurate, with the intention of capturing more concrete results we apply same scenario analysis on a smaller instance set. This set includes 100 customers, 25 suppliers and 25 candidate facility locations.Notice that since we work on a small instance set, we also need to adapt our pre-selected heuristic parameters. In order make a fair comparison and evaluate all results under same conditions, we reduce the values of those parameters by keeping previous data size ratios as shown in Table 10.

Table 10: Values assigned to heuristic parameters

Notation	**Assigned Figures**	**Description**
N	7	a pre-selected number which limits the size of $cand_m(N)$
s	3	limits the number of iteration without improvement as tc
n_1	3	Both n_1 and n_2 are used in *TPCIA* where $N = n_1 + n_2$
n_2	4	

Again we start to our analysis on small instance set with base scenario as illustrated in Table 11. We realize that CPLEX computational time for the exact algorithm is still quite big, even if most of the test instances manage to reach to default gap. In this study, we aim to solve large instances and the reason we also investigate small instance set is to demonstrate that performance of our heuristics are not inefficient. For that reason, we believe that by incorporating a small gap into CPLEX, we can still get what we want and save time. After observed CPLEX results, we distinguish that solution time is getting slower after 1% gap. Therefore, remaining results for small instance set are obtained with inclusion of 1% gap. In order to show the difference after this gap inclusion, we also solve base scenario of small instance set one more time as presented in Table 12. Both tables show that *TPCIA* provide significantly better results than *IA*. Also notice that computational times of heuristics are way shorter.

Table 11: Comparison of results obtained by heuristic algorithms and mathematical model for base scenario

Data Type	CPLEX		IA		TPCIA	
100×25×25	**Time(sec)**	**Gap**	**Time(sec)**	**Gap**	**Time(sec)**	**Gap**
Data1	43200.0	2.23%	156.8	4.00%	135.8	1.78%
Data2	23290.5	0.01%	83.6	1.76%	275	0.68%
Data3	16975.8	0.01%	132.9	3.63%	141.2	1.05%
Data4	40596.1	0.01%	293.3	1.87%	247.2	0.01%
Data5	43200.0	3.07%	138.7	4.03%	153	3.13%
Data6	43200.0	0.26%	27.5	2.22%	210.1	1.78%
Data7	24099.4	0.01%	52.2	1.22%	270.8	0.33%
Data8	10105.9	0.01%	12.9	1.98%	25.7	0.92%
Data9	9952.6	0.01%	11.3	1.07%	33.3	0.61%
Data10	43200.0	0.18%	15.9	0.63%	162.8	0.91%
Average	**29782.0**	**0.58%**	**92.5**	**2.24%**	**165.5**	**1.12%**

Table 12: Comparison of results obtained by heuristic algorithms and mathematical model for base scenario with 1% optimality gap

Data Type	CPLEX		IA		TPCIA	
100×25×25	**Time(sec)**	**Gap**	**Time(sec)**	**Gap**	**Time(sec)**	**Gap**
Data1	43200.0	2.23%	156.8	4.00%	135.8	1.78%
Data2	4313.6	1.00%	83.6	2.28%	275.0	1.19%
Data3	2632.4	1.01%	132.9	4.08%	141.2	1.49%
Data4	14235.7	1.00%	293.3	2.20%	247.2	0.42%
Data5	43200.0	3.07%	138.7	4.03%	153.0	3.13%
Data6	16245.5	1.00%	27.5	2.68%	210.1	2.24%
Data7	8096.7	0.92%	52.2	1.76%	270.8	0.76%
Data8	1635.3	1.00%	12.9	2.52%	25.7	1.45%
Data9	603.1	1.01%	11.3	1.47%	33.3	1.01%
Data10	14207.6	1.01%	15.9	1.28%	162.8	1.57%
Average	**14837.0**	**1.32%**	**92.5**	**2.63%**	**165.5**	**1.50%**

Tables 13 and 14 present a scenario where transportation cost is doubled and tripled respectively. Similar to our previous remark during our investigation on large instance set, although computational times are already small, increasing travel rates reduce solution time considerably for both heuristics. Additionally, we notice that same remark may also apply to CPLEX results since there is a serious decrease in solution times compared to previous scenario.

Table 13: Comparison of results obtained by heuristic algorithms and mathematical model for scenario where transportation cost is doubled

Data Type	**CPLEX**		**IA**		**TPCIA**	
100×25×25	**Time(sec)**	**Gap**	**Time(sec)**	**Gap**	**Time(sec)**	**Gap**
Data1	3407.2	1.01%	24.7	1.73%	133.6	1.29%
Data2	3511.1	0.94%	116.7	1.88%	111.8	1.07%
Data3	1054.5	1.00%	31.8	2.09%	125.2	1.83%
Data4	16125.5	1.01%	56.7	2.36%	65.0	0.99%
Data5	3101.8	0.81%	94.9	2.77%	88.6	0.93%
Data6	2456.5	1.01%	57.7	2.20%	67.3	1.29%
Data7	5012.2	1.01%	20.4	1.36%	51.1	0.71%
Data8	3912.3	1.00%	21.5	2.67%	26.0	1.59%
Data9	1767.2	1.02%	10.3	1.37%	32.8	1.31%
Data10	2942.1	1.01%	12.1	1.23%	29.1	1.06%
Average	**4329.0**	**0.98%**	**44.7**	**1.97%**	**73.1**	**1.21%**

Table 14: Comparison of results obtained by heuristic algorithms and mathematical model for scenario where transportation cost is tripled

Data Type	**CPLEX**		**IA**		**TPCIA**	
100×25×25	**Time(sec)**	**Gap**	**Time(sec)**	**Gap**	**Time(sec)**	**Gap**
Data1	276.1	1.00%	15.1	1.75%	38.5	1.20%
Data2	2950.3	0.90%	20.7	1.17%	27.0	1.38%
Data3	497.8	1.01%	25.9	1.18%	108.6	1.28%
Data4	1165.1	0.98%	27.6	1.38%	47.8	1.37%
Data5	3012.2	1.01%	17.7	0.30%	39.5	0.33%
Data6	567.9	1.00%	28.1	0.30%	34.7	0.52%
Data7	789.1	0.86%	10.5	1.46%	28.2	0.47%
Data8	1783.4	0.99%	14.5	2.26%	13.8	0.80%
Data9	3001.2	1.01%	8.9	1.30%	28.7	0.86%
Data10	2145.5	1.01%	10.3	0.88%	24.9	0.95%
Average	**1618.9**	**0.98%**	**17.9**	**1.20%**	**39.2**	**0.92%**

Tables 15 and 16 provide a scenario where fixed costs are doubled and tripled respectively. Again computational times of heuristic algorithms increased significantly compared to earlier scenarios. However, this time around we also notice a degradation in solution quality. Table 15 shows that performance of heuristic algorithm is not that efficient as expected to be. Table 16 on the other hand, indicates that *IA* was inadequate to handle this scenario.

Table 15: Comparison of results obtained by heuristic algorithms and mathematical model for scenario where fixed costs are doubled

Data Type	**CPLEX**		**IA**		**TPCIA**	
100×25×25	**Time(sec)**	**Gap**	**Time(sec)**	**Gap**	**Time(sec)**	**Gap**
Data1	43200.0	1.78%	150.0	6.53%	165.6	2.95%
Data2	5593.3	1.01%	81.5	2.78%	458.7	2.18%
Data3	3350.0	1.01%	212.9	4.57%	596.6	2.40%
Data4	6456.8	1.01%	314.8	2.99%	1153.5	1.59%
Data5	22732.1	1.01%	267.4	5.35%	639.0	3.81%
Data6	25005.4	1.01%	195.4	3.15%	210.4	4.21%
Data7	19679.8	1.01%	184.3	4.11%	298.2	2.01%
Data8	5634.8	1.01%	16.6	1.73%	26.6	1.50%
Data9	3187.2	1.01%	18.6	2.02%	27.0	0.87%
Data10	14267.8	1.01%	23.7	3.56%	88.5	1.66%
Average	**14910.7**	**1.09%**	**146.5**	**3.68%**	**366.4**	**2.32%**

Table 16: Comparison of results obtained by heuristic algorithms and mathematical model for scenario where fixed costs are tripled

Data Type	**CPLEX**		**IA**		**TPCIA**	
100×25×25	**Time(sec)**	**Gap**	**Time(sec)**	**Gap**	**Time(sec)**	**Gap**
Data1	43200.0	2.32%	111.2	7.27%	130.9	3.34%
Data2	8472.1	1.00%	185.7	4.95%	477.3	1.92%
Data3	3774.0	1.01%	244.7	7.93%	382.0	1.02%
Data4	8958.9	1.01%	327.7	5.58%	97.1	1.00%
Data5	12827.3	1.01%	211.4	5.75%	102.7	1.01%
Data6	8210.9	1.01%	190.0	4.83%	505.2	1.01%
Data7	11438.6	1.01%	186.9	4.02%	432.9	1.03%
Data8	7902.1	1.01%	32.0	3.94%	91.5	3.41%
Data9	3967.8	1.01%	54.5	2.48%	62.7	1.28%
Data10	10298.2	1.01%	79.3	4.87%	124.8	1.02%
Average	**11905.0**	**1.14%**	**162.3**	**5.16%**	**240.7**	**1.60%**

Again our final scenarios are doubling and tripling both transportation cost and fixed costs on small instance set as shown in Table 17 and 18 respectively. As expected, solution times of those scenarios are lower than only increasing fixed costs but higher than only increasing trasportation cost. Especially the results from Table 18 looks decent enough.

Table 17: Comparison of results obtained by heuristic algorithms and mathematical model for scenario where transportation cost and fixed costs are doubled

Data Type	**CPLEX**		**IA**		**TPCIA**	
100×25×25	**Time(sec)**	**Gap**	**Time(sec)**	**Gap**	**Time(sec)**	**Gap**
Data1	5761.4	1.01%	49.0	3.57%	280.2	1.15%
Data2	2140.0	0.93%	184.7	1.79%	76.7	0.57%
Data3	1148.7	1.01%	52.6	4.83%	206.3	2.09%
Data4	4567.2	0.98%	90.2	2.02%	292.0	1.45%
Data5	4965.7	1.01%	186.9	2.29%	251.7	2.23%
Data6	12942.0	1.01%	87.1	1.73%	109.3	2.03%
Data7	5466.5	0.96%	45.3	3.72%	191.5	0.86%
Data8	1089.2	1.01%	14.6	2.40%	23.6	1.31%
Data9	603.2	0.95%	9.0	2.03%	51.3	1.40%
Data10	4212.1	1.00%	23.2	2.26%	195.9	1.26%
Average	**4289.6**	**0.99%**	**74.3**	**2.66%**	**167.9**	**1.43%**

Table 18: Comparison of results obtained by heuristic algorithms and mathematical model for scenario where transportation cost and fixed costs are tripled

Data Type	**CPLEX**		**IA**		**TPCIA**	
100×25×25	**Time(sec)**	**Gap**	**Time(sec)**	**Gap**	**Time(sec)**	**Gap**
Data1	5457.5	1.01%	54.7	1.49%	249.6	1.09%
Data2	1097.9	0.87%	383.9	2.05%	76.4	0.72%
Data3	4042.2	1.01%	346.3	2.69%	346.2	1.02%
Data4	4451.1	1.01%	348.9	1.68%	294.1	0.90%
Data5	4841.2	0.98%	166.0	1.45%	462.2	0.86%
Data6	4966.7	1.01%	167.2	1.57%	125.6	1.62%
Data7	4021.2	1.01%	194.4	2.67%	299.5	1.07%
Data8	3412.2	1.00%	201.3	1.12%	27.0	0.98%
Data9	798.2	0.99%	65.2	1.13%	85.3	0.69%
Data10	3981.2	1.01%	245.2	0.85%	189.9	1.52%
Average	**3706.9**	**0.99%**	**217.3**	**1.67%**	**215.6**	**1.05%**

Apart from these, we generate an additional third instance set. This set includes 1000 customers, 20 suppliers and 100 candidate facility locations. Again 7 different scenario is implemented as presented in Tables 19 to 25. Our previous remarks mostly comply with this new data set. As always, scenarios where we increase fixed cost rates looks most challenging.

Table 19: Comparison of results obtained by heuristic algorithms and mathematical model for base scenario

Data Type	CPLEX		IA		TPCIA	
1000×20×100	**Time(sec)**	**Gap**	**Time(sec)**	**Gap**	**Time(sec)**	**Gap**
Data1	43200.0	16.07%	1446.6	12.27%	9720.4	10.14%
Data2	43200.0	15.69%	1283.3	11.98%	7397.2	11.34%
Data3	43200.0	14.82%	1327.8	13.05%	8939.2	11.59%
Data4	43200.0	16.53%	1654.9	12.43%	9092.6	12.51%
Data5	43200.0	15.10%	1210.6	13.65%	6873.6	13.96%
Data6	43200.0	14.69%	1728.4	11.45%	9470.7	10.78%
Data7	43200.0	16.23%	2004.1	13.30%	9550.2	11.45%
Data8	43200.0	16.84%	1961.5	12.45%	9635.6	11.19%
Data9	43200.0	16.05%	2298.3	12.71%	12601.5	10.32%
Data10	43200.0	14.03%	1482.5	12.10%	5381.4	12.57%
Average	**43200.0**	**15.60%**	**1639.8**	**12.54%**	**8866.2**	**11.58%**

Table 20: Comparison of results obtained by heuristic algorithms and mathematical model for scenario where transportation cost is doubled

Data Type	CPLEX		IA		TPCIA	
1000×20×100	**Time(sec)**	**Gap**	**Time(sec)**	**Gap**	**Time(sec)**	**Gap**
Data1	43200.0	9.33%	1728.2	8.77%	5753.8	8.83%
Data2	43200.0	8.14%	1217.3	8.61%	5681.7	7.97%
Data3	43200.0	9.94%	1405.3	9.06%	1952.8	8.15%
Data4	43200.0	9.17%	1584.3	7.35%	6024.5	8.29%
Data5	43200.0	9.53%	1346.2	8.62%	2571.4	8.59%
Data6	43200.0	8.19%	1323.6	8.17%	4012.5	7.42%
Data7	43200.0	10.26%	1789.8	8.26%	6361.9	8.04%
Data8	43200.0	8.73%	1371.6	8.43%	1841.9	8.25%
Data9	43200.0	9.05%	1814.9	7.82%	5727.3	7.31%
Data10	43200.0	9.90%	1522.2	8.95%	3069.2	9.52%
Average	**43200.0**	**9.22%**	**1510.3**	**8.40%**	**4299.7**	**8.24%**

Table 21: Comparison of results obtained by heuristic algorithms and mathematical model for scenario where transportation cost is tripled

Data Type	CPLEX		IA		TPCIA	
1000×20×100	**Time(sec)**	**Gap**	**Time(sec)**	**Gap**	**Time(sec)**	**Gap**
Data1	43200.0	6.65%	1244.4	5.95%	5707.6	5.62%
Data2	43200.0	6.91%	1220.1	5.92%	2460.8	5.98%
Data3	43200.0	7.03%	1326.7	6.17%	5916.3	5.93%
Data4	43200.0	6.19%	1421.0	6.03%	3926.4	5.65%
Data5	43200.0	6.48%	1275.4	5.68%	6160.4	5.21%
Data6	43200.0	6.93%	1292.5	5.97%	2737.8	5.99%
Data7	43200.0	6.49%	1307.2	6.01%	4354.5	5.67%
Data8	43200.0	6.80%	1415.1	6.27%	2819.0	5.48%
Data9	43200.0	6.26%	1214.8	5.70%	3571.9	5.52%
Data10	43200.0	6.84%	1261.9	6.14%	3175.6	5.96%
Average	**43200.0**	**6.66%**	**1297.9**	**5.98%**	**4083.0**	**5.70%**

Table 22: Comparison of results obtained by heuristic algorithms and mathematical model for scenario where fixed costs are doubled

Data Type	CPLEX		IA		TPCIA	
1000×20×100	Time(sec)	Gap	Time(sec)	Gap	Time(sec)	Gap
Data1	43200.0	11.93%	3078.2	9.61%	10491.9	9.57%
Data2	43200.0	14.66%	6899.0	12.02%	14443.4	10.87%
Data3	43200.0	12.28%	4519.4	9.87%	12631.5	9.21%
Data4	43200.0	14.21%	6268.5	9.93%	12077.2	9.97%
Data5	43200.0	14.97%	5619.3	12.46%	10466.1	10.52%
Data6	43200.0	11.76%	3512.7	9.09%	9179.4	9.75%
Data7	43200.0	12.83%	6035.2	10.95%	10178.2	9.00%
Data8	43200.0	13.61%	5719.6	12.58%	14628.1	9.45%
Data9	43200.0	11.04%	3857.0	10.16%	9419.4	9.33%
Data10	43200.0	12.73%	4215.2	11.52%	9928.0	11.64%
Average	**43200.0**	**13.00%**	**4972.4**	**10.82%**	**11344.3**	**9.93%**

Table 23: Comparison of results obtained by heuristic algorithms and mathematical model for scenario where fixed costs are tripled

Data Type	CPLEX		IA		TPCIA	
1000×20×100	Time(sec)	Gap	Time(sec)	Gap	Time(sec)	Gap
Data1	43200.0	8.98%	5356.6	8.91%	13046.9	8.58%
Data2	43200.0	10.69%	6076.6	8.63%	14042.6	8.87%
Data3	43200.0	10.27%	5891.1	9.24%	13194.5	8.66%
Data4	43200.0	9.14%	5791.5	8.11%	14028.1	8.19%
Data5	43200.0	9.61%	5318.2	8.59%	14837.3	8.33%
Data6	43200.0	11.45%	6021.9	8.51%	10073.5	8.41%
Data7	43200.0	9.76%	5180.0	9.23%	10591.7	8.14%
Data8	43200.0	9.55%	4720.5	8.23%	12509.1	8.10%
Data9	43200.0	10.62%	6625.2	8.97%	11631.0	8.06%
Data10	43200.0	8.87%	5116.2	8.16%	13071.5	8.25%
Average	**43200.0**	**9.89%**	**5609.8**	**8.66%**	**12702.6**	**8.36%**

Table 24: Comparison of results obtained by heuristic algorithms and mathematical model for scenario where transportation cost and fixed costs are doubled

Data Type	CPLEX		IA		TPCIA	
1000×20×100	Time(sec)	Gap	Time(sec)	Gap	Time(sec)	Gap
Data1	43200.0	13.80%	3716.8	10.59%	11961.1	10.41%
Data2	43200.0	12.20%	4464.1	9.58%	9793.0	8.42%
Data3	43200.0	14.30%	4951.5	10.92%	9857.2	10.83%
Data4	43200.0	11.40%	3410.1	8.12%	10977.2	8.18%
Data5	43200.0	11.10%	5206.4	9.26%	13845.1	8.50%
Data6	43200.0	12.50%	5328.2	9.44%	12484.1	9.15%
Data7	43200.0	11.30%	5667.1	9.10%	9784	8.87%
Data8	43200.0	13.70%	3526.7	10.81%	9249.1	9.52%
Data9	43200.0	14.10%	3488.3	10.75%	13145.4	9.82%
Data10	43200.0	13.00%	4379.9	9.19%	9563.4	9.04%
Average	**43200.0**	**12.74%**	**4413.9**	**9.78%**	**11066.0**	**9.27%**

Table 25: Comparison of results obtained by heuristic algorithms and mathematical model for scenario where transportation cost and fixed costs are tripled

Data Type	**CPLEX**		**IA**		**TPCIA**	
1000×20×100	**Time(sec)**	**Gap**	**Time(sec)**	**Gap**	**Time(sec)**	**Gap**
Data1	43200.0	8.56%	5238.1	7.96%	12146.8	7.20%
Data2	43200.0	10.06%	5646.1	9.56%	13503.0	8.62%
Data3	43200.0	9.52%	5840.7	8.92%	12565.3	8.18%
Data4	43200.0	10.12%	5093.9	9.22%	12267.6	8.46%
Data5	43200.0	8.58%	5891.8	7.98%	12716.5	7.18%
Data6	43200.0	8.67%	5340.6	7.97%	11406.6	7.14%
Data7	43200.0	10.28%	5547.3	9.68%	13832.4	8.80%
Data8	43200.0	8.85%	5168.9	7.35%	11339.0	6.53%
Data9	43200.0	10.35%	5895.7	9.75%	13030.2	8.84%
Data10	43200.0	9.50%	5067.7	8.80%	10473.3	7.90%
Average	**43200.0**	**9.45%**	**5473.1**	**8.72%**	**12328.1**	**7.89%**

To give you a brief idea about computational burden of each phase of *TPCIA*, we also want to share another additional information. According to computational results, first phase of *TPCIA* lasts around 20% on average of entire *TPCIA* time.

Our computational results show that when the portion of transportation costs are high, our heuristic algorithms manage to find very efficient results in terms of both solution quality and computational time. It is actually not surprising because in order to further simplify the problem, our algorithms solve *SP2* with a candidate facility set $cand_m(N)$. Initialization process for assigning this facility set for each customer is selecting closest N facilities to customer m. Our intuiton is that when travel rates are high, it is natural to be more inclined to open closest facilities and it makes $cand_m(N)$ a good fit.

We discern that in most case, it is more challenging for our heuristics to handle problems with large facility opening costs. That environment is reflected to our results by affecting whether the solution time, solution quality or both. In most case, we observe a significant increase in solution time and in some cases even our solution quality is influenced negatively (especially *IA*). Obviously, when opening costs are high, it is usual to open less facility than normal. When such case occurs, determining a candidate facility set and moreover initializing that set based on closest ones may limit possible alternatives. That is especially true for *IA* since differentiation of facility sets is really difficult during iterations. *TPCIA* on the other hand, has more chance to differentiate its initial facility sets by trying alternative facility scenarios during *Phase 1*. Apart from these, it is important to underline that long computational runs of our algorithms do not emerge from only problem difficulty. Since both of our algorithms have iterative based steps, as the number of iteration increases, solution time also increases. For example, average computational time of a scenario where transportation cost is doubled equals to 7093.1 for *TPCIA* as shown in Table 4. On the other hand, average computational time for scenario where the fixed costs are doubled equal to 12369.2 as presented in Table 6. However, when we check the iteration count, we notice that second scenario do almost twice as many iterations on average compared to first one. So from that perspective, it is natural for the second scenario to have longer run. From that perspective our intuition is that, with bigger fixed costs, converging may be more difficult since opening decision of a facility is more threatening.

When we compare both of our heuristics, we notice that *TPCIA* provide better results in general by sacrificing from computational time. In some cases, it may not be that beneficial to afford time-consuming *TPCIA* iterations for only a slight improvement (see Tables such as 21, 23, 24). When each result is examined one by one, it is easy to discover that *TPCIA* have more tendency to capture a significantly better result compared to *IA*. To be more precise, let us consider Tables 7, 8 and 9. We notice that in Table 7, *IA* outperforms *TPCIA* in 5 data instance out of 10 while for Tables 8 and 9, this number equals to 4. However, when overall results are examined, *TPCIA* stands out with significantly better results compared to *IA*. It is mainly because in each scenario type, there are always a couple of instances where *TPCIA* provide substantially better results. Separately, it is crucial to adress the importance of pre-selected heuristic parameters; N, n_1 and n_2. It becomes more evident during our tests with small instance set. As it is stated, in order to make a fair comparison with large instances and evaluate all instance sets under the same conditions, while assigning values to those parameters we look out to keep a constant ratio designated based on instance size. For that reason, we decrease our N value to 7 from 25 while evaluating small instances. However, it is not actually possible to be sure that whether we manage to ensure same conditions or not. Because selecting N value as 25 out of 100 candidate facility for a large instance or using N as 7 out of 25 candidate facility may not mean similar cases and one of them may cause a much tighter situation. To evaluate the importance N selection, we additionally solve base scenario of small instance with an alternative N value. In this alternative, we increase N's value only by 1 and equate it to 8. Our average *IA* result is improved from 2.24% to 1.78%. On the other hand, equating n_1 to 4 by increasing it 1 for *TPCIA* only bring a very small overall improvement from 1.12% to 1.09%. However, as a contradictory attempt, decreasing n_2 to 3 from 4 for *TPCIA* would result a redcution of 0.57% in *TPCIA* solution quality.

Notice that the results presented for heuristic algorithms use a 1000 seconds time limit while solving *S1* and *S2*. We want to mention that by increasing that time limit, it may be possible to further increase the solution quality in exchange for additional computational time. Though our preliminary results show that it produces only a slight improvement.

In search of further improvement for our network structure, we also implement our heuristic results in exact MIP model as a starting solution. Starting values for MIP is established by only feeding binary variables from our final *TPCIA* results. In Table 26, a preliminary run for small instance set is performed to have a brief idea about possible outcomes.

Table 26: Preliminary Comparison with MIP start for small instance set with 1% optimality gap

Data Type	**CPLEX**		**CPLEX with Starting Solution**	
Base Data	**Time(sec)**	**Gap**	**Time(sec)**	**Gap**
Data1	43200.0	2.23%	43200.0	1.57%
Data2	4313.6	1.00%	3871.4	1.00%
Data3	2632.4	1.01%	2579.1	0.99%
Data4	14235.7	1.00%	12988.5	1.01%
Data5	43200.0	3.07%	43200.0	1.79%
Data6	16245.5	1.00%	14631.2	1.01%
Data7	8096.7	0.92%	4554.8	1.01%
Data8	1635.3	1.00%	2274.3	0.98%
Data9	603.1	1.01%	869.5	1.01%
Data10	14207.6	1.01%	13214.7	1.00%
Average	**14837.0**	**1.32%**	**14138.4**	**1.14%**

The preliminary results show that starting solutions do not necessarily improve computational time. Our intuition is that feeding some initial variables may lead to find good solutions quicker especially in instances where achieving a good result is comparably more difficult. On the other hand, after found a good solution, improvement time is getting slower. However, we believe that for concrete remarks, it is necessary to perform further tests.

CHAPTER VI

CONCLUSION

In this thesis, we investigate a two echelon supply chain network where the distribution facilities procure the end-product from the suppliers which offer quantity discount and satisfy the customer demand. Our objective is to select suppliers and locate the distribution facilities in a given planning horizon in order to minimize purchasing, distribution and facility opening costs. To address the problem on hand, we initially formulate it as a mixed integer programming problem. Additionally, to handle large instances, we develop two heuristic algorithms: *Iterative Algorithm* (*IA*) and *Two-Phase Clustered Iterative Algorithm* (*TPCIA*). In order to simplify the problem, a specific candidate facility set where each customer can only be served from facilities included to their own facility set is created. *IA* is an MIP based iterative algorithm where we solve each echelon in an iterative manner by using those candidate sets. *TPCIA* is an extended version of *IA* and consists of two phases. First phase involves generation of specific clusters in order to reduce facility set defined in problem network. Second phase solve the MIP of each echelon iteratively by using clusters generated in first phase. Finally we test the performance of our algorithms by implementing different scenarios on randomly generated instances. We observe that results of our heuristic algorithms which have a computational run of 1 to 3 hours , outperform 12-hour solution of exact MIP formulation by 3% on average.

Bibliography

[1] M. T. Melo, S. Nickel, and F. S. Da Gama, "Dynamic multi-commodity capacitated facility location: a mathematical modeling framework for strategic supply chain planning," *Computers & Operations Research*, vol. 33, no. 1, pp. 181–208, 2006.

[2] Z. Shen, "Integrated supply chain design models: a survey and future research directions," *Journal of Industrial and Management Optimization*, vol. 3, no. 1, pp. 1–27, 2007.

[3] M. T. Melo, S. Nickel, and F. Saldanha-da Gama, "Facility location and supply chain management–a review," *European Journal of Operational Research*, vol. 196, no. 2, pp. 401–412, 2009.

[4] R. Z. Farahani, M. Abedian, and S. Sharahi, "Dynamic facility location problem," in *Facility Location*, pp. 347–372, Springer, 2009.

[5] S. D. Jena, J.-F. Cordeau, and B. Gendron, "Solving a dynamic facility location problem with partial closing and reopening," *Computers & Operations Research*, vol. 67, pp. 143–154, 2016.

[6] B. M. Beamon, "Supply chain design and analysis: Models and methods," *International journal of production economics*, vol. 55, no. 3, pp. 281–294, 1998.

[7] M. B. Ayhan, "A fuzzy ahp approach for supplier selection problem: A case study in a gear motor company," *International Journal of Managing Value and Supply Chains*, vol. 4, no. 3, 2013.

[8] G. Schmidt and W. E. Wilhelm, "Strategic, tactical and operational decisions in multi-national logistics networks: a review and discussion of modelling issues," *International Journal of Production Research*, vol. 38, no. 7, pp. 1501–1523, 2000.

[9] S. H. Huang and H. Keskar, "Comprehensive and configurable metrics for supplier selection," *International Journal of Production Economics*, vol. 105, no. 2, pp. 510–523, 2007.

[10] R. Florez-Lopez, "Strategic supplier selection in the added-value perspective: A ci approach," *Information Sciences*, vol. 177, no. 5, pp. 1169–1179, 2007.

[11] A. Mendoza and J. A. Ventura, "A serial inventory system with supplier selection and order quantity allocation," *European Journal of Operational Research*, vol. 207, no. 3, pp. 1304–1315, 2010.

[12] S. H. Ghodsypour and C. O'Brien, "A decision support system for supplier selection using an integrated analytic hierarchy process and linear programming," *International Journal of Production Economics*, vol. 56, pp. 199–212, 1998.

[13] L. J. Krajewski and L. P. Ritzman, *Operations management: strategy and analysis*. Pearson College Division, 2001.

[14] C. A. Weber, J. R. Current, and W. Benton, "Vendor selection criteria and methods," *European Journal of Operational Research*, vol. 50, no. 1, pp. 2–18, 1991.

[15] M. B. Ayhan and H. S. Kilic, "A two stage approach for supplier selection problem in multi-item/multi-supplier environment with quantity discounts," *Computers & Industrial Engineering*, vol. 85, pp. 1–12, 2015.

[16] M. S. Daskin, *Network and discrete location: models, algorithms, and applications*. John Wiley & Sons, 2011.

[17] C. L. Munson and M. J. Rosenblatt, “Coordinating a three-level supply chain with quantity discounts,” *IIE Transactions*, vol. 33, no. 5, pp. 371–384, 2001.

[18] R. M. Ebrahim, J. Razmi, and H. Haleh, “Scatter search algorithm for supplier selection and order lot sizing under multiple price discount environment,” *Advances in Engineering Software*, vol. 40, no. 9, pp. 766–776, 2009.

[19] G. Wang, S. H. Huang, and J. P. Dismukes, “Product-driven supply chain selection using integrated multi-criteria decision-making methodology,” *International Journal of Production Economics*, vol. 91, no. 1, pp. 1–15, 2004.

[20] R. J. Dolan, “Quantity discounts: Managerial issues and research opportunities,” *Marketing Science*, vol. 6, no. 1, pp. 1–22, 1987.

[21] Y. Hinojosa, J. Puerto, and F. R. Fernández, “A multiperiod two-echelon multicommodity capacitated plant location problem,” *European Journal of Operational Research*, vol. 123, no. 2, pp. 271–291, 2000.

[22] S. Tragantalerngsak, J. Holt, and M. Rönnqvist, “An exact method for the two-echelon, single-source, capacitated facility location problem,” *European Journal of Operational Research*, vol. 123, no. 3, pp. 473–489, 2000.

[23] D. Caglar, C.-L. Li, and D. Simchi-Levi, “Two-echelon spare parts inventory system subject to a service constraint,” *IIE Transactions*, vol. 36, no. 7, pp. 655–666, 2004.

[24] H. Wong, D. V. Oudheusden, and D. Cattrysse, “Two-echelon multi-item spare parts systems with emergency supply flexibility and waiting time constraints,” *IIE Transactions*, vol. 39, no. 11, pp. 1045–1057, 2007.

[25] M. D. Rossetti and S. Thomas, "Object-oriented multi-indenture multi-echelon spare parts supply chain simulation model," *International Journal of Modelling and Simulation*, vol. 26, no. 4, pp. 359–369, 2006.

[26] S. Kolay, G. Shaffer, and J. A. Ordover, "All-units discounts in retail contracts," *Journal of Economics & Management Strategy*, vol. 13, no. 3, pp. 429–459, 2004.

[27] M. Motta, "Michelin ii - the treatment of rebates," *Cases in European competition policy*, pp. 29–49, 2009.

[28] J. Bughin and M. Chui, "Evolution of the networked enterprise: Mckinsey global survey results," *McKinsey Quarterly*, vol. 29, pp. 34–41, 2013.

[29] S. A. Alumur, B. Y. Kara, and M. T. Melo, "Location and logistics," in *Location Science*, pp. 419–441, Springer, 2015.

[30] C. Canel and B. M. Khumawala, "A mixed-integer programming approach for the international facilities location problem," *International Journal of Operations & Production Management*, vol. 16, no. 4, pp. 49–68, 1996.

[31] J. E. Torres-Soto and H. Uster, "Dynamic-demand capacitated facility location problems with and without relocation," *International Journal of Production Research*, vol. 49, no. 13, pp. 3979–4005, 2011.

[32] A. Klose and A. Drexl, "Facility location models for distribution system design," *European Journal of Operational Research*, vol. 162, no. 1, pp. 4–29, 2005.

[33] S. H. Owen and M. S. Daskin, "Strategic facility location: A review," *European Journal of Operational Research*, vol. 111, no. 3, pp. 423–447, 1998.

[34] M. K. Boujelben, C. Gicquel, and M. Minoux, "A milp model and heuristic approach for facility location under multiple operational constraints," *Computers & Industrial Engineering*, vol. 98, pp. 446–461, 2016.

[35] F. M. Bass, "A new product growth for model consumer durables," *Management Science*, vol. 15, no. 5, pp. 215–227, 1969.

[36] G. Şahin and H. Süral, "A review of hierarchical facility location models," *Computers & Operations Research*, vol. 34, no. 8, pp. 2310–2331, 2007.

[37] C. H. Aikens, "Facility location models for distribution planning," *European Journal of Operational Research*, vol. 22, no. 3, pp. 263–279, 1985.

[38] Z. Drezner and H. W. Hamacher, *Facility location*. Springer-Verlag New York, NY, 1995.

[39] C. S. Revelle, H. A. Eiselt, and M. S. Daskin, "A bibliography for some fundamental problem categories in discrete location science," *European Journal of Operational Research*, vol. 184, no. 3, pp. 817–848, 2008.

[40] H.-b. Ro and D.-w. Tcha, "A branch and bound algorithm for the two-level uncapacitated facility location problem with some side constraints," *European Journal of Operational Research*, vol. 18, no. 3, pp. 349–358, 1984.

[41] L. Kaufman, M. V. Eede, and P. Hansen, "A plant and warehouse location problem," *Journal of the Operational Research Society*, vol. 28, no. 3, pp. 547–554, 1977.

[42] S. Tragantalerngsak, J. Holt, and M. Rönnqvist, "Lagrangian heuristics for the two-echelon, single-source, capacitated facility location problem," *European Journal of Operational Research*, vol. 102, no. 3, pp. 611–625, 1997.

[43] R. Z. Farahani, M. Hekmatfar, B. Fahimnia, and N. Kazemzadeh, "Hierarchical facility location problem: Models, classifications, techniques, and applications," *Computers & Industrial Engineering*, vol. 68, pp. 104–117, 2014.

[44] C. Canel, B. M. Khumawala, J. Law, and A. Loh, "An algorithm for the capacitated, multi-commodity multi-period facility location problem," *Computers & Operations Research*, vol. 28, no. 5, pp. 411–427, 2001.

[45] Y. Hinojosa, J. Kalcsics, S. Nickel, J. Puerto, and S. Velten, "Dynamic supply chain design with inventory," *Computers & Operations Research*, vol. 35, no. 2, pp. 373–391, 2008.

[46] R. Manzini and E. Gebennini, "Optimization models for the dynamic facility location and allocation problem," *International Journal of Production Research*, vol. 46, no. 8, pp. 2061–2086, 2008.

[47] R. M. Soland, "Optimal facility location with concave costs," *Operations Research*, vol. 22, no. 2, pp. 373–382, 1974.

[48] D. L. Kelly and B. M. Khumawala, "Capacitated warehouse location with concave costs," *Journal of the Operational Research Society*, vol. 33, pp. 817–826, 1982.

[49] L. Dupont, "Branch and bound algorthm for a facility location problem with concave site dependent costs," *International Journal of Production Economics*, vol. 112, pp. 245–254, 2008.

[50] J. V. Broek, P. Schutz, L. Stougie, and A. Tomasgard, "Location of slaughterhouses under economies of scale," *European Journal of Operational Research*, vol. 175, pp. 740–750, 2006.

[51] J. Harkness and C. ReVelle, "Facility location with increasing production costs," *European Journal of Operational Research*, vol. 145, no. 1, pp. 1–13, 2003.

[52] W. Xia and Z. Wu, "Supplier selection with multiple criteria in volume discount environments," *Omega*, vol. 35, no. 5, pp. 494–504, 2007.

[53] W. Benton, "Quantity discount decisions under conditions of multiple items, multiple suppliers and resource limitations," *International Journal of Production Research*, vol. 29, no. 10, pp. 1953–1961, 1991.

[54] H. Stadtler, "A general quantity discount and supplier selection mixed integer programming model," *OR Spectrum*, vol. 29, no. 4, pp. 723–744, 2007.

[55] G. J. Burke, J. Carrillo, and A. J. Vakharia, "Heuristics for sourcing from multiple suppliers with alternative quantity discounts," *European Journal of Operational Research*, vol. 186, no. 1, pp. 317–329, 2008.

[56] A. H. Lee, H.-Y. Kang, and C.-M. Lai, "Solving lot-sizing problem with quantity discount and transportation cost," *International Journal of Systems Science*, vol. 44, no. 4, pp. 760–774, 2013.

[57] J. Shu, T. Wu, and K. Zhang, "Warehouse location and two-echelon inventory management with concave operating cost," *International Journal of Production Research*, vol. 53, no. 9, pp. 2718–2729, 2015.

VITA

Görkem Emirhüseyinoğlu received the Bachelor of Science degree in Industrial Engineering from Özyeğin University, Istanbul, Turkey in 2014. After graduation, he joined Master of Science Program in Industrial Engineering at Özyeğin University and currently studies under supervision of Assoc. Prof. Ali Ekici. He has been working as a teaching assistant since and assisted in courses: Productions Systems Analysis, Simulation Modelling and Analysis, Risk Management, Project Management. His research interest mainly revolves around logistic systems.