ROUTING AND INVENTORY LOADING PROBLEM FOR HETEROGENEOUS VEHICLE FLEET WITH COMPARTMENTS

A Thesis

by

Türkay Umut Yılmaz

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
May, 2018

ROUTING AND INVENTORY LOADING PROBLEM FOR HETEROGENEOUS VEHICLE FLEET WITH COMPARTMENTS

Approved by:

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

Assistant Professor Dilek Günneç
Department of Industrial Engineering
Özyeğin University

Associate Professor Şule Itır Satoğlu
Department of Industrial Engineering
Istanbul Technical University

Date Approved: 4 May 2018

“Entia non sunt multiplicanda praeter necessitatem.”

William of Ockham

ABSTRACT

This thesis focuses on multi compartment heterogeneous vehicle fleet routing problems for the case of incompatible products dedicated to separately within the compartments. In order to supply the different product demand of the customers, delivery plans are being prepared with minimum logistic costs while incompatible products between compartments. While logistic cost is accepted as total distance in a unit period of time, objective function is formed by combination of problems in both routing and inventory loading. Although Vehicle Routing Problems (VRP) and variants can be seen frequently in the literature, Multiple Compartment Heterogeneous Vehicle Routing Problems are still being under investigation. As the discussed problem in thesis is composed of combination of two different NP-Hard problems, comprehensive mathematic model is proposed. Constructed model ensures minimum routing cost for each vehicle in use and minimum number of deliveries per a unit period. We propose two phase approach which contains clustering methodology, heuristics both routing and inventory loading problem and set partitioning problem all together iteratively. In small scale problem our solution approach obtained optimal solution comparing to mathematical model. For large scale problems, mathematical models cannot give a feasible solution. For this reason, Tabu Search methodology which is used for heterogeneous vehicle routing problem in literature has been applied. The performance of two approaches were compared. Suggested algorithm is producing rapid and qualified results especially for companies which are planning the product delivery such as food, fuel, live animal or chemicals.

ÖZETÇE

Bu tezde, çok kompartmanlı heterojen araç filolu dağıtım problemlerinde karışmayan ürünlere odaklanılmıştır. Müşterilerden gelen farklı tipteki ürün taleplerini karşılamak için en az lojistik maliyeti esas alınarak kompartmanlarda karışmayacak bir şekilde dağıtım planı hazırlanmaktadır. Lojistik maliyeti olarak; birim periyottaki katedilen uzaklık miktarlarının toplamı alındığından; amaç fonksiyonu hem rotalama hem de envanter yükleme probleminin birleşimi şeklindedir. Araç Rotalama Problemi (ARP) ve varyantları geniş bir literatüre sahip olduğu halde, Çok Kompartmanlı Heterojen Filo Rotalama Problemi alanı hâla araştırmaya açıktır. Tez kapsamında ele alınan problem iki ayrı NP-Zor problemin birleşimi olduğundan kapsamlı bir matematiksel model önerilmiştir. Oluşturulan model sayesinde her bir araç için en kısa rotalama hesaplanırken aynı zamanda birim periyottaki teslimat sayısını en aza indirgemeye çalışmaktadır. Envanterlerin araçların kompartmanlara atanması ve araçların rotalanması için iki aşamalı bir yaklaşım önermekteyiz. İki aşamalı çözüm yaklaşımımızda kümeleme metotları, rotalama ve yükleme için geliştirilen sezgisel algoritmalar ve küme bölme problemi birlikte kullanılmıştır. Matematiksel modelin performanslarını test etmek için sayısal deney hazırlanmıştır. Küçük ölçekli problem örnekleri için iki aşamalı geliştirilen sezgisel yöntem sayesinde küçük ölçekli problemler için optimal sonuçları matematiksel modelden daha hızlı bir şekilde bulduğu görülmüştür. Büyük ölçekli problem için matematiksel model çözüm verememektedir. Bu yüzden literatürde heterojen filolar için kullanılan Tabu Arama metodolojisi uygulanmış ve iki yaklaşımın performansları test edilmiştir. Gıda, yakıt, canlı hayvan veya kimyasal ürün sevkiyat planlaması yapan şirketler için; önerilen algoritma, hızlı ve kaliteli sonuçlar üretmektedir.

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to my thesis supervisor Assoc. Prof. Ali Ekici for his invaluable guidance, patience, encouragement and support throughout this study. It has been an honor and a great privilige to work with him. I have learned a lot of from him and I appreciate all his contributions of time and ideas to make master experience productive.

I am grateful to my CEO Dr. Onur Yazgan for his support and consideration to keep in line both business and academic life.

I would also like to thank my loving family, Hatice Yılmaz, Kemal Yılmaz and Tonguç Utku Yılmaz, for their support from home even far away they were always my motivation. My love Şebnem Erakyol for her never ending support, patience, optimisim for making me feel good in my diffucult times.

To all other people that their names are not mentioned here who contribute to this study; I would like to thank them all.

Contents

List of Tables

List of Figures

LIST OF ABBREVIATIONS

ALNS .... An Adaptive Large Neighborhood Search
FPP .... Fixed Partition Policy
GAA .... Greedy Assignment Algorithm
GRASP .... Greed Randomized Adaptive Search Procedure
IRP .... Inventory Routing Problem
ISC .... Iterative Sweep Algorithm
MILP .... Mixed Integer Linear Programming
ML .... Maximum Level
MPLP .... Multi Product Loading Problem
MPLPF .... Multi Product Loading Problem with Frequency
MIRP .... Maritime Inventory Routing Problem
NP-Hard .... Non-Deterministic Polynomial-Time Hard
OR .... Operations Research
OU .... Order-up-to-Level
PVRP .... Periodic Vehicle Routing Problem
RILPSVC .... Routing and Inventory Loading Problem for Single Vehicle with Compartments
RILPHVFC ...... Routing and Inventory Loading Problem for Heterogenous Vehicle Fleet with Compartments
RSC .... Random Subset Algorithm
SC .... Sweep Algorithm
SPP .... Set Partition Problem
TS .... Tabu Search
TSP .... Traveling Salesman Problem
VMI .... Vendor Menaged Inventory
VRP .... Vehicle Routing Problem
ZIO .... Zero Inventory Ordering

Chapter I

INTRODUCTION

Supply chain is a network of organizations that are comprised in moving a product or service from supplier to an end customer. Supply chain activities contain on raw material flow to finished goods which are delivered to an end customer. Importance of supply chain management is increasing day by day. As a reason, possible integrated examination of trade-offs in supply chain could improve overall performance of the system. Decision stages of supply chain management are strategic, tactical and operational. Improvement of supply chain must be integrated every stage to sustain business. However, improvement and development on strategic stage or level is potentially costly and complex which leads companies and institutions focus on tactical and operational stage of supply chain management. Decisions on tactical and operational level could be achieved by lower investments and efforts, even it made same effect comparing to strategic decision-making. For the illustration, principal components of logistic costs are which are inventory costs and transportation costs, representing successively one third and one fifth of logistic cost according to study [1]. Thus, an improvement on inventory and transportation management activities have essential role on increasing productivity and operational efficiencies. Researchers investigated problems such as Vehicle Routing Problem (VRP), Periodic Vehicle Routing Problem (PVRP) and Inventory Routing Problem (IRP) which are important and well known combinatorial problems, in order to minimize logistic cost meanwhile satisfying customer preferences and service level requirements.

One of the most studied combinatorial optimization problem is the VRP which exists in many distribution and transportation system with significant economic benefits.

Many organizations and companies in the area of transportation and logistic comply with varieties of VRP every day. Furthermore, optimizing operations such as deliveries or periodic pickups are made to a set of customers over a time horizon could be a cause of significant cost savings in many supply chains such as e.g. waste collection and grocery distribution. This problem is defined as the PVRP.

Vendor Managed Inventory (VMI) is an interesting concept of supply chain collaboration. Under the Vendor Managed Inventory concept, the distributors or suppliers take over responsibility in decision process to replenish customer according to inventory data provided to suppliers by customers. The IRP is based on idea that integrated inventory and distribution management provide a degree of freedom for constructing efficient routes while optimizing inventory across the supply chain. The objective of the IRP is to combine inventory management and transportation activities at the same time because of minimizing inefficiency of solving underlying inventory and vehicle routing sub-problems separately. Importance of IRP arises when working on Supply Chain integration.

In reality, customer requests delivery of multiple commodities which need to keep separately during storage and transportation in supply chain practice as grocery distribution, petroleum distribution and waste collection. As a result of that, multi compartment vehicles are in use to deliver commodities to customers and these vehicle have compartments which are dedicated to storage for different commodities. In literature, tendency is rising as simplifying IRP by considering single product (commodity) models or multi-products (commodities) models with joint storage in vehicle ignoring compartments in vehicle and inventories. Moreover in real world, the organizations and companies have heterogeneous vehicle fleet which ensures flexibility in investments and operations. Therefore most models assume that each customer delivered by vehicle or vehicle fleet which are homogenous structure and having homogenous compartments. The main contribution of this thesis is to investigate the

IRP in a multi compartment settings of vehicles that are heterogeneous. Each compartments of vehicle is dedicated to a specific customer demand, mixing products in compartments is not allowed, and delivery pattern of vehicle has static period structure. Moreover, our model is designed to choose which vehicle is whether in use or not.

The rest of this thesis is organized as follow; Chapter 2 provides comprehensive literature review related IRP and we classified academic studies of IRP according to our problem settings. Chapter 3 explains our problem structure in a broader sense with example and includes mathematical model. In the following Chapter 4, we investigate our problem in two part, the first part is dedicated to single vehicle version of the problem and the second part is dedicated to multi vehicle version of the problem. We have proved that even single version problem is very Np-Hard could not be solved in mathematical model and in order to reach optimal solution must be divided into two sub problems. We propose novel heuristic to solve loading problem and use existing heuristic (Christofides) to solve routing problem for the single vehicle version of the problem. Moreover, we proposed two-phase solution approach in order to solve multi vehicle version of the problem. In this approach we benefit from cluster technique to generate desirable solution space and integrated with set partition problem to choose best solution across the solutions. However, due to the limited existing work on multi compartment in heterogeneous vehicle structure, there is no available benchmark set. Therefore we have to compare our solution efficiency with other well-known metaheuristic Tabu Search. This is the second contribution of this thesis to academic literature. Finally, Chapter 5 provides computational results of proposed algorithms and Chapter 6 concludes obtained results with mentioning future research areas.

Chapter II

LITERATURE REVIEW

The problem we studied in this thesis is a variant of IRP. The IRP integrates vehicle routing, delivery scheduling and inventory management decisions. The IRP has been gained importance in VMI system which is withstanding and state of art business practice and strategy. The VMI is based on collaboration between a suppliers and its customers whose aim is creating business value and reducing logistic costs. Based on demand and inventory provided by customers to supplier, Supplier takes on responsibility of managing inventory of customers regarding to delivery periods and replenishment quantities of demands. In the VMI environment, replenishment strategy varies under the supply chain policies [2, 3, 4]. The VMI practice is absolutely win-win strategy for both parties. The buyers can not afford to control efficiently their inventories, as a result the supplier can choose best option for them regarding to distribution and production costs with right resource allocation. Three questions arise in the IRP for the side of suppliers which are (1) how much to deliver to customer when delivery accrued (2) how to construct route for costumers' orders in a delivery and (3) which time to serve a set of customers.

The IRP has been worked widely in Operations Research (OR) literature in many fields. For example, in maritime transportation, this problem is stated as Maritime Inventory Routing Problem (MIRP) which is a special class of IRP with maritime settings. Even if the MIRP is the subset of IRP, MIRP is widely studied by many researchers. The IRP is firstly introduced in the seminal-paper by [5] and they studied on the distribution of industrial gases from a central depot to customers by a fleet vehicles. Their aim is to minimize transportation costs in miles with no stock-outs.

They proposed Langrangian-relaxation based algorithm to solve mixed integer program. Moreover non-linear integer programming formulation is studied for minimizing total transportation, inventory holding and shortage cost [6]. The methodology they used is decomposing the main problem into inventory allocation problem and Travel Salesman Problem in a way of solving two parts iteratively. We refer the reader to excellence surveys [7, 8, 9, 10, 11] for MIRP, [12, 13, 14] for IRP.

To construct topology of the IRP, we prefer classify of our problem in eleven criteria based on the study [12, 13], additional criterias will be added as in Table 1.

Table 1: Classification for the IRP

Criteria	Possible Options		
Demand	Deterministic	Stochastic	
Planning Horizon	Single Period	Multiple Period	Infinite Time
Replenishment Policy	Order-up-to Level	Maximum Level	Zero Inventory Ordering
Shipping Strategy	Direct Shipping	Multiple Shipping	
Inventory Decision	Back-order	Lost Sales	Non-negative
Number of Products	One	Multiple	
Replenishment Strategy	Periodic(Static)	Cyclic	
Structure of Delivery	One-to-one	One-to-Many	Many-to-Many
Fleet Size	Single	Multiple	Unconstrained
Fleet Characteristic	Homogenous	Heterogeneous	
Compartments Characteristic	No Compartments	Homogenous Compartments	Heterogeneous Compartments

Without loss of generality, we prefer not to add an objective function as criteria because flexibility of problems vary related to strategic decision process. Generally, objective function is the minimization of vehicle routing and inventory costs. Vehicle routing cost can be categorized as follow: (i) transportation cost when traversing each edge [15, 16, 17] (ii) a fixed cost per stop at a customer location [18, 19] (iii) a fixed cost accrued when each vehicle is dispatched [20, 21, 22], and (iv) a fixed cost of fleet size as decision variable either use or not [23]. The inventory costs can be categorized as follow: (i) holding cost at each customer location [24, 25, 26, 27], (ii) shortage cost (negative inventory cost) [6, 28] and finally (iii) ordering cost prefering external sources as producing instead of in house production [22, 29].

In the Table 1, demand is the major criteria effecting the IRP in terms of complexity. The demand can be either deterministic [16, 25, 29, 30] or stochastic [28, 31, 32].

Most of the paper in OR worked on deterministic in a way the demand in each period is satisfied instantly. In addition to that some researchers assumed the demand is realized on a continuous time basis.

Another major criteria for the IRP is planning horizon which changes problem structure eventually very much. The planning horizon can be single period [6, 33, 34], multi-period [15, 16, 25, 35] and infinite time [5, 20, 24, 28, 29, 30]. Generally the IRP is long term problem which deals with effects of short-term planning.

Replenishment strategy is defined as pre-established rules to replenish customers. There are several replenishment policies but most of them categorized as Order-up-to-Level(OU), Maximum Level(ML) and Zero Inventory Ordering (ZIO). OU is the policy where customer's inventory raised to its maximum level whenever each delivery accrued. Therefore, the quantity to be delivered to customer is the difference between its maximum capacity and its current inventory level [15, 25, 36]. Conversely to OU policy, in the maximum-level (ML) policy, any quantity could be delivered as much as the maximum level determined by the customer which is not exceeded [16, 19, 37, 38, 39, 40]. Moreover in ZIO policy, in each period, only the set of customers whose inventory drop at zero is replenished [41, 42]. In addition to that Fixed Partition Policy (FPP) is widely used. In the FPP policy, the set of customers is partitioned into a number of clusters such that each cluster replenishes independently. Delivered amount could be restricted either OU or ML policy [3, 20, 29, 43, 44]. Direct shipping refers to each customer visited by only one tour or vehicle in a way of repeated certain frequency [19, 20, 22, 32]. On the other hand multiple shipping refers to each customer can be visited sequentially in a tour. In addition to that some cases customer can be visited by multiple tours which means splitting the delivery to customer among different tours. Therefore, multiple shipping provides better solution to direct shipping because of flexibility; however it has more complexity than direct shipping problem [15, 16, 19, 22, 28].

Inventory decision is other pre-defined structure of problem. If the inventory can be negative, then back-ordering occurs which means corresponding demand will be procured when new shipments are delivered [16, 31, 45, 46, 47, 48, 49, 50]. In case negative inventory under no back-ordering, then considering demand will be considered as lost sales [18, 19]. In both situation there may be a penalty for the stock-out. Non allowing negative inventory ensure no-stock outs [25, 50, 51, 52]. The number of product is other criteria of the IRP. Most studies work on one product [16, 25, 29, 36], however quite few study work on multi-product settings as well [18, 30, 53, 54].
The structure of delivery is crucial settings for IRP and can be categorized as two parts:(i) periodic (static) policy whose replenishment and delivery schedules are repeated same pattern for all periods [19, 40, 41, 55, 56] and (ii) cyclic policies where the proposed delivery and replenishment schedules can change from periods to periods. Therefore cyclic policy are more flexible and ensures more cost effective solution comparing to static policy [15, 16, 36, 37, 38, 51, 57, 58, 59, 60].

Also the customer, supplier numbers can be varied. Therefore, desired distribution process could be one-to-one which there is only one customer and one supplier, one-to-many where one supplier is responsible for all customers, finally many-to-many [50, 61] with several suppliers serve to several customers. The fleet can be heterogeneous and homogenous whose number could be defined as single, multiple and unconstrained. Moreover, the vehicle could have compartments in such that homogenous and heteregenous [62]. In the IRP settings compartment structure is used as capacity constraints. As far as our knowledge there is no study which ensures to schedule compartment allocation problem. Main study settings for single product deterministic IRP can be found in Table 2.

Table 2: Classification of main studies on single product deterministic IRP

Study	***Planning Horizon***	***Replenishment Policy***	***Shipping Strategy***	***Inventory Decision***	***Replenishment Strategy***	***Delivery Structure***	***Fleet***	***Fleet Size***	***Compartment***
Bertazzi et al. [2002]	Finite	OU	Multiple	Non-negative	Cyclic	1-M	Hom.	Sing.	No Comp
Adelman [2003]	Infinite	ML	Multiple	Lost sales	Cyclic	1-M	Hom.	Mult.	No Comp
Campbell and Savelsbergh [2004]	Finite	ML	Multiple	Non-negative	Cyclic	1-M	Hom.	Mult.	No Comp
Gaur and Ficher [2004]	Finite	ML	Multiple	Non-negative	Periodic	1-M	Het.	Mult.	No Comp
Aghezzaf et al. [2006]	Infinite	ML	Multiple	Lost sales	Cyclic	1-M	Hom.	Mult.	No Comp
Archetti et al. [2007]	Finite	OU	Multiple	Non-negative	Cyclic	1-M	Hom.	Sing.	No Comp
Raa and Aghezaaf [2008]	Finite	ML	Multiple	Non-negative	Periodic	1-M	Hom.	Sing.	No Comp
Bertazzi [2008]	Finite	ML	Direct	Non-negative	Cyclic	1-1	Hom.	Mult.	No Comp
Savelsbergh and Song [2008]	Finite	ML	Multiple	Lost sales	Cyclic	M-M	Hom.	Mult.	No Comp
Raa and Aghezaaf [2009]	Infinite	ML	Multiple	Non-negative	Periodic	1-M	Hom.	Mult.	No Comp
Abdelmaguid et al. [2009]	Finite	ML	Multiple	Back-order	Cyclic	1-M	Het.	Mult.	No Comp
Bard and Nananukul [2010]	Finite	ML	Multiple	Non-negative	Cyclic	1-M	Hom.	Mult.	No Comp
Solyali and Sral [2011]	Finite	OU	Multiple	Non-negative	Cyclic	1-M	Hom.	Sing.	No Comp
Coelho et al. [2012]	Finite	OU- ML	Multiple	Non-negative	Cyclic	1-M	Hom.	Sing.	No Comp
Coelho et al. [2012]	Finite	OU	Multiple	Non-negative	Cyclic	1-M	Hom.	Mult.	No Comp
Archetti et al. [2012]	Finite	ML	Multiple	Non-negative	Cyclic	1-M	Hom.	Sing.	No Comp
Aksen et al. 2012	Finite	OU	Multiple	Non-negative	Periodic	1-M	Hom.	Sing.	No Comp
Coelho and Laporte [2013]	Finite	ML	Multiple	Non-negative	Cyclic	1-M	Hom.	Mult.	No Comp
Coelho and Laporte [2014]	Finite	ML	Multiple	Non-negative	Cyclic	1-M	Hom.	Mult.	No Comp
Ekici et al. [2015]	Finite	ML	Multiple	Non-negative	Periodic	1-M	Hom.	Mult.	No Comp
Desaulniers et al. [2016]	Finite	ML	Multiple	Non-negative	Cyclic	1-M	Hom.	Mult.	No Comp
Raa, B. [2016]	Finite	OU	Multiple	Non-negative	Periodic	1-M	Hom.	Mult.	No Comp

Routing and inventory loading problem for heterogeneous vehicle fleet with multi compartments (RILPHVFC) we studied in this thesis. In our problem settings, the objective is to find a static replenishment, delivery and loading scheduling which minimize total logistic cost without stock-out at the customers. The demand or consumption rate at each customer is continuous and deterministic. We solved our problem in one period however thanks to static period policy we can repeat one period structure as long as we want. Replenishment strategy is ZIO under restriction of instant delivery. One depot which has an unlimited supply of product serving un-capacitated customers with a single product. Each customer is served by only one vehicle vice versa in one tour. We combined ZIO policy with FPP where the set of customers is partitioned into a number of clusters such that each cluster is served independently. We assume that there are multiple hetoregeneous vehicles which has heteregeneous compartments which can be used or not in the scheduling. RILPHVFC criterias related to IRP can be summarized as in Table 3 follows:

Table 3: Classification of RILPHVFC

Criteria	Features
Demand	Deterministic
Planning Horizon	Single Period
Replenishment Policy	Zero Inventory Ordering
Shipping Strategy	Multiple Shipping
Inventory Decision	Non-negative
Number of Products	One
Replenishment Strategy	Periodic(Static)
Delivery Structure	One-to-Many
Fleet Size	Multiple
Fleet Characteristic	Heterogeneous
Compartments Characteristic	Heterogeneous Compartments

As far as our knowledge, RILPHVFC has not been studied before. In the OR literature, there are several solution approach to solve IRP with deterministic demand for single product which are stated as Table 4. Moreover, most proposed algorithms decompose the problem into two phase (i) vehicle routing and (ii) inventory control.

We are inspired to construct our methodology from the work [55]. In their work, they divide customers into clusters. Therefore, each of cluster is assigned to a vehicle. Each vehicle serves only a single cluster of customer. In addition to that, they propose three cluster algorithm one of the Iterative Random Subset Clustering . We use this cluster technique in our two-phase strategy in order to reach better clustering. In their Second Phase, they develop algorithms to generate feasible delivery, which are also cost efficient for each cluster. Using three algorithm which are based on constructive heuristic, integer programming and network flow formulation finds solution to obtain the final set of delivery routes and volumes according to demand.

The other similar approach to use two-phase methodology in the work [16]. In the first phase, they solve an integer programming to determine the customers to be served on each day of planning horizon. Thanks to FPP, they can reduce number of routes. Moreover, they applied set partitioning problem to select clusters and embodied with insertion heuristic to solve the vehicle routing problem. Finally, they incorporate Greedy Random Adaptive Search Procedure (GRASP) to generate more solutions and choose the best one.

The last but not the least, it is worth to mention here the work of [63]. The problem is studied on mainly VRP. Which part distinguishes them from others is heterogeneous vehicle fleet and multi product. Moreover, heterogeneous vehicle fleet have compartment structure. They also have to consider assignment of compartments to products. They propose Reactive Tabu Search Algorithm (RTS), which is well known metaheuristic. Firstly, they develop constructive heuristic and the improvement based on neighborhood structure or we can say that moves. In addition to that all moves has guiding mechanism which penalize long distance in routes. In the neighborhood generation neighborhood reduction strategy is applied to enhance computational efficiency. Finally, they also make Tabu Search armed with reactive mechanism and local search to reach more solution space in order to satisfy diversification [64].

Put in the nutshell, we also implement Tabu Search meta-heuristic to compare with two-phase approach according to our problem settings in order to evaluate performance in absolute sense.

Table 4: Solution methodologies of main studies on single product deterministic IRP

Study	***Solution Methodology***
Bertazzi et al. [2002]	Two-phase Decomposition
Adelman [2003]	Dynamic Programming, Semi-Markov Decision Process
Campbell and Savelsbergh [2004]	Route-based MIP formulation; Limited number of routes; Two-Phase Decomposition Algorithm
Gaur and Ficher [2004]	Fixed Partition Policy; Weighted Matching; Heuristic
Aghezzaf et al. [2006]	Column Generation Based Heuristic
Archetti et al. [2007]	Branch-and-cut Algorithm
Raa and Aghezaaf [2008]	Column Generation for Distribution Patterns
Bertazzi [2008]	Worst-case Analysis
Savelsbergh and Song [2008]	Reduction and Separation Heuristic, MILP
Raa and Aghezaaf [2009]	Heuristic; Column Generation Algorithm
Abdelmaguid et al. [2009]	MILP, Lagrangian Relaxation and Benders' Decomposition
Bard and Nananukul [2010]	Branch-and-price Algorithm
Solyali and Sral [2011]	Branch-and-cut Algorithm, Strong Formulation, Priori-based Tour Heuristic
Coelho et al. [2012]	ALNS
Coelho et al. [2012]	MILP Formulation; Matheuristic; ALNS
Archetti et al. [2012]	Tabu Search, Local Improvement Heuristic
Aksen et al. [2012]	MILP Formulation
Coelho and Laporte [2013]	Branch-and-cut Algorithm
Coelho and Laporte [2014]	Valid Inequalities; Branch-and-cut Algorithm
Ekici et al. [2015]	Fixed Partition Policy, Iterative Clustering; Two-phase Decomposition
Desaulniers et al. [2016]	Branch-and-price-and-cut Algorithm
Raa, B. [2016]	Multi Start Heuristic, Slot Selection Process

Chapter III

PROBLEM DESCRIPTION

The problem we studied in this thesis is such an extension of basic inventory model. Even if growing importance on proper management of replenishment process is increased, our problem has not received much attention so far. We consider distribution system, which has one depot and many geographically dispersed customers. Moreover each of customers order demands at constant, deterministic however specific rates. In a practical manner these customer could be retailers of franchising company, which face external demands. We consider a set of commodities each having its own demand per period. These commodities or products in terms of physical volume must be transported from depot to a customer via heterogeneous vehicle fleet with compartments. The compartment structure of vehicle fleet differs from each other. Distribution process for this problem is illustrated in Figure 1.

The fundamental restriction we impose that no two demands (products) may be transported within the same compartment and customer may not be served by no more than one vehicle. Allowing mixing products in a compartment may cause hazardous reaction or as it may ruin individuals products. We wish to determine feasible replenishment strategies both minimizing replenishment strategy and routing patterns. Moreover we wish to find the type of products to which each vehicles compartments assigned and the amount of that product loaded into specific vehicle compartments regarding to route pattern. Successively with determining such an assignment, we have to determine both tour length and replenishment time (or frequency in a unit time) strictly ensure that no shortage are encountered at the customer site. Without loss of generality, we assume the holding cost are only incurred at the customers' sites therefore we did not take into account holding cost.

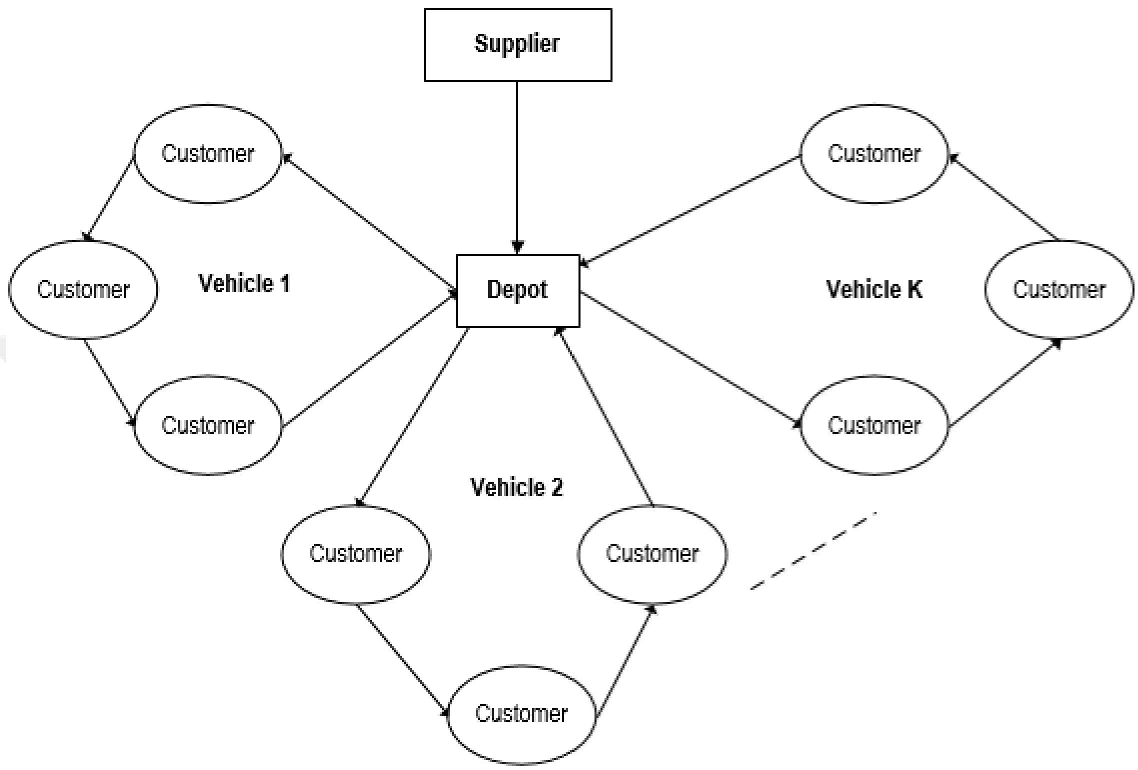


Figure 1: Distribution Process for Heterogenous Vehicle Fleet

In this study, we examined static policy is one which a vehicle fleet uses a delivery every unit of time which can be a day, week or a proper fixed time duration. Thereon we designate this unit of time as period. In addition to that, the vehicle fleet use the same loading and route pattern for all periods. Consequently the objective is to determine the best loading policy that satisfied product demands in every periods for all in use vehicle by using minimum setup cost which is multiplication of route cost and number of deliveries per unit time, for all trucks.

In this part, we present a formal definition of RILPHVFC under consideration. We described the locations who are customers demanding only one product which is different product than the others. Our problem is defined on a complete undirected network graph. An undirected graph $G = \{V, E\}$ be given which consists of a vertex set $V = \{0, 1, ..., N\}$ including one depot (node 0) and set N customers, and an edge set $E = \{(i, j) : i, j \in V\}$, representing the edges which can be travelled between the different locations. Each edges (i, j) has a cost $(c_{ij} = c_{ji}) \in E$ and the set of costs satisfies the triangle inequality. Each customer has a demand of $d_i : i \in V\backslash\{0\}$ units of volume per unit time. We assume that K heterogenous vehicles with different compartment capacity structures. In addition to that q_{mk} is defined for capacity of compartment m of vehicle k. Deliveries to the customers are made via routes that start and end depot as illustrated in Figure 1. As we explained in beginning of this section all trucks use static policy in which each customer is visited by only one vehicle in each period.

Forgoing part, we present an example in order to explain how to minimize logistic cost per unit time for each vehicle regarding to tour length and replenishment time. To illustrate this concept firstly we explain how to calculate frequency. Frequency is calculated minimize the maximum customers' demand over assigned capacity to this demand within all possible assignment. For example let two customers with demand $(10, 20)$ and one vehicle with two compartment capacity $(20, 5)$. Optimal assignment for this example will be first product to second compartment, second product to first compartment. Therefore vehicle frequency is calculated minimum of maximum frequency among the assignments. In our case $min((max(10/5 = 2; 20/20 = 1), (max(20/5 = 4; 10/20 = 1/2))$ is 2 which means this truck has to deliver two times in unit time in order to obtain no shortage at customers' sites.

Note that the optimal frequency need not necessarily be an optimal because this is only optimal in static policy. We stated delivery policies such as static and cyclic. Static policy is repeated in every period therefore cyclic policy leads better result in terms of objective value. For the illustration, consider the example, we have two customers with demands $(3, 1)$ and two compartments such as $(3, 3)$. An optimal frequency solution would obviously repeat every period, delivering three units of customer-1's demand in compartment-1 and one unit of customer-2's demand in compartment-2. However, in cyclic policy, the optimal solution would deliver three unit of customer-2's demand in compartment-1 and three units of customer-1's demand in compartment-2 during in the first period. For the second period, the configuration would change; in the next two periods no more customer-2's demand needs to be transported. In addition, in period 3, no delivery must be made before the delivery pattern repeats. In static policy, we schedule total three deliveries; in the cyclic policy, we schedule two deliveries. Therefore static policy frequency is 1 versus cyclic policy frequency is 2/3.

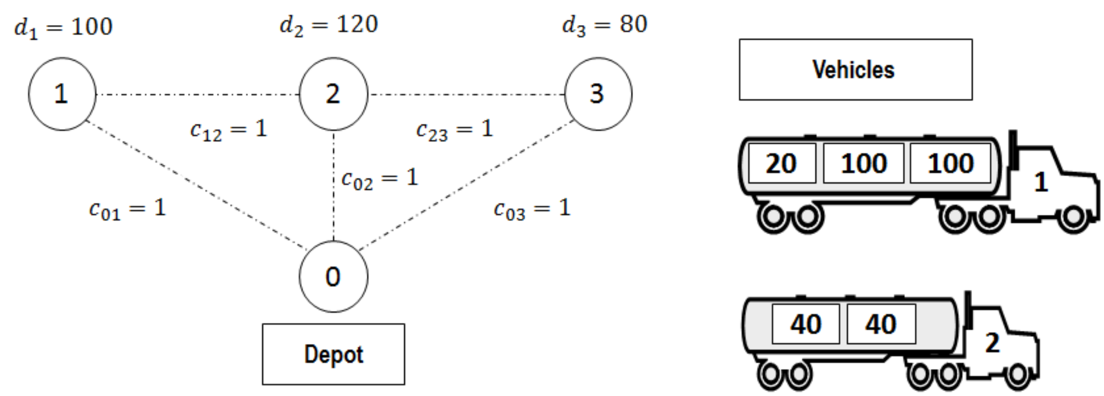


Figure 2: Simple Illustration Example

To calculate logistic cost we introduce frequency f_k for each vehicle represents numbers of deliveries per unit time. By all means, logistic cost is multiplication of frequency and route cost for each vehicle. Therefore, in the example provided in Figure 2 we want to explain how we calculate logistic cost for our problem. In this example we have three customers and two vehicles. If we generate all solutions according to our problem settings, there are only eight scenarios for our problem which are presented in the Table 5 and Table 6. For the sake of simplicity, we demonstrated optimal frequencies instead of showing all assignment among the compartments to demands assignments. In the Table 7 , the optimal solution is 1 and 2 customers' demand

Table 5: Scenarios for Vehicle-1

Vehicle 1					
Scenario #	***Products***	***Solution Representation***	***Route Length***	***Frequency***	***Cost***
#1	(1,2,3)	(3-2-1)	4	Max (100/100, 120/100,80/20)=4	4*4=16
#2	(1-2)	(2-2-1)	3	Max(120 / (100+20), 100 /100)=1	3*1=3
#3	(2-3)	(2-2-3)	3	Max(120 / (100+20), 80 /100)=1	3*1=3
#4	(1-3)	(1-1-3)	3	Max(100 / (100+20), 80 /100)=5/6	3*5/6= 15/6
#5	(1)	(1-1-1)	2	Max(100/220) =5/11	2*5/11=10/11
#6	(2)	(2-2-2)	2	Max(120/220) = 6/11	2*6/11=12/11
#7	(3)	(3-3-3)	2	Max(80 / 220)=4/11	2*4/11=8/11
#8	no assgn.	infeasible	infeasible	infeasible	infeasible

Table 6: Scenarios for Vehicle-2

Vehicle 2					
Scenario #	***Products***	***Solution Representation***	***Route Length***	***Frequency***	***Cost***
#1	no assgn.	-	0	0	0
#2	(3)	(3-3)	2	Max(80/80)=1	2*1=2
#3	(1)	(1-1)	2	Max(100/80)=5/4	2*5/4=10/4
#4	(2)	(2-2)	2	Max(120 / 80))= 6/4	2*6/4=3
#5	(2-3)	(2-3)	3	Max(100/40,80/40) =5/2	3*5/2=15/2
#6	(1-3)	(1-3)	3	Max(120/40,80/40)=3	3*3=9
#7	(1-2)	(1-2)	3	Max(120/40, 100/ 40)= 3	3*3=9
#8	(1,2,3)	infeasible	infeasible	infeasible	infeasible

assigned to vehicle 1, 3 customer's demand assigned to vehicle 2 with objective cost 5 per unit time. Solution representations for compartments are (2-2-1) for vehicle-1 and (3-3) for vehicle-2.

Table 7: Objective Costs for All Scenarios

All Results			
Scenario #	***Vehicle 1 Cost***	***Vehicle 2 Cost***	***Objective***
#1	16.00	0	16.00
#2	3.00	2	5.00
#3	3.00	2.25	5.25
#4	2.50	3	5.50
#5	0.91	7.5	8.41
#6	1.09	9	10.09
#7	0.73	9	9.73
#8	infeasible	infeasible	infeasible

3.1 *Multi-Product Loading Problem with Frequency*

Logistic cost of vehicle contains frequency within itself, therefore firstly we demonstrate Multi Product Loading with Frequency (MILPF) which is fundemantal part of mathematical formulation in order to obtain frequencies of vehicles. Supposing that one vehicle which has m compartments and serve n different customers who demand only one product. It is not allowed to be mixed two products into same compartment. The objective is maximizing the minimum replenishment time. The mathematical formulation is as follows:

Parameters:

M: Set of compartments,

N: Set of customers,

d_i: Demand rate for customer $i \in N$,

q_m: Capacity of compartment $m \in M$,

Decision Variables:

x_{im}: 1, if customer demands i is assigned to compartment m ; otherwise 0 : $i \in N$, $m \in M$,

T: replenishment time for static replenishment strategy

$$\textbf{Max} \quad T \tag{1}$$

s.t.

$$\frac{\sum_{m \in M} x_{im} q_m}{d_i} \geq T \quad \forall i \in N \tag{2}$$

$$\sum_{i \in N} x_{im} = 1 \quad \forall m \in M \tag{3}$$

$$x_{im} \in \{0, 1\} \quad \forall m \in M, \forall i \in N \tag{4}$$

$$T \geq 0 \tag{5}$$

The objective (1) is clearly to maximize common replenishment time in static policy. Constraints (2) require that ample compartment space to be devoted to each demand. Constraints (3) require that each compartment is dedicated to exactly one product. Finally, Constraints (4) and (5) dictate the structures and sign restrictions of the decision variables. This problem is called as Multi Product Loading Problem (MPLP) and it is firstly studied in the work which proposed Lagrangian Relaxation to solve Maxmin problem [65].

Since our main concern is frequency which is multiplicative inverse of common replenishment time T. We have to transform above Maxmin problem to Minmax problem which is called Multi Product Loading Problem with Frequency (MPLPF). For the sake of simplicity, we prefer to describe new decision variables and parameters in here to avoid from repeating previous existing definitions. Mathematical formulation of MPLPF is as follows:

New Parameter:

B: Sufficiently big number to enforce some constraints non-effective.

New Decision Variables:

f: frequency in static period

u_{im}: auxillary variable for linearization of $\mathbf{x_{im}} * \mathbf{f}$; $\forall i \in N$, $\forall m \in M$

$$\textbf{Min} \quad f \tag{6}$$

s.t.

$$d_i \leq \sum_{m \in M} u_{im} q_m \qquad \forall i \in N \tag{7}$$

$$\sum_{i \in N} x_{im} = 1 \qquad \forall m \in M \tag{8}$$

$$u_{im} \leq f \qquad \forall m \in M, \forall i \in N \tag{9}$$

$$u_{im} \leq B(x_{im}) \qquad \forall m \in M, \forall i \in N \tag{10}$$

$$u_{im} \geq f - B(1 - x_{im}) \qquad \forall m \in M, \forall i \in N \tag{11}$$

$$x_{im} \in \{0, 1\} \qquad \forall m \in M, \forall i \in N \tag{12}$$

$$f \geq 0 \quad ; \quad u_{im} \geq 0 \qquad \forall m \in M, \forall i \in N \tag{13}$$

The objective (6) is clearly to minimize common frequency or number of deliveries per unit period time in static policy. Constraints (7) require that ample compartment space to be devoted to each demand. Constraints (8) require that each compartment is dedicated to exactly one product. Constraints (9),(10) and (11) ensure linearity in $x_{im} * f$. Finally, Constraints (12) and (13) dictate the structures and sign restrictions of the decision variables.

Theorem 1. *MPLPF is strongly NP-Hard problem.*

Proof. To prove NP-Hardness, we must first define a decision problem (D-MLPLF) related to MPLPF and prove that is NP-Complete. We demonstrate that there exists a strongly NP-complete problem that polynomially reduces to D-MLPF from The 3-Partition problem exclusively [66]. Given a multiset S of $n = 3m$ positive integers $(a_1, a_2, ..., a_{3m})$ such that $P/4 \leq a_j \leq P/2$ for $j = 1, ..., 3m$ and such that $a_1 + a_2 + a_3 + ... + a_{3m} = mP$. The problem is to decide whether a given multiset S of integers can be partitioned into triples that all have the same sum. Can S be partitioned into m subset $S_1, S_2, ..., S_m$ such that the sum of the numbers in each subset is equal? In this case, each subset S_i is forced to consist of exactly three elements and sum of the numbers in each subset equals P. From an instance of 3-partition under the conditions above, we generate an instance of our problem as follows. Assuming that we have 3m compartments with size integers $(a_1, a_2, ..., a_{3m})$. The sum of all these numbers is mP. In addition to that we have m customers with same demand rates (d). The feasibility question of whether there is an assignment of the comopartments to items such that there is a feasible solution with d/P is equivalent to finding a solution to the corresponding 3-Partition Problem. To sum up, Decision Problem (D-MLPF) is strongly NP-Complete since 3-Partition Problem is strongly NP-Complete. □

3.2 Mathematical Model of RILPHVFC

In this section, we present mathematical model for the RILPHVFC. As it was explained in the beginning of this chapter, the distribution system has a fleet of heterogeneous vehicles having heterogenous compartments where a compartment can carry and dedicated only one customer's demand. Our objective is to minimize overall logistic cost according to problem settings in Table 3. Mathematical model of RILPHVFC is presented as below:

Parameters:

K: Set of vehicles,

M_k: Set of compartments in vehicle k; $k \in K$,

d_i: Demand rate for customer i; $i \in V \setminus \{0\}$,

q_{mk}: Capacity of compartment m of vehicle k; $m \in M_k$, $k \in K$,

c_{ij}: Non-negative distance between customer i and j; $i \in V$, $j \in V$,

B: Sufficiently big number to enforce some constraints non-effective.

Decision Variables:

x_{ikm}: 1, if customer i is assigned to compartment m of vehicle k; otherwise 0 : $i \in V \setminus \{0\}$, $m \in M_k$,$k \in K$.

y_{ik}: 1, if demand of customer i is delivered by vehicle k; otherwise 0 : $i \in V$, $k \in K$.

w_{ijk}: 1,if edge (i, j) is used once by vehicle k; otherwise 0 : $i, j \in V$, $k \in K$.

f_k: Frequency of vehicle k in static period: $k \in K$.

u_i: Auxiliary variable defined for customers i in order to eliminate subtours : $i \in V \setminus \{0\}$.

$$\text{Min} \quad \sum_{k\in K}\sum_{i\in V}\sum_{j\in V}(w_{ijk}c_{ij}f_k) \tag{14}$$

s.t.

$$\sum_{k\in K} y_{ik} = 1 \qquad \forall i \in V\backslash\{0\} \tag{15}$$

$$x_{ikm} \leq y_{ik} \qquad \forall i \in V\backslash\{0\}, \forall m \in M_k, \forall k \in K \tag{16}$$

$$\sum_{m\in M_k} x_{ikm} \geq y_{ik} \qquad \forall i \in V\backslash\{0\}, \forall k \in K \tag{17}$$

$$\sum_{i\in V\backslash\{0\}} x_{ikm} \leq 1 \qquad \forall m \in M_k, \forall k \in K \tag{18}$$

$$\sum_{k\in K}\sum_{m\in M_k} x_{ikm} \geq 1 \qquad \forall i \in V\backslash\{0\} \tag{19}$$

$$B(1-y_{ik}) + \frac{\sum_{m\in M_k} x_{ikm}q_{mk}}{d_i} f_k \geq 1 \qquad \forall i \in V\backslash\{0\}, \forall k \in K \tag{20}$$

$$\sum_{i\in V}\sum_{k\in K} w_{ijk} = 1 \qquad \forall j \in V\backslash\{0\} \tag{21}$$

$$\sum_{i\in V}\sum_{k\in K} w_{jik} = 1 \qquad \forall j \in V\backslash\{0\} \tag{22}$$

$$\sum_{j\in V} w_{ijk} = \sum_{j\in V} w_{jik} \qquad \forall i \in V, \forall k \in K \tag{23}$$

$$\sum_{j\in\backslash\{0\}} w_{0jk} \leq 1 \qquad \forall k \in K \tag{24}$$

$$1 \leq u_i \leq |V| \qquad \forall i \in V\backslash\{0\} \tag{25}$$

$$u_i - u_j + |V|\sum_{k\in K} w_{ijk} \leq |V| - 1 \qquad \forall i,j \in V\backslash\{0\} \tag{26}$$

$$w_{ijk} \leq y_{ik} \qquad \forall i \in V\backslash\{0\}, \forall k \in K \tag{27}$$

$$x_{ikm} \in \{0,1\} \qquad \forall i \in V\backslash\{0\}, \forall k \in K, \forall m \in M_k \tag{28}$$

$$y_{ik} \in \{0,1\} \qquad \forall i \in V\backslash\{0\}, \forall k \in K \tag{29}$$

$$w_{ijk} \in \{0,1\} \qquad \forall i,j \in V, \forall k \in K \tag{30}$$

$$f_k \geq 0 \qquad \forall k \in K \tag{31}$$

$$u_i \geq 0 \qquad \forall i \in V\backslash\{0\} \tag{32}$$

Furthermore, (14) and (20) equations making our problem as non-linear problem. Nonetheless (14) and (20) includes only multiplication of binary and continuous decision variables which can be transform into linear representation. Therefore we introduce two new decision variables in order to obtain MIP formulation as follow:

$$s_{ijk} = \text{auxillary variable for}(w_{ijk} * f_k) \qquad \forall i, j \in V, \forall k \in K \tag{33}$$

$$t_{ikm} = \text{auxillary variable for}(x_{ikm} * f_k) \quad \forall i \in V \setminus \{0\}, \forall m \in M_k, \forall k \in K \tag{34}$$

Implementing two auxillary (33) and (34) decision variables we can formulate Mixed Integer Problem (MIP) Formulation as follow:

$$\textbf{Min} \qquad \sum_{k \in K} \sum_{i \in V} \sum_{j \in V} (s_{ijk} c_{ij}) \tag{35}$$

s.t.

$$\sum_{k \in K} y_{ik} = 1 \qquad \forall i \in V \setminus \{0\} \tag{36}$$

$$x_{ikm} \leq y_{ik} \qquad \forall i \in V \setminus \{0\}, \forall m \in M_k, \forall k \in K \tag{37}$$

$$\sum_{m \in M_k} x_{ikm} \geq y_{ik} \qquad \forall i \in V \setminus \{0\}, \forall k \in K \tag{38}$$

$$\sum_{i \in V \setminus \{0\}} x_{ikm} \leq 1 \qquad \forall m \in M_k, \forall k \in K \tag{39}$$

$$\sum_{k \in K} \sum_{m \in M_k} x_{ikm} \geq 1 \qquad \forall i \in V \setminus \{0\} \tag{40}$$

$$B(1 - y_{ik}) + \sum_{m \in M_k} t_{ikm} q_{mk} \geq d_i \quad \forall i \in V \setminus \{0\}, \forall k \in K \tag{41}$$

$$\sum_{i\in V}\sum_{k\in K} w_{ijk} = 1 \qquad \forall j \in V\backslash\{0\} \tag{42}$$

$$\sum_{i\in V}\sum_{k\in K} w_{jik} = 1 \qquad \forall j \in V\backslash\{0\} \tag{43}$$

$$\sum_{j\in V} w_{ijk} = \sum_{j\in V} w_{jik} \qquad \forall i \in V, \forall k \in K \tag{44}$$

$$\sum_{j\in\backslash\{0\}} w_{0jk} \leq 1 \qquad \forall k \in K \tag{45}$$

$$1 \leq u_i \leq |V| \qquad \forall i \in V\backslash\{0\} \tag{46}$$

$$u_i - u_j + |V|\sum_{k\in K} w_{ijk} \leq |V| - 1 \qquad \forall i,j \in V\backslash\{0\} \tag{47}$$

$$w_{ijk} \leq y_{ik} \qquad \forall i \in V\backslash\{0\}, \forall k \in K \tag{48}$$

$$t_{ikm} \leq f_k \qquad \forall i \in V\backslash\{0\}, \forall m \in M_k, \forall k \in K \tag{49}$$

$$t_{ikm} \leq Bx_{ikm} \qquad \forall i \in V\backslash\{0\}, \forall m \in M_k, \forall k \in K \tag{50}$$

$$t_{ikm} \geq f_k - B(1 - x_{ikm}) \qquad \forall i \in V\backslash\{0\}, \forall m \in M_k, \forall k \in K \tag{51}$$

$$s_{ijk} \leq f_k \qquad \forall i,j \in V, \forall k \in K \tag{52}$$

$$s_{ijk} \leq Bw_{ijk} \qquad \forall i,j \in V, \forall k \in K \tag{53}$$

$$s_{ijk} \geq f_k - B(1 - w_{ijk}) \qquad \forall i,j \in V, \forall k \in K \tag{54}$$

$$x_{ikm} \in \{0,1\}, y_{ik} \in \{0,1\} \qquad \forall i \in V\backslash\{0\}, \forall k \in K, \forall m \in M_k \tag{55}$$

$$w_{ijk} \in \{0,1\} \qquad \forall i,j \in V, \forall k \in K \tag{56}$$

$$f_k \geq 0, u_i \geq 0 \qquad \forall k \in K, \forall i \in V\backslash\{0\} \tag{57}$$

$$t_{ijm} \geq 0 \qquad \forall i \in V\backslash\{0\}, \forall m \in M_k, \forall k \in K \tag{58}$$

$$s_{ijk} \geq 0 \qquad \forall i,j \in V, \forall k \in K \tag{59}$$

In the MIP model of RILPHVFC, the objective function (35) minimizes total logistic costs which are the multiplication of transportation cost and frequency (the number of deliveries per unit time) for each vehicle. Constraint set (36) ensures each customer is served by exactly one vehicle. Constraint set (37) ensures that assignment of customer's products to compartments in which the vehicle serves that customer. Constraint set (38) forces each customers assigned to vehicle must be assigned to at least one compartment of that vehicle. Constraint set (39) ensures no two demands mixing in same compartment. Constraint set (40) procures each customers demand must be served by at least one compartment. Constraint set (41) enforces minimize maximum frequency for each vehicle. Constraint sets (42) and (43) ensure that number of entrance and departure for each node equals exactly one. Constraint set (44) guarantees continuity in routes. Constraint set (45) makes sure that there can be only one assignment of edge between depot and customer to vehicle. Constraint sets (46) and (47) enforce sub-tour eliminations in routes. Constraint set (48) ensures that customer's products in the vehicle must be in the vehicles' route. Constraint sets (49), (50) and (51) linearize multiplication of x_{ikm} and f_k. Constraint sets (52), (53) and (54) linearize multiplication of w_{ijk} and f_k. Finally, constraint sets (55), (56), (57), (58), (59) dictate the structures and sign restrictions of the decision variables.

Chapter IV

METHODOLOGY

In this chapter, we construct solution framework for RILPHVFC. Initially we demonstrate single vehicle version of the problem which is fundamental to our problem.

4.1 Routing and Inventory Loading Problem in Single Vehicle with Compartments

Assuming that only one vehicle serves geographically dispersed customers under the same restriction in the Chapter 3. The vehicle has numbers of compartments which is greater than numbers of customers which ensures feasibility. RILPHVFC's Mathematical Formulation can be used for this problem. Moreover, there can be some formulation technique regarding to single vehicle version which is tightened and shorten. For the sake of simplicity we are avoiding specific formulation of this problem in here. We refer the reader this specific formulation on Appendix A.1. Moreover, the problem we studied the vehicle number and compartments structures is input of the problem and we continue with same formulation.

Observation 1. *Routing and Inventory for Single Vehicle with Compartments (RILPSVC) can decompose two sub-problem which are Travelling Salesman Problem (TSP) and Multi-Product Loading Problem with Frequency (MPLPF).*

The logistic cost is multiplication of frequency and routing cost. Therefore if only one vehicle serves all customers, routing cost is obviously TSP tour and frequency is obviously obtained from MPLPF. Therefore if we solve simultaneously solve two problem then multiplying two objective leads original problem's objective. The RILPSVC is illustrated in Figure 3.

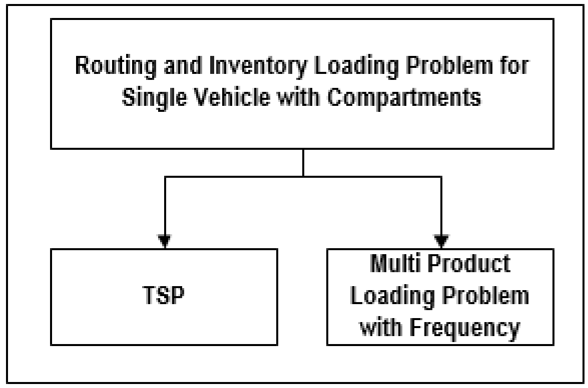


Figure 3: Two Subproblems of RILPSVC

In Industrial Engineering Literature, TSP is the most famous problem which is NP-Hard problem. We do not want to occupy place for showing formulation. Besides, we use Miller-Tucker-Zemlin (MTZ) formulation to handle sub-tour elimination which performs faster than constructing all subsets [67].

4.1.1 Solution Approach for Routing and Loading Problem for Single Vehicle with Compartments

As we stated in the Section 4.1, single vehicle version of the problem could be solved dividing problem into two sub problem and multiplying these two subproblem's objective value as an objective value. Even single version problem could be solved by two separately mathematical problem for small size problems in reasonable time, it could not solved for large scale problem because of the nature of the NP-Hardness. Therefore, we prefer to propose heuristic in case of larger problem to ensure computational efficiency. In the heuristic we proposed has two parts as routing part and loading part as follow:

4.1.1.1 Routing Problem

In the routing part, we use Christofides Algorithm to get TSP tour. We choose Christofides Algorithm which has the best approximation ratio that has been proven for the TSP on general metric space. The Pseudocode of the Christofides Algorithm is represented in Algorithm 1. In addition to that, we empowered Christofides Algorithm with improvement moves such as 1-0 insertion, 1-1 switch and 2-Opt after short cutting phase in order to get more desirable route.

Algorithm 1 *Pseudocode of Christofides Algorithm*

Input: RILPSVC instance on a map of $D \times D$ unit square geographically dispersed locations and depot.
Output: TSP tour which starts from depot and returns to depot by visiting all customer only once
Calculate Minimum Spanning Tree T
Define the set of vertices O with odd degree in T
Construct the subgraph of G using only the odd vertices of O
Solve a minimum-weight perfect matching M in this subgraph
Unite matching and spanning tree $M \cup T$ to construct an Eulerian multigraph E
Calculate Euler tour E
Apply short cutting repeated vertices in E
Perform 1-0 insertion, 1-1 switch and 2-Opt improvement until no more improvement

4.1.1.2 Loading Problem

In the loading part we proposed novel heuristic called Greedy Assignment Heuristic to solve MPLPF problem to assign customers' demands to compartments. The Pseudocode of Greedy Assignment Algorithm is represented in Algorithm 2. In addition to that, illustrative example is presented in Appendix A.2. This algorithm firstly starts with initial solution. We use two phenomenon for initial solution, first one is that sorted both compartments capacities and demands in decreasing order then iteratively determines customer which has maximum frequency assigned next unassigned compartment to that customer. Second one is randomly assign compartments to customers demand. We combine these two methodologies together which could be considered as multi run process. Multi run process is limited as number of customer. Moreover, we apply improvement phase to obtain more promising solution, therefore at the improvement phase iteratively determine the worst customer who has maximum frequency then searching other customers' to switch their compartments whether it decrease maximum frequencies between customers. If switching customers' compartments leads decrease on maximum frequency, it will try to overall sorting assigned compartments in a decreasing way. If it leads again decreasingly on maximum frequency. Algorithm accepts last sorting phase then it will try until there is no improvement. Finally, it selects minimum objective value from solutions of multi run process.

The last but not least, we stated one more time, we proposed two methodologies in face of whole mathematical formulation. First one is exact method to solve two problem by mathematical model and multiplying these two sub problem as an objective function. We called this method as TSPMPLPF in further notations. The other methodology is heuristic combined Christofides Algorithm and Greedy Assignment Algorithm. For the sake of simplicity, we called this methodology as C-GAA in further notations.

Algorithm 2 *Pseudocode of Greedy Assignment Algorithm*

Input: Demand rates d_i , $i \in N$; Compartment capacities q_i , $i \in M$; Compartments assigned to demand rates P_i , $i \in N$; Sum of compartments capacities assigned to demand rates K_i , $i \in N$

Notations: Frequency of demand rates f_i , $i \in N$; Assignment of compartments to customer(demand rates) x_{ij} : (if compartment assigned to demand rates(customer) equals 1 or 0, , $i \in N$,$j \in M$),

Output: Minmax frequence among the demand rates' frequencies

for m=1 to $|N'|$ \\Multi run process **do**
  Sort the demand rates in descending order as N'
  Sort the compartment capacity in descending order as M'
  if m =1 **then**
    Set all $x_{ij} = 0$ for all $i \in N$,$j \in M$
    for y=0 to $|N'|$, y++ **do**
      $x_{yy} = 1$ $y \in N'$,$y \in M'$
      Calculate f_i, K_i, $i \in N'$
    end for
    for y=$|N'|$ to $|M'|$, y++ **do**
      $f_{i^*} = max\{f_i|i \in N'\}$
      $x_{i^*y} = 1$ $y \in M'$
      Calculate f_i, K_i, $\forall i \in N'$
    end for
  else
    Randomly assign compartments to customers and calculate f_i, K_i,$\forall i \in N'$
  end if
  Start Improvement Stage
  Δ represent improvement on decrease in the $\mathbf{max\{f_i\}}, i \in N'$
  repeat
    $\Delta = 0$; Select demand index which has $f_{i^*} = max\{f_i|i \in N'\}$
    Ancillary Notations: W for define minimum decrease, $W = 0$;
    for s=1 to $|P_i|$ **do**
      for k=1 to $|N'|$ *and* $i \neq k$ **do**
        for l=1 to $|P'_k|$ **do**
          if $q_s < q_l$ **then**
            Switch compartments and update W
          end if
        end for
      end for
    end for
    Keep best improvement on Δ regarding to W and switch compartment positions.
    Sorting K_i in descending leads improvement on if Δ changes in decreasing, then change compartment assignments vice versa
  until No more improvement on Δ
end for
Select minimum frequence from results of multi run process

4.2 Solution Approach for Routing and Inventory Loading Problem for Heterogeneous Vehicle Fleet with Compartments

In this section, we provide two-phase approach as illustrated in Figure 4 to solve RILPHVC. In the first phase, we cluster customers and determine both routing cost and frequencies for each cluster corresponding to each vehicle type. Therefore possibilities of cost of serving customers in each cluster are total numbers of vehicles. In the second phase, we solve a set partitioning problem to select best solution among the solution pool.

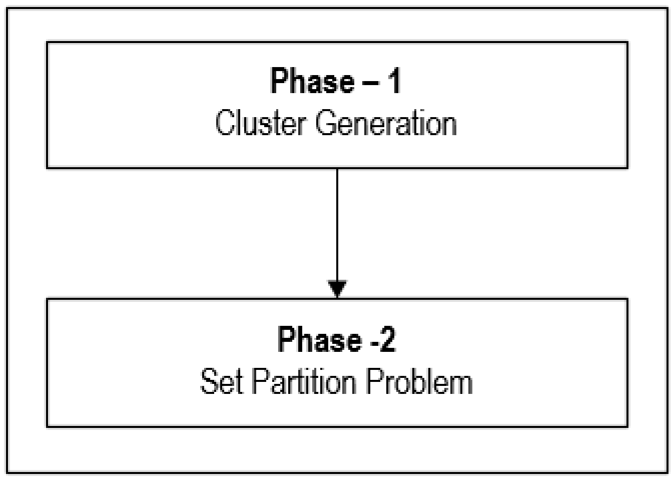


Figure 4: An illustration of Two-phase Approach

4.2.1 Cluster Generation

In this section, we provide some cluster generation methodologies, which are Sweep Algorithm, Iterative Sweep Algorithm and Random Subset Generation. All methodologies are based on create non-overlapping and potentially good cluster regarding to distance which enhanced the routing part. In our problem setting, we restrict cluster size as maximum compartment number among vehicles. For the illustration, In Figure 5, supposing that we have only five vehicle, there is only one feasible solution which consists of (Cluster 1-2-3-4) which are non-overlapping (unique) and we have to use 4 vehicle.

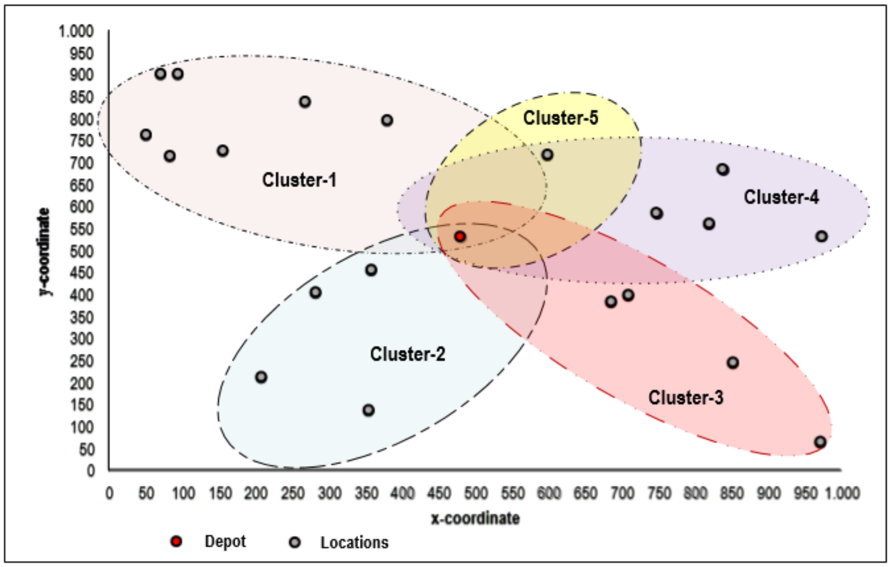


Figure 5: An Illustration of Cluster Generation

4.2.1.1 Sweep Cluster Algorithm

Sweep Algorithm is well known cluster generation technique for routing problem. The idea of this methodology determines polar coordinates of each location and select each location as starting point and sweeping area based on this starting point and clusters locations by one by in a clockwise or counterclockwise [68]. However, this technique does not provide enough clusters number, which eventually effects solution quality in terms of diversification. Pseudocode of Sweep Algorithm is presented in Algorithm-3.

Algorithm 3 *Sweep Cluster Algorithm Pseudocode*

```
1: Input: RILPHVFC instance on a map of D × D unit square geographically
   dispersed locations and depot.
2: Output: Cluster set S contains partitions of customer locations i ∈ V\{0}
3: Calculate polar coordinate system(angle θ,radius ρ) for all location i ∈ V\{0}
   based on as if depot locations is (0,0)
4: Sort all customers in non-decreasing order of θ
5: Create empty candidate set S
6: repeat
7:    Create set M
8:    for i=0 to MaximumClusterSize do
9:       Add nearest 'i' locations to set M
10:      if Is M in candidate set S then
11:         i ← i + 1 and clear set M go to Step 8
12:      end if
13:      Add M to subset S
14:   end for
15: until All customer is set as starting point
```

4.2.1.2 Iterative Sweep Cluster Algorithm

Iterative Sweep Algorithm is novel and developed from the idea of Sweep Algorithm but it explores much more solution space comparing to Sweep Algorithm. This algorithm needs to two inputs that are regions radian and starting angles degree. We define these degrees as (3, 5, 15, 30, 45, 60, 90, 120, 180 and 360) which are denominators of 360. Iterative Sweep Algorithm use each starting angle and convert coordinates to new coordinates as if these starting angles would zero angle. Then divide coordinates as radians degree based on this new coordinates, then cluster locations in each region based on their distance to depot iteratively. Thanks to constructing different regions we can reach much more diversified space comparing to Sweep Algorithm. Pseudocode of Iterative Sweep Algorithm is presented in Algorithm 4.

Algorithm 4 *Iterative Sweep Cluster Algorithm Pseudocode*

```
1: Input: RILPHVFC instance on a map of D × D unit square geographically
   dispersed locations and depot.
   Input-2: Starting angle sets A, Sector radian sets L
2: Output: Cluster set S contains partitions of customer locations i ∈ V\{0}
3: Calculate polar coordinate system(angle θ,radius ρ) for all location i ∈ V\{0}
   based on as if depot locations is (0,0)
4: Create empty candidate set S
5: Sort all customers in non-decreasing order of θ
6: for i=0 to |A| do
7:   for j=0 to |L| do
8:     Divide graph into sectors "L_j radians"
9:     Sort all customers in non-decreasing order of distances to depot's location
       for each sectors
10:    repeat
11:      for k=0 to MaximumClusterSize do
12:        Create set M which k successive order elements of ordered list in sector
           L_j
13:        for l=0 to |M| do
14:          if Is M_l in candidate set in S then
15:            l ← l + 1 , go to Step 13
16:          end if
17:        end for
18:        Add M_L to candidate set
19:      end for
20:    until All sector is scanned.
21:  end for
22: end for
```

4.2.1.3 Random Subset Cluster Algorithm

Random Subset Generation is based on randomly generated cluster regarding to geographically close locations [55]. At each iteration algorithm generates random base point then it calculate distances to base location to each locations. After that assigns probability to each location based on these distances. Assignment of customer location to cluster proceeding from high probability which enable us to intensification in solution space. On the other hand, MPLPF is independent of distance therefore we used extra randomness to adding non-promising customer location into cluster which enables us to diversification in solution space. In the assignment process we generate random number between (0-1) for each location, if customer location probability is greater than random number, it will be selected into cluster. As stated earlier we want to generate cluster which are non-overlapping. As a result of that we ensure each subset is unique. Moreover cluster generation process is restricted by pre-specified number of iterations. Termination rule for our algorithm is either cumulative 200,000 unsuccessful or 60,000 successful cluster generation trials. Pseudocode of Random Subset Algorithm is presented in Algorithm 5.

Algorithm 5 *Random Subset Cluster Generation*

1: **Input:** RILPHVFC instance on a map of $D \times D$ unit square geographically dispersed locations and depot.
2: **Output:** Cluster set S contains partitions of customer locations $i \in V \backslash \{0\}$
3: **repeat**
4:   Pick a random point b, the *base point*, on the map.
5:   Calculate c_{bi} for all $i \in V \backslash \{0\}$.
6:   Set the probability p_i of each customer location i for the selection process.

$$p_i \leftarrow \begin{cases} 0.9 & \text{if } c_{bi} \leq \frac{D}{10}, \\ 0.7 & \text{if } \frac{D}{10} < c_{bi} \leq \frac{D\times 3}{10} \\ 0.5 & \text{if } \frac{D\times 3}{10} < c_{bi} \leq \frac{D\times 5}{10} \\ 0.2 & \text{if } \frac{D\times 5}{10} < c_{bi} \leq \frac{D\times 7}{10} \\ 0 & \text{otherwise.} \end{cases}$$

7:   Order by descending these probabilities
8:   Create empty set S
9:   **repeat**
10:     Create candidate set M
11:     Generate random number $r \in (0, 1)$ for selection process.
12:     **if** $p_i \geq r$ **then**
13:       Add location i to M
14:     **end if**
15:   **until** M size is equal to maximum number of compartments having by vehicles
16:   **if** Is candidate set M in S **then**
17:     Go to the Step 3.
18:   **end if**
19:   Add candidate set M to S
20: **until** A termination condition is reached.

4.2.2 Set Partitioning Problem

In this subsection, we demonstrate how to integrate our problems into Set Partitioning Problem (SPP) in order to select best subset of generated clusters. Before solving set partitioning problem, we apply pre-processing technique to prepare problem environment. The reason is that some of subset size is greater than vehicles total compartment sizes that causes infeasibility. Therefore, we extract some cluster to vehicle assignment from set partition problem in order to handle infeasibility. Parameter of our problem as follow:

Parameters:

S: Set of all generated clusters,

S_j: j^{th} cluster in $S, j \in \{1, ..., |S|\}$,

K: Set of vehicles,

N: Set of locations except depot,

y_{jk}: feasibility 0-1 matrix define whether cluster j can be served by vehicle k or not, $j \in \{1, ..., |S|\}, k \in \{1, ..., |K|\}$,

x_{ij}: coverage 0-1 matrix define whether i location is in cluster j or not, $j \in \{1, ..., |S|\}, i \in \{1, ..., |N|\}$,

c_{jk}: cost of serving cluster j by vehicle k, $j \in \{1, ..., |S|\}, i \in \{1, ..., |N|\}$,

4.2.2.1 Cluster Cost

Before mathematical model, we want to state pre-processing algorithm for generate c_{jk}, y_{jk} parameters. In this pre-processing code we calculate all cost of serving each cluster which is satisfied feasibility. The cost of serving cluster is actually RILPSVC problem. In the Section 4.1.1., we provide two methodologies. The cost of serving cluster could be calculated as either TSPMPLPF or C-GAA. The Pseudocode of Pre-Processing is presented in Algorithm 6.

Algorithm 6 *Pre-Poccessing Pseudocode.*

```
Input: RILPHVC instance on a map of D × D unit square geographically dispersed locations and depot.
Input-2: Method ← (TSPMPLPF) or (C − GAA) \\To define which method is applied to calculate the logistic cost of cluster
Output: c_jk, y_jk ,
for j=1 to |N| do
  for k=1 to |K| do
    if S_j is feasible for vehicle k then
      c_jk ← Method
      y_jk ← 1;   else y_jk ← 0;
    end if
  end for
end for
```

Decision Variables:

$$s_{jk} = \begin{cases} 1, & \text{if the generated cluster } j \text{ is served by vehicle } k \\ 0, & \text{otherwise.} \end{cases} \qquad j \in \{1, ..., |S|\}, k \in \{1, ..., |K|\}$$

The set-partitioning problem as follows:

$$\text{SPP: Min} \quad \sum_{j \in S, k \in S} c_{jk} s_{jk} \tag{60}$$

s.t.

$$\sum_{j \in S, k \in K} x_{ij} s_{jk} = 1 \quad \forall i \in N \tag{61}$$

$$\sum_{k \in K} s_{jk} \leq 1 \quad \forall j \in S \tag{62}$$

$$s_{jk} \leq y_{jk} \quad \forall k \in K, \forall j \in S \tag{63}$$

$$s_{jk} \in 0, 1 \quad \forall k \in K, \forall j \in S \tag{64}$$

The objective (60) is to find minimal cost partition from S. Constraints (61) ensure that each location's demand must be met exactly once by one tour. Constraints (62) enforce location must be served by only one vehicle. Constraints (63) make sure that no vehicle serves tour which has number of customers greater than its number of compartments. Constraints (64) dictate structures and sign restrictions of the decision variables.

4.3 Tabu Search

In this section, we worked through other solution approach Tabu Search(TS), which can be adapted to our problem from the work [63].

TS is a well-known local search metaheuristic, one of the most effective heuristic for tackling with VRP and inventory related variations. The TS algorithm starts with initial solution whether feasible or infeasible. Algorithm generates neighborhood by the help of move operators for current solution. The current solution is selected by best objective value, fitness value, in the neighborhood at each iteration. Previous move or solution is kept in a tabu list, which ensures to avoid cycling, stuck in local optima by prohibiting moves in tabu list for a certain number of iterations called tabu-tenure. The tabu tenure of moves is updated at the end of each iteration. The best neighborhood solution is selected as the new current solutions as long as it is not labeled as tabu. Nevertheless, it is in the tabu list, it can have better fitness value than the overall best feasible solution i.e. incumbent solution so far which can be selected as the current solution even if it is in the tabu list. This is the aspiration criteria, which break ties. The algorithm is continued until termination criteria have been met [69].

Tabu Search presents basic framework to construct local search methodology, which must be adapted to problem specific in a tailored way. Our Tabu Search algorithm has 4 move operators to create neighborhood and contain Local Search (LS) algorithm embedded into it. The initial solution is obtained by constructive heuristic. We applied reduction technique to provide performance in terms of complexity and reactive mechanism which can change tabu tenure and disable reduction strategy in order to diversify more solution space.

4.3.1 Solution Representation

Designing solution representation is essential in constructing metaheuristic. We prefer to binary matrix to represent our solution as customer index and vehicle index, which are sufficient to explore solution space related to our problem. We define binary matrix such as each column represents customer index, each row represents vehicle index. If element of matrix equals 1, related customer assign to that vehicle. It is not allowed to assign one customer to multi vehicles therefore each sum of row must equal to 1. To illustrate this phenomenon, suppose that 9 customer with 5 vehicles each vehicle has at least 9 compartments, one feasible solution is as follow:

Customer Index \ Vehicle Index	0	1	2	3	4
0	0	1	0	0	0
1	0	0	1	0	0
2	0	0	0	1	0
3	0	0	0	1	0
4	1	0	0	0	0
5	0	0	1	0	0
6	0	1	0	0	0
7	1	0	0	0	0
8	0	0	0	1	0

Figure 6: Illustration of Tabu Search Solution Representation

According to Figure 6, Vehicle 0 serves Customer 4-7, Vehicle 1 serves Customer 0-6, Vehicle 2 serves Customer 1-5, Vehicle 3 serves Customer 2-3-8, Vehicle 4 is not in use.

4.3.1.1 Fitness Value of Solution Representation

Each solutions fitness value (objective value) could be calculated with Section 4.1.1. In order to calculate fitness value, turn solution representation into clusters as if customers being served by vehicle index. After that, use single vehicle version of problem whether use exact method (TSPMPLPF) or heuristic (C-GAA). In order to achieve objective value calculation, select rows index for each column element which equals to 1 and then calculate objective value for related column, after that summing all rows objective value reveals fitness value of solution representation.

4.3.2 Initial Solution

Tabu Search mechanism starts with initial solution. Starting as possible as good solution reveals better performance and well convergence. In our constructive heuristic, infeasible solution is not allowed. We tested several constructive heuristic and we decided to use Constructive Heuristic presented in Algorithm 7 as follow:

Algorithm 7 *Constructive Heuristic Pseudocode*

Input: Demand rates of customers, Vehicles, Method to evaluate calculation (TSPMPLPF or C-GAA)
Output: Initial Solution
Sort the demand rates in descending order as D
Ancillary Notation: if x_{ij}=1 demand i assigned to vehicle j otherwise x_{ij}=0, $\forall i \in D, \forall j \in V$
Set x_{ij}=0 , $\forall i \in D, \forall j \in V$
for s=1 to $|D|$ **do**
    Try all possible assignment as if $x_{sk} = 1$ which ensure feasibility,$\forall k \in V$
    Calculate objective values for each possible assignment
    Select minimum objective value among possible assignments and related to vehicle index (k^*);
    x_{sk^*}=1
end for

4.3.3 Neighborhood Generation

The fundemental of metaheuristic is creating good search space and evaluating its neighborhood by predetermined rules. The neighborhood of the current solutions is comprised by some move operators at each iteration in Tabu Search. Move operators is designed in different ways which are generally depends on solution representation. If move operators was used successively, it could searched every solution space. In our solution representation, we neglected compartment assignment which was solved by single vehicle version of problem. We defined four moves, which are 1-0 Move, 2-0 Move, 1-1 Move and 2-2 Move. All moves interact with other routes therefore no move in same route. The working mechanism of move operators is described as below and illustration of move operators in Figure 7:

1-0 Move: Select a served customer from vehicle, remove from its current vehicle and assigned to different vehicle.

2-0 Move: Select served two customer from same vehicle, remove them from their current vehicle and assigned to different vehicle.

1-1 Move: Select served two customers from different vehicles and swap their vehicle assignments.

2-2 Move: Select served two customer from same vehicle and other served two customer from different vehicle. Swap their vehicle assignment pair to pair.

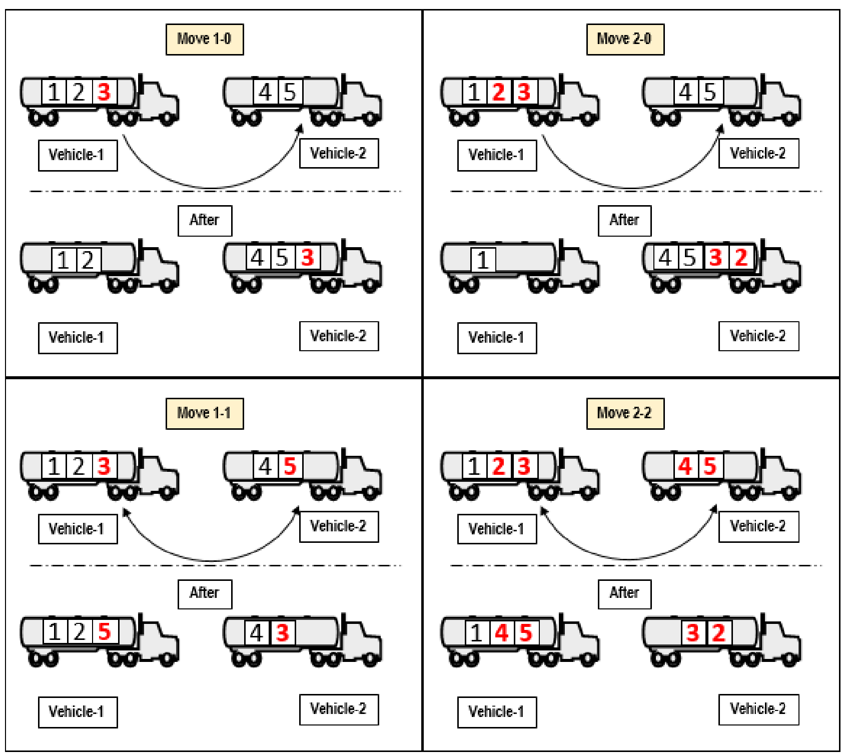


Figure 7: Illustration of Move Operators

4.3.3.1 Feasibility Control

In our solution frame, only 1-0 Move and 2-0 Move operators can make solution infeasible. In case of assigning customers to vehicle, which has fewer compartments than total number of customer assigned, is not allowed. Therefore, we ensure feasibility at each neighborhood generation process.

4.3.3.2 Neighborhood Reduction Strategy

Due to the high complexity of the IRP's, even metaheuristics need a significant amount of computational time when solving large-scale problems. In addition, there is a nearly general agreement on that researchers should implement the neighborhood reduction strategy to focus on desirable neighborhoods in order to reduce computation [70, 71].

To reduce neighborhood size, we adapted reduction technique into Move 1-0 and 2-0 operators. We have consensus that moving products from vehicle has greater objective value to other vehicle has lower objective value leads more improvement. Therefore, we design empirical study to test our consensus. In empirical study, we generate 100 instance, which has 30 customer, 15 vehicle, and compartment number in vehicles varies between 2-5. Then, we apply constructive heuristic and generate all possible move 1-0 and 2-0 and reduction strategy. We tested neighborhood size and best objective value. Reduction strategy has same objective value as all possible neighborhood in %86 of instances while decreasing neighborhood size as %46 of instances. The working mechanism of neighborhood reduction strategy is presented in Algorithm 8. The related empirical study is presented in Appendix A.3

4.3.4 Local Search

Diversification and intensification are two concept must be accomplish within Tabu Search mechanism [69]. Our main approach for diversification is move operators. Tabu Search armed with Local Search creates more sequential move which could not

Algorithm 8 *Neighborhood Reduction Pseudocode*

```
1: Input: Solution, Method to evaluate calculation (TSPMPLPF or C-GAA)
2: Output: Neighborhood List
3: Create empty set of Neighborhood List N
4: Decompose solution into for each vehicle's solution as V
5: for i=1 to |V| − 1 do
6:    for j=i+1 to |V| do
7:       Create empty candidate list S
8:       Generate and Move 1-0 and Move 2-0 for all products from Vehicle i and
         assigned to Vehicle j if ensure feasibility.
9:       Add candidates generated in Step 8 into S
10:      Calculate all candidates Objective Value in the S
11:      Add candidate list S into N
12:   end for
13: end for
```

be create by iteration after iteration. Therefore we applied Local Search Procedure within Tabu Search in order to get intensification. For that purpose, we used only 1-1 Move and 2-2 Move operators as Local Search operators. These operators create neighborhood of best candidate constructed from neighborhood from 1-0 Move and 2-0 Move Operators.

Implementing Local Search without pre-defined rules and only integrated sequential move operators reminds of brute force calculation [69]. Therefore we implemented sub procedure such as guiding mechanism which continuously identifies low-quality features and tries to emphasize specific qualified solution space. As we stated in Chapter 3, our problem is combined two sub problem. Therefore, we orient move operators in two layer which want to improve two sub problem side by side.

The idea behind two layer is that selecting desirable demands or customers while switching their assignment in their vehicles leads improvement on either routing problem or loading problem. For example, in the routing part we evaluate switching customers on vehicles lead decrease on Minimum Spanning Tree comparing to previous case. If switching customers leads decrease on sum of two vehicle Minimum Spanning Tree, we accept this move. In loading part, we sort vehicles in decreasing order by their frequencies. We allow switching customers, which are greater demand rates from higher frequencies of vehicle to less demand rates from fewer frequencies of vehicles. The two layer local search is presented in Algorithm 9. We tested Local Search within Tabu Search versus Tabu Search which includes also Move 1-1, 2-2. Our tested instances has same characteristic features as in Section 4.3.3.2 and convergence of Local Search presented in Figure 8 and numerical results in Appendix A.4.

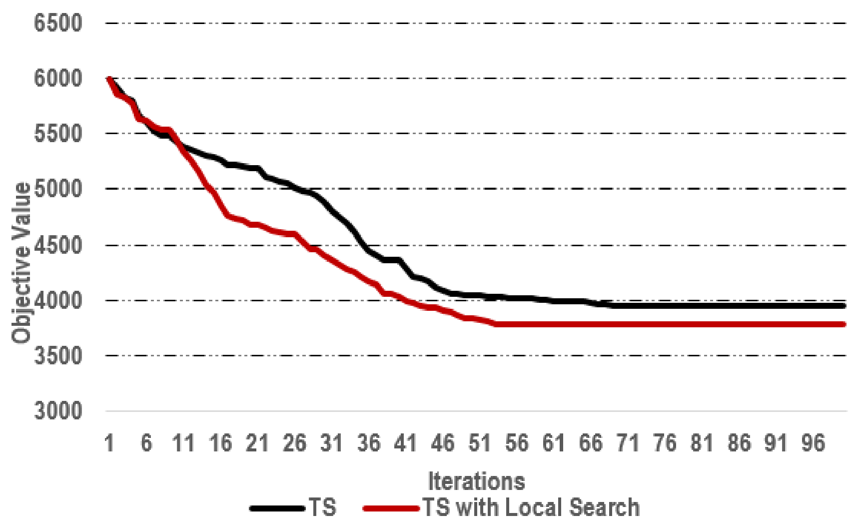


Figure 8: Convergence of TS versus TS with Local Search

4.3.5 Determination of Tabu Search Parameters

In this section, we discuss on determination parameters of Tabu Search such as tabu list, tabu tenure, termination criteria, aspiration criteria and reactive mechanism.

As the current solution is updated throughout the iterations, we decided to construct tabu restriction therefore, solutions visited before would not be selected repeatedly. In the academic literature there are several techniques to employ tabu list property such as keeping move indexes, solutions etc. We designed our two tabu lists according to each move indexes. In the Move 1-0 operator, two index needed first one is which product will assign to which vehicles. In the Move 2-0 operator, we insert two-row entry into tabu list. Suppose that Customer 1 in vehicle 6 assigned to

Algorithm 9 *Neighborhood Generation in Local Search*

```
1: Input: Best Solution from 4.3.3, Method to evaluate calculation (TSPMPLPF
   or C-GAA)
2: Output: Neighborhood List
3: Create empty set of Neighborhood List N
4: Decompose solution into for each vehicle's solution as V_j^f frequencies and V_j^r
   routing cost ,∀j ∈ V
   First Layer (Routing)
5: for s=1 to |V| − 1 do
6:    for k=s+1 to |V| do
7:       Create empty candidate list S_1
8:       Generate and Move 1-1 and Move 2-2 switching customers between Vehicle
         s and Vehicle k if improvement in a decreasing way on sum of Minimum
         Spanning Tree (MST) of Vehicle s,k
9:       Add candidates generated in Step 8 into S_1
10:      Calculate all candidates Objective Value in the S_1
11:      Add candidate list S_1 into N
12:   end for
13: end for
   Second Layer (Loading)
14: Sort V_i^f frequencies in a descending order, i∈ V
15: for s=1 to |V| − 1 do
16:   for k=s+1 to |V| do
17:      Create empty candidate list S_2
         Notation: d_ji refers demand rates of customer i assigned to vehicle j,
                   L_s refers location set which is served by vehicle V_s
18:      Generate and Move 1-1 and Move 2-2 switching customers between Vehicle
         s and Vehicle k, in such that d_si > d_kl ∀i ∈ L_s, ∀l ∈ L_k
19:      Add solutions generated in Step 18 into S_2
20:      Calculate all candidates Objective Value in the S_2
21:      Add candidate list S_2 into N
22:   end for
23: end for
```

Vehicle 5 with Move 1-0 operator, and Customer 2,3 assigned to Vehicle 4 with Move 2-0 operator, associated Tabu list as follow:

Table 8: Structure of Tabu List

Tabu List		
Index	Customer	Vehicle
1	6	5
2	2	4
3	3	4

In the Move 1-1 and Move 2-2 operator we need two row entry according each pair switching which means that two Move 1-0 Operator. For example suppose that Customer 2 in vehicle 1 and Customer 3 in vehicle 3 changes according to Move 1-1 operator. Then two row entries will insert to tabu list as follow:

Table 9: Move 1-1 Operator in Tabu List

Tabu List		
Index	Customer	Vehicle
1	1	3
2	3	2

The tabu tenure is assigned random integer values in the interval [5, MaximumTabuTenure] where MaxTabuTenure is 20. We use termination criteria for our Tabu Search is the maximum permissible number of iterations during which incumbent solution does not improve. Maximum non improvement number is determined as 50 which adapted from work [63, 72].

As we stated earlier in the beginning of the section, the solution whose move operator is in tabu list can be executed for the next best candidate if it produces a solution with a lower logistic cost than the incumbent's logistic cost (current best logistic cost so far) as integrated into Tabu Search as an aspiration criterion.

The reactive mechanism is first proposed by the study in [73]. The main idea of reactive mechanism is adapting tabu tenure dynamically in order to free search trajectory from limited part of the search space instead of avoiding closed search repetition or cycles. The robustness of the reactive mechanism is demonstrated in the studies [74, 75]. The reactive mechanism does not change only tabu tenure, it can control turn off reduction strategy. We prefer to use reactive mechanism in a two way. After limited number of iterations, if there is no improvement on incumbent solution, we turn off reduction strategy in neighborhood generation with Move 1-0 and Move 2-0 and change tabu tenure in randomly as [5, MaximumTabuTenure] to handle existing Tabu List element we used First In First Out (FIFO) strategy in order to rearrange tabulist. Effectiveness of the reactive mechanism is tested on 30 instance and provide overall %3.71 improvement on average. The numerical results are presented in Appendix A.5. We limit number of iterations as half of the termination number to activate reactive mechanism.

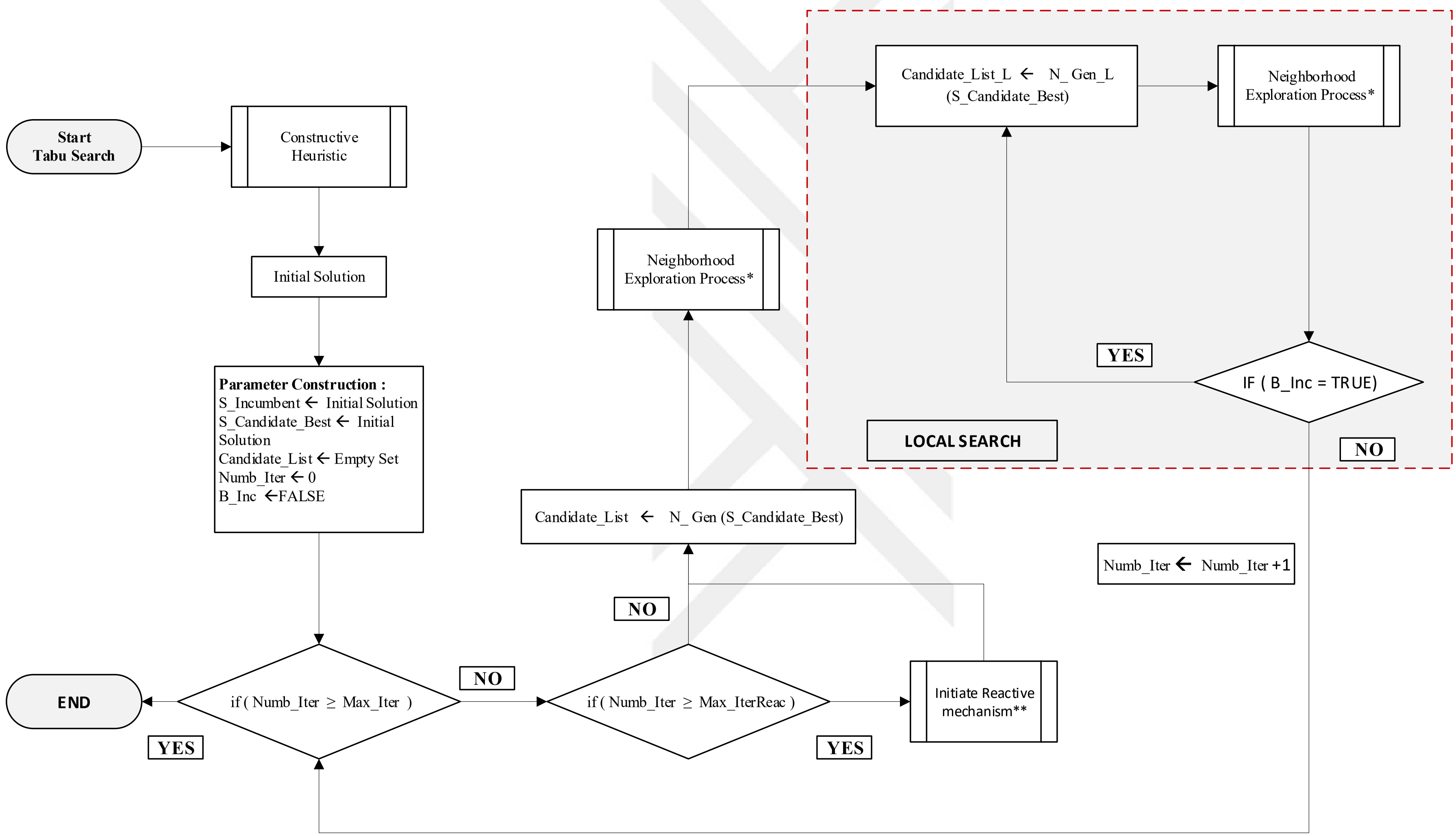
4.3.6 Tabu Search Framework

In this subsection, we provide flowchart in Figure 9 to describe the steps of our TS implementation. For the beginning, we give notations used in flowchart.

Numb_Iter:	Number of iterations which the incumbent solutions does not improve
Max_Iter:	Maximum number of iterations which the incumbent solutions does not improve
Max_IterReac:	Maximum number of iterations to initiate reactive mechanisim
OV():	Objective value

S_Incumbent:	Incumbent solution
S_Candidate:	Candidate solution
S_Candidatebest:	Best candidate solution in neighborhood
Candidate_List:	Candidate list which is generated by move operators (Move 1-0, Move 2-0)
Candidate_List_L:	Candidate list which is generated in Local Search by move operators (Move 1-1, Move 2-2)
MoveList:	Move lists associated to Candidate_List
MoveList_L:	Move lists associated to Candidate_List_L
TL:	Tabu list
BestMove:	Best move associated with best candidate solution in neighborhood
B_Inc:	Boolean in order to follow incumbent solution whether change or not
R_active:	Boolean in order to follow reactive mechanism whether active or not
N_Gen:	Neighborhood Generation
N_Gen_L:	Neighborhood Generation in Local Search

Figure 9: The Flowchart of Tabu Search



* The Flowchart of Neighborhood Exploration Subprocess is presented in Appendix A

** The Flowchart of Reactive Mechanism Subprocess is presented in Appendix A

4.4 Lower Bound Approaches

In the area of optimization, there must be conducted studies on lower/upper bound approaches with intention to understand nature of the problem and comparison issues. By the help of lower/upper bound approaches, researcher figures out boundaries and limitations even for the large scaled problems. In this section, we worked on lower bound approaches according to our problem. In general, constructing effective lower/upper bound on optimization problem is very time consuming and hard comparing to developing solution approach.

4.4.1 Single Vehicle Version of the Problem

Observation 2. *The multiplication of objective values (Minimum Spanning Tree and Relaxed MPLPF) is a lower bound of single vehicle version of the problem.*

Suppose that objective value of single version problem is OV which can be composed in such that $OV = OV^R * OV^F$ where OV^R is routing cost and OV^F is frequency. Since OV^R and OV^F are non-negative number, it is sufficient to find two lower bound corresponding each subproblem and multiplication of these lower bound reveals lower bound of the original problem. Minimum Spanning Tree is lower bound of Travel Salesman Problem. In addition to that if we allow mixing product allocation into compartments, we are relaxing Constraint (58) and the optimum solution of relaxing problem is unique because of each customer having same frequency such that $\frac{\sum_{i \in N} d_i}{\sum_{m \in M} q_m}$. Therefore lower bound of RILPSVC is $(\frac{\sum_{i \in N} d_i}{\sum_{m \in M} q_m} * MST)$.

4.4.2 Multi Vehicle Version of the Problem

In this subsection, we work on lower bound according to multi vehicle version of the problem. Multi vehicle version of problem is very complex problem hence proved lower bound is not easy to find. We tested several approaches and decided on using following lower bound technique.

Observation 3. *The objective value in such that vehicle with maximum compartment capacity travels each location and only serves customer with minimum demand is lower bound of multi vehicle version of problem.*

Suppose that optimum objective value of original problem is OV which can be decompose into objective values for each vehicles such as $OV^* = OV_1^f * OV_1^r + OV_2^f * OV_2^r + ... + OV_n^f * OV_n^r \quad | \quad \forall n \in |V|$ where f stands for frequency and r stands for route cost. Suppose that OV^{f_L} is related frequence which is minimum customer demand over maximum vehicle capacity (sum of all its compartments). The OV^{f_L} ensure minimum frequency can be obtained according to problem settings. Therefore $OV^{f_L} \leq OV_n^f \quad | \quad \forall n \in |V|$. Moreover we can derive the distribution property of multiplication over sum operation such as $OV^L = OV^{f_L} \times \{OV_1^r + OV_2^r + ... + OV_n^r\} \quad | \quad \forall n \in |V|$. However the $\sum_{n \in |V|} OV_n^r$ is not known because we have no clue which vehicle serves to which customer. However we can construct lower bound on $\sum_{n \in |V|} OV_n^r$ such as grand TSP which visit all customer. We define Minimum Spanning Tree related to grand TSP tour cost as OV^{r_L}. Therefore lower bound OV^L equals to $OV^L = OV^{f_L} \times OV^{r_L}$. All objective costs are non-negative number as a result of that we can proved observation by derived inequalities as below:
$0 \leq OV^L \leq OV^{f_L} \times \{\sum_{n \in |V|} OV_n^r\} \leq \{OV_1^f * OV_1^r + OV_2^f * OV_2^r + ... + OV_n^f * OV_n^r\} \quad | \quad \forall n \in |V|\}$

Chapter V

COMPUTATIONAL STUDY

In this chapter, we conduct a computational study on randomly generated instances to assess performances of all proposed algorithms in terms of solution quality and computational performances. Firstly, we explain computational environment, instance generation and then introduce results and findings related to proposed approaches.

5.1 Computational Environment and Instance Generation

As we stated earlier, our problem has not worked in the OR literature. As a reason, there is no available benchmark data to compare with our solution methodology performance. Therefore, we firstly want to mention our data generation. We have to need six input for the instance generation, which are as follow:

- Geographical Map Size (X_{max}, Y_{max})
- Customer location number $(N_{location})$
- Minimum-maximum demand size for each customer (D_{min}, D_{max})
- Vehicle number $(N_{vehicle})$
- Compartment number (CN_{min}, CN_{max})
- Minimum-maximum compartment size (Cap_{min}, Cap_{max})

First, we generate our map according to (X_{max}, Y_{max}) coordinates. Then we generate random numbers both $(0, X_{max})$ and $(0, Y_{max})$ for each location. Then, we generate random number (D_{min}, D_{max}) for demand rate for each customer location. Further we generate random number (CN_{min}, CN_{max}) to assign compartment number for each vehicle. For the last stage we generate random number (Cap_{min}, Cap_{max}) for all compartments in the vehicles. For the sake of simplicity, we use $(1000 * 1000)$ unit square map for generating locations' coordinates. For generation of demand rates, we prefer to construct demand rates such that less and more varied which randomly generated numbers between (1000, 2000), (1000, 4000). Moreover, we use same structure in generation of compartment capacities such that less and more varied which randomly generated numbers between (1000, 2000), (1000, 4000). If not defined we prefer to generate both demand rates and compartment capacities in interval (1000, 3000).

The complexity of single vehicle version of problem mainly depends on number of locations (customers) and number of compartments. Therefore, we have to define dynamic properties regarding to compartment numbers. The number of compartments are randomly generated between $[N_{location}, (2 \times N_{location})]$ which is applied for only generating for single vehicle version of problem. In addition to that, we performed analysis on number of locations such as 5, 10, 20, 30, 40 and 50.

Instance generation of multi vehicle version of the problem differs from single version because of necessity of vehicle numbers. Number of vehicles is another factor which is directly effecting complexity. Since our problem decides which vehicle in use or not, invastigation of vehicle number must adapted in dynamically changing instances. We assign random number between $(\lceil N_{location}/CN_{min} \rceil, \lceil 2 \times N_{location}/CN_{min} \rceil)$ before generating other variables. Plus, we use minimum compartment number which ensures feasibility on further process. Thanks to that, we enchance stability in analysis of number of vehicles factor. For the part of generating compartment structure, we address studies in the literature. The number of compartments is varied on from studies to studies. However, general conception on number of compartments is less than six. Moreover, we can refer famous studies [76], [77] ,[78], [79] which are respectively use interval of compartments numbers as 2, (2-3), (2-5) and

(2-6). Proposed solution methodologies were programmed in C# language in Visual Studio 2015 Community platform and executed on a laptop with Intel (R) Pentium Dual CPU at 2.33 GHZ, 2GB of RAM on Windows 7. In order to evaluate exact solutions, we have run the developed mathematical models using Cplex 12.6 32 bit via Concert Technology with C#. Cplex execution time and termination are controlled by parameter setting. For that purpose, a relative optimality gap and termination time are set to 0.000001 and 3 hours CPU time. As a result, Concert Technology will terminates mathematical model either gap reaches optimality gaps value or execution time reaches to termination time. For the purpose of memory management, we use node file parameter as “2” in order to handle $\ll out\,of\,memory \gg$ situation.

5.2 Single Vehicle Version of the Problem

In this section, we provide the results of our computational study on single vehicle version of problem, we tested our Christofides with Greedy Assignment Algorithm (C-GAA) with respect to the exact solution found by mathematical model (TSPM-PLPF) on diversified size of problems. For small instances, mathematical model found optimal solution whereas it could not find the optimal solution in large instances. For the purpose of comparison and sensitivity analysis, we conduct assessments in two ways such as optimality gap and computational time. We started analysis instances with normal demand rates and compartment capacities. The evaluation of results is based on two ways in such that optimality gap in percentage and computational time in seconds. In order to obtain optimality gap, initially we used lower bound in presented in the Section 4.4, however it did not provide tightened and effective lower bound. Therefore, it is better to use lower bound which is acquired from Cplex after predetermined time to end. This lower bound can be reached by method called $\ll GetBestObjValue \gg$ in Concert Technology Library. As we stated in the Section 5.1, our Cplex algorithm terminates after 3 hour execution time for both TSP and

MPLPF problems. It is worth to emphasize one more time here, we employ two mathematical model in order to get optimal solution therefore maximum execution time is 6 hours for each single vehicle version of the problem instances.

Not only we provide optimality gaps both C-GAA and TSPMPLPF algoritm, but also we provide optimality gap for lower bound presented in the Section 4.4. In the Table 17, we present comparisonal result according to proposed algorithm for single vehicle version of problem on original instances. The first two columns represent Instance name and number of locations. Following three columns defines optimality gaps of C-GAA, TSPMPLF and Lower Bound comparing to lower bound of the Cplex. Negative signs means that lower bound approach in Section 4.4 is lower than the lower bound obtained from Cplex.

According to results presented in Table 10, we observe that the small instances (5-10 number of locations) are solved to optimality by TSPMPLPF in a few seconds. TSPMPLPF could not solve all problem in optimality and it takes 10,961 seconds on average to solve instances which have more than 20 number of locations. On the other hand, C-GAA is very effective in terms of computational efficiency even large instances solved in a less than a second. Moreover C-GAA has average optimality gap 0% and 0.30% for small instances respectively 5 and 10 number of locations. C-GAA has average 6.99% optimality gap whereas TSPMPLPF has average 4.10% on instances with more than 20 customers and the maximum difference of optimality gap between C-GAA and TSPMPLF is 4.50%. In the worst case ,optimality gap of C-GAA and TSPMPLPF are 19.23% and 14.73%. Unfortunately, our proposed lower bound is not effective which is far away from optimal with average -25.15% on overall instances.

Table 10: The optimality gaps and computational times of C-GAA on single vehicle instances on original instances

		Optimality Gap(%)			Time in Seconds		
Instance	Loc #	C-GAA	TSPMPLPF	Lower Bound	C-GAA	TSPMPLPF	Lower Bound
Ins 1	5	0.00%	0.00%	-47.07%	0.01	0.09	0.00
Ins 2	5	0.00%	0.00%	-41.05%	0.01	0.05	0.00
Ins 3	5	0.00%	0.00%	-35.57%	0.01	0.05	0.00
Ins 4	5	0.00%	0.00%	-35.86%	0.01	0.05	0.00
Ins 5	5	0.00%	0.00%	-34.74%	0.01	0.17	0.00
Ins 6	10	0.00%	0.00%	-33.65%	0.02	0.78	0.04
Ins 7	10	1.49%	0.00%	-26.82%	0.02	10.90	0.00
Ins 8	10	0.00%	0.00%	-37.79%	0.02	0.54	0.00
Ins 9	10	0.01%	0.00%	-38.53%	0.01	0.69	0.00
Ins 10	10	0.00%	0.00%	-37.10%	0.01	0.95	0.00
Ins 11	20	3.25%	1.67%	-17.80%	0.04	11,308.19	0.02
Ins 12	20	5.68%	2.79%	-15.10%	0.02	11,039.98	0.02
Ins 13	20	0.42%	0.00%	-40.38%	0.03	2,257.96	0.00
Ins 14	20	3.60%	0.00%	-37.66%	0.03	3,532.21	0.00
Ins 15	20	10.62%	5.55%	-23.54%	0.02	11,289.79	0.00
Ins 16	30	5.20%	2.50%	-18.94%	0.01	10,721.51	0.00
Ins 17	30	2.30%	0.00%	-20.44%	0.02	5,692.12	0.00
Ins 18	30	9.30%	4.90%	-27.33%	0.05	11,166.51	0.01
Ins 19	30	9.87%	6.30%	-24.88%	0.04	11,158.26	0.00
Ins 20	30	5.20%	3.80%	-29.08%	0.04	10,721.49	0.00
Ins 21	40	6.40%	1.94%	-16.34%	0.06	11,649.39	0.02
Ins 22	40	2.40%	2.02%	-14.06%	0.12	11,078.91	0.06
Ins 23	40	5.64%	1.64%	-13.79%	0.09	11,002.47	0.03
Ins 24	40	6.70%	2.98%	-10.50%	0.17	10,802.55	0.00
Ins 25	40	15.64%	11.90%	-17.34%	0.09	15,802.16	0.00
Ins 26	50	4.23%	2.90%	-13.54%	0.07	15,185.92	0.01
Ins 27	50	6.30%	3.60%	-13.10%	0.08	12,698.58	0.02
Ins 28	50	11.20%	8.80%	-9.89%	0.10	13,410.20	0.03
Ins 29	50	6.72%	3.94%	-9.06%	0.11	11,892.00	0.02
Ins 30	50	19.23%	14.73%	-13.41%	0.23	16,817.19	0.01

In addition we tested single vehicle version of problem with other characteristics such as less and more varied demand rates, compartment capacities. We provided these results on Appendix B.1. We prefer to present here summary of statistic measures according to all results in Table 11. According to Table 11, difference interval of demand rates and compartment capacities do not significantly influence neither C-GAA performance or TSPMPLPF performances in terms of optimality. The main reason is that MPLPF problem complexity increases with ratio such that number of compartments over number of locations. Thus we conclude that C-GAA is very effective in solving single version of problem in terms of solution quality and computational time.

Table 11: Statistical measures of C-GAA and TSPMPLPF performances on Optimality Gap(%) in all results

			Number of Locations					
			5	10	20	30	40	50
Less varied demand	C-GAA	Average	0.00%	0.47%	4.87%	6.42%	8.66%	10.29%
		Median	0.00%	0.00%	3.40%	5.30%	9.43%	8.16%
		Min	0.00%	0.00%	2.71%	3.67%	3.10%	6.21%
		Max	0.00%	2.19%	8.90%	11.20%	13.23%	16.23%
	TSPMPLPF	Average	0.00%	0.00%	2.06%	3.93%	5.25%	6.25%
		Median	0.00%	0.00%	1.72%	3.90%	4.98%	4.51%
		Min	0.00%	0.00%	0.00%	0.00%	1.90%	3.54%
		Max	0.00%	0.00%	4.50%	7.12%	9.63%	11.41%
More varied demand	C-GAA	Average	0.00%	0.39%	5.05%	6.77%	8.95%	10.66%
		Median	0.00%	0.00%	4.00%	6.90%	8.33%	9.23%
		Min	0.00%	0.00%	2.34%	5.25%	6.34%	5.78%
		Max	0.00%	1.96%	8.35%	9.23%	14.21%	18.21%
	TSPMPLPF	Average	0.00%	0.00%	2.04%	3.70%	6.25%	7.07%
		Median	0.00%	0.00%	1.65%	3.17%	6.31%	6.94%
		Min	0.00%	0.00%	0.00%	1.95%	2.54%	2.87%
		Max	0.00%	0.00%	4.62%	7.34%	10.12%	13.21%
Less varied comp. cap.	C-GAA	Average	0.00%	0.41%	5.23%	6.91%	8.80%	10.18%
		Median	0.00%	0.00%	3.78%	6.85%	9.75%	9.93%
		Min	0.00%	0.00%	1.80%	2.75%	4.67%	5.80%
		Max	0.00%	2.04%	9.12%	12.54%	12.33%	14.68%
	TSPMPLPF	Average	0.00%	0.00%	2.05%	3.89%	6.68%	6.51%
		Median	0.00%	0.00%	1.48%	3.14%	7.31%	5.91%
		Min	0.00%	0.00%	0.00%	1.73%	2.12%	2.16%
		Max	0.00%	0.00%	4.67%	8.47%	9.25%	12.94%
More varied comp. cap.	C-GAA	Average	0.00%	0.46%	4.61%	6.63%	8.41%	10.24%
		Median	0.00%	0.00%	3.91%	5.40%	8.07%	8.81%
		Min	0.00%	0.00%	1.35%	5.00%	6.33%	6.26%
		Max	0.00%	1.56%	8.62%	10.54%	12.78%	17.03%
	TSPMPLPF	Average	0.00%	0.00%	1.88%	3.79%	5.07%	6.19%
		Median	0.00%	0.00%	1.85%	4.02%	4.56%	4.91%
		Min	0.00%	0.00%	0.00%	0.00%	3.08%	3.19%
		Max	0.00%	0.00%	4.35%	6.71%	9.44%	12.84%

5.3 Multi Vehicle Version of the Problem

In this section, we test performance of our proposed approaches (Two-phase approach, Tabu Search). In order to evaluate performances, we design enumeration frame, which gives optimal solution according to multi vehicle version of the problem in small size instances even taking many computational times. According to Section 5.2, since single version problem could be solved in reasonable amount of time by commercial solver. If we generate all subsets less than or equal to maximum compartment size among vehicles, after that we apply single version of problem for each subset for each vehicle, which ensures all possibilities. Finally, we can apply SPP into all subsets in order to find best assignment with minimum logistic cost. However, enumeration frame only work for instances which have at most 25-customer locations and max 6-compartment size because of increasing number of subsets and complexity. In the lights of enumeration frame, we can test our proposed algorithms and lower bound performances for small size problems. For the large size problems, we compare our proposed approaches with each others.

Table 12 provides a summary of the results on the main test instances which has compartment numbers between 2 and 5, demand rates, compartment capacities between 1000 and 3000. In Table 12, the first column is the instance name, the second column is the number of vehicles and third column is the number of locations corresponding to instances. We evaluate four solution approaches which three of them are clustering based two-phase approach and the other is Tabu Search. All of the solution approaches using C-GAA algorithm in order to solve single version of problem within. Clustering based Two-phase solution approaches are namely Sweep Algorithm (SC), Iterative Sweep Algorithm (ISC) and Random Subset Algorithm (RSC). The Tabu Search Algorithm is presented as TS. The following five columns (SC, ISC, RSC, TS and Lower Bound) presents the optimality gaps of these approaches with respect to results obtained by enumeration. We called enumeration as AC-TSPMPLPF. The remaining columns presents computational times of these approaches and enumeration.

The average optimality gap of SC is about 1.82% on all instances whereas 6.83% in the worst case. The average computational time for SC is 2.33 seconds. The average optimality gap of ISC is 0.24% on all instances whereas 1.68% in the worst case. The average computational time for ISC is 6.85 seconds. The average optimality gap of RSC is about 0.03% on all instances whereas 0.76% in the worst case. The average computational time for RSC is 42.78 seconds. Moreover the average optimality gap of TS is 0.10% whereas 1.32% in the worst case. The average computational time for TS is 110.10 seconds. The average computational time for obtaining optimal solution in all instances is 2,771 seconds. More, it takes 6,605 seconds to get optimal solution on average more than 20 customers. Which shows that both clustering based two phase solution approaches and TS are very fast in terms of computational time. On the contrary, the average optimality gap of Lower Bound Approach is -91.78%. Since lower bound approach for single vehicle version problem fails in terms of optimality gaps, it is not surprising that lower bound of multi vehicle version of problem is very weak; nevertheless, it is the first proven lower bound so far. To sum up, we observed that RSC outperform significantly all the solution approaches. Next, we investigate how our algorithms perform on instance with different characteristic such as less/more varied compartment capacities and demand rates. For the sake of simplicity, we prefer to present here summary of statistic measures of all results according to generated instances. The detailed results of different characteristic instances are presented Appendix B.1.

Table 12: The optimality gaps and computational times of heuristics on multi vehicle instances in where compartment numbers range is (2-5)

			Optimality Gap (%)					Time in Seconds				
Instance	Vec. #	Loc #	SC	ISC	RSC	TS	Lower Bound	SC	ISC	RSC	TS	AC-TSPMPLPF
Ins1	3	5	0.00%	0.00%	0.00%	0.00%	-91.24%	0.05	0.10	0.13	0.21	1.55
Ins2	4	5	0.00%	0.00%	0.00%	0.00%	-88.35%	0.06	0.11	0.13	0.24	1.40
Ins3	5	5	0.00%	0.00%	0.00%	0.00%	-89.29%	0.07	0.12	0.13	0.41	1.52
Ins4	5	5	0.00%	0.00%	0.00%	0.00%	-88.23%	0.08	0.12	0.15	0.37	1.24
Ins5	5	5	0.00%	0.00%	0.00%	0.00%	-92.37%	0.12	0.13	0.15	0.31	1.52
Ins6	5	5	0.00%	0.00%	0.00%	0.00%	-91.81%	0.14	0.08	0.17	0.25	1.81
Ins7	5	5	0.00%	0.00%	0.00%	0.00%	-88.16%	0.14	0.07	0.17	0.41	1.64
Ins8	6	10	2.50%	0.00%	0.00%	0.00%	-93.21%	0.38	0.61	0.85	2.02	33.37
Ins9	6	10	0.00%	0.00%	0.00%	0.00%	-91.02%	0.48	0.82	0.96	3.11	40.41
Ins10	7	10	1.25%	0.00%	0.00%	0.00%	-89.20%	0.56	0.83	1.20	4.21	44.90
Ins11	10	10	0.34%	0.00%	0.00%	0.00%	-92.72%	0.57	0.99	1.23	4.30	46.34
Ins12	10	10	0.78%	0.00%	0.00%	0.00%	-88.30%	0.58	1.08	1.41	3.44	50.53
Ins13	10	10	3.78%	0.00%	0.00%	0.00%	-92.03%	0.60	1.30	1.41	1.92	51.31
Ins14	10	10	0.91%	0.00%	0.00%	0.00%	-92.35%	0.63	1.43	1.67	3.88	85.99
Ins15	8	15	0.26%	0.00%	0.00%	0.00%	-92.28%	1.01	1.28	3.24	5.25	452.56
Ins16	10	15	3.60%	0.00%	0.00%	0.00%	-92.51%	1.70	2.37	4.75	8.02	454.92
Ins17	10	15	2.57%	0.38%	0.00%	0.00%	-88.80%	1.78	2.85	9.21	25.44	517.05
Ins18	10	15	1.28%	0.00%	0.00%	0.00%	-91.04%	1.81	3.00	15.88	22.19	540.58
Ins19	13	15	4.01%	1.25%	0.00%	0.00%	-93.87%	1.97	3.44	16.01	28.90	635.84
Ins20	15	15	1.85%	0.00%	0.00%	0.00%	-92.42%	2.12	7.25	17.17	43.59	765.79
Ins21	15	15	0.48%	0.00%	0.00%	0.00%	-91.63%	2.37	7.82	19.83	29.74	806.86
Ins22	11	20	0.89%	0.00%	0.00%	0.00%	-90.98%	1.88	6.76	11.97	18.61	2,604.33
Ins23	12	20	1.39%	0.98%	0.00%	0.00%	-90.10%	2.02	6.89	14.69	28.22	2,718.41
Ins24	14	20	2.59%	0.00%	0.24%	0.23%	-91.06%	2.63	7.57	16.88	24.43	2,926.32
Ins25	16	20	1.77%	0.00%	0.00%	0.00%	-90.72%	2.65	7.57	18.38	27.73	2,832.10
Ins26	17	20	1.99%	0.00%	0.00%	0.00%	-90.66%	3.07	8.30	23.43	34.98	3,000.61
Ins27	17	20	2.32%	0.00%	0.00%	0.00%	-90.34%	3.16	8.76	29.90	62.40	3,134.50
Ins28	20	20	5.90%	1.68%	0.00%	1.05%	-91.57%	3.49	8.82	30.76	74.44	3,551.03
Ins29	15	25	1.50%	0.07%	0.00%	0.00%	-95.38%	2.81	9.64	39.06	63.08	8,631.76
Ins30	17	25	2.61%	0.00%	0.00%	0.00%	-93.86%	3.47	10.78	99.18	272.74	8,649.33
Ins31	18	25	3.21%	1.63%	0.00%	0.45%	-96.62%	3.84	12.59	129.35	280.56	10,351.64
Ins32	18	25	2.93%	1.57%	0.76%	1.32%	-95.44%	5.88	21.49	200.62	326.61	9,982.21
Ins33	20	25	6.83%	0.76%	0.00%	0.47%	-94.79%	7.66	27.42	215.68	599.39	10,175.68
Ins34	21	25	3.45%	0.00%	0.00%	0.00%	-93.39%	10.25	29.41	263.18	744.54	10,684.46
Ins35	25	25	1.91%	0.00%	0.00%	0.00%	-95.33%	11.23	38.19	308.35	675.90	13,228.38

Table 13: Statistical measures of solution approaches' performances on Optimality Gap(%) and Computational Time in all results

		Optimality Gaps(%)					Computational Time in Seconds				
		SC	ISC	RSC	TS	Lower B,	SC	ISC	RSC	TS	AC-TSPMPLPF
less varied Demand	Average	1.99%	0.26%	0.03%	0.13%	-91.10%	2.05	6.90	42.13	85.54	2,832.38
	Median	2.07%	0.00%	0.00%	0.00%	-91.98%	1.31	4.20	9.43	20.92	455.21
	Min	0.00%	0.00%	0.00%	0.00%	-95.34%	0.06	0.09	0.13	0.17	1.19
	Max	7.02%	2.32%	0.67%	1.57%	-85.07%	10.25	42.21	371.22	970.73	13,017.60
more varied demand	Average	2.11%	0.32%	0.04%	0.14%	-86.32%	2.12	6.98	46.84	94.51	2,898.89
	Median	1.81%	0.00%	0.00%	0.00%	-87.40%	1.52	3.32	14.61	22.46	581.55
	Min	0.00%	0.00%	0.00%	0.00%	-92.15%	0.06	0.09	0.13	0.16	1.42
	Max	8.02%	1.51%	0.75%	1.22%	-72.31%	7.23	32.66	323.12	854.98	12,865.67
less varied comp, cap,	Average	1.98%	0.34%	0.04%	0.12%	-93.38%	2.25	6.96	43.89	87.75	2,813.62
	Median	1.89%	0.00%	0.00%	0.00%	-93.23%	1.73	4.00	9.07	19.75	388.75
	Min	0.00%	0.00%	0.00%	0.00%	-96.00%	0.07	0.12	0.13	0.15	1.14
	Max	7.12%	1.90%	0.68%	1.27%	-90.36%	8.40	38.01	300.12	834.94	11,702.03
more varied comp, cap,	Average	2.09%	0.35%	0.03%	0.13%	-89.68%	2.33	8.42	44.31	99.21	2,888.21
	Median	2.08%	0.00%	0.00%	0.00%	-89.04%	1.79	3.21	13.71	24.41	369.75
	Min	0.00%	0.00%	0.00%	0.00%	-95.86%	0.04	0.07	0.10	0.17	1.16
	Max	6.31%	1.22%	0.41%	1.02%	-85.69%	9.98	51.53	318.12	871.02	12,731.55

Examination of Table 13 reveals that both less/more varied compartment capacities and demand rates do not effect significantly solution quality in terms of optimality gap and computational times. In general, demand rates or vehicle capacity effect IRP because of changing further planning delivery schedules. According to static policy which deals with one scheduling at time which ensures stability on changing compartment capacities and demand rates. Minor differences on optimality gaps on RSC is based on instance specific features.

Next, we examine analysis on instances in where compartment number ranges is between 2 and 6 and the results are presented in Table 14. Eventually, computational times of solution approaches and enumeration is increase because of increasing complexity. The average optimality gap of SC is about 2.72% on all instances whereas 10.23% in the worst case. The average computational time for SC is 0.08 minutes. The average optimality gap of ISC is 0.31% on all instances whereas 2.12% in the worst case. The average computational time for ISC is 0.28 minutes. The average optimality gap of RSC is about 0.11% on all instances whereas 1.02% in the worst case. The average computational time for RSC is 3.99 minutes. Moreover the average optimality gap of TS is 0.29% whereas 2.25% in the worst case. The average computational time for TS is 6.43 minutes. The average computational time for obtaining optimal solution in all instances is 509 minutes. In the worst case, it takes 2,803 minutes (approximately 46 hours) to find optimal solution. Computational times of SC and ISC is less than RSC and TS. However, RSC ensure solution quality and outperform SC, ISC and TS in terms of optimality gap.

Finally, we test performances solution approaches on instances where maximum number of customers is 50 and their compartment number range is between 2 and 10. For the comparison, enumeration is not possible. Therefore, we evaluate performances of solution approaches with each other. So far, RSC has the least optimality gap, we compare other approaches with results obtained by RSC. We presented results in

Table 15. According to Table 15, The relative gap to RSC of SC is about 5.52% on all instances whereas 14.82% in the worst case. The average computational time for SC is 0.26 minutes. The relative gap to RSC of ISC is about 1.19% on all instances whereas 4.23 % in the worst case. The average computational time for SC is 1.64 minutes. The relative gap to RSC of TS is about 1.42% on all instances whereas 6.03 % in the worst case. The average computational time for TS is 34.41 minutes. Computational times of RSC and TS is increasing very much such that worst case 78.13 minutes for RSC and 93.25 minutes for TS. It is interesting thing that overall average optimality gap of ISC is less than TS. The main reason is that constructing TS parameters mainly based on compartment number range (2-5). Moreover, even if solution quality of the ISC is worse than RSC, it provides well speed according to its solution quality.

In conclusion, our proposed solution approaches perform well according to their optimality gaps where compartment number ranges are (2-5) and (2-6). The best performer is RSC and it provides good quality solution comparing to other solution approaches.

Table 14: The optimality gaps and computational times of heuristics on multi vehicle instances in where compartment numbers range is (2-6)

			Optimality Gap (%)					Time in Minutes				
Instance	Vec. #	Loc #	SC	ISC	RSC	TS	Lower Bound	SC	ISC	RSC	TS	AC-TSPMPLPF
Ins176	3	5	0.00%	0.00%	0.00%	0.00%	-91.47%	0.01	0.01	0.01	0.01	0.02
Ins177	3	5	0.00%	0.00%	0.00%	0.00%	-92.55%	0.01	0.01	0.01	0.01	0.02
Ins178	4	5	0.00%	0.00%	0.00%	0.00%	-95.13%	0.01	0.01	0.01	0.01	0.03
Ins179	4	5	0.00%	0.00%	0.00%	0.00%	-94.50%	0.01	0.01	0.01	0.01	0.03
Ins180	4	5	0.00%	0.00%	0.00%	0.00%	-95.14%	0.01	0.01	0.01	0.01	0.03
Ins181	5	5	0.91%	0.00%	0.00%	0.00%	-92.68%	0.01	0.01	0.01	0.01	0.04
Ins182	5	5	0.00%	0.00%	0.00%	0.00%	-92.49%	0.01	0.01	0.01	0.01	0.04
Ins183	5	10	1.37%	0.11%	0.00%	0.00%	-92.15%	0.06	0.09	0.23	0.33	7.72
Ins184	5	10	1.01%	0.20%	0.00%	0.00%	-91.78%	0.06	0.17	0.44	0.80	7.86
Ins185	9	10	3.04%	0.00%	0.00%	0.00%	-91.26%	0.07	0.11	0.40	0.55	14.32
Ins186	9	10	2.62%	0.00%	0.00%	0.00%	-91.97%	0.06	0.10	0.41	0.58	13.81
Ins187	9	10	0.94%	0.00%	0.00%	0.00%	-95.72%	0.05	0.16	0.71	0.98	13.01
Ins188	10	10	2.32%	0.30%	0.00%	0.00%	-93.20%	0.04	0.10	0.43	0.95	15.43
Ins189	10	10	1.92%	0.00%	0.00%	0.24%	-94.57%	0.04	0.10	0.45	0.60	14.43
Ins190	10	15	1.84%	0.00%	0.00%	0.12%	-93.23%	0.04	0.19	0.43	0.57	22.90
Ins191	11	15	1.52%	0.00%	0.00%	0.00%	-91.20%	0.07	0.31	2.46	3.31	35.23
Ins192	13	15	2.57%	0.54%	0.28%	0.28%	-92.26%	0.07	0.23	2.85	4.15	46.97
Ins193	13	15	1.80%	0.00%	0.00%	0.00%	-92.05%	0.06	0.20	2.31	4.91	40.18
Ins194	14	15	2.12%	0.00%	0.00%	0.52%	-92.53%	0.06	0.20	2.18	2.93	33.03
Ins195	14	15	3.96%	1.23%	0.00%	0.00%	-94.20%	0.07	0.18	2.60	6.40	46.39
Ins196	16	15	4.48%	0.00%	0.00%	0.00%	-93.89%	0.09	0.35	5.52	7.46	92.47
Ins197	11	20	2.33%	0.00%	0.00%	0.00%	-95.56%	0.12	0.39	8.05	10.82	231.72
Ins198	13	20	2.70%	0.14%	0.00%	0.23%	-93.81%	0.13	0.48	7.57	17.49	237.90
Ins199	15	20	4.83%	0.00%	0.00%	0.00%	-94.00%	0.13	0.38	6.62	11.71	274.50
Ins200	16	20	4.33%	0.71%	0.41%	0.71%	-93.85%	0.10	0.43	5.16	9.24	247.66
Ins201	16	20	1.99%	0.00%	0.00%	0.00%	-93.38%	0.10	0.34	6.31	8.94	292.80
Ins202	18	20	5.83%	2.12%	0.71%	2.25%	-95.92%	0.11	0.33	5.59	7.46	329.40
Ins203	19	20	4.02%	0.00%	0.00%	0.00%	-92.72%	0.12	0.37	6.27	9.12	347.70
Ins204	13	25	3.43%	0.00%	0.00%	0.00%	-95.10%	0.14	0.54	10.08	13.27	1,584.93
Ins205	16	25	2.24%	0.00%	0.00%	0.00%	-92.40%	0.16	0.58	11.75	22.18	1,949.37
Ins206	17	25	2.90%	0.85%	0.31%	0.72%	-92.75%	0.13	0.68	12.55	17.19	2,072.29
Ins207	17	25	7.21%	0.00%	0.00%	0.00%	-94.38%	0.14	0.76	11.30	15.44	2,072.15
Ins208	19	25	4.34%	1.23%	0.00%	2.11%	-93.73%	0.14	0.69	9.70	14.40	2,315.95
Ins209	22	25	6.21%	1.31%	1.02%	1.23%	-95.00%	0.14	0.49	8.06	19.29	2,681.45
Ins210	23	25	10.23%	2.01%	0.97%	1.78%	-91.61%	0.14	0.72	9.27	13.83	2,803.21

Table 15: The relative gaps to RSC and computational times of heuristics on multi vehicle instances in where compartment numbers range is (2-10)

			Relative Gap to RSC (%)				Time in Minutes			
Instance	Vec. #	Loc #	SC	ISC	RSC	TS	SC	ISC	RSC	TS
Ins211	3	5	0.00%	0.00%	0.00%	0.00%	0.00	0.00	0.00	0.00
Ins212	3	5	0.00%	0.00%	0.00%	0.00%	0.00	0.00	0.00	0.00
Ins213	3	5	0.00%	0.00%	0.00%	0.00%	0.00	0.00	0.00	0.00
Ins214	4	5	0.00%	0.00%	0.00%	0.00%	0.00	0.00	0.00	0.00
Ins215	5	5	0.00%	0.00%	0.00%	0.00%	0.00	0.00	0.00	0.00
Ins216	7	10	1.76%	0.07%	0.00%	0.00%	0.01	0.02	0.10	0.15
Ins217	8	10	2.65%	0.00%	0.00%	0.00%	0.01	0.02	0.09	0.14
Ins218	8	10	5.11%	2.01%	0.00%	0.91%	0.02	0.02	0.11	0.17
Ins219	9	10	1.81%	0.00%	0.00%	0.00%	0.04	0.07	0.38	0.43
Ins220	9	10	3.76%	0.00%	0.00%	2.73%	0.06	0.07	0.43	0.74
Ins221	16	20	3.33%	0.00%	0.00%	0.00%	0.27	0.88	15.72	15.81
Ins222	17	20	5.09%	1.23%	0.00%	1.92%	0.24	0.88	25.07	18.39
Ins223	18	20	9.67%	2.43%	0.00%	0.00%	0.06	0.20	21.13	26.62
Ins224	19	20	4.82%	0.00%	0.00%	3.91%	0.27	1.08	30.08	30.70
Ins225	20	20	5.04%	0.90%	0.00%	0.00%	0.35	1.08	17.10	32.55
Ins226	16	30	5.02%	1.23%	0.00%	1.93%	0.29	1.35	41.11	37.37
Ins227	19	30	4.12%	0.00%	0.00%	0.00%	0.37	1.49	42.72	43.83
Ins228	26	30	9.12%	3.01%	0.00%	3.73%	0.46	1.78	43.05	44.25
Ins229	27	30	5.12%	0.00%	0.00%	0.00%	0.60	2.12	43.90	45.63
Ins230	29	30	8.02%	2.61%	0.00%	2.72%	0.18	3.55	54.23	45.89
Ins231	26	40	6.23%	0.00%	0.00%	0.00%	0.57	1.62	44.58	46.47
Ins232	32	40	8.23%	2.02%	0.00%	4.03%	0.73	2.82	61.76	48.09
Ins233	34	40	10.01%	4.23%	0.00%	0.00%	0.82	4.32	72.69	57.33
Ins234	34	40	7.23%	0.00%	0.00%	3.23%	0.24	4.95	76.16	57.80
Ins235	39	40	10.92%	3.98%	0.00%	4.21%	0.29	5.13	78.23	69.36
Ins236	32	50	7.23%	0.00%	0.00%	0.00%	0.37	2.98	46.25	75.22
Ins237	34	50	8.12%	3.01%	0.00%	2.04%	0.35	2.93	51.71	79.63
Ins238	37	50	11.23%	3.79%	0.00%	0.00%	0.35	2.93	58.89	79.65
Ins239	43	50	7.23%	1.03%	0.00%	5.07%	0.32	2.18	70.32	82.77
Ins240	50	50	14.82%	4.12%	0.00%	6.03%	0.48	4.75	74.23	93.25

Chapter VI

CONCLUSIONS

In this study, we investigated a distribution system in which geographically dispersed customers request one type of commodity. The supplier delivers these commodities by a heterogeneous fleet of vehicle which ensuring that there is no stock out at customers' side. The commodities must be stored separately in heterogeneous compartments during transportation. Our main objective is to minimize total logistic cost regarding to static replenishment policy. We presented two mathematical formulation for both single and multi vehicle version of problems. Hence our problem is very complex, we would rather solve dividing two sub-problem by commercial solver than solving whole problem. We also develop a heuristic algorithm called Christofides-Greedy Assignment Algorithm (C-GAA) for the single vehicle version problem. For the multi vehicle version of the problem, we proposed two-phase solution approaches.

In the two-phase solution approach, we firstly implement clustering algorithms such as Sweep, Iterative Sweep and Random Subset Clustering Algorithms which classify customer locations into cluster and in the second phase solves a set partitioning problem to determine the best set of clusters among the all generated ones based on total logistic cost calculated by C-GAA algorithm. Furthermore, we also implement Tabu Search Algorithm (TS) for multi vehicle version of the problem in order to compare with two-phase solution approaches in large size of problems.

We test performances of our proposed solution approaches on randomly generated instances with various characteristics. For the single version problem, C-GAA is also very effective regarding to optimality and computational times. For multi vehicle version of the problem, we use enumeration technique in small size problems to get optimal solution because of weakness of the lower bound. Comparison of solution approaches reveals that Random Subset Algorithm has outperformed the other solution approaches in terms of optimality. As a managerial insight, the Iterative Sweep Algorithm is very effective in terms of solution quality regarding to computational times comparing with other solution approaches.

As this is the first work to study on the RILPHVC, no benchmark is available and our results will be examined as benchmark data for further comparison by researchers. There are multiple progression for future studies. Firstly, our problem characteristics could be extended with changing replenishment policy as cyclic, partial delivery, holding cost, customer preferences and so on. Since the complexity of problem is very complex, the future research could focus on formal tightened lower bound or exact algorithm as obvious step regarding to our problem. Finally, the new heuristic approaches or existing heuristics combined in a new metaheuristic frame is a essential path for future researches. It shall be enlightening to see how such algorithm would ensure speed and quality compared to algorithms presented in this thesis.

Appendix A

SUPPLEMENTARY

A.1 Tightened Mathematical Model for Routing and Inventory Loading Problem in Single Vehicle with Compartments

Parameters:

M: Set of compartments,

d_i: Demand rate for customer $i : i \in V \backslash \{0\}$,

q_m: Capacity of compartment $m : m \in M$,

c_{ij}: Non-negatve distance between customer i and j: $i \in V, j \in V$,

B: Sufficiently big number to enforce some constraints non-effective.

Decision Variables:

x_{im}: 1, if customer i is assigned to compartment m ; otherwise 0 : $i \in V \backslash \{0\}, m \in M$.

w_{ij}: 1,if edge (i, j) is used once by vehicle; otherwise 0 : $i, j \in V$.

f: Frequency in static period.

u_i: Auxiliary variable defined for customers i in order to eliminate subtours :$i \in V \backslash \{0\}$.

s_{ij} = auxillary variable for $(w_{ij} * f) : \forall i, j \in V$

t_{im} =auxillary variable for $(x_{im} * f) : \forall i \in V \backslash \{0\}, \forall m \in M$

$$\text{MIP: Min} \quad \sum_{i\in V}\sum_{j\in V}(s_{ij}c_{ij}) \tag{65}$$

s.t.

$$\sum_{i\in V} w_{ij} = 1 \quad \forall i \in V \tag{66}$$

$$\sum_{j\in V} w_{ji} = 1 \quad \forall i \in V \tag{67}$$

$$u_1 = 1 \tag{68}$$

$$2 \leq u_i \leq |V| \quad \forall i \in V\backslash\{0\} \tag{69}$$

$$u_i + u_j + 1 \leq |V|(1 - x_{ij}) \quad \forall i, j \in V\backslash\{0,1\} \tag{70}$$

$$\sum_{i\in V\backslash\{0\}} x_{im} = 1 \quad \forall m \in M \tag{71}$$

$$d_i \leq \sum_{m\in M} t_{im}q_m \quad \forall i \in V\backslash\{0\} \tag{72}$$

$$t_{im} \leq f \quad \forall m \in M, \forall i \in V\backslash\{0\} \tag{73}$$

$$t_{im} \leq B(x_{im}) \quad \forall m \in M, \forall i \in V\backslash\{0\} \tag{74}$$

$$t_{im} \geq f - B(1 - x_{im}) \quad \forall m \in M, \forall i \in V\backslash\{0\} \tag{75}$$

$$s_{ij} \leq f \quad \forall i, j \in V \tag{76}$$

$$s_{ij} \leq Bw_{ij} \quad \forall i, j \in V \tag{77}$$

$$s_{ij} \geq f - B(1 - w_{ij}) \quad \forall i, j \in V \tag{78}$$

$$x_{im} \in \{0, 1\} \quad \forall i \in \backslash\{0\}, \forall m \in M \tag{79}$$

$$w_{ij} \in \{0, 1\} \quad \forall i, j \in V \tag{80}$$

$$f \geq 0, u_i \geq 0 \quad \forall i \in V\backslash\{0\} \tag{81}$$

$$t_{im} \geq 0 \quad \forall i \in V\backslash\{0\}, \forall m \in M \tag{82}$$

$$s_{ij} \geq 0 \quad \forall i, j \in V \tag{83}$$

A.2 Illustration of Greedy Assignment Algorithm

Suppose that there are five customer with demand rates/unit time $d_i = 165, 155, 150, 135, 130$ eleven compartments with capacity $q_m = (1700, 1650, 1525, 1350, 1325, 1250, 1125, 750, 650, 575)$. This is the first run of the multi run process, but randomly assignment process is used to same improvement stages.

Initial Assignment:

X_{1j} = {1,0,0,0,0,0,0,0,1,0,0} with K_1 =2450 f_1 =165 / 2450 =0.06735

X_{2j} = {0,1,0,0,0,0,0,0,0,1,1} with K_2 =2850 f_2 =155 / 2850 =0.05439

X_{3j} = {0,0,1,0,0,0,1,0,0,0,0} with K_3 =2775 f_3 =150 / 2775 =0.05405

X_{4j} = {0,0,0,1,0,1,0,0,0,0,0} with K_4 =2600 f_4 =135 / 2775 =0.05192

X_{5j} = {0,0,0,0,1,0,0,1,0,0,0} with K_5 =2600 f_5 =130 / 2450 =0.05306

f = max {0.06735, 0.05439, 0.05405, 0.05192, 0.05306}

f = 0.06735 which is 1^{st} customer can be improved by changing compartment with other assigned compartment.

Improvement Phase:

To illustrate more precisely, we show only all possible changes in the Step 0 with calculation, forgoing steps we prefer briefly summarize changes.

Step 0:

Possible Changes:

x_{19} can be interchanged with x_{22} and,

F' =max(165/3350, 155/1950, 150/2775, 135/2600, 130/2450);

F'= 0.07949

Δ =F'-F = 0.01214 (increasing)

x_{19} can be interchanged with x_{33} and,

F' =max(165/3225, 155/2850, 150/2000, 135/2600, 130/2450);

F'= 0.07500

Δ =F'-F = 0.00765 (increasing)

x_{19} can be interchanged with x_{37} and,

F' =max(165/2950, 155/2850, 150/2275, 135/2600, 130/2450);

F'= 0.06593

Δ =F'-F = -0.00141 (decreasing)

x_{19} can be interchanged with x_{44} and,

F' =max(165/3050, 155/2850, 150/2775, 135/2000, 130/2450);

F'= 0.06750

Δ =F'-F = 0.00015 (increasing)

x_{19} can be interchanged with x_{46} and,

F' =max(165/2950, 155/2850, 150/2775, 135/2100, 130/2450);

F'= 0.06429

Δ =F'-F =-0.00306 (decreasing)

x_{19} can be interchanged with x_{55} and,

F' =max(165/3025, 155/2850, 150/2775, 135/2600, 130/1875);

F'= 0.06933

Δ =F'-F = 0.00199 (increasing)

x_{19} can be interchanged with x_{58} and,

F' =max(165/2825, 155/2850, 150/2775, 135/2600, 130/2075);

F'= 0.06265

Δ =F'-F =-0.00470 (decreasing)

The best improvement x_{19} with x_{58} and after changing we checked if sorting algorithm leads better improvement as follow: K_i = {2825,2850,2775,2600,2075} sorting as K_i' = {2850,2825,2575,2600,2075} does not lead improvement on Δ or decreasing in f. Therefore, we can iterate next step.

Step 1:

f = 0.06265 which is 5^{st} customer can be improved by changing compartment with other assigned compartment.

Best improvement on f is changing x_{55} with x_{33}. The new objective value is 0.005840.

After applied changes made, sum of compartments to each customers as follow:

K_i = {2825,2850,2575,2600,2275} sorting as K_i' = {2850,2825,2600,2575,2075} lead improvement Δ= 0.00051 and the new objective value is 0.057894.

Step 2:

f=0.057894 which is 1^{st} customer can be improved by changing compartment with other assigned compartment.

Best improvement on f is changing x_{11} with x_{21}. The new objective value is 0.005769.

We checked if sorting algorithm leads better improvement as follow:

K_i = {2900,2775,2600,2575,2275} which is ordered by descending no sorting is needed.Therefore, we can iterate next step.

Step 3:

f=0.0.005769 which is 3^{st} customer can be improved by changing compartment with other assigned compartment.

Best improvement on f is changing x_{36} with x_{45}. The new objective value is 0.005714.

We checked if sorting algorithm leads better improvement as follow:

K_i = {2900,2775,2675,2500,2275} which is ordered by descending no sorting is needed.Therefore, we can iterate next step.

Step 4:

f=0.005714 which is 5^{st} customer can be improved by changing compartment with other assigned compartment. There is no improvement changing 5^{st} compartment with other customer comperments. Since Δ does not change decreasing way our algorithm terminates iteration.

A.3 Empirical Study on Reduction Strategy

#	Objective Value of Reduction Strategy	Best Objective Value of All Possible Candidates	Neighborhood Size of All Possible Candidates	Neighborhood Size of Reduction Strategy
1	2,869	2,869	114	55
2	2,739	2,739	113	54
3	1,660	1,660	134	64
4	3,491	3,357	131	63
5	3,276	3,276	106	51
6	3,901	3,901	135	65
7	1,645	1,552	134	64
8	3,586	3,586	103	49
9	2,507	2,507	127	61
10	4,488	4,488	143	69
11	1,960	1,960	111	53
12	4,245	4,245	127	61
13	1,694	1,645	150	72
14	3,509	3,509	141	68
15	2,415	2,415	108	52
16	3,614	3,614	146	70
17	3,622	3,622	129	62
18	1,896	1,806	109	52
19	3,363	3,363	109	52
20	4,371	4,047	106	51
21	2,979	2,979	135	65
22	3,201	3,201	106	51
23	2,908	2,693	144	69
24	2,385	2,385	122	59
25	2,140	2,140	128	61
26	2,986	2,986	117	56
27	3,030	3,030	107	51
28	4,377	4,377	117	56
29	4,368	4,368	143	69
30	1,985	1,927	141	68
31	2,490	2,490	111	53
32	3,475	3,475	120	58
33	1,557	1,557	110	53
34	2,364	2,364	131	63
35	2,564	2,564	148	71
36	3,430	3,430	116	56
37	3,239	3,239	104	50
38	4,324	4,324	103	49
39	2,144	2,144	112	54
40	1,874	1,802	144	69
41	3,622	3,622	148	71
42	3,527	3,527	121	58
43	2,757	2,757	115	55
44	4,136	4,136	108	52
45	2,011	2,011	127	61
46	3,475	3,475	116	56
47	2,348	2,348	127	61
48	4,144	4,144	110	53
49	2,978	2,978	125	60

#	Objective Value of Reduction Strategy	Best Objective Value of All Possible Candidates	Neighborhood Size of All Possible Candidates	Neighborhood Size of Reduction Strategy
50	4,244	4,244	145	70
51	4,278	4,278	146	70
52	1,648	1,600	109	52
53	2,696	2,696	114	55
54	4,000	4,000	102	49
55	1,818	1,818	128	61
56	3,987	3,987	138	66
57	3,180	3,180	136	65
58	2,654	2,654	119	57
59	4,462	4,462	117	56
60	3,718	3,508	112	54
61	2,220	2,220	133	64
62	3,361	3,361	111	53
63	2,614	2,614	107	51
64	2,653	2,653	147	71
65	1,779	1,779	126	60
66	4,311	4,311	126	60
67	1,898	1,898	123	59
68	3,860	3,860	123	59
69	1,716	1,716	137	66
70	1,951	1,951	131	63
71	2,816	2,607	120	58
72	2,460	2,460	135	65
73	2,355	2,355	120	58
74	1,836	1,836	131	63
75	2,551	2,551	104	50
76	3,405	3,405	139	67
77	4,024	4,024	127	61
78	1,889	1,889	131	63
79	2,379	2,266	147	71
80	2,890	2,890	109	52
81	3,219	3,219	127	61
82	3,432	3,432	111	53
83	4,008	4,008	140	67
84	3,353	3,353	135	65
85	3,336	3,336	147	71
86	3,511	3,511	132	63
87	2,351	2,351	128	61
88	2,174	2,111	117	56
89	3,957	3,957	112	54
90	2,632	2,632	109	52
91	3,855	3,855	125	60
92	3,830	3,830	121	58
93	2,653	2,503	135	65
94	2,845	2,845	113	54
95	3,493	3,493	140	67
96	4,289	4,289	140	67
97	1,595	1,595	136	65
98	3,046	3,046	101	48
99	1,591	1,591	112	54
100	3,453	3,453	120	58
Average	**3,000**	**2,981**	**125**	**60**

A.4 Empirical Study on Local Search

This emprical study is based on one instance which has 30 customer with 15 vehicles with compartments number varying between 2-5. Demand rates and compartments capacities are randomly generated betweeen (1000-3000).

The Objective Value Changes through Iterations		
Iteration Number	***TS***	***TS with Local Search***
1	6,000	6,000
2	5,907	5,907
3	5,826	5,826
4	5,799	5,799
5	5,658	5,658
6	5,609	5,609
7	5,528	5,528
8	5,484	5,484
9	5,484	5,484
10	5,434	5,434
11	5,389	5,389
12	5,357	5,357
13	5,338	5,338
14	5,309	5,309
15	5,290	5,290
16	5,262	5,262
17	5,227	5,227
18	5,223	5,223
19	5,204	5,204
20	5,193	5,193
21	5,193	5,193
22	5,114	5,114
23	5,091	5,091
24	5,063	5,063
25	5,056	5,056
26	5,020	5,020
27	4,985	4,985
28	4,968	4,968
29	4,949	4,949
30	4,889	4,889
31	4,805	4,805
32	4,752	4,752
33	4,689	4,689
34	4,615	4,615
35	4,527	4,527
36	4,441	4,441
37	4,408	4,408
38	4,363	4,363
39	4,360	4,360
40	4,359	4,359
41	4,286	4,286
42	4,207	4,207
43	4,199	4,199
44	4,163	4,163
45	4,117	4,117
46	4,082	4,082
47	4,061	4,061
48	4,058	4,058
49	4,051	4,051
50	4,048	4,048

The Objective Value Changes through Iterations		
Iteration Number	***TS***	***TS with Local Search***
51	4,039	3,816
52	4,036	3,808
53	4,033	3,787
54	4,026	3,787
55	4,021	3,787
56	4,020	3,787
57	4,017	3,787
58	4,012	3,787
59	4,003	3,787
60	3,998	3,787
61	3,994	3,787
62	3,992	3,787
63	3,986	3,787
64	3,983	3,787
65	3,983	3,787
66	3,974	3,787
67	3,964	3,787
68	3,955	3,787
69	3,950	3,787
70	3,946	3,787
71	3,943	3,787
72	3,943	3,787
73	3,943	3,787
74	3,943	3,787
75	3,943	3,787
76	3,943	3,787
77	3,943	3,787
78	3,943	3,787
79	3,943	3,787
80	3,943	3,787
81	3,943	3,787
82	3,943	3,787
83	3,943	3,787
84	3,943	3,787
85	3,943	3,787
86	3,943	3,787
87	3,943	3,787
88	3,943	3,787
89	3,943	3,787
90	3,943	3,787
91	3,943	3,787
92	3,943	3,787
93	3,943	3,787
94	3,943	3,787
95	3,943	3,787
96	3,943	3,787
97	3,943	3,787
98	3,943	3,787
99	3,943	3,787
100	3,943	3,787

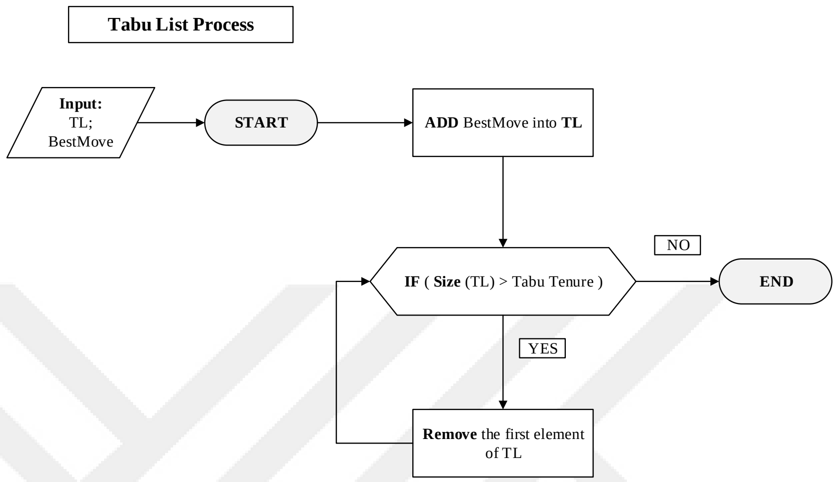
A.5 *Empirical Study on Reactive Mechanism*

This emprical study is based on instances which have 30 customer with 15 vehicles with compartments number varying between 2-5. Demand rates and compartments capacities are randomly generated betweeen (1000-3000).

Instance No	TS Objective Value	R-TS Objective Value	GAP
1	3,493.21	3,470.30	1%
2	1,545.91	1,537.74	1%
3	4,227.05	3,955.20	6%
4	2,104.56	2,036.67	3%
5	1,477.48	1,358.10	8%
6	3,874.97	3,803.11	2%
7	2,716.15	2,534.03	7%
8	3,938.93	3,705.26	6%
9	3,595.59	3,595.59	0%
10	1,980.16	1,909.91	4%
11	3,448.16	3,429.73	1%
12	2,560.46	2,555.65	0%
13	3,325.21	2,993.83	10%
14	1,933.26	1,913.67	1%
15	3,037.68	2,916.73	4%
16	4,416.93	4,315.76	2%
17	1,843.03	1,828.81	1%
18	4,315.71	4,239.92	2%
19	3,141.30	2,856.43	9%
20	3,589.28	3,338.85	7%
21	3,042.74	2,917.28	4%
22	3,750.12	3,631.88	3%
23	2,947.92	2,758.39	6%
24	4,276.93	4,053.65	5%
25	3,262.24	2,986.36	8%
26	3,146.00	3,062.00	3%
27	3,836.66	3,821.81	0%
28	2,973.51	2,872.81	3%
29	1,980.63	1,955.84	1%
30	3,788.67	3,680.42	3%

Average	**3.71%**
Stand Dev,	**2.90%**
Minimum	**0.00%**
1st Quartile	**0.95%**
Median	**3.19%**
3rd Quartile	**6.43%**
Maximum	**9.97%**

A.6 The Flowchart of Neighborhood Exploration and Reactive Mechanism



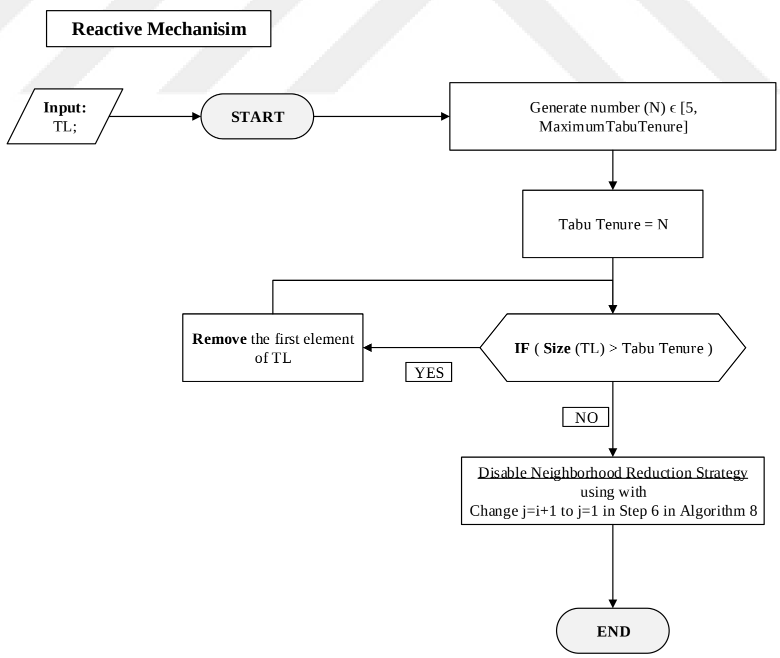
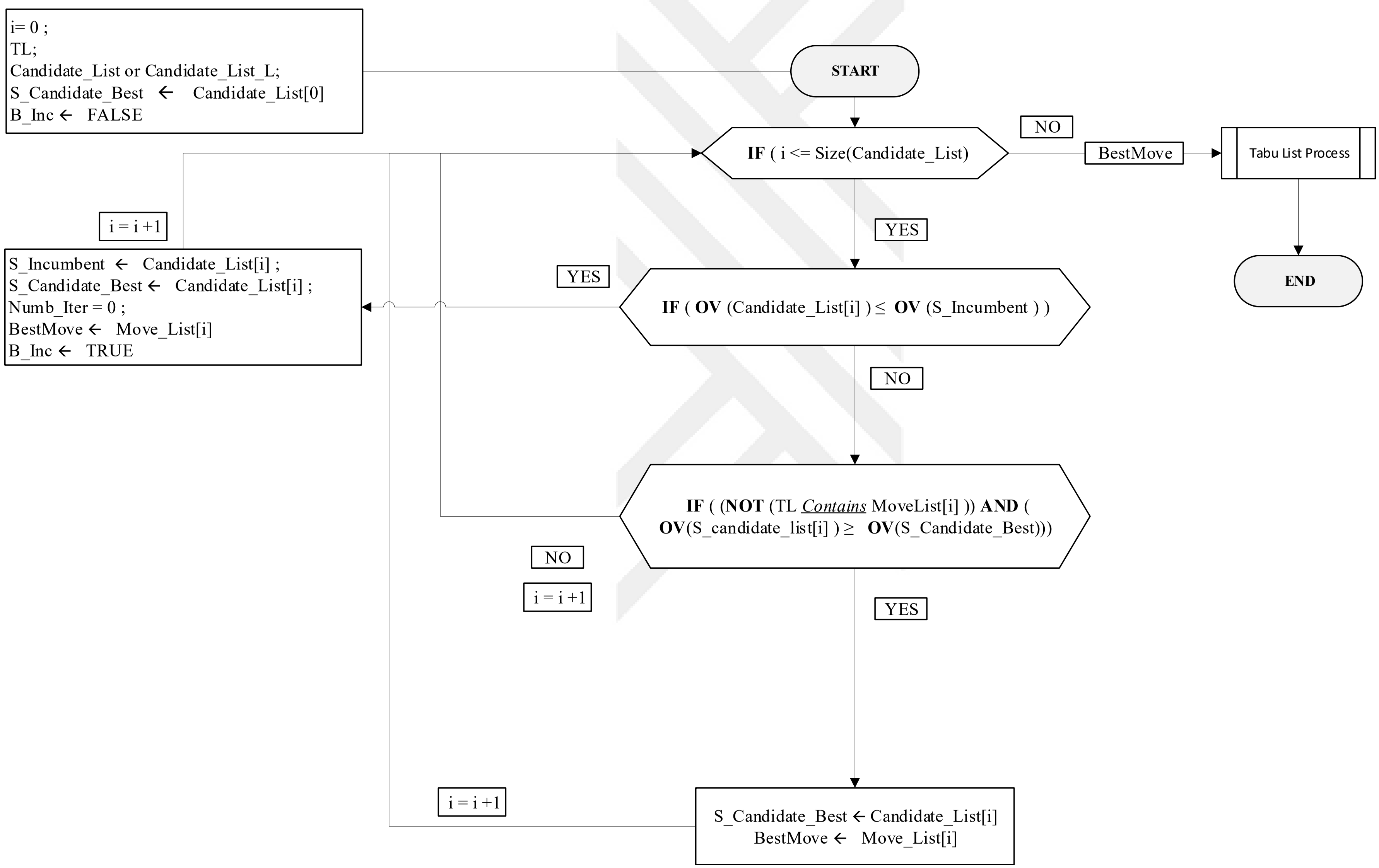


Figure 10: The Flowchart of Neighborhood Exploration



Appendix B

RESULTS

B.1 Single and Multi Vehicle(s) Version of Problem Results

Table 16: The optimality gaps and computational times of C-GAA on single vehicle instances with less varied demands

		Optimality Gap(%)			Time in Seconds		
Instance	Loc #	C-GAA	TSPMPLPF	Lower Bound	C-GAA	TSPMPLPF	Lower Bound
Ins 31	5	0.00%	0.00%	-36.06%	0.03	0.34	0.00
Ins 32	5	0.00%	0.00%	-25.37%	0.02	0.37	0.00
Ins 33	5	0.00%	0.00%	-33.22%	0.02	0.30	0.00
Ins 34	5	0.00%	0.00%	-51.67%	0.02	0.08	0.00
Ins 35	5	0.00%	0.00%	-33.94%	0.02	0.23	0.00
Ins 36	10	0.00%	0.00%	-24.47%	0.01	1.95	0.00
Ins 37	10	0.00%	0.00%	-22.90%	0.01	0.89	0.00
Ins 38	10	2.19%	0.00%	-22.21%	0.01	20.90	0.00
Ins 39	10	0.00%	0.00%	-36.26%	0.00	0.43	0.00
Ins 40	10	0.15%	0.00%	-26.73%	0.01	1.46	0.00
Ins 41	20	8.90%	3.46%	-22.45%	0.06	10,946.57	0.00
Ins 42	20	2.93%	0.64%	-19.11%	0.11	10,806.61	0.00
Ins 43	20	6.40%	4.50%	-20.73%	0.06	11,630.27	0.02
Ins 44	20	2.71%	1.72%	-20.43%	0.12	10,908.15	0.01
Ins 45	20	3.40%	0.00%	-30.10%	0.18	4,003.09	0.00
Ins 46	30	4.80%	3.90%	-21.20%	0.10	11,642.72	0.02
Ins 47	30	3.67%	0.00%	-16.90%	0.06	1,099.51	0.02
Ins 48	30	7.12%	5.40%	-17.23%	0.11	11,512.61	0.03
Ins 49	30	11.20%	7.12%	-33.80%	0.09	11,119.53	0.01
Ins 50	30	5.30%	3.24%	-29.32%	0.12	10,975.83	0.02
Ins 51	40	11.24%	6.91%	-16.23%	0.14	11,874.41	0.02
Ins 52	40	6.30%	1.90%	-14.12%	0.12	13,365.97	0.02
Ins 53	40	3.10%	2.81%	-23.65%	0.21	13,693.30	0.02
Ins 54	40	9.43%	4.98%	-19.94%	0.14	11,745.82	0.03
Ins 55	40	13.23%	9.63%	-21.91%	0.22	11,353.52	0.03
Ins 56	50	16.23%	11.41%	-27.28%	0.13	12,143.52	0.01
Ins 57	50	6.21%	3.54%	-13.67%	0.24	12,950.13	0.02
Ins 58	50	7.62%	4.51%	-17.21%	0.21	12,262.13	0.03
Ins 59	50	8.16%	3.68%	-12.96%	0.30	17,160.80	0.02
Ins 60	50	13.24%	8.10%	-11.69%	0.30	12,887.05	0.01

Table 17: The optimality gaps and computational times of C-GAA on single vehicle instances with more varied demands

		Optimality Gap(%)			Time in Seconds		
Instance	Loc #	C-GAA	TSPMPLPF	Lower Bound	C-GAA	TSPMPLPF	Lower Bound
Ins 61	5	0.00%	0.00%	-34.86%	0.00	0.12	0.00
Ins 62	5	0.00%	0.00%	-42.13%	0.03	0.12	0.00
Ins 63	5	0.00%	0.00%	-35.95%	0.00	0.25	0.00
Ins 64	5	0.00%	0.00%	-41.63%	0.00	0.27	0.00
Ins 65	5	0.00%	0.00%	-32.72%	0.02	0.11	0.00
Ins 66	10	0.00%	0.00%	-45.19%	0.10	1.60	0.00
Ins 67	10	0.00%	0.00%	-39.65%	0.01	1.27	0.00
Ins 68	10	1.96%	0.00%	-28.21%	0.02	4.00	0.00
Ins 69	10	0.00%	0.00%	-33.40%	0.00	0.81	0.00
Ins 70	10	0.00%	0.00%	-39.39%	0.00	0.71	0.00
Ins 71	20	8.35%	3.91%	-22.25%	0.01	10,938.66	0.02
Ins 72	20	7.20%	4.62%	-25.91%	0.19	11,488.49	0.01
Ins 73	20	4.00%	0.00%	-34.06%	0.08	2,004.34	0.00
Ins 74	20	3.38%	0.00%	-27.37%	0.06	1,009.52	0.00
Ins 75	20	2.34%	1.65%	-27.01%	0.12	11,196.66	0.00
Ins 76	30	9.23%	7.34%	-31.24%	0.18	10,897.56	0.02
Ins 77	30	7.13%	3.74%	-21.43%	0.07	11,509.41	0.02
Ins 78	30	5.25%	1.95%	-26.91%	0.12	10,956.66	0.03
Ins 79	30	5.32%	3.17%	-19.87%	0.13	10,421.06	0.02
Ins 80	30	6.90%	2.32%	-14.91%	0.14	11,805.19	0.02
Ins 81	40	6.34%	2.54%	-22.12%	0.15	13,501.17	0.04
Ins 82	40	7.24%	4.14%	-19.81%	0.15	11,693.35	0.02
Ins 83	40	8.63%	6.31%	-15.87%	0.12	13,321.02	0.01
Ins 84	40	14.21%	10.12%	-24.17%	0.15	13,754.60	0.02
Ins 85	40	8.33%	8.12%	-18.15%	0.13	11,654.56	0.03
Ins 86	50	11.56%	7.01%	-21.53%	0.18	11,554.58	0.02
Ins 87	50	5.78%	2.87%	-12.78%	0.15	10,632.91	0.01
Ins 88	50	8.54%	5.31%	-13.65%	0.14	12,839.49	0.03
Ins 89	50	18.21%	13.21%	-16.31%	0.20	10,456.31	0.02
Ins 90	50	9.23%	6.94%	-15.32%	0.17	13,291.00	0.04

Table 18: The optimality gaps and computational times of C-GAA on single vehicle instances with less varied compartment capacity

		Optimality Gap(%)			Time in Seconds		
Instance	Loc #	C-GAA	TSPMPLPF	Lower Bound	C-GAA	TSPMPLPF	Lower Bound
Ins 91	5	0.00%	0.00%	-38.41%	0.02	0.12	0.00
Ins 92	5	0.00%	0.00%	-33.58%	0.02	0.14	0.00
Ins 93	5	0.00%	0.00%	-49.72%	0.02	0.11	0.00
Ins 94	5	0.00%	0.00%	-33.29%	0.00	0.16	0.00
Ins 95	5	0.00%	0.00%	-37.38%	0.02	0.12	0.00
Ins 96	10	0.00%	0.00%	-32.34%	0.01	1.96	0.00
Ins 97	10	0.00%	0.00%	-51.27%	0.01	0.62	0.00
Ins 98	10	0.00%	0.00%	-24.03%	0.01	1.17	0.00
Ins 99	10	0.00%	0.00%	-42.24%	0.01	0.59	0.00
Ins 100	10	2.04%	0.00%	-38.91%	0.00	0.62	0.00
Ins 101	20	9.12%	4.67%	-20.44%	0.01	10,846.27	0.00
Ins 102	20	2.65%	0.92%	-17.84%	0.12	11,221.12	0.00
Ins 103	20	3.78%	0.00%	-18.70%	0.09	4,856.22	0.02
Ins 104	20	1.80%	1.48%	-14.80%	0.14	11,564.55	0.01
Ins 105	20	8.80%	3.21%	-14.24%	0.09	11,472.94	0.00
Ins 106	30	2.75%	1.90%	-17.45%	0.17	11,981.88	0.02
Ins 107	30	9.25%	4.22%	-14.21%	0.19	10,750.10	0.02
Ins 108	30	6.85%	3.14%	-15.24%	0.14	11,803.26	0.03
Ins 109	30	12.54%	8.47%	-18.32%	0.10	11,335.74	0.02
Ins 110	30	3.17%	1.73%	-16.31%	0.10	11,441.58	0.02
Ins 111	40	9.75%	8.31%	-19.43%	0.17	10,856.12	0.03
Ins 112	40	10.24%	7.31%	-18.22%	0.19	12,401.15	0.02
Ins 113	40	7.02%	6.42%	-17.42%	0.19	12,551.07	0.01
Ins 114	40	4.67%	2.12%	-14.32%	0.20	13,302.77	0.02
Ins 115	40	12.33%	9.25%	-21.03%	0.18	10,334.30	0.03
Ins 116	50	11.79%	5.91%	-20.80%	0.24	10,167.81	0.03
Ins 117	50	14.68%	12.94%	-19.21%	0.19	14,548.26	0.02
Ins 118	50	8.71%	4.12%	-17.63%	0.21	13,898.60	0.04
Ins 119	50	5.80%	2.16%	-19.27%	0.21	10,835.32	0.05
Ins 120	50	9.93%	7.42%	-23.02%	0.18	14,179.26	0.03

Table 19: The optimality gaps and computational times of C-GAA on single vehicle instances with more varied compartment capacity

		Optimality Gap(%)			Time in Seconds		
Instance	Loc #	C-GAA	TSPMPLPF	Lower Bound	C-GAA	TSPMPLPF	Lower Bound
Ins 121	5	0.00%	0.00%	-37.23%	0.01	0.12	0.00
Ins 122	5	0.00%	0.00%	-37.31%	0.02	0.13	0.00
Ins 123	5	0.00%	0.00%	-35.76%	0.01	0.18	0.00
Ins 124	5	0.00%	0.00%	-38.74%	0.00	0.12	0.00
Ins 125	5	0.00%	0.00%	-34.34%	0.02	0.15	0.00
Ins 126	10	0.00%	0.00%	-32.99%	0.01	1.77	0.00
Ins 127	10	0.00%	0.00%	-33.24%	0.01	1.08	0.00
Ins 128	10	0.73%	0.00%	-26.12%	0.01	2.59	0.00
Ins 129	10	0.00%	0.00%	-37.40%	0.01	0.64	0.00
Ins 130	10	1.56%	0.00%	-38.00%	0.01	0.83	0.00
Ins 131	20	8.62%	4.35%	-21.35%	0.03	10,942.62	0.01
Ins 132	20	3.91%	1.85%	-18.47%	0.12	11,130.55	0.01
Ins 133	20	1.35%	0.00%	-27.40%	0.07	9,130.55	0.01
Ins 134	20	3.05%	0.00%	-23.90%	0.09	7,930.55	0.01
Ins 135	20	6.10%	3.21%	-25.27%	0.11	11,130.55	0.00
Ins 136	30	5.00%	3.20%	-20.07%	0.13	11,270.14	0.02
Ins 137	30	5.40%	0.00%	-18.67%	0.07	8,221.11	0.02
Ins 138	30	6.99%	4.02%	-22.07%	0.11	11,339.56	0.03
Ins 139	30	10.54%	6.71%	-22.38%	0.09	11,138.90	0.02
Ins 140	30	5.25%	5.04%	-22.70%	0.11	11,208.70	0.02
Ins 141	40	8.08%	4.73%	-17.88%	0.15	11,761.90	0.03
Ins 142	40	6.77%	3.08%	-16.17%	0.13	12,047.25	0.02
Ins 143	40	6.33%	4.56%	-16.65%	0.15	12,936.04	0.01
Ins 144	40	8.07%	3.55%	-17.13%	0.16	12,524.29	0.02
Ins 145	40	12.78%	9.44%	-19.59%	0.15	11,504.04	0.03
Ins 146	50	11.67%	6.46%	-21.17%	0.15	11,849.05	0.02
Ins 147	50	6.26%	3.57%	-13.39%	0.17	12,824.35	0.02
Ins 148	50	8.81%	4.91%	-15.43%	0.17	13,124.85	0.03
Ins 149	50	7.44%	3.19%	-14.64%	0.20	11,363.66	0.02
Ins 150	50	17.03%	12.84%	-14.37%	0.20	13,735.13	0.02

Table 20: The optimality gaps and computational times of heuristics on multi vehicle instances in where compartment numbers range is (2-5) with less varied demand

			Optimality Gap (%)					Time in Seconds				
Instance	Vec. #	Loc #	SC	ISC	RSC	TS	Lower Bound	SC	ISC	RSC	TS	AC-TSPMPLPF
Ins36	3	5	0.00%	0.00%	0.00%	0.00%	-91.30%	0.09	0.09	0.13	0.18	1.98
Ins37	4	5	0.00%	0.00%	0.00%	0.00%	-87.60%	0.08	0.10	0.13	0.41	1.81
Ins38	5	5	0.00%	0.00%	0.00%	0.00%	-92.16%	0.07	0.11	0.14	0.38	1.53
Ins39	5	5	0.00%	0.00%	0.00%	0.00%	-95.34%	0.08	0.13	0.15	0.17	1.94
Ins40	5	5	0.00%	0.00%	0.00%	0.00%	-92.68%	0.06	0.10	0.15	0.48	1.19
Ins41	5	5	0.00%	0.00%	0.00%	0.00%	-86.21%	0.07	0.11	0.15	0.30	1.60
Ins42	5	5	0.00%	0.00%	0.00%	0.00%	-95.17%	0.09	0.14	0.17	0.34	1.31
Ins43	6	10	1.67%	0.00%	0.00%	0.00%	-92.24%	0.35	1.31	1.08	1.63	36.37
Ins44	7	10	1.23%	0.00%	0.00%	0.00%	-92.66%	0.42	1.60	1.62	1.70	41.19
Ins45	7	10	2.60%	0.00%	0.00%	0.00%	-85.34%	0.45	1.22	1.38	2.72	54.74
Ins46	7	10	2.28%	0.00%	0.00%	0.00%	-94.03%	0.29	0.88	1.35	1.47	57.42
Ins47	8	10	0.64%	0.00%	0.00%	0.00%	-91.95%	0.33	1.29	1.21	1.53	64.89
Ins48	8	10	3.02%	0.00%	0.00%	0.00%	-85.07%	0.31	1.30	1.07	3.23	85.99
Ins49	9	10	2.09%	0.00%	0.00%	0.00%	-89.45%	0.61	0.90	1.08	3.71	33.47
Ins50	8	15	2.76%	0.06%	0.00%	0.00%	-85.51%	0.57	2.03	4.92	10.38	313.05
Ins51	10	15	0.92%	0.00%	0.00%	0.00%	-85.49%	0.83	2.05	8.62	12.93	394.24
Ins52	11	15	3.14%	0.90%	0.00%	0.41%	-93.40%	1.18	4.12	8.87	20.92	757.20
Ins53	12	15	1.89%	0.00%	0.00%	0.00%	-93.39%	1.31	4.20	9.43	17.53	553.02
Ins54	12	15	1.41%	0.00%	0.00%	0.00%	-86.63%	1.40	4.54	12.57	27.96	687.71
Ins55	12	15	1.78%	0.42%	0.00%	0.00%	-91.72%	1.66	4.98	13.12	37.85	359.32
Ins56	14	15	2.38%	0.00%	0.00%	0.00%	-93.31%	1.76	5.63	13.19	27.34	455.21
Ins57	13	20	2.09%	0.00%	0.00%	0.00%	-90.01%	2.35	6.42	17.87	31.14	3,134.02
Ins58	13	20	2.07%	1.02%	0.00%	0.00%	-91.98%	2.44	6.53	19.97	29.89	2,985.41
Ins59	17	20	1.71%	0.00%	0.00%	1.09%	-89.52%	2.51	6.86	23.33	42.74	2,221.96
Ins60	17	20	2.30%	0.00%	0.00%	0.00%	-93.58%	3.22	6.94	25.84	49.13	2,357.04
Ins61	20	20	2.87%	0.75%	0.42%	0.00%	-91.76%	3.28	7.13	26.54	44.33	2,301.17
Ins62	20	20	3.12%	0.00%	0.00%	0.00%	-90.44%	3.83	7.34	31.11	51.18	3,641.69
Ins63	20	20	4.06%	1.45%	0.00%	0.00%	-90.37%	3.99	7.65	30.17	78.16	2,908.47
Ins64	13	25	1.97%	0.00%	0.00%	0.00%	-89.81%	1.60	11.73	105.21	240.93	6,952.19
Ins65	15	25	3.56%	0.96%	0.00%	0.00%	-93.65%	3.35	10.78	81.16	126.28	8,507.76
Ins66	20	25	3.97%	1.09%	0.00%	0.99%	-93.15%	3.48	12.59	103.23	153.61	11,153.68
Ins67	21	25	3.17%	0.00%	0.00%	0.35%	-92.74%	5.88	21.49	115.77	159.18	10,846.46
Ins68	22	25	1.46%	0.00%	0.00%	0.00%	-95.18%	5.95	27.42	181.77	266.12	12,576.25
Ins69	25	25	7.02%	2.32%	0.67%	1.57%	-92.87%	7.66	29.41	260.90	577.36	13,017.60
Ins70	25	25	2.35%	0.00%	0.00%	0.00%	-92.78%	10.25	42.21	371.22	970.73	12,624.38

Table 21: The optimality gaps and computational times of heuristics on multi vehicle instances in where compartment numbers range is (2-5) with more varied demand

			Optimality Gap (%)					Time in Seconds				
Instance	Vec. #	Loc #	SC	ISC	RSC	TS	Lower Bound	SC	ISC	RSC	TS	AC-TSPMPLPF
Ins71	3	5	0.00%	0.00%	0.00%	0.00%	-85.19%	0.09	0.09	0.13	0.16	1.72
Ins72	3	5	0.00%	0.00%	0.00%	0.00%	-87.01%	0.08	0.10	0.17	0.41	1.70
Ins73	4	5	0.00%	0.00%	0.00%	0.00%	-84.00%	0.07	0.11	0.13	0.38	1.42
Ins74	4	5	0.00%	0.00%	0.00%	0.00%	-76.80%	0.08	0.13	0.16	0.17	1.49
Ins75	4	5	0.00%	0.00%	0.00%	0.00%	-76.55%	0.06	0.10	0.15	0.48	1.66
Ins76	6	5	0.00%	0.00%	0.00%	0.00%	-72.31%	0.07	0.11	0.14	0.30	1.66
Ins77	6	5	0.00%	0.00%	0.00%	0.00%	-78.43%	0.09	0.14	0.13	0.34	1.64
Ins78	6	10	1.45%	0.00%	0.00%	0.00%	-90.71%	0.46	0.81	1.18	1.63	35.65
Ins79	6	10	2.60%	0.00%	0.00%	0.00%	-87.73%	0.57	1.20	1.52	1.70	42.57
Ins80	7	10	1.74%	0.00%	0.00%	0.00%	-86.57%	0.43	0.85	0.91	2.72	59.90
Ins81	8	10	0.90%	0.00%	0.00%	0.00%	-90.44%	0.60	1.28	0.92	1.47	85.36
Ins82	8	10	0.00%	0.00%	0.00%	0.00%	-87.40%	0.60	1.21	1.51	1.53	42.72
Ins83	10	10	1.28%	0.00%	0.00%	0.00%	-87.48%	0.62	0.63	0.90	3.23	43.63
Ins84	10	10	3.60%	0.00%	0.00%	0.00%	-85.05%	0.53	1.42	1.18	3.71	63.92
Ins85	8	15	4.81%	1.31%	0.00%	0.00%	-88.39%	1.53	7.41	7.55	21.21	368.15
Ins86	9	15	1.81%	0.67%	0.00%	0.00%	-88.98%	2.09	1.37	9.45	13.20	459.55
Ins87	11	15	3.56%	0.00%	0.00%	0.48%	-85.18%	1.47	3.22	11.24	17.26	532.55
Ins88	12	15	1.63%	0.00%	0.00%	0.00%	-88.19%	1.69	7.16	14.61	22.46	619.55
Ins89	13	15	0.80%	0.15%	0.00%	0.38%	-89.27%	1.47	3.32	17.54	34.25	581.55
Ins90	14	15	0.63%	0.00%	0.00%	0.00%	-85.04%	1.52	3.56	18.55	50.66	605.42
Ins91	16	15	3.72%	0.94%	0.00%	0.00%	-86.30%	1.26	2.64	16.85	23.53	677.00
Ins92	11	20	0.89%	0.00%	0.00%	0.00%	-90.07%	3.49	8.68	23.78	55.89	2,708.01
Ins93	12	20	3.60%	0.84%	0.00%	0.00%	-85.53%	1.89	6.05	19.55	55.10	2,857.28
Ins94	12	20	3.97%	0.98%	0.00%	0.00%	-88.53%	3.15	7.21	25.00	35.05	2,912.19
Ins95	14	20	2.87%	1.29%	0.24%	0.71%	-89.34%	2.70	8.11	20.48	39.14	3,171.73
Ins96	14	20	2.60%	0.00%	0.00%	0.00%	-89.92%	3.06	7.61	28.13	39.74	3,275.58
Ins97	19	20	2.54%	0.75%	0.00%	0.85%	-88.20%	6.15	8.56	35.68	57.44	3,332.63
Ins98	20	20	2.97%	0.00%	0.00%	0.00%	-85.59%	3.02	8.44	31.57	67.51	3,471.55
Ins99	14	25	3.05%	0.00%	0.00%	0.00%	-88.03%	3.38	22.16	101.55	224.62	9,705.15
Ins100	15	25	2.76%	0.00%	0.00%	0.00%	-88.98%	6.35	19.31	114.55	321.08	9,871.54
Ins101	15	25	3.69%	1.23%	0.75%	1.22%	-85.38%	6.18	32.66	265.78	451.03	10,230.55
Ins102	15	25	1.22%	0.00%	0.00%	0.00%	-85.75%	4.56	13.77	107.64	239.29	10,853.55
Ins103	17	25	3.22%	0.34%	0.00%	0.00%	-85.11%	4.92	24.29	159.47	215.60	11,139.73
Ins104	18	25	3.97%	1.14%	0.00%	0.22%	-92.15%	2.84	16.46	323.12	854.98	12,865.67
Ins105	21	25	8.02%	1.51%	0.47%	0.96%	-91.50%	7.23	24.06	278.25	450.48	10,837.19

Table 22: The optimality gaps and computational times of heuristics on multi vehicle instances in where compartment numbers range is (2-5) with less varied compartment capacities

			Optimality Gap (%)					Time in Seconds				
Instance	Vec. #	Loc #	SC	ISC	RSC	TS	Lower Bound	SC	ISC	RSC	TS	AC-TSPMPLPF
Ins106	3	5	0.00%	0.00%	0.00%	0.00%	-93.23%	0.08	0.12	0.15	0.29	1.14
Ins107	4	5	1.05%	0.00%	0.00%	0.00%	-92.49%	0.07	0.12	0.14	0.24	1.24
Ins108	4	5	0.00%	0.00%	0.00%	0.00%	-92.68%	0.07	0.13	0.16	0.26	1.29
Ins109	4	5	0.00%	0.00%	0.00%	0.00%	-90.54%	0.11	0.12	0.16	0.36	1.34
Ins110	5	5	0.00%	0.00%	0.00%	0.00%	-92.72%	0.08	0.13	0.17	0.50	1.34
Ins111	5	5	0.00%	0.00%	0.00%	0.00%	-93.98%	0.07	0.14	0.13	0.15	1.75
Ins112	5	5	0.00%	0.00%	0.00%	0.00%	-93.32%	0.14	0.13	0.14	0.45	1.82
Ins113	7	10	1.83%	0.00%	0.00%	0.00%	-95.57%	0.52	0.80	1.32	5.07	39.22
Ins114	8	10	0.44%	0.00%	0.00%	0.00%	-95.18%	0.59	0.67	1.59	2.04	40.85
Ins115	8	10	3.41%	0.00%	0.00%	0.00%	-92.28%	0.60	1.18	0.95	1.65	48.04
Ins116	8	10	0.00%	0.00%	0.00%	0.00%	-92.71%	0.62	1.40	1.04	1.50	56.95
Ins117	9	10	1.22%	0.00%	0.00%	0.00%	-92.96%	0.54	0.72	1.36	2.40	59.33
Ins118	11	10	0.94%	0.00%	0.00%	0.00%	-92.48%	0.57	0.65	1.38	3.60	65.37
Ins119	11	10	2.38%	0.00%	0.00%	0.00%	-95.66%	0.50	0.86	1.43	4.09	85.99
Ins120	8	15	1.89%	0.32%	0.00%	0.00%	-90.98%	1.89	6.73	7.78	22.86	229.41
Ins121	12	15	2.55%	0.00%	0.00%	0.00%	-95.93%	1.11	2.14	17.93	43.13	352.68
Ins122	14	15	0.85%	0.00%	0.00%	0.00%	-95.83%	2.03	1.97	6.52	11.11	368.46
Ins123	15	15	2.97%	0.41%	0.22%	0.52%	-93.99%	1.64	4.00	4.91	7.71	388.75
Ins124	15	15	1.95%	0.84%	0.00%	0.35%	-96.00%	1.18	3.74	9.07	19.75	474.72
Ins125	16	15	4.65%	1.28%	0.00%	0.00%	-95.51%	2.14	5.70	10.60	14.96	618.22
Ins126	16	15	0.79%	0.00%	0.00%	0.15%	-93.12%	1.73	6.16	15.95	24.25	682.69
Ins127	11	20	0.56%	0.00%	0.00%	0.00%	-90.36%	2.02	8.32	12.19	22.70	2,314.55
Ins128	12	20	3.94%	1.22%	0.35%	0.41%	-94.58%	3.49	6.85	20.38	29.06	2,457.16
Ins129	17	20	1.97%	0.22%	0.00%	0.00%	-94.09%	2.84	7.10	13.85	31.17	2,605.15
Ins130	18	20	6.25%	1.90%	0.00%	0.00%	-94.90%	3.26	7.53	16.61	23.61	2,705.55
Ins131	19	20	1.89%	0.00%	0.00%	0.39%	-92.72%	2.74	8.64	21.84	44.05	2,992.17
Ins132	20	20	2.42%	0.35%	0.00%	0.65%	-92.53%	3.05	7.16	22.01	49.59	3,105.55
Ins133	20	20	1.95%	0.14%	0.00%	0.00%	-94.19%	2.59	7.16	17.16	47.76	3,705.55
Ins134	15	25	3.65%	0.45%	0.00%	0.00%	-93.56%	4.11	12.63	120.22	185.85	9,704.24
Ins135	15	25	2.84%	1.22%	0.00%	0.00%	-93.43%	8.40	27.91	135.52	215.89	10,175.68
Ins136	16	25	3.55%	0.18%	0.00%	0.00%	-95.76%	5.28	22.56	210.55	290.56	10,684.46
Ins137	19	25	2.24%	0.54%	0.00%	1.27%	-90.63%	8.29	38.01	224.57	531.78	10,684.46
Ins138	21	25	7.12%	1.81%	0.00%	0.42%	-91.04%	6.05	13.40	146.17	214.00	10,891.16
Ins139	21	25	3.22%	0.79%	0.68%	0.00%	-91.52%	5.37	11.37	300.12	834.94	11,228.25
Ins140	23	25	0.81%	0.24%	0.00%	0.00%	-91.91%	4.85	27.23	191.91	384.01	11,702.03

Table 23: The optimality gaps and computational times of heuristics on multi vehicle instances in where compartment numbers range is (2-5) with more varied compartment capacities

			Optimality Gap (%)					Time in Seconds				
Instance	Vec. #	Loc #	SC	ISC	RSC	TS	Lower Bound	SC	ISC	RSC	TS	AC-TSPMPLPF
Ins141	3	5	0.00%	0.00%	0.00%	0.00%	-92.24%	0.05	0.10	0.10	0.24	1.16
Ins142	3	5	0.00%	0.00%	0.00%	0.00%	-91.91%	0.05	0.11	0.20	0.28	1.26
Ins143	4	5	0.00%	0.00%	0.00%	0.00%	-85.69%	0.04	0.12	0.13	0.32	1.28
Ins144	5	5	0.00%	0.00%	0.00%	0.00%	-91.87%	0.10	0.12	0.24	0.19	1.34
Ins145	5	5	0.00%	0.00%	0.00%	0.00%	-85.95%	0.12	0.13	0.11	0.74	1.38
Ins146	5	5	0.00%	0.00%	0.00%	0.00%	-86.74%	0.08	0.08	0.13	0.17	1.49
Ins147	6	5	0.00%	0.00%	0.00%	0.00%	-88.37%	0.09	0.07	0.18	0.64	1.62
Ins148	6	10	0.47%	0.00%	0.00%	0.00%	-95.86%	0.51	1.08	1.33	5.07	33.25
Ins149	7	10	2.52%	0.00%	0.00%	0.00%	-87.77%	0.59	1.32	1.59	2.03	48.12
Ins150	7	10	0.56%	0.00%	0.00%	0.00%	-85.86%	0.60	1.06	0.95	1.65	63.14
Ins151	8	10	3.21%	0.00%	0.00%	0.00%	-89.04%	0.62	0.82	1.04	1.49	63.81
Ins152	8	10	0.91%	0.00%	0.00%	0.00%	-88.12%	0.54	0.62	1.37	2.41	65.91
Ins153	10	10	0.76%	0.00%	0.00%	0.00%	-88.20%	0.56	0.74	1.38	3.59	67.45
Ins154	10	10	1.09%	0.00%	0.00%	0.00%	-86.23%	0.50	0.85	1.43	4.10	85.99
Ins155	9	15	0.45%	0.00%	0.00%	0.00%	-90.86%	2.11	1.55	10.23	24.41	213.44
Ins156	10	15	3.22%	0.00%	0.00%	0.71%	-92.08%	1.04	6.60	18.02	34.76	236.54
Ins157	13	15	4.25%	1.20%	0.08%	0.23%	-87.12%	2.25	4.37	14.51	20.43	240.76
Ins158	14	15	1.01%	0.00%	0.00%	0.00%	-90.03%	1.66	5.96	18.89	46.41	369.75
Ins159	14	15	2.13%	0.81%	0.00%	0.00%	-86.98%	1.06	2.27	19.88	35.68	566.73
Ins160	14	15	4.05%	0.22%	0.00%	0.12%	-91.72%	1.79	2.25	13.71	34.27	681.37
Ins161	16	15	3.13%	0.24%	0.00%	0.00%	-89.40%	2.35	3.21	12.86	26.78	815.41
Ins162	15	20	1.52%	0.00%	0.00%	0.00%	-93.86%	2.01	8.44	12.19	28.87	3,210.72
Ins163	16	20	2.55%	0.43%	0.00%	0.00%	-90.58%	3.48	7.82	20.31	56.60	2,499.54
Ins164	16	20	5.56%	1.22%	0.22%	1.02%	-87.76%	2.85	7.34	13.89	19.43	3,340.46
Ins165	17	20	1.50%	0.94%	0.00%	0.15%	-95.44%	3.27	7.01	16.65	24.30	2,925.28
Ins166	17	20	3.99%	0.66%	0.00%	0.00%	-86.88%	2.74	8.43	21.79	30.03	2,520.60
Ins167	19	20	3.19%	1.21%	0.15%	0.22%	-86.05%	3.04	7.10	22.05	58.78	3,489.38
Ins168	20	20	2.91%	0.00%	0.00%	0.00%	-87.93%	2.59	7.01	17.15	29.74	3,555.46
Ins169	14	25	2.08%	0.13%	0.00%	0.00%	-91.76%	3.05	38.97	175.56	402.74	8,765.68
Ins170	14	25	2.95%	0.00%	0.00%	0.00%	-90.76%	6.08	22.61	125.70	250.02	9,302.19
Ins171	15	25	4.22%	1.22%	0.13%	0.21%	-86.67%	5.55	13.40	154.18	348.75	10,005.24
Ins172	18	25	1.35%	0.84%	0.00%	0.00%	-88.57%	9.98	25.40	211.57	556.63	11,403.38
Ins173	20	25	3.40%	0.94%	0.41%	0.52%	-93.13%	8.45	30.88	205.21	370.20	11,519.24
Ins174	21	25	3.69%	1.22%	0.21%	0.31%	-92.67%	5.17	25.40	318.12	871.02	12,257.46
Ins175	23	25	6.31%	1.12%	0.00%	0.94%	-94.69%	6.47	51.53	118.10	179.63	12,731.55

Bibliography

[1] Y.-y. Tseng *et al.*, "The role of transportation in logistics," *Eastern Asia Society for Transportation Studies*, 2004.

[2] H. L. Lee and S. Whang, "The whose, where and how of inventory control design," *Supply Chain Management Review*, vol. 12, no. 8, 2008.

[3] J. Bramel and D. Simchi-Levi, *The logic of logistics: theory, algorithms, and applications for logistics management*. Springer New York, 1997.

[4] A. Angulo, H. Nachtmann, and M. A. Waller, "Supply chain information sharing in a vendor managed inventory partnership," *Journal of business logistics*, vol. 25, no. 1, pp. 101–120, 2004.

[5] W. J. Bell, L. M. Dalberto, M. L. Fisher, A. J. Greenfield, R. Jaikumar, P. Kedia, R. G. Mack, and P. J. Prutzman, "Improving the distribution of industrial gases with an on-line computerized routing and scheduling optimizer," *Interfaces*, vol. 13, no. 6, pp. 4–23, 1983.

[6] A. Federgruen and P. Zipkin, "A combined vehicle routing and inventory allocation problem," *Operations Research*, vol. 32, no. 5, pp. 1019–1037, 1984.

[7] D. J. Papageorgiou, G. L. Nemhauser, J. Sokol, M.-S. Cheon, and A. B. Keha, "Mirplib–a library of maritime inventory routing problem instances: Survey, core model, and benchmark results," *European Journal of Operational Research*, vol. 235, no. 2, pp. 350–366, 2014.

[8] M. Christiansen and K. Fagerholt, "Maritime inventory routing problems maritime inventory routing problems," in *Encyclopedia of optimization*, pp. 1947–1955, Springer, 2008.

[9] D. Ronen, "Marine inventory routing: Shipments planning," *Journal of the Operational Research Society*, vol. 53, no. 1, pp. 108–114, 2002.

[10] D. Ronen, "Cargo ships routing and scheduling: Survey of models and problems," *European Journal of Operational Research*, vol. 12, no. 2, pp. 119–126, 1983.

[11] N. H. Moin and S. Salhi, "Inventory routing problems: a logistical overview," *Journal of the Operational Research Society*, vol. 58, no. 9, pp. 1185–1194, 2007.

[12] L. C. Coelho, J.-F. Cordeau, and G. Laporte, "Thirty years of inventory routing," *Transportation Science*, vol. 48, no. 1, pp. 1–19, 2013.

[13] H. Andersson, A. Hoff, M. Christiansen, G. Hasle, and A. Løkketangen, "Industrial aspects and literature survey: Combined inventory management and routing," *Computers & Operations Research*, vol. 37, no. 9, pp. 1515–1536, 2010.

[14] J. Ward and X. Zhaï, "Joint transportation and inventory problems in supply chains: A review," tech. rep., Working paper. Krannert school of Management, University of Chicago, 2004.

[15] L. Bertazzi, G. Paletta, and M. G. Speranza, "Deterministic order-up-to level policies in an inventory routing problem," *Transportation Science*, vol. 36, no. 1, pp. 119–132, 2002.

[16] A. M. Campbell and M. W. Savelsbergh, "A decomposition approach for the inventory-routing problem," *Transportation science*, vol. 38, no. 4, pp. 488–502, 2004.

[17] G. Desaulniers, F. Errico, S. Irnich, and M. Schneider, "Exact algorithms for electric vehicle-routing problems with time windows," *Operations Research*, vol. 64, no. 6, pp. 1388–1405, 2016.

[18] W. W. Qu, J. H. Bookbinder, and P. Iyogun, "An integrated inventory–transportation system with modified periodic policy for multiple products," *European Journal of Operational Research*, vol. 115, no. 2, pp. 254–269, 1999.

[19] B. Raa and E.-H. Aghezzaf, "A practical solution approach for the cyclic inventory routing problem," *European Journal of Operational Research*, vol. 192, no. 2, pp. 429–441, 2009.

[20] L. M. A. Chan and D. Simchi-Levi, "Probabilistic analyses and algorithms for three-level distribution systems," *Management Science*, vol. 44, no. 11-part-1, pp. 1562–1576, 1998.

[21] G. Hiermann, J. Puchinger, S. Ropke, and R. F. Hartl, "The electric fleet size and mix vehicle routing problem with time windows and recharging stations," *European Journal of Operational Research*, vol. 252, no. 3, pp. 995–1018, 2016.

[22] Q.-H. Zhao, S. Chen, and C.-X. Zang, "Model and algorithm for inventory/routing decision in a three-echelon logistics system," *European Journal of Operational Research*, vol. 191, no. 3, pp. 623–635, 2008.

[23] B. Raa and E.-H. Aghezzaf, "Designing distribution patterns for long-term inventory routing with constant demand rates," *International Journal of Production Economics*, vol. 112, no. 1, pp. 255–263, 2008.

[24] S. Anily and A. Federgruen, "Two-echelon distribution systems with vehicle routing costs and central inventories," *Operations Research*, vol. 41, no. 1, pp. 37–47, 1993.

[25] C. Archetti, L. Bertazzi, A. Hertz, and M. G. Speranza, "A hybrid heuristic for an inventory routing problem," *INFORMS Journal on Computing*, vol. 24, no. 1, pp. 101–116, 2012.

[26] F. Niakan and M. Rahimi, "A multi-objective healthcare inventory routing problem; a fuzzy possibilistic approach," *Transportation Research Part E: Logistics and Transportation Review*, vol. 80, pp. 74–94, 2015.

[27] R. G. Van Anholt, L. C. Coelho, G. Laporte, and I. F. Vis, "An inventory-routing problem with pickups and deliveries arising in the replenishment of automated teller machines," *Transportation Science*, vol. 50, no. 3, pp. 1077–1091, 2016.

[28] A. J. Kleywegt, V. S. Nori, and M. W. Savelsbergh, "Dynamic programming approximations for a stochastic inventory routing problem," *Transportation Science*, vol. 38, no. 1, pp. 42–70, 2004.

[29] J. Jung and K. Mathur, "An efficient heuristic algorithm for a two-echelon joint inventory and routing problem," *Transportation Science*, vol. 41, no. 1, pp. 55–73, 2007.

[30] S. Viswanathan and K. Mathur, "Integrating routing and inventory decisions in one-warehouse multiretailer multiproduct distribution systems," *Management Science*, vol. 43, no. 3, pp. 294–312, 1997.

[31] D. Adelman, "A price-directed approach to stochastic inventory/routing," *Operations Research*, vol. 52, no. 4, pp. 499–514, 2004.

[32] A. J. Kleywegt, V. S. Nori, and M. W. Savelsbergh, "The stochastic inventory routing problem with direct deliveries," *Transportation Science*, vol. 36, no. 1, pp. 94–118, 2002.

[33] A. Federgruen, G. Prastacos, and P. H. Zipkin, "An allocation and distribution model for perishable products," *Operations Research*, vol. 34, no. 1, pp. 75–82, 1986.

[34] B. Golden, A. Assad, and R. Dahl, "Analysis of a large scale vehicle routing problem with an inventory component," *Large scale systems*, vol. 7, no. 2-3, pp. 181–190, 1984.

[35] P. Jaillet, J. F. Bard, L. Huang, and M. Dror, "Delivery cost approximations for inventory routing problems in a rolling horizon framework," *Transportation Science*, vol. 36, no. 3, pp. 292–300, 2002.

[36] O. Solyalı and H. Süral, "A branch-and-cut algorithm using a strong formulation and an a priori tour-based heuristic for an inventory-routing problem," *Transportation Science*, vol. 45, no. 3, pp. 335–345, 2011.

[37] J. F. Bard and N. Nananukul, "A branch-and-price algorithm for an integrated production and inventory routing problem," *Computers & Operations Research*, vol. 37, no. 12, pp. 2202–2217, 2010.

[38] L. C. Coelho and G. Laporte, "A branch-and-cut algorithm for the multi-product multi-vehicle inventory-routing problem," *International Journal of Production Research*, vol. 51, no. 23-24, pp. 7156–7169, 2013.

[39] G. Desaulniers, J. G. Rakke, and L. C. Coelho, "A branch-price-and-cut algorithm for the inventory-routing problem," *Transportation Science*, vol. 50, no. 3, pp. 1060–1076, 2015.

[40] V. Gaur and M. L. Fisher, "A periodic inventory routing problem at a supermarket chain," *Operations Research*, vol. 52, no. 6, pp. 813–822, 2004.

[41] L. Bertazzi, L. M. A. Chan, and M. G. Speranza, "Analysis of practical policies for a single link distribution system," *Naval Research Logistics (NRL)*, vol. 54, no. 5, pp. 497–509, 2007.

[42] L. M. A. Chan, A. Federgruen, and D. Simchi-Levi, "Probabilistic analyses and practical algorithms for inventory-routing models," *Operations Research*, vol. 46, no. 1, pp. 96–106, 1998.

[43] J. Li, F. Chu, and H. Chen, "A solution approach to the inventory routing problem in a three-level distribution system," *European Journal of Operational Research*, vol. 210, no. 3, pp. 736–744, 2011.

[44] H.-Z. Zheng, D.-H. Chu, D.-C. Zhan, and X.-F. Xu, "A heuristic solution approach for vmi cyclic inventory routing problem," in *Intelligent Systems, 2009. GCIS'09. WRI Global Congress on*, vol. 2, pp. 503–508, IEEE, 2009.

[45] M. Dror and M. Ball, *Inventory routing: Reduction from an annual to short period problem.* Université de Montréal, Centre de recherche sur les transports, 1984.

[46] M. Dror, M. Ball, and B. Golden, "A computational comparison of algorithms for the inventory routing problem," *Annals of Operations Research*, vol. 4, no. 1, pp. 1–23, 1985.

[47] S. Anily and A. Federgruen, "One warehouse multiple retailer systems with vehicle routing costs," *Management Science*, vol. 36, no. 1, pp. 92–114, 1990.

[48] G. Gallego and D. Simchi-Levi, "On the effectiveness of direct shipping strategy for the one-warehouse multi-retailer r-systems," *Management Science*, vol. 36, no. 2, pp. 240–243, 1990.

[49] M. Christiansen, "Decomposition of a combined inventory and time constrained ship routing problem," *Transportation science*, vol. 33, no. 1, pp. 3–16, 1999.

[50] L. Bertazzi and M. G. Speranza, "Continuous and discrete shipping strategies for the single link problem," *Transportation Science*, vol. 36, no. 3, pp. 314–325, 2002.

[51] C. Archetti, L. Bertazzi, G. Laporte, and M. G. Speranza, "A branch-and-cut algorithm for a vendor-managed inventory-routing problem," *Transportation Science*, vol. 41, no. 3, pp. 382–391, 2007.

[52] D. Aksen, O. Kaya, F. S. Salman, and Y. Akça, "Selective and periodic inventory routing problem for waste vegetable oil collection," *Optimization Letters*, vol. 6, no. 6, pp. 1063–1080, 2012.

[53] P. Chandra, "A dynamic distribution model with warehouse and customer replenishment requirements," *Journal of the Operational Research Society*, pp. 681–692, 1993.

[54] P. Parthanadee and R. Logendran, "Periodic product distribution from multi-depots under limited supplies," *Iie Transactions*, vol. 38, no. 11, pp. 1009–1026, 2006.

[55] A. Ekici, O. Ö. Özener, and G. Kuyzu, "Cyclic delivery schedules for an inventory routing problem," *Transportation Science*, vol. 49, no. 4, pp. 817–829, 2014.

[56] B. Raa, "Fleet optimization for cyclic inventory routing problems," *International Journal of Production Economics*, vol. 160, pp. 172–181, 2015.

[57] L. C. Coelho and G. Laporte, "Improved solutions for inventory-routing problems through valid inequalities and input ordering," *International Journal of Production Economics*, vol. 155, pp. 391–397, 2014.

[58] L. C. Coelho, J.-F. Cordeau, and G. Laporte, "The inventory-routing problem with transshipment," *Computers & Operations Research*, vol. 39, no. 11, pp. 2537–2548, 2012.

[59] L. C. Coelho, J.-F. Cordeau, and G. Laporte, "Consistency in multi-vehicle inventory-routing," *Transportation Research Part C: Emerging Technologies*, vol. 24, pp. 270–287, 2012.

[60] J. Bukchin and S. C. Sarin, "A cyclic policy for the loading of multiple products on a vehicle with different compartment sizes," *IIE transactions*, vol. 36, no. 7, pp. 641–653, 2004.

[61] M. Savelsbergh and J.-H. Song, "An optimization algorithm for the inventory routing problem with continuous moves," *Computers & operations research*, vol. 35, no. 7, pp. 2266–2282, 2008.

[62] D. Popović, M. Vidović, and G. Radivojević, "Variable neighborhood search heuristic for the inventory routing problem in fuel delivery," *Expert Systems with Applications*, vol. 39, no. 18, pp. 13390–13398, 2012.

[63] Q. Wang, Q. Ji, and C.-H. Chiu, "Optimal routing for heterogeneous fixed fleets of multicompartment vehicles," *Mathematical Problems in Engineering*, vol. 2014, 2014.

[64] Ç. Koç, T. Bektaş, O. Jabali, and G. Laporte, "Thirty years of heterogeneous vehicle routing," *European Journal of Operational Research*, vol. 249, no. 1, pp. 1–21, 2016.

[65] U. Yuceer, "A multi-product loading problem: a model and solution method," *European Journal of Operational Research*, vol. 101, no. 3, pp. 519–531, 1997.

[66] M. R. Garey and D. S. Johnson, "Computers and intractability: a guide to the theory of np-completeness. 1979," *San Francisco, LA: Freeman*, vol. 58, 1979.

[67] C. E. Miller, A. W. Tucker, and R. A. Zemlin, "Integer programming formulation of traveling salesman problems," *Journal of the ACM (JACM)*, vol. 7, no. 4, pp. 326–329, 1960.

[68] B. E. Gillet and L. R. Miller, "A heuristic algorithm for the vehicle-dispatch problem," *Operations research*, vol. 22, no. 2, pp. 340–349, 1974.

[69] F. Glover and M. Laguna, "Tabu search," in *Handbook of Combinatorial Optimization*, pp. 3261–3362, Springer, 2013.

[70] F. Li, B. Golden, and E. Wasil, "A record-to-record travel algorithm for solving the heterogeneous fleet vehicle routing problem," *Computers & Operations Research*, vol. 34, no. 9, pp. 2734–2742, 2007.

[71] B. Almada-Lobo, J. F. Oliveira, and M. A. Carravilla, "Production planning and scheduling in the glass container industry: A vns approach," *International Journal of Production Economics*, vol. 114, no. 1, pp. 363–375, 2008.

[72] E. Rolland, D. A. Schilling, and J. R. Current, "An efficient tabu search procedure for the p-median problem," *European Journal of Operational Research*, vol. 96, no. 2, pp. 329–342, 1997.

[73] R. Battiti and G. Tecchiolli, "The reactive tabu search," *ORSA journal on computing*, vol. 6, no. 2, pp. 126–140, 1994.

[74] N. A. Wassan, A. H. Wassan, and G. Nagy, "A reactive tabu search algorithm for the vehicle routing problem with simultaneous pickups and deliveries," *Journal of combinatorial optimization*, vol. 15, no. 4, pp. 368–386, 2008.

[75] I. H. Osman and N. A. Wassan, "A reactive tabu search meta-heuristic for the vehicle routing problem with back-hauls," *Journal of Scheduling*, vol. 5, no. 4, pp. 263–285, 2002.

[76] A. El Fallahi, C. Prins, and R. W. Calvo, "A memetic algorithm and a tabu search for the multi-compartment vehicle routing problem," *Computers & Operations Research*, vol. 35, no. 5, pp. 1725–1741, 2008.

[77] P. V. Silvestrin and M. Ritt, "An iterated tabu search for the multi-compartment vehicle routing problem," *Computers & Operations Research*, vol. 81, pp. 192–202, 2017.

[78] M. Alinaghian and N. Shokouhi, "Multi-depot multi-compartment vehicle routing problem, solved by a hybrid adaptive large neighborhood search," *Omega*, 2017.

[79] T. Henke, M. G. Speranza, and G. Wäscher, "The multi-compartment vehicle routing problem with flexible compartment sizes," *European Journal of Operational Research*, vol. 246, no. 3, pp. 730–743, 2015.

[80] D. Barnes-Schuster and Y. Bassok, "Direct shipping and the dynamic single-depot/multi-retailer inventory system," *European Journal of Operational Research*, vol. 101, no. 3, pp. 509–518, 1997.

[81] J. Bramel and D. Simchi-Levi, "A location based heuristic for general routing problems," *Operations research*, vol. 43, no. 4, pp. 649–660, 1995.

[82] E.-H. Aghezzaf, B. Raa, and H. Van Landeghem, "Modeling inventory routing problems in supply chains of high consumption products," *European Journal of Operational Research*, vol. 169, no. 3, pp. 1048–1063, 2006.

[83] T. F. Abdelmaguid, M. M. Dessouky, and F. Ordóñez, "Heuristic approaches for the inventory-routing problem with backlogging," *Computers & Industrial Engineering*, vol. 56, no. 4, pp. 1519–1534, 2009.

VITA

Türkay Umut Yılmaz was born in Ankara, Turkey on February 7, 1990. He graduated from Ankara Atatürk Anatolian High School in 2008. He received the BSc degree in Industrial Engineering from Galatasaray University in June 2013. In February 2014, he entered Graduate School of Engineering at Özyeğin University, in Istanbul. He worked as a Teaching and Research Assistant throughout his MSc degree. He also received MA degree in Business Administration from Bahcesehir University in June 2016. He is currently working as Quantitative Analyst in Fina Enerji Holding A.Ş which is a subsidary of Fiba Holding A.Ş.