TRANSPORTATION PLANNING FOR THE RELIEF ITEMS DISTRIBUTION DURING AN EMERGENCY

A Thesis

by

Emre Çankaya

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
January 2017

TRANSPORTATION PLANNING FOR THE RELIEF ITEMS DISTRIBUTION DURING AN EMERGENCY

Approved by:

Associate Professor Okan Örsan Özener, Advisor
Department of Industrial Engineering
Özyeğin University

Assistant Professor İhsan Yanıkoğlu
Department of Industrial Engineering
Özyeğin University

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

Assistant Professor Mehmet Önal
Department of Industrial Engineering
Işık University

Date Approved: 3 January 2017

Associate Professor Burcu Balçık
Department of Industrial Engineering
Özyeğin University

To my family

ABSTRACT

We study the inventory slack routing problem (ISRP) to improve planning of distribution of the relief supplies, which is a critical issue in emergency preparedness and response. The authority officials construct their distribution plan in case of emergencies (such as terrorist attacks, hurricanes, earthquakes or natural disasters, etc) in order to minimize the risk for human lives. Thus, unlike inventory routing problem (IRP) and vehicle routing problem (VRP), the objective of the ISRP is to maximize minimum slack, that is time until a dispensing site affected by the disaster runs out of supplies. This difference in the objective function requires a solution approach that is significantly different than the ones that are proposed in the literature for IRP or other routing problems. We propose a two-phase approach which includes clustering and routing to develop distribution planning of relief items. We conduct an extensive computational study on randomly generated instances in order to assess the performances of the proposed algorithms and compare the performances of the proposed algorithms with respect to two existing solution algorithms in the literature in terms of solution quality.

Keywords: inventory routing; emergency planning; medication distribution

ÖZETÇE

Acil durumlara hazırlık ve müdahalelerde önemli bir konu olan yardım malzemeleri dağıtımını iyileştirmek için Envanter Gevşeklik (Slack) Rotalama Problemi'ni (ISRP) çalışmaktayız. Yetkili makamlar, terör saldırıları, kasırgalar, depremler veya doğal felaketler gibi acil durumlarda insan hayatının riskini en aza indirgemek için dağıtım planı oluştururlar. Envanter Rotalama Problemi (IRP) ve Araç Rotalama Problemi'nden (VRP) farklı olarak Envanter Gevşeklik Rotalama Problemi'nin (ISRP) amacı en düşük gevşeklik değerini en yüksek değere çıkarmaktır, bu gevşeklik değeri de bir bölgedeki tedariklerin tükenene kadar geçen süreye eşittir. Amaç fonksiyonunda ki bu farklılık, IRP veya diğer rotalama problemleri için literatürde önerilenlerden önemli ölçüde farklı bir çözüm yaklaşımını gerektirir. Yardım malzemeleri dağıtım planlamasını geliştirmek için kümeleme ve rotalama içeren iki aşamalı bir yaklaşım önerilmiştir. Önerilen algoritmaların performanslarını değerlendirmek ve çözüm kalitesi açısından performanslarını literatürde mevcut olan algoritmalarla karşılaştırmak için rastgele oluşturulmuş örnekler üzerinde kapsamlı bir çalışma yürütülmüştür.

Anahtar kelimeler: envanter rotalama; acil durum planlaması; ilaç dağıtımı

ACKNOWLEDGEMENTS

Firstly, I would like to express my sincere gratitude to my advisors Okan Örsan Özener and Ali Ekici for their valuable support of my M.Sc. study and related research throughout the completion of this thesis.

In addition to my advisors, I would like to thank my office roommates who help me during the my thesis discussions.

Finally, I would like to thank my family for supporting me throughout writing this thesis.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION

Inventory slack routing problem (ISRP) is a variant of an inventory routing problem (IRP) where the goal is to maximize the minimum slack in inventory as oppose to minimizing total cost. The "slack of a dispensing site" is defined as time until that particular dispensing site runs out of supplies if no deliveries are made between now and then. This time value will be known as "slack". The minimum slack is important while distributing relief supplies for those in need in a case of emergency as lack of relief supplies such as food, water, medicines, tents, blankets might result in loss of lives. Therefore, while responding to the diasters or other emergencies, it is crucial to deliver the relief supplies to dispensing points in a timely manner. Solving ISRP efficiently provides a good routing plan that ensures the relief supplies to be delivered timely and so prevents supplies from being stock out in any point in time.

In recent years, emergency preparedness operations have become much more important due to the increasing number of major disasters or terrorist attacks. Each year, several natural disasters occur in the world and these natural disasters affect millions of people. About 4.4 million people were affected by disasters during 1994-2013 and more than one million people died. The total cost of these disasters was estimated to be 2 trillion US dollars [1]. In some specific cases, such as the earthquake in Haiti, the number of people affected by the disaster might be considerably. For instance, based on the statistics by the government of Haiti, at least 230000 people were dead and at least 300000 people were injured because of the earthquake occurred in January 2010. In this earthquake, over one million people were displaced and most of them had to live in shelters for a long time. From a monetary perspective, the total cost of this earthquake for government of Haiti was about 8 billion US dollars. In order to mitigate such drastic effects of disasters, especially in disasters of this magnitude, efficient and effective response planning is required. Fueled by this need, there have been an increasing number of studies in the literature focusing on humanitarian logistics and disaster mitigation activities.

Humanitarian logistics consist of four main phases: (i) mitigation, (ii) preparedness, (iii) response, and (iv) recovery [2]. The mitigation phase, also called as prevention, includes any activities to prevent a disaster or minimize the destructive effects of the disasters. Mitigation activities should be planned and executed preferably long before an emergency. Preparedness is set of activities that prepares the society to react properly when an emergency occurs. Training, exercising and planning are the fundamental elements of preparedness. Emergency personnel may respond much better due to the preparedness activities when an emergency occurs. Similar to the mitigation phase, the preparedness activities should be carried out before the disaster occurrence. The response phase is the deployment of the resources and taking necessary actions to preserve life and prevent the negative effects of the disasters to the nature and the community. Unlike the preparedness phase, the response activities occur during a disaster. The recovery phase includes activities executed after the disaster occurrence and the objective of such activities is to return the affected area to normal as quickly as possible.

In this study, we focus on the response phase of out of four phases in humanitarian logistic activities as illustrated in Figure 1. More specifically, we focus on the distribution planning of emergency relief supplies to the affected people who are in need. Our objective is to improve the planning of the distribution during an emergency. In response to an emergency, the supplies to be distributed may include food, water, tents, medicines, blankets or tarp and the humanitarian organizations are responsible for distribution these supplies to the dispensing sites or to the other facilities at the right quantity and at the right time. The distribution plan should be executed in a swift manner. Otherwise, it may result in human suffering and potential deaths in disaster areas. For example, in Haiti, more than eight thousand people died due to cholera caused by bad living conditions at tent camps and lack of enough medicines during living here [1].

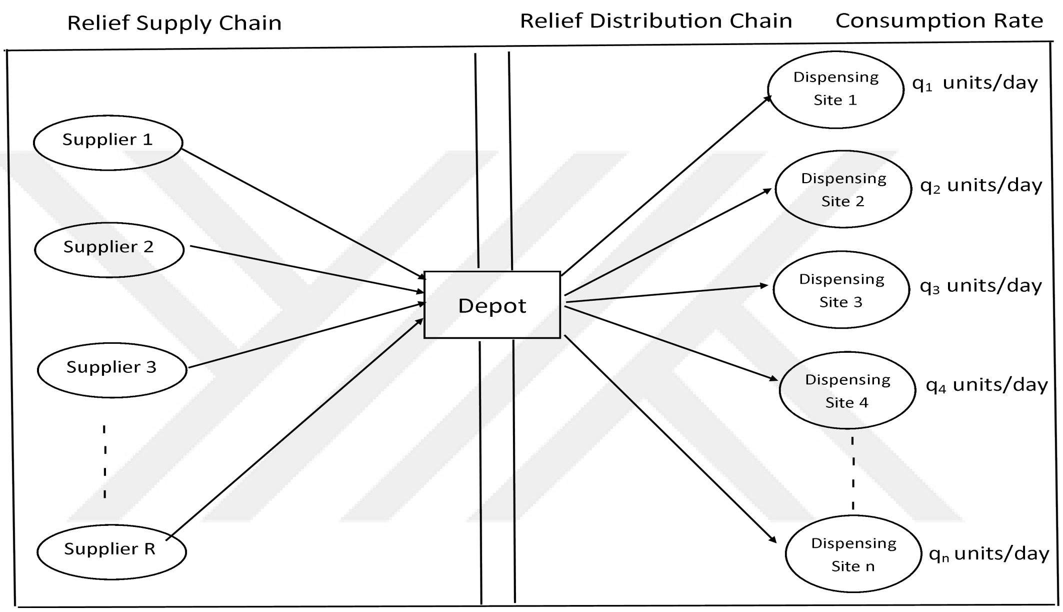

Figure 1: Illustration of an example relief logistic network after a disaster

Constructing of an efficient and an effective distribution plan is difficult for several reasons. First, there exists a limited amount of supply and therefore it may not be possible send very large amount relief supplies to any locations in the affected area. Under limit supply availability, the first and foremost task to determine the optimal allocation of resources to the affected areas. Unfortunately, this task is not a trivial one as it depends on several factors such as the total amount of available relief supplies, the relief supplies that will be available in latter stages, the number of people affected in the area, and finally the distribution of these relief supplies to the affected areas. The second is the routing of the relief supply vehicles, which is not only complicated but also affects the first task as well. When a vehicle visits a dispensing site, the amount to be delivered to the particular dispensing site depends on the amount available on the vehicle, the amount remaining on the dispensing site, the relief items usage rate of the dispensing site and the items requested by the dispensing sites that will be visited later by the same vehicle. The usage rate being non-uniform at the dispensing sites depending on several factors such as type and impact of the disasters, demographics and some uncertainties is one of the main reasons that makes the task on hand challenging. Not only the allocation of resources becomes complicated but also the visiting order of the dispensing sites suddenly becomes a factor in this decision process. Particulary, given that the number of vehicles limited, the visiting order of the dispensing sites and the timing of deliveries are the key decisions in a distribution planning process. Last but not least, the distribution planning is a task that spans multiple periods and the decisions given in one particular period have significant affects in the decision given in the subsequent periods. For instance, it may be preferable to deliver a large quantity of relief supplies to a dispensing site and not visiting this dispensing site in the subsequent periods as opposed to delivering small quantities but visiting the dispensing site more frequently (e.g. every period). Such a tradeoff depends on several factors such as the number of vehicles, the number of dispensing sites and their usage rates, the amount of available relief supplies, etc. and should be analyzed in detail while constructing the most effective distribution plan.

In order to construct such a distribution plan, we propose a two-phase approach to the uncapacitated version of the problem. We consider both the single vehicle and multiple vehicle versions of the problem. The proposed of the two-phase approach consists of the clustering and routing phases. In the clustering phase, we use an exact solution method to partition the dispensing sites into clusters. In the second phase, we implement the two different ideas to estimate the minimum slack by determining the delivery routes and schedule for each cluster. We implement the first idea in two stages (i) construct an initial tour using a well-known algorithm, Clarke and Wright algorithm, and solve an exact model using a commercial optimization solver to estimate the minimum slack as an initial solution (ii) we improve the routing by using 2-opt and insertion algorithms and solve the problem with exact model using a commercial optimization at each step iteratively. In the second idea, we solve the problem with exact model using a commercial optimization by determining the delivery routes using that model.

The rest of the paper is organized as follows. In Chapter 2, we review the related works in the literature. In Chapter 3, we introduce a statement of the ISRP and we present an illustrative example to further explain the problem. In Chapter 4, we discuss both the single vehicle and multi vehicle versions of the problem and explain the two-phase solution approach. We first describe the algorithm used in the clustering phase. Next, we explain the two different algorithms used in the second phase. In Chapter 5, we computationally demonstrate how our proposed algorithms perform under different problem settings and compare their performance with the benchmark algorithms proposed in the literature. Concluding remarks are provided in Chapter 6.

CHAPTER II

LITERATURE REVIEW

The problem under consideration is a variant of the *Inventory Routing Problem* (IRP). IRP is a well-studied problem in the literature, and depending on the application different variants are analyzed. Early works studying IRP can be found in [3, 4, 5].

IRP is a general version of well-known *Vehicle Routing Problem* (VRP) where the decision maker has the flexibility to determine the delivery volume and delivery time to dispensing sites over a given planning horizon in order to satisfy the demand at the dispensing sites. The objective in IRP is to minimize the total cost which may include (i) transportation cost, (ii) fixed cost of the vehicles used, (iii) inventory holding cost, and (iv) backordering cost. Although the decision maker the flexibility of deciding how much and when to deliver to the customers, finding the optimal solution in such a setting is quite challenging even for very small instances [6].

Several of variants of IRP are studied in the literature. These variants differ in terms of the objective function, the replenishment strategy and the nature of the demand. Objective function may contain vehicle routing and inventory related costs. Vehicle routing cost may include the following parts: (i) a fixed cost included when each vehicle is dispatched [7, 8, 9], (ii) transportation cost for traversing each edge [10, 11, 12], (iii) fixed cost per stop at a customer point [13, 14] and (iv) fixed cost of a vehicle when fleet size is a decision [15]. Inventory related cost may contain three parts: (i) holding cost at each customer point [16, 17, 18, 19], (ii) shortage cost [4, 20] and (iii) ordering cost if we procure products from an external source of produce in house [9, 21]. Some studies in the literature focus on specific strategies and analyze the problem under these strategies in order to simplify the problem. These strategies include (i) fixed partition policy where customers are assigned to nonoverlapping clusters and each cluster is visited by one vehicle [22, 23], (ii) zero inventory ordering policy where a customer's inventory is replenished only if its inventory is zero [7, 24], (iii) power of two policy where inventory of each customer is replenished at a multiple of two of constant reorder interval [21, 25], (iv) order up to level policy where customer's inventory is replenished up to its maximum level when customer is served [17, 26]. Another differentiating characteristic of different variants of IRP is the demand structure. While some studies assume that the demand is deterministic [17, 21, 27], some analyze the routing and delivery schedule decisions under stochastic demand [28, 29, 30, 31, 32].

IRP studies in the literature focus on two different settings for demand realization: discrete time versus continuous time. In the first case, it is assumed that the delivery is performed at the beginning of a period and demand is realized at the end of a period [17, 26, 33]. In the second case, demand is realized continuously throughout the period rather than instantly at the end of the period [34, 35].

In our study, we focus on delivering the products to dispensing sites where demand is realized on a continuous time basis. Different from the discrete demand case, the visiting time of each demand point in each period significantly affects the solution in continuous time demand case. Given the set of demand points to be visited on a given day in the discrete case, the problem turns into a traveling salesman problem (TSP). But in our case, even if the dispensing sites to be visited are known in a period, the visiting order significantly affects the solution. A cost-efficient solution like TSP may result in shortage at certain dispensing sites if they are visited later in the tour. For example, if a certain dispensing site needs a delivery immediately (due to low inventory), then visiting that dispensing site first regardless of its distance to depot may give a better solution.

Another characteristic of different variants of IRP is the planning horizon. Some studies consider a single period while some study multi-period setting. When a single period is considered, the problem becomes an extension of VRP and the solution techniques for VRP may be adapted to solve the single period version [4, 36, 37]. Unlike IRP with a single period, we make decisions over a planning horizon in IRP with multiple periods [17, 19, 34, 38]. In the multi-period case, the decision we make in one period affects the decisions in the following periods. For example, if we deliver a large quantity to a customer, we may skip this customer in the following period or if we deliver a small quantity to a customer, we may have to visit this customer again in the following period.

The problem analyzed in this thesis is an application of relief distribution operations which is extensive study in recent years due to disasters. Many studies in the literature have been done to develop models to improve relief distribution operations. A literature review and analysis of operations research models in transportation relief items have been conducted by [39]. Relief transportation of food items from a distribution center to a number of camps assuming a single mode of transportation are considered by [40]. A linear programming model is developed while minimizing the transportation cost or maximizing the amount of food delivered. In study [41], a mathematical model is developed for helicopter mission planning during a disaster relief operation. The problem is decomposed hierarchically into two sub-problems where tactical decisions are made at the top level, and the operational routing and loading decisions are made at the base level. Mixed integer programming models are developed for operational and tactical problems, which are solved by an iterative coordination heuristic. In study [42], the authors analyze the transportation of multiple commodities on a network with time windows to minimize loss of life. They formulate a multi-commodity, multi-modal network flow with time windows, and two solution methods are presented. A similar model is developed by including uncertainties in study [43]. The authors develop a two-stage stochastic programming model for transportation planning in disaster response. They solve the response problem with stochastic estimates of transportation capacities, supply availabilities and demand, based on alternative disaster scenarios in the first phase. Actual values are used as they are revealed in the second stage. The two-phase approach is tested on real instances. Logistics planning in emergency situations that involve dispatching commodities to distribution centers of affected areas is examined by [44]. They formulate a, multi-commodity network flow and it addresses a dynamic time-dependent transportation problem, and repetitively derives a solution at given time intervals to represent ongoing aid delivery. They develop an iterative Lagrangian relaxation algorithm and also a greedy heuristic to solve the problem. In study [45], a mixed integer programming model that determines delivery schedules for vehicles and equitably allocates resources, based on supply, vehicle capacity, and delivery time restrictions while minimizing the sum of routing costs and penalty costs for backordered demand for last mile distribution in relief operations. Relief items are categorized into two main groups such as Type 1 and Type 2 depend on their demand characteristics. Type 1 items are critical items and their demand occurs once at the beginning of the planning horizon. Type 2 items are consumed regularly and their demand occurs periodically over the planning horizon. If a Type 2 item is not satisfied on time, backordered is not allowed and it increases the penalty cost because of the lost demand. Hence, excess amount of Type 2 items is held for consumption for future periods.

As reviewed above, the relief distribution literature has different settings. The problem the most related to our setting is the one studied by [45] in terms of demand characteristics. Our problem addresses a similar setting with holding excess inventory for future periods, but it differs from this study in terms of the objective function. The main concern of our problem is to deliver the relief items (such as medicines) as quickly as possible. The limited availability of relief items at the distribution facility adds an additional constraint. Furthermore, a larger slack in inventory is necessary to hedge against the uncertainties in travel times and demand. Hence, the objective is to maximize the minimum slack (similar to safety stock), which is the time until a dispensing site runs out of supply, among all dispensing sites to develop a more robust plan.

The problem we analyzed in this thesis is called the *Inventory Slack Routing Problem* (ISRP). In our setting, we consider a single item (such as medicine) whose demand is deterministic at the dispensing sites over a multi-period planning horizon. Items arrive at the distribution facility in batches and the amount of each batch may be different in each period. Since, humanitarian organizations procure items from different donors or suppliers and these items become available at different periods. The problem most related to our setting is the one studied by [46, 47, 48, 49]. The ISRP with multi depot is considered in study [48] and the ISRP with single depot is considered in studies [46, 47, 49]. Both of two studies [46, 49] propose a two-stage approach: (i) routing stage where routes are created for each vehicle, and (ii) scheduling stage where the visiting time and delivery amount of each site are determined. In both studies, these two stages are solved independent of each other. In study [46], first each site is assigned to a single vehicle in order to create a single "giant" tour. This tour is created using two heuristic approaches: nearest neighbor algorithm and 2-opt algorithm. Then, this route is separated into clusters based on the capacity of the vehicles. In the scheduling stage, the delivery decisions are made after the routes are created. Delivery quantities are determined on each route such that the slack of all sites on the following day are equal. Study [49] modified the routing stage of the algorithm proposed by [46]. First, sites are sorted in a decreasing order of their consumption rates. They assign a single site to each vehicle based on the order of consumption rates. Then, the remaining sites are assigned to a vehicle based on the order of consumption rates again. Remaining sites are inserted to a tour that has the shortest duration between deliveries to these sites after insertion. After all tours are formed, the authors use some insertion ideas to improve the solution. In the scheduling part, the same approach proposed by [46] is followed. In study [47], the authors integrated the adaptive large neighborhood search method in addition to methods in study [46]. In study [48], first all dispensing sites are classified into a set of single depots ISRP which are solved using the same idea in study [49] and, then a reassignment step is implemented to improve the initial solution.

We extend the same problem of [46] and [49] by developing a mathematical model and presenting a new heuristic approach. Unlike [46] and [49], we handle the two (routing and scheduling) stages of the problem in an integrated manner. To do that, we first generate random clusters of the dispensing sites (possible overlapping), then estimate the minimum slack of each cluster by finding a feasible solution, and then solve a set partitioning problem in order select the best (nonoverlapping) subset of generated clusters.

CHAPTER III

PROBLEM DEFINITION

In this part, we present a formal definition of ISRP under consideration. We described the locations who are in need as dispensing site. The problem is defined on a Euclidean graph $G = (V_0, E)$, where $V_0 = \{0, 1, 2, ..., n\}$ is the set of dispensing sites ($V = \{1, 2, ..., n\}$), and the depot (0), E is the set of edges connecting the nodes in V_0. The dispensing sites in our setting represent the affected areas. The travel time (in minutes) between two dispensing sites is denoted by c_{ij}, and the travel time between two nodes is equal to the distance between two nodes divided by the constant speed of each vehicle. The dispensing sites are served a single relief item (such as medicine) from a single depot by a vehicle. The items to be delivered to the dispensing sites arrive in batches denoted as B_t throughout the T periods. In our setting, the length of the planning horizon T is given ($\mathcal{T} = \{1, 2, ..., T\}$), and the batches become available at the beginning of the each period. Demand at each dispensing site is constant and denoted by q_i units per period. The storage capacity of each dispensing site is assumed as infinite. Additionally, there is an initial inventory at each dispensing site, and it is denoted by I_i.

We assume that K homogenous uncapacitated vehicles are available to serve dispensing sites. Deliveries to the dispensing sites are made via routes that start and end at the depot, as illustrated in Figure 2. Dispensing sites operate L hours per period and the time period of length L equals 24 hours per period. For practicality purposes, we assume that we determine specific routes to ensure the safety of roads and each vehicle follows the same tour in each period. Each dispensing site is visited by only one vehicle in each period.

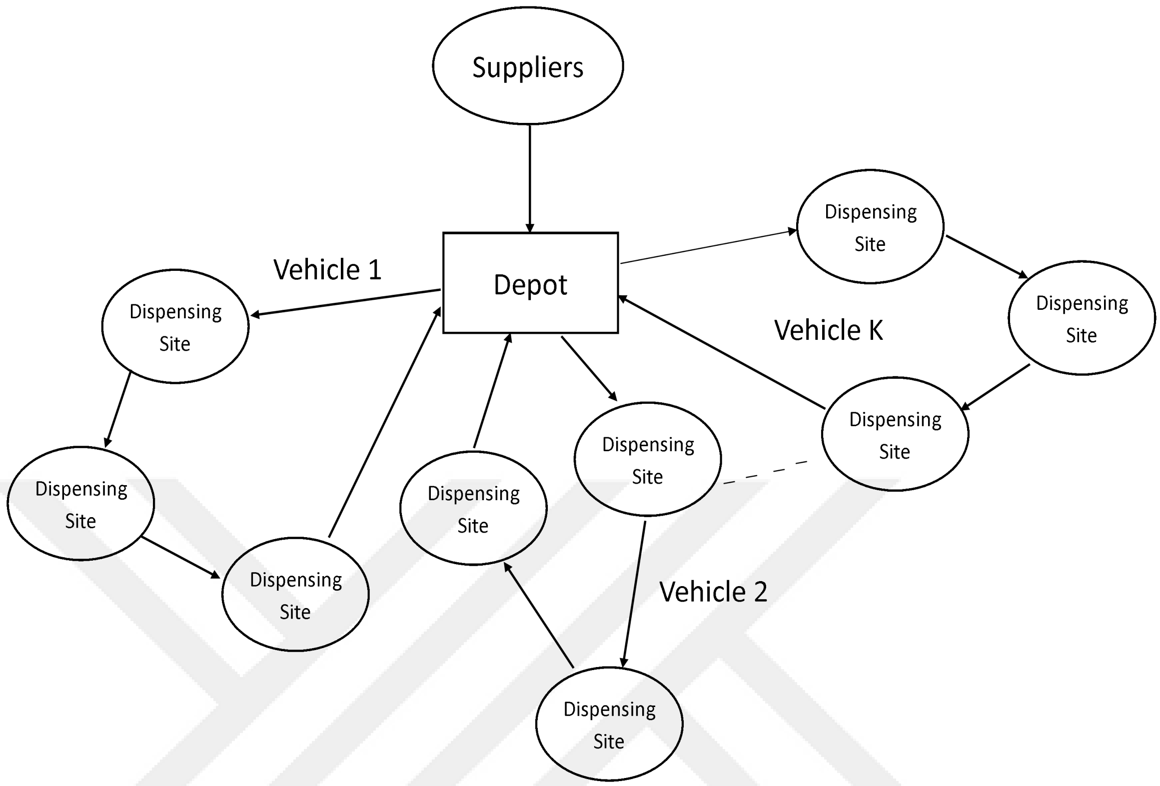

Figure 2: Distribution process in ISRP

The relief items delivered to depot become available in batches at different periods. Although the capacity of each vehicle is unlimited, we can deliver a limited quantity of the relief items to the dispensing sites in each period due to availability of the items.

Next, we present an example in order to explain the slack calculations in ISRP. Evaluating the minimum slack over T periods needs calculating the slack before each delivery. In the example provided in Figure 3, we have a single vehicle and two dispensing sites to be served over three periods. At the beginning of each period, the vehicle starts the tour at the depot, then it visits dispensing site 1 and dispensing site 2, in this order. The vehicle returns to the depot at the end of the tour. The initial amount of inventory and the daily consumption rate of each dispensing site are represented in Figure 3.

There are deliveries to the depot at the beginning of each period with amounts

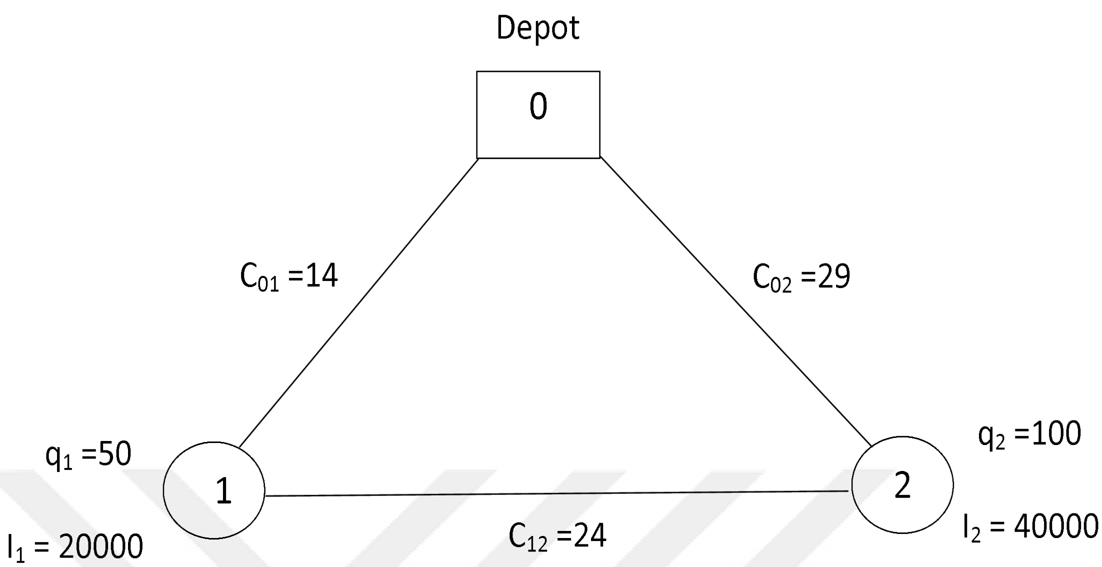

Figure 3: A simple example provided to illustrate the ISRP

Table 1: Delivery quantities to each dispensing site for each period

Disp. Site	q_i(per minutes)	Quantity	Quantity	Quantity
1	50	66000	80000	72000
2	100	132000	160000	144000

198000, 240000 and 216000. The amount of the delivery to each dispensing site is provided in Table 1. Vehicle starts the route at the beginning of the each period. Table 2 shows the visiting time (denoted by k_i) of each dispensing site, and the minimum slack (before delivery) for each dispensing site in each period. The slack for dispensing site i in the period t is denoted by S_{it} and the slack for each dispensing site in the period 1 can be calculated as follows:

$S_{11} = \frac{20000-(14)(50)}{50} = 386$

$S_{21} = \frac{40000-(38)(100)}{100} = 362$

Slack variables for the following periods are presented in Table 2. The minimum slack is 242 minutes and it occurs in the second period.

Unlike the classical routing problems, we do not focus on minimizing total travel time in ISRP. Our goal is to maximize the minimum slack at the dispensing sites. We

Table 2: Slack calculations all times in minutes before delivery

Disp. Site	k_i(in minutes)	Slack	Slack	Slack
1	14	386	266	426
2	38	362	242	402

present a simple example with three dispensing sites in Figure 4 to illustrate how the difference in the objective function affects the solution in ISRP.

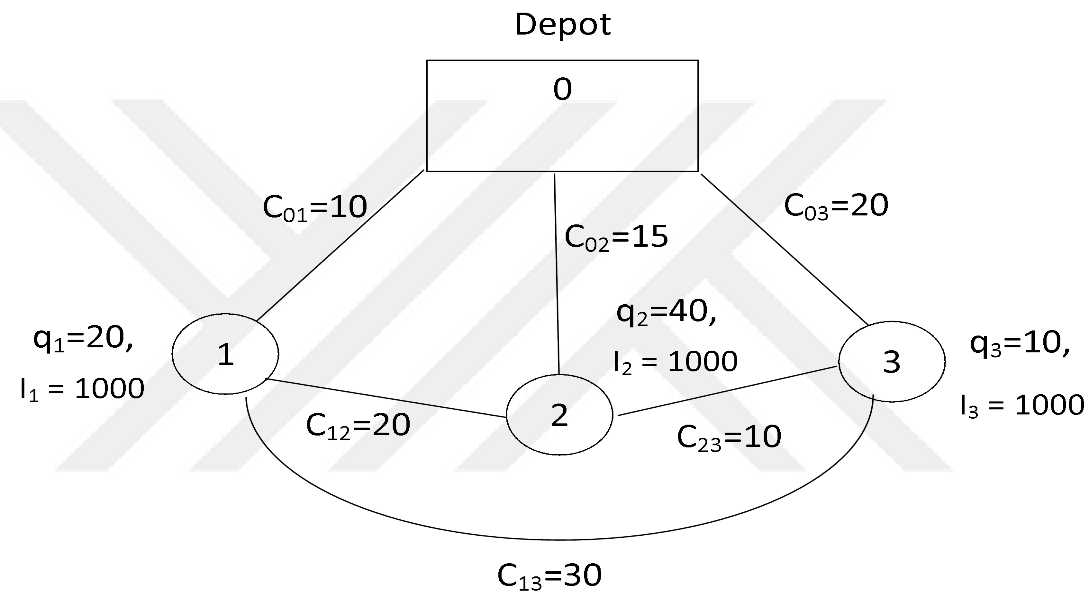

Figure 4: A simple example provided to illustrate the problem structure

In this example, we have a single vehicle and three dispensing sites to be served over two periods. The travel times between the depot and dispensing sites are shown in Figure 4. The initial inventory is 1000 items for each dispensing site. The consumption rates for dispensing site 1, dispensing site 2 and dispensing site 3 are 20 items per minute, 40 items per minute and 10 items per minute, respectively. There are deliveries to the depot at the beginning of each period with amounts 105000 and 105000. The amount of the delivery to each dispensing site is provided in Table 3. The possible scheduling plans and the minimum slack of each dispensing site are presented in Table 4.

Table 4 shows that Plan 1 and Plan 6 have the shortest travel times. However,

Table 3: Delivery quantities to each dispensing site

Disp. Site	Day 1	Day 2
1	30000	30000
2	60000	60000
2	15000	15000

Table 4: Slack calculations in minutes before delivery

Plan		Route	Duration	Slack Values When Delivery Arrives		
				Disp. Site 1	Disp. Site 2	Disp. Site 3
1	Day 1	0-1-2-3-0	60	40	-5	60
	Day 2	0-1-2-3-0	60	62	59.875	66
2	Day 1	0-1-3-2-0	65	40	-25	60
	Day 2	0-1-3-2-0	65	62	59.375	66
3	Day 1	0-2-1-3-0	85	15	10	35
	Day 2	0-2-1-3-0	85	60.75	60.250	63.5
4	Day 1	0-2-3-1-0	65	-5	10	75
	Day 2	0-2-3-1-0	65	59.75	60.250	67.5
5	Day 1	0-3-1-2-0	85	0	-45	80
	Day 2	0-3-1-2-0	85	60	58.875	68
6	Day 1	0-3-2-1-0	60	0	-5	80
	Day 2	0-3-2-1-0	60	60	59.875	68

the minimum slack of dispensing site 2 is -5 in Plan 1 and Plan 6. It means that dispensing site 2 experiences shortage in these distribution plans. The solution found in Plan 3 is the optimal solution. As it can be seen in the example above, minimizing the total travel time or total cost may not give the optimal solution for ISRP.

CHAPTER IV

METHODOLOGY

4.1 Single Vehicle Version of ISRP

In this section, we first analyze the single vehicle version of the problem. We assume that the set of dispensing sites are given and we have a single vehicle to visit each dispensing site in each period. Batches which are available at the beginning of each period are given and we present that how to maximize the minimum slack by delivering the relief items in each period. We made some observations to solve the problem easily. In the first observation, we observe that there is always an optimal solution that visits each dispensing site in each period.

Observation 1. *There exists an optimal solution such that every dispensing site is visited in each period.*

Take any solution that does not visit all the dispensing sites in each period can be converted to a solution that visits each dispensing site in each period. We can do that by inserting the unvisited dispensing site at the end of each tour in each period which is not going to change the slack. But it gives a solution where all the dispensing sites are visited in each period. This shows that there is always an optimal solution visiting each dispensing site in each period.

Next, we make an observation about the period in which the minimum slack occurs when the slacks at the dispensing sites are equal at the beginning. In order to make this observation, we first define the total amount of batches, total amount of daily consumptions of the dispensing sites and the difference between the total amount of batches and the total amount of daily consumptions. We define some notations for these definitions in the following equations. Then, we will call the period in which the difference is minimum as critical period. Based on this critical period, we make the following Observation 2.

Let D_t be the total amount of the batches to be delivered to the depot until period t. It is calculated using the following equation.

$$D_t = \sum_{a=1}^{t} B_a \tag{1}$$

Let C_t be the total amount of daily consumptions included all dispensing sites until period t. It is calculated using the following equation:

$$C_t = \left(\sum_{i=1}^{n} q_i \right) t \tag{2}$$

Let H_t be the difference between the cumulative batches and cumulative daily consumptions up to period t. It is calculated using the following equation:

$$H_t = D_t - C_t \tag{3}$$

Then, we calculate the difference between the total batches and the total daily consumptions up to period t and we call the period where the difference is minimum as critical period ($t^* = argmin_{t \in \mathcal{T}}(H_t = D_t - C_t)$). Next, we show that the minimum slack occurs in the period right after the critical period when the slacks at the dispensing sites are equal at the beginning.

Observation 2. *The minimum slack occurs in the period right after the critical period when the slacks at the dispensing sites are equal at the beginning.*

Proof. Suppose that the minimum slack does not occur in the period right after the critical period. In order to prove our observation and disprove the claim given here, let's analyze the tour for the period right after the critical period. We first follow Observation 1 and add the unvisited dispensing sites to the end of the tour in order to calculate its slacks. As a second step, by starting from initial period and following the tour from the first step, we will redistribute the items for each period in a way such that the slacks for all dispensing sites on the next period are equal.

Let $v_i^{'}$ be the visiting times of the dispensing sites in the tour which is analyzed and S_{i,t^*+1} be the slack in the period right after the critical period.

The difference between the total amount of batches and the total daily consumptions up to period t^* is less than or equal to the difference between the total amount of batches and the total daily consumptions up to the period t^*-1 as in the following:

$$D_{t^*} - C_{t^*} \leq D_{t^*-1} - C_{t^*-1} \tag{4}$$

The equation (4) can be rewritten as follows:

$$D_{t^*} - D_{t^*-1} \leq C_{t^*} - C_{t^*-1} \tag{5}$$

We know that $C_{t^*} - C_{t^*-1}$ is equal to the daily consumption as in the following equation:

$$C_{t^*} - C_{t^*-1} = \sum_{i=1}^{n} q_i \tag{6}$$

Then, the equation (5) can be rewritten as:

$$D_{t^*} - D_{t^*-1} \leq \sum_{i=1}^{n} q_i \tag{7}$$

If we distribute the items for the critical period in a way such that the slacks for all dispensing sites on the next period are equal, the slacks will be as:

$$S_{i,t^*+1} = S_{it^*} + \frac{\left(\frac{D_{t^*}-D_{t^*-1}-\sum_{i=1}^{n} v_i^{'} q_i}{\sum_{i=1}^{n} q_i}\right) q_i + v_i^{'} q_i - q_i}{q_i} \tag{8}$$

Hence $\left(\frac{D_{t^*}-D_{t^*-1}-\sum_{i=1}^{n} v_i^{'} q_i}{\sum_{i=1}^{n} q_i}\right) q_i + v_i^{'} q_i$ is less than or equal to the q_i, the slack in the period right after critical period is less than or equal to the slack in the critical period.

If we implement this policy for each period, the minimum slack will be occur in the period right after the critical period.

□

Next, using the observation about the critical period, we prove that there is always an optimal solution where the same tour is performed by the vehicle in each period when the slacks at the dispensing sites are equal at the beginning.

Observation 3. *There is always an optimal solution where the same tour is performed in each period when the slacks at the dispensing sites are equal at the beginning.*

Proof. We assume that we have an optimal solution where the minimum slack occurs in the period right after the critical period. Let 1,2,3,...,n be the visiting tour of the dispensing sites. Then, v_i be the visiting times of the dispensing sites in this tour. Let S_{i,t^*+1} be the slack in the period right after the critical period.

Assume that we perform the same tour in the following period. The difference between the total amount of batches and the total daily consumptions up to period t^* is less than or equal to the difference between the total amount of batches and the total daily consumptions up to the period t^*+1 as in the following:

$$D_{t^*} - C_{t^*} \leq D_{t^*+1} - C_{t^*+1} \tag{9}$$

The equation above can be also rewritten in the following equation:

$$D_{t^*+1} - D_{t^*} \geq C_{t^*+1} - C_{t^*} \tag{10}$$

We know that $C_{t^*+1} - C_{t^*}$ is equal to the daily consumption of all the dispensing sites as in the following equation:

$$C_{t^*+1} - C_{t^*} = \sum_{i=1}^{n} q_i \tag{11}$$

Then, the equation (10) can be rewritten as:

$$D_{t^*+1} - D_{t^*} \geq \sum_{i=1}^{n} q_i \tag{12}$$

By using this inequality, we can say that the total amount to be distributed to the dispensing sites in the period right after the critical period are higher than or equal to the daily consumption. Hence, if we deliver $D_{t^*+1} - D_{t^*}$ divided by q_i amount to dispensing site i in the period right after the critical period, the slack in the following the that period will be as:

$$S_{i,t^*+2} = S_{i,t^*+1} + \frac{\left(\frac{D_{t^*+1} - D_{t^*}}{\sum\limits_{i=1}^{n} q_i}\right) q_i - q_i}{q_i} \tag{13}$$

Hence $\left(\frac{D_{t^*+1} - D_{t^*}}{\sum\limits_{i=1}^{n} q_i}\right)$ is higher than or equal to the 1, the slack in period $t^* + 2$ is higher than or equal to the the minimum slack which occurs in the period right after the critical period.

Assume that we perform the same tour in the critical period. As in the following equation, the difference between total amount of batches and total daily consumption up to period t^* is less than or equal to the difference between total amount of batches and total daily consumption up to period up to period $t^* - 1$. So, we can follow the same framework as we explained above.

$$D_{t^*} - C_{t^*} \leq D_{t^*-1} - C_{t^*-1} \tag{14}$$

The equation (14) can also be rewritten again as follows:

$$D_{t^*} - D_{t^*-1} \leq C_{t^*} - C_{t^*-1} \tag{15}$$

We know again that $C_{t^*} - C_{t^*-1}$ is equal to the daily consumption as in the following equation:

$$C_{t^*} - C_{t^*-1} = \sum_{i=1}^{n} q_i \tag{16}$$

Then, the equation (15) can be rewritten as:

$$D_{t^*} - D_{t^*-1} \leq \sum_{i=1}^{n} q_i \tag{17}$$

By using this inequality, we can say that the total amount to be distributed to the dispensing sites in the critical period are less than or equal to the daily consumption. Hence, if we deliver $D_{t^*} - D_{t^*-1}$ divided by q_i amount to dispensing site i, the slack in the period right after the critical period will be as follows:

$$S_{i,t^*+1} = S_{it^*} + \frac{\left(\frac{D_{t^*}-D_{t^*-1}}{\sum_{i=1}^{n} q_i}\right) q_i - q_i}{q_i} \tag{18}$$

Hence $\left(\frac{D_{t^*}-D_{t^*-1}}{\sum_{i=1}^{n} q_i}\right)$ is less than or equal to 1, the slack in the critical period is higher than or equal to the minimum slack in the period right after critical period. □

Based on these observations, we develop a mixed integer mathematical programming model called MIP in order to find the optimal distribution plan using the same tour in each period. Parameters and decision variables are represented as follows:

Parameters

B_t = amount of the relief items delivered to the depot in period t $\quad t \in \mathcal{T}$

q_i = consumption rate at the dispensing site i in units per period $\quad i \in V$

I_i = initial inventory at the dispensing site i $\quad i \in V$

Decision Variables

$$x_{ij} = \begin{cases} 1, & \text{if the vehicle travels from node } i \text{ to node } j \\ 0, & \text{otherwise.} \end{cases} \qquad i,j \in V_0$$

d_{it} = amount of the relief items delivered to the dispensing site i in period t $\quad i \in V, t \in \mathcal{T}$

S_{it} = the minimum slack at the dispensing site i in period t $\quad i \in V, t \in \mathcal{T}$

k_i = visiting time of the dispensing site i in each period $\quad i \in V_0$

u_i = auxiliary variable defined for dispensing site i in order to eliminate subtours $\quad i \in V$

z = the minimum slack over all dispensing sites over all periods

The slack formulation can be represented as follows:

$$\text{MIP: Max} \quad z \tag{19}$$

s.t.

$$\sum_{i \in V} d_{it} \le B_t \quad \forall t \in \mathcal{T} \tag{20}$$

$$S_{i1} = \frac{I_i - k_i \frac{q_i}{L}}{q_i} \quad \forall i \in V \tag{21}$$

$$S_{it} = S_{i,t-1} + \frac{d_{it} - q_i}{q_i} \quad \forall i \in V, t \in \mathcal{T} - \{1\} \tag{22}$$

$$z \le S_{it} \quad \forall i \in V, t \in \mathcal{T} \tag{23}$$

$$\sum_{i \in V_0} x_{ij} = 1 \quad \forall j \in V_0 \tag{24}$$

$$\sum_{j \in V_0} x_{ij} = 1 \quad \forall i \in V_0 \tag{25}$$

$$u_i - u_j + (n+1)x_{ij} \le n \quad \forall i,j \in V \tag{26}$$

$$k_j \ge k_i - M(1 - x_{ij}) + c_{ij} \quad \forall i,j \in V_0 \tag{27}$$

$$k_0 = 0 \tag{28}$$

$$d_{it} \ge 0 \quad \forall i \in V, t \in \mathcal{T} \tag{29}$$

$$k_i \ge 0 \quad \forall i \in V \tag{30}$$

$$u_i \ge 0 \quad \forall i \in V \tag{31}$$

$$x_{ij} \in \{0,1\} \quad \forall i,j \in V_0 \tag{32}$$

The aim of the objective function is to maximize the minimum slack. Constraints (20) ensure that the amount delivered to the dispensing sites in any period cannot be greater than the amount shipped to to depot in that period. Constraints (21) and (22) calculate the minimum slack for each dispensing site in each period. Constrains (23) calculate the overall minimum slack. Constraints (24) and (25) ensure that each dispensing site is visited once. Constraints (26) prevent subtours. Constraints (27) calculate the visiting time for each dispensing site. Constraint (28) ensures that the time when each vehicle leaves the depot is equal to zero. Finally, the integrality and sign restrictions are represented by Constraints between (29)-(32).

4.1.1 Solution Approach for Single Vehicle Version of ISRP

ISRP is a strongly NP-hard problem which takes a long time in order to obtain an exact solution.

Theorem 1. *ISRP is a strongly NP-hard problem.*

Proof. We prove NP-hardness of ISRP by reduction from the Hamiltonian Path problem. We prove that the ISRP is a strongly NP-hard by creating an instance with n dispensing sites and one depot. In this instance, the travel time from the depot to the each dispensing site is M and travel time between two dispensing sites is denoted as c_{ij}. Dispensing sites consume product at a given rate is denoted as q_i in units per minute. Entire items is not available at the beginning of the distribution and items become available in batches throughout T periods. Moreover, each dispensing site has five times of daily consumption rate as initial inventory. In this problem, the aim is to find the maximum minimum slack and the vehicle leaves the depot once a day. When dispensing sites are delivered once a day, the dispensing site is visited recently gives the minimum slack. Thus, our problem turns into the Hamiltonian path problem by visiting each dispensing site exactly once. In literature Hamiltonian Path problem is a strongly NP-hard problem, so our problem is a strongly NP-hard problem. □

Since ISRP is a strongly NP-hard problem, we focus on developing an heuristic algorithm. We propose an algorithm called Iterative Heuristic Algorithm (IHA) to find a good solution for a single version of ISRP. In IHA, based on the observations above we focus on solutions where the same tour is performed in each period. The main idea in IHA is to construct an initial tour and then, improve the tour using the 2-opt and insertion ideas iteratively. First, we construct an initial tour using Clarke and Wright algorithm [50]. Although our objective is to maximize the minimum slack, due to nature of the problem, tours with low traveling time tend to give better results. Hence, we use one of the most popular algorithms developed for VRP. We fix the visiting times for each dispensing site in each period after constructing the initial tour. Let $\bar{k_i}$ be the visiting time for each dispensing site and then $\bar{k_i}$ be the parameter different than decision variable k_i in MIP formulation. Thus, Constraints between (24)-(28) and between (30)-(32) are ignored and the model turns into a new model called LP as follows:

$$\text{LP: Max} \quad z \tag{33}$$

s.t.

$$\sum_{i \in V} d_{it} \leq B_t \qquad \forall t \in \mathcal{T} \tag{34}$$

$$S_{i1} = \frac{I_i - \bar{k_i}\frac{q_i}{L}}{q_i} \qquad \forall i \in V \tag{35}$$

$$S_{it} = S_{i,t-1} + \frac{d_{it} - q_i}{q_i} \qquad \forall i \in V, t \in \mathcal{T} - \{1\} \tag{36}$$

$$z \leq S_{it} \qquad \forall i \in V, t \in \mathcal{T} \tag{37}$$

$$d_{it} \geq 0 \qquad \forall i \in V, t \in \mathcal{T} \tag{38}$$

Then, we solve the LP model to obtain an initial feasible solution and next, we try to improve the initial solution in the improvement part.

4.1.2 Improvement

After constructing an initial solution, we apply two steps in order to improve the initial solution: (i) 2-opt and (ii) insertion.

4.1.2.1 2-opt Step

In 2-opt step, we use 2-opt algorithm in which two nonsuccessive edges is deleted of a tour and then we exchange each pair of edges. We implement these 2-opt exchanges iteratively and we solve the LP model to determine the minimum slack at each step. We calculate the improvement in increasing the minimum slack over all possible pair of edges and choose the one that improves the solution by increasing minimum slack. 2-opt algorithm is implemented until there is no improvement. We illustrate the idea in Figure 5. The steps of the 2-opt is provided in Algorithm 1.

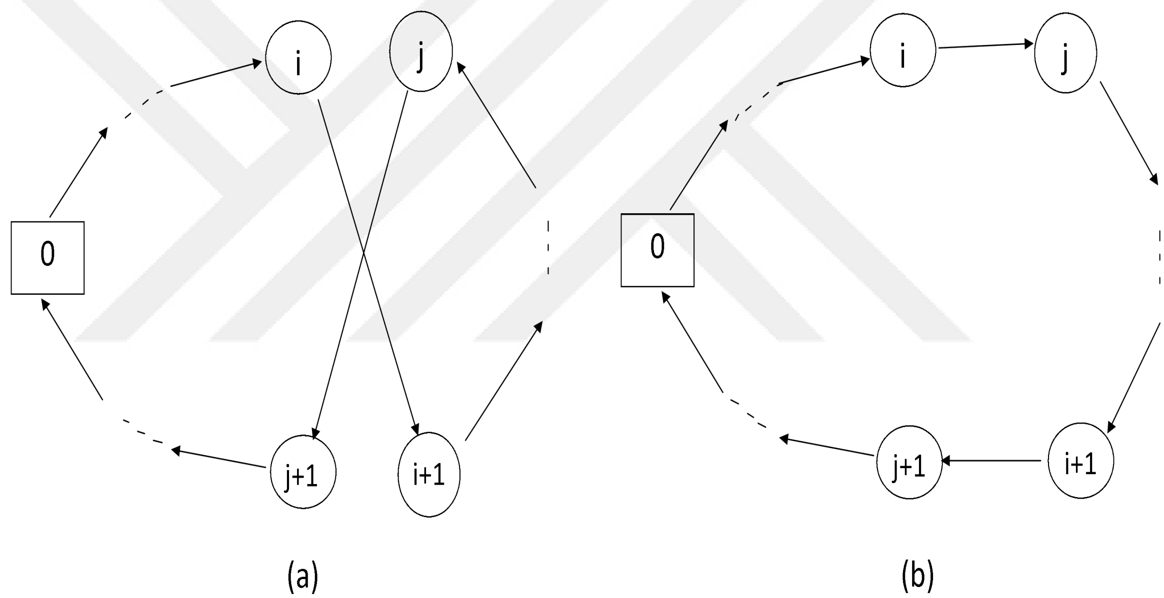

Figure 5: An illustration of 2-opt step: (a) before implementing 2-opt step between pairs of edges (i,i+1) and (j,j+1), (b) after implementing 2-opt step between pairs of edges (i,i+1) and (j,j+1)

4.1.2.2 Insertion Step

In insertion step, we consider all nodes of a tour and delete each of them in order. We try to insert the deleted node into the tour between a consecutive nodes of this tour. We illustrate the idea in Figure 6 where we delete node j of a tour and we try to insert it between two consecutive nodes i and i+1. We consider all possible consecutive nodes to be inserted in a tour and preserve the one that improves the solution by increasing minimum slack.

Algorithm 1 *Steps of the 2-opt*

Input: H = (Constructed tour by Clarke and Wright algorithm for the first step, then H equals the best tour obtained by the second algorithm for other steps)
solve LP to determine the minimum slack (S)
$\alpha = 10000$
while $\alpha > 0$ **do**
    for $i \in V$ **do**
        for $j \in V - \{i\}$ **do**
            $H^{'}$ = (best tour found by using 2-opt)
            solve LP model to determine the minimum slack ($S^{'}$)
            if $S^{'} \geq S$ **then**
                $\alpha = S' - S$
                $S = S^{'}$
                $H = H^{'}$
            end if
        end for
    end for
end while
return H
return S

We implement the algorithm until there is no improvement. The steps of the insertion is provided in Algorithm 2.

4.1.3 Implementation

These two steps are implemented in a cyclic manner as illustrated in Figure 7. First, we start with an initial tour constructed by Clarke and Wright algorithm and then implement 2-opt and insertion algorithms, respectively until there is no improvement. If the improvement does not continue, final solution is returned.

4.2 Multi Vehicle Version of ISRP

In this part, we analyze the multi vehicle version of the ISRP and it is used as a subroutine in two-phase approach, as illustrated in Figure 8. In the first phase, we first cluster the dispensing sites and determine the delivery volume for each cluster. In the second phase, we estimate the minimum slack by determining the delivery routes and schedule for each cluster based on the delivery volume. Then, we solve a set partitioning problem to select the best subset of generated clusters. We implement the two different ideas in the second phase

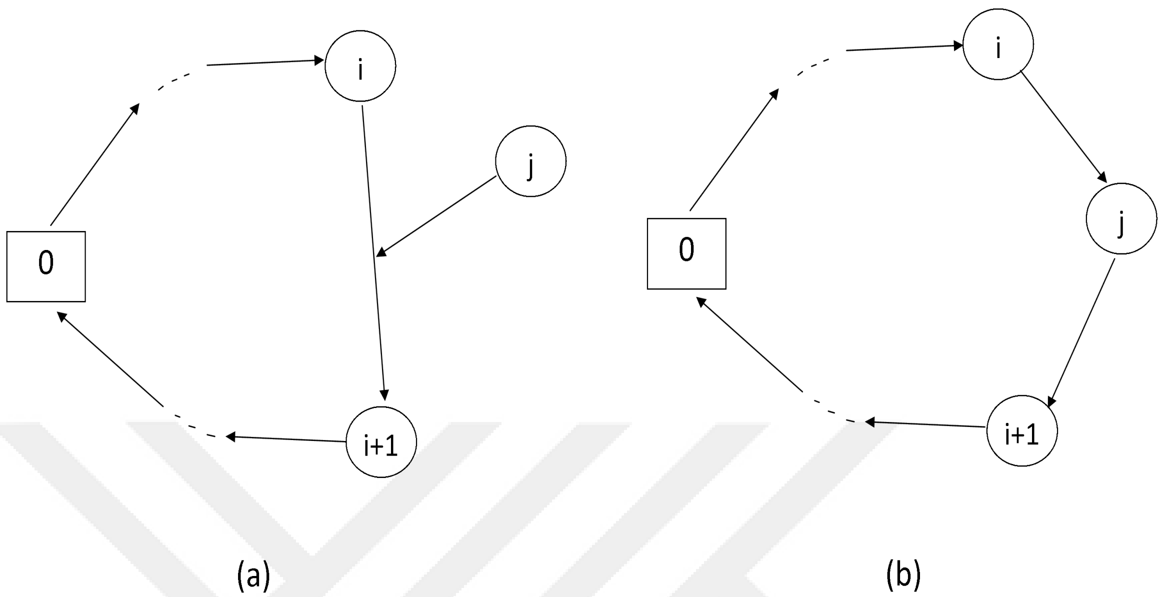

Figure 6: An illustration of insertion step: (a) before inserting removed node j between nodes i and i+1, (b) after inserting node j between nodes i and i+1

Algorithm 2 *Steps of the Insertion*

Input: H = (the tour found at the end of the Algorithm 1), S = (equals the maximum slack obtained in Algorithm 1)
$\gamma = 10000$
while $\gamma \geq 0$ **do**
    for $i \in V$ **do**
        for $j \in V - \{i\}$ **do**
            $H^{'}$ = (best tour found by insertion)
            solve LP model to determine the minimum slack ($S^{'}$)
            if $S' \geq S$ **then**
                $\gamma = S' - S$
                $S = S^{'}$
                $H = H^{'}$
            end if
        end for
    end for
end while
return H
go to the Algorithm 1

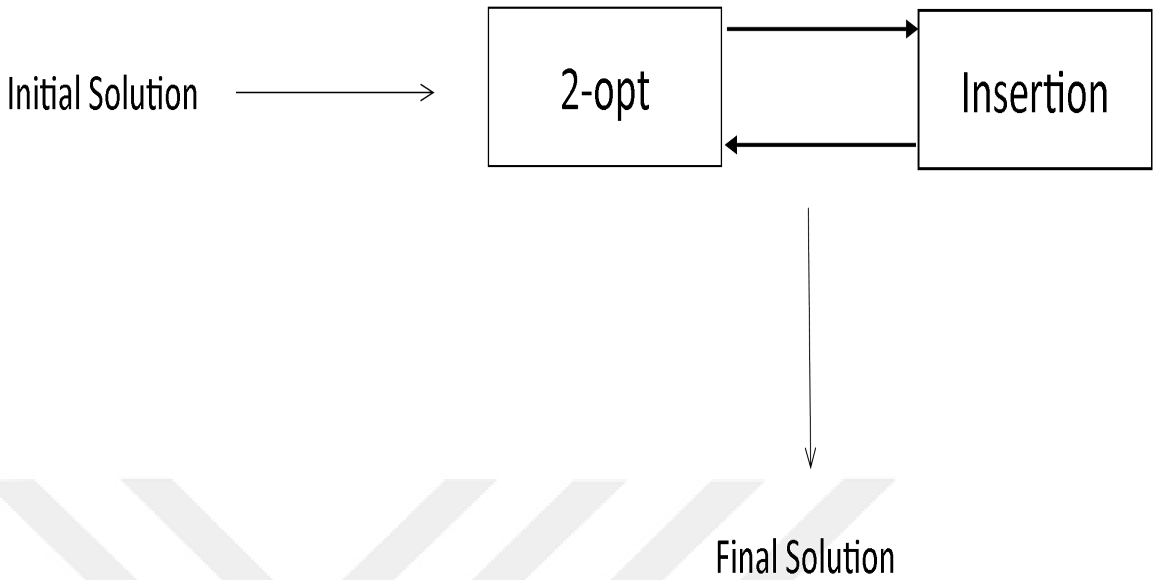

Figure 7: An illustration of the improvement steps implementation

to estimate the minimum slack of each cluster will be explained in the following section. We consider the same assumptions as the single vehicle version of the problem and we explain two-phase approach under these assumptions.

4.2.1 Cluster Generation

We generate a pre-specified number of random clusters which are potentially good. We determine a feasible distribution plan (delivery routes and schedule) to estimate minimum slack for each generated cluster in the second phase, then we solve a set partitioning problem in order to choose a set of clusters among all generated clusters. The steps of the generating clusters are represented in Algorithm 3.

4.2.2 Delivery Routes and Schedule:

In the second phase, we assume that clusters are determined for each vehicle and focus on determining the feasible distribution plan (included delivery routes and schedule) for each cluster. For this aim, we need to determine the delivery volume for each cluster. To do that, we implement the following idea: first, we sum up the consumption rates of the dispensing sites in a given cluster which is denoted as f_Γ and then, we divide this value to the total

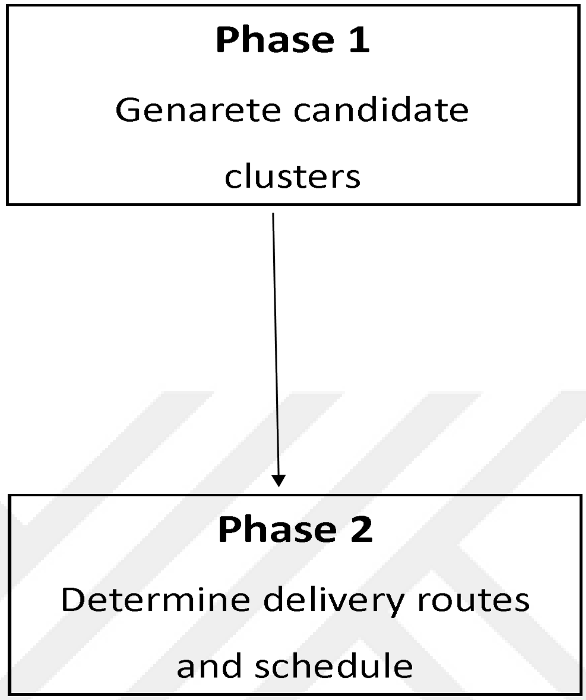

Figure 8: An illustration of the two-phase approach

Algorithm 3 *Clustering* procedure.

Input: An ISRP instance on a region of $D \times D$ mile square dispensing sites and the depot
Output: A partition of the dispensing site set V
repeat
    Pick a random point b, the *base point*, on the map.
    Calculate c_{bi} for all $i \in V$.
    Set the probability p_i of selecting dispensing site i for the cluster to be generated as:

$$p_i \leftarrow \begin{cases} 1 - \sqrt{\frac{c_{bi}}{D}} & \text{if } \; c_{bi} \leq \frac{D}{4}, \\ 0 & \text{otherwise.} \end{cases}$$

    Generate the cluster by selecting dispensing sites based on their respective probabilities. That is, dispensing site i will be in the generated cluster with probability p_i.
until A termination condition is reached.

---

consumption rates for all dispensing sites. Thus, we determine the weighted consumption rate denoted as w_Γ for each cluster. We allocate the relief items based on the weighted consumption rates to each cluster for each period. Delivery volume for a cluster in a period is equal to the relief items in a depot for that period times the weighted consumption rate for that cluster, as provided in Algorithm 4.

The second phase has two stages: (i) construction stage and (ii) improvement stage. In construction stage, we determine the delivery routes and amount of the relief items to be delivered to each dispensing site. In the second stage, we try to increase minimum slack by improving delivery schedule.

Algorithm 4 *Delivery volume for each cluster for a period*

1: **Input:** $\Gamma = given\ \ cluster$
2: calculate the total consumption rates for cluster Γ $(f_\Gamma = \sum\limits_{i\in\Gamma} q_i)$
3: calculate the weighted consumption rates for cluster Γ $(w_\Gamma = \frac{f_\Gamma}{\sum\limits_{i\in V} q_i})$
4: calculate the delivery volume for cluster Γ for each period ($Z_{\Gamma t} = B_t w_\Gamma$)

For the first idea, we assume that each cluster turns into the ISRP with a single vehicle version and solve the each cluster using the IHA. The only difference is that each cluster has the relief items which equal to the its delivery volume. Then, we implement the IHA to determine the delivery routes and schedule, but there is an only difference in LP model in terms of the amount of delivered items to the dispensing sites for each cluster in a period. We modify the constraint (34) in LP model and we call the new model as LP1 as in the following:

$$\text{LP1: Max} \quad z \tag{39}$$

s.t.

$$\sum_{i\in\Gamma} d_{it} \leq Z_{\Gamma t} \qquad \forall t \in \mathcal{T} \tag{40}$$

$$S_{i1} = \frac{I_i - \bar{k}_i \frac{q_i}{L}}{q_i} \qquad \forall i \in \Gamma \tag{41}$$

$$S_{it} = S_{i,t-1} + \frac{d_{it} - q_i}{q_i} \qquad \forall i \in \Gamma, t \in \mathcal{T} - \{1\} \tag{42}$$

$$z \leq S_{it} \qquad \forall i \in \Gamma, t \in \mathcal{T} \tag{43}$$

$$d_{it} \geq 0 \qquad \forall i \in \Gamma, t \in \mathcal{T} \tag{44}$$

For the second idea, we consider different approach rather than IHA for each generated cluster. Unlike IHA, we solve a mixed integer programming similar to MIP to determine the delivery routes and schedule. We modify the constraint (20) in MIP model since each generated cluster has the relief items which equal to the its delivery volume and we call the new model as MIP1 as in the following:

$$\text{MIP1: Max} \quad z \tag{45}$$

s.t.

$$\sum_{i \in \Gamma} d_{it} \leq Z_{\Gamma t} \quad \forall t \in \mathcal{T} \tag{46}$$

$$S_{i1} = \frac{I_i - k_i \frac{q_i}{L}}{q_i} \quad \forall i \in \Gamma \tag{47}$$

$$S_{it} = S_{i,t-1} + \frac{d_{it} - q_i}{q_i} \quad \forall i \in \Gamma, t \in \mathcal{T} - \{1\} \tag{48}$$

$$z \leq S_{it} \quad \forall i \in \Gamma, t \in \mathcal{T} \tag{49}$$

$$\sum_{i \in \Gamma} x_{ij} = 1 \quad \forall j \in \Gamma \tag{50}$$

$$\sum_{j \in \Gamma} x_{ij} = 1 \quad \forall i \in \Gamma \tag{51}$$

$$u_i - u_j + (n^{'} + 1) x_{ij} \leq n^{'} \quad \forall i, j \in \Gamma \tag{52}$$

$$k_j \geq k_i - M(1 - x_{ij}) + c_{ij} \quad \forall i, j \in \Gamma \tag{53}$$

$$k_0 = 0 \tag{54}$$

$$d_{it} \geq 0 \quad \forall i \in \Gamma, t \in \mathcal{T} \tag{55}$$

$$k_i \geq 0 \quad \forall i \in \Gamma \tag{56}$$

$$u_i \geq 0 \quad \forall i \in \Gamma \tag{57}$$

$$x_{ij} \in \{0, 1\} \quad \forall i, j \in \Gamma \tag{58}$$

4.2.2.1 Set Partitioning Problem

In this part, we solve a set partitioning problem (SPP) in order to select the best subset of generated clusters. The set of the all generated clusters are labeled as J. We use ϑ_{ij} denote the parameter of the dispensing site i to determine whether it is selected in cluster j or not.

Parameters

$$\vartheta_{ij} = \begin{cases} 1, & \text{if the dispensing site } i \text{ is selected in generated cluster } j \\ 0, & \text{otherwise.} \end{cases} \quad i \in V, j \in J$$

S_j = estimated minimum slack value of the generated cluster j $\quad j \in J$

Decision Variables

$$y_j = \begin{cases} 1, & \text{if the generated cluster } j \text{ is selected} \\ 0, & \text{otherwise.} \end{cases} \qquad j \in J$$

$\nu =$ the minimum slack over all the generated clusters

The set partitioning formulation can be represented as follows:

$$\text{SPP: Max} \quad \nu \tag{59}$$

s.t.

$$\sum_{j \in J} \vartheta_{ij} y_j = 1 \qquad \forall i \in V \tag{60}$$

$$\sum_{j \in J} y_j \leq K \tag{61}$$

$$S_j y_j - M y_j - \nu \geq -M \quad \forall j \in J \tag{62}$$

$$y_j \in \{0, 1\} \qquad \forall j \in J \tag{63}$$

4.2.2.2 Improvement Stage

At this stage, delivery schedule constructed in the first stage is tried to improve and we keep the orders of the routes same. Now, a set of clusters are determined by solving the set partitioning problem and then, we fix the visiting time of each dispensing site (denoted as $k_i^{'}$) for each period. Let $k_i^{'}$ be the visiting time of each dispensing site and it is the parameter in the improvement model. We improve the delivery volume for each cluster and solve the problem again based on the model in the following:

$$\text{Max} \quad z \tag{64}$$

s.t.

$$\sum_{i \in V} d_{it} \leq B_t \qquad \forall t \in \mathcal{T} \tag{65}$$

$$S_{i1} = \frac{I_i - k_i' \frac{q_i}{L}}{q_i} \qquad \forall i \in V \tag{66}$$

$$S_{it} = S_{i,t-1} + \frac{d_{it} - q_i}{q_i} \qquad \forall i \in V, t \in \mathcal{T} - \{1\} \tag{67}$$

$$z \leq S_{it} \qquad \forall i \in V, t \in \mathcal{T} \tag{68}$$

$$d_{it} \geq 0 \qquad \forall i \in V, t \in \mathcal{T} \tag{69}$$

CHAPTER V

COMPUTATIONAL STUDY

In this chapter, we conduct an computational study on randomly generated instances in order to assess the performance of the all proposed algorithms in terms of solution quality. We classify the instances according to their sizes and divide into two categories: small and large instances. For the small instances, we assume there is only a single vehicle available for serving the dispensing sites and hence we solve the single vehicle version of the ISRP and however, for the large instances, we solve the generic version of the ISRP with multiple vehicles. In small instances, we compare our solutions with the corresponding optimal solution as these problems are small enough to be solved to optimality using a commercial solver.

In large instances, we assess the performance of our proposed two-phase approach in comparison with the two existing algorithms in the literature, namely DVI Algorithm [46] and Improved Heuristic [49]. The benchmark of this comparison is an upper bound developed for the ISRP, hopefully tight enough to assess the performances accurately. In order to identify such an upper bound, we assume a sufficient number vehicles for each instance so that we can dedicated a vehicle for each dispensing site at each period. Thus, each vehicle visits only one dispensing site at each period and the visiting time of each dispensing site equals to the travel time between depot and that particular dispensing site. We calculate the percentage gap of the algorithms, our two-phase approach, DVI Algorithm, and Improved Heuristic, with respect to this upper bound solution and compare the performances based on these relative gap values.

5.1 Generating Instances

We performed our experiments on randomly generated 30 small instances and 30 large instances. The small instances are generated on a region of 1,000 $\times$ 1,000 mile square with

8, 9, or 10 dispensing sites to serve. The large instances are also generated on a region of 1,000 $\times$ 1,000 mile square with 30, 40, or 50 dispensing sites to serve.

In both the small and the large instances, the dispensing sites have certain consumption rates, which represent the demand in units per period for each dispensing site. Only single type of relief item is assumed to be delivered to the dispensing sites and the consumption rates of the relief items at each dispensing site are generated randomly between [0,144000]. We assume that the storage capacity of the dispensing sites is infinite and each dispensing site is visited only once by a vehicle at each period. We also assume that the deliveries of relief items to the depot occur at the beginning of the each period and the delivery volumes are generated uniformly between 80% and 120% of the total daily consumption of the all dispensing sites for each instance. Note that the delivery volume to the depot at each period may vary significantly complicating the problem further. The travel time between the sites (the dispensing sites and the depot) are based on the Euclidean distances between the sites and we assume that the vehicles travel with a constant speed of 10. Next, we calculate the travel times between sites by dividing the distances by this constant speed.

As mentioned before, we construct a feasible distribution plan over a planning horizon to deliver to the dispensing sites and the planning horizon is equal to the T periods. We assume that each vehicle repeats the same tour over these T periods. Our objective is to maximize the minimum slack over T period. In our experiments, we assume that the planning horizon has 15 periods, $T = 15$. The reason for selecting only a single value for T and not performing a sensitivity analysis is that the planning horizon does not have a significant effect on the solution time or the quality of the proposed solution.

5.2 Experimental Results

In this part, we present the results of our computational experiments. First, we test the performance of Iterative Heuristic Algorithm with respect to the exact solution found by solving MIP model in small instances. Next, we test the performance of our two-phase approach in large instances with respect to the upper bound and compare the relative performance to DVI Algorithm and Improved Heuristic.

For small instances, we solve the MIP model using ILOG CPLEX 12.5.1 solver and implement Iterative Heuristic Algorithm using $C++$ and ILOG CPLEX 12.5.1 solver. For each instance, we manage to identify a feasible solution using our proposed approach and the corresponding optimality gaps values are presented in Table 5.

According to the results, we observe that the instances are solved to optimality in a few minutes using the commercial solver and the average optimality gap of the solutions provided by IHA is only 0.93% and less than 3% in the worst case. The average computational time of IHA is about 4 minutes which is a reasonable time to solve the small instances. Thus, we conclude that Iterative Heuristic Algorithm is very effective in solving the small instances of ISRP.

Next, we test the performance of our two-phase approach in comparison with the two existing algorithms in the literature. The basis of comparison is the relative gap value with respect to the upper bound. In our two-phase approach, we implement two different subroutines, namely Clustering based-Iterative Heuristic Algorithm (C-IHA) and Clustering based-MIP1 (C-MIP1), in the routing phase and obtain two sets of results. The algorithms are implemented using $C++$ and ILOG CPLEX 12.5.1 solver. In our instances, the dispensing sites have an initial inventory value equal to the daily consumption value. For every base instance, we generate three more copies, namely varying initial inventory instance, high distance instance and varying delivery volume instance. In varying initial inventory instances so as to observe the effects of having dispensing sites with varying characteristics, we generate the initial inventory values randomly; either 100% or 200% of the daily consumption value. In high distance instances, we double the distances among the sites in order to test how the "distance factor" affects the performance of our solution methodology. In varying delivery volume instances, we generate the delivery volume values uniformly between 60% and 110% of the total daily consumption of the all dispensing sites. Finally, we also perform a sensitivity analysis on the number vehicles in order to assess the effects of vehicle availability on the performance of proposed and benchmark algorithms.

Table 6 provides a summary of the results on the original test instances. In Table 6 the first column is the instance number, the second column is the number of dispensing

Table 5: The optimality gap of Iterative Heuristic Algorithm on small instances

Instance	# of Disp. Sites	IHA
1	8	0.00%
2	8	0.39%
3	8	0.60%
4	8	1.19%
5	8	0.00%
6	8	0.34%
7	8	2.89%
8	8	0.46%
9	8	0.00%
10	9	0.31%
11	9	0.52%
12	9	0.00%
13	9	0.53%
14	9	2.23%
15	9	1.90%
16	9	0.94%
17	9	0.13%
18	9	1.99%
19	9	0.40%
20	9	0.71%
21	10	1.83%
22	10	1.30%
23	10	1.21%
24	10	0.00%
25	10	1.29%
26	10	2.10%
27	10	0.63%
28	10	1.17%
29	10	1.04%
30	10	1.77%

sites served, and, the third column is the number of vehicles available in the corresponding instance. The fourth column (C-IHA) presents the gap of the solution obtained by Clustering based- IHA with respected to the upper bound. Similarly, the fifth (C-MIP1), the sixth (DVI Alg.) and the seventh (Impr. Heurs.) columns present the gap values of the algorithms, Clustering based- IHA, Clustering based-MIP1, DVI Algorithm and Improved Heuristic, respectively. Each row of Table 6 corresponds to a different instance of the problem. We first observe that all of the algorithms manage to find a feasible solution for each instance. The average upper bound gap of the C-IHA is about 2.38% (5.30% in the worst case) and the average computational times is about 24.28 minutes (61.36 minutes in the worst case) over all the instances. It provides a better solution in 17 out of 30 instances compared to DVI Algorithm and Improved Heuristic. We observe that the average upper bound gap of the C-MIP1 is about 1.05% (1.99% in the worst case) and the average computational times is about 36.05 minutes (71.45 minutes in the worst case) over all the instances. It provides a better solution in all of the instances compared to DVI Algorithm and Improved Heuristic. The average upper bound gap values of DVI Algorithm and Improved Heuristic are 3.28% (6.39% in the worst case) and 3.70% (6.69% in the worst case), respectively and the average computational times of DVI Algorithm and Improved Heuristic are 12.23 and 11.74 minutes (14.29 and 13.46 minutes in the worst case), respectively. Therefore, in the original instances, our proposed algorithms (especially Clustering based-MIP1) significantly outperform the algorithms proposed in the literature.

Next, we decrease the numbers of available vehicles for each of the original instances and present the results in Table 7. Even though the average upper bound gap values the C-IHA and C-MIP1 have increased, 3.10% and 1.73% (6.40% and 3.31% in the worst case), they still outperform DVI Algorithm and Improved Heuristic with the average upper bound gap values of 3.72% and 4.49% (6.84% and 7.73% in the worst case), respectively. C-IHA manages to find a better solution in 20 out of 30 instances and once again C-MIP1 finds a better solution in all the instances compared to DVI Algorithm and Improved Heuristic. The average computational times for C-IHA and C-MIP1 are 62.28 and 157.34 minutes (2.11 and 7.07 hours in the worst case) and the average computational times for DVI Algorithm

Table 6: The performance of the algorithms on the original test instances

Instance	# of Disp. Sites	# of Vehicles	C-IHA	C-MIP1	DVI Alg.	Impr. Heurs.
1	30	6	0.21%	0.21%	3.29%	5.43%
2	30	6	3.09%	1.10%	3.86%	3.31%
3	30	6	1.49%	1.49%	5.00%	6.20%
4	30	6	4.11%	1.37%	1.56%	3.03%
5	30	6	3.62%	0.93%	1.95%	1.77%
6	30	6	0.60%	0.53%	4.06%	6.02%
7	30	6	2.99%	1.11%	3.16%	2.65%
8	30	6	2.80%	1.45%	3.07%	3.16%
9	30	6	1.26%	1.11%	3.55%	5.47%
10	30	6	3.39%	1.24%	3.39%	2.94%
11	40	8	2.21%	0.95%	2.37%	2.53%
12	40	8	0.79%	0.39%	1.86%	1.96%
13	40	8	0.30%	0.30%	6.39%	6.69%
14	40	8	1.74%	1.36%	5.36%	3.63%
15	40	8	1.88%	0.89%	1.78%	3.46%
16	40	8	4.49%	1.44%	4.33%	6.25%
17	40	8	5.30%	1.99%	5.30%	5.74%
18	40	8	2.79%	1.21%	2.33%	3.07%
19	40	8	2.16%	1.29%	4.17%	4.17%
20	40	8	0.22%	0.22%	5.81%	3.65%
21	50	10	0.74%	0.15%	2.44%	5.62%
22	50	10	1.30%	0.80%	2.81%	1.90%
23	50	10	1.86%	0.88%	1.86%	2.55%
24	50	10	3.76%	1.80%	2.94%	3.10%
25	50	10	3.65%	1.82%	4.48%	2.82%
26	50	10	2.63%	0.63%	1.90%	2.63%
27	50	10	3.12%	1.40%	3.28%	2.81%
28	50	10	2.91%	1.30%	2.11%	2.31%
29	50	10	1.44%	0.83%	1.89%	2.04%
30	50	10	4.45%	1.17%	1.99%	4.10%

and Improved Heuristic is 15.06 and 13.52 minutes (16.48 and 15.24 minutes in the worst case), respectively.

The instances with varying characteristics dispensing sites presents a different challenge as the initial inventory affects the routing decision in the earlier periods in the planning horizon. We present the results in Table 8. The performances of the proposed algorithm improves in these instances and the main reason is the upper bound becomes tighter. Specifically, in 15 out of 30 instances our proposed mechanism manage the find the same minimum slack value with the upper bounding mechanism hence the upper bound gap of the solution is 0.0%. This means that the minimum slack occurs in a dispensing site, which is visited first by a vehicle. Since, the initial inventory at dispensing sites are not uniform in these instances, some dispensing site may one-day of supply in their inventories, whereas the others have two-day of supply. The dispensing site, which has the lowest initial inventory value, is visited first to increase its slack value immediately and hence the minimum slack value is equal to the that of the upper bounding mechanism. Due to this phenomena, not surprisingly, the average upper bound gap values the C-IHA and C-MIP1 have decreased, 1.04% and 0.61% (3.10% and 2.09% in the worst case), compared to the original instances. Also, they still outperform DVI Algorithm and Improved Heuristic with the average upper bound gap values of 2.34 and 3.28 (6.07% and 9.60% in the worst case), respectively. C-IHA manages to find a better solution in 24 out of 30 instances and C-MIP1 finds a better solution in 29 out of 30 instances compared to DVI Algorithm and Improved Heuristic. The average computational time of C-IHA and C-MIP1 are 21.25 and 32.10 minutes (29.06 and 52.28 minutes in the worst case), respectively. The average computational time of DVI Algorithm and Improved Heuristic are 11.92 and 11.24 minutes (13.46 and 12.18 minutes in the worst case), respectively.

Next, we decrease the numbers of available vehicles for each of the varying initial inventory instances and present the results in Table 9. Even though the average upper bound gap values the C-IHA and C-MIP1 have increased once again, 1.36% and 0.77% (3.84% and 2.28% in the worst case), they still outperform DVI Algorithm and Improved Heuristic with the average upper bound gap values of 2.79 and 4.26 (6.07% and 7.77% in the worst

Table 7: The performance of the algorithms on the original test instances with decreased number of vehicles

Instance	# of Disp. Sites	# of Vehicles	C-IHA	C-MIP1	DVI Alg.	Impr. Heurs.
1	30	5	0.43%	0.29%	4.57%	6.29%
2	30	5	3.53%	1.43%	4.08%	3.86%
3	30	5	1.79%	1.79%	3.58%	6.57%
4	30	5	4.59%	1.86%	2.35%	3.42%
5	30	5	3.72%	1.21%	2.32%	3.25%
6	30	5	1.58%	1.50%	4.14%	6.32%
7	30	5	3.50%	1.37%	3.93%	3.50%
8	30	5	3.16%	1.90%	3.44%	4.43%
9	30	5	2.51%	2.36%	3.91%	4.73%
10	30	5	4.41%	1.58%	3.96%	4.30%
11	40	6	2.29%	1.10%	2.21%	2.92%
12	40	6	1.67%	0.49%	0.59%	1.67%
13	40	6	1.34%	1.34%	4.31%	6.17%
14	40	6	2.57%	2.04%	3.17%	2.87%
15	40	6	2.67%	1.78%	3.26%	4.74%
16	40	6	5.93%	2.56%	4.81%	6.89%
17	40	6	6.40%	3.31%	6.84%	7.73%
18	40	6	3.72%	1.86%	3.35%	4.10%
19	40	6	3.02%	2.16%	3.74%	4.02%
20	40	6	2.08%	2.08%	3.80%	4.10%
21	50	7	2.59%	1.85%	4.14%	4.88%
22	50	7	1.80%	1.30%	4.01%	3.41%
23	50	7	2.06%	1.18%	3.43%	4.42%
24	50	7	3.76%	2.12%	3.43%	3.92%
25	50	7	4.64%	2.99%	6.80%	4.98%
26	50	7	3.44%	1.27%	2.26%	4.26%
27	50	7	3.12%	2.03%	5.15%	4.84%
28	50	7	3.31%	1.60%	3.41%	4.31%
29	50	7	2.11%	1.36%	2.19%	2.42%
30	50	7	5.15%	2.11%	4.33%	5.39%

Table 8: The performance of the algorithms on the varying initial inventory test instances

Instance	# of Disp. Sites	# of Vehicles	C-IHA	C-MIP1	DVI Alg.	Impr. Heurs.
1	30	6	0.00%	0.00%	5.75%	4.07%
2	30	6	2.80%	1.21%	2.88%	3.18%
3	30	6	0.00%	0.00%	1.91%	4.48%
4	30	6	2.51%	2.09%	1.26%	5.09%
5	30	6	2.34%	0.80%	4.46%	1.54%
6	30	6	0.00%	0.00%	6.07%	6.85%
7	30	6	0.00%	0.00%	0.00%	0.00%
8	30	6	0.00%	0.00%	0.00%	5.51%
9	30	6	0.00%	0.00%	0.34%	7.02%
10	30	6	1.75%	0.95%	2.54%	2.15%
11	40	8	0.00%	0.00%	4.97%	6.13%
12	40	8	0.00%	0.00%	1.59%	1.15%
13	40	8	0.00%	0.00%	1.15%	3.44%
14	40	8	0.00%	0.00%	0.89%	1.64%
15	40	8	1.58%	1.21%	1.58%	2.94%
16	40	8	3.10%	1.50%	2.90%	3.60%
17	40	8	2.28%	1.80%	3.11%	3.23%
18	40	8	0.00%	0.00%	0.07%	0.37%
19	40	8	1.82%	0.96%	2.98%	2.88%
20	40	8	0.00%	0.00%	5.57%	4.48%
21	50	10	0.00%	0.00%	4.00%	1.24%
22	50	10	3.03%	1.41%	2.22%	1.70%
23	50	10	1.02%	0.65%	0.87%	0.73%
24	50	10	2.11%	1.48%	2.22%	2.64%
25	50	10	1.95%	1.33%	2.98%	2.16%
26	50	10	0.00%	0.00%	0.07%	5.70%
27	50	10	2.10%	1.20%	2.20%	2.20%
28	50	10	1.00%	0.86%	1.65%	1.79%
29	50	10	0.00%	0.00%	2.39%	4.29%
30	50	10	1.89%	0.86%	1.63%	3.44%

case), respectively. C-IHA manages to find a better solution in 20 out of 30 instances and C-MIP1 finds a better solution in 29 out of 30 instances compared to DVI Algorithm and Improved Heuristic. The average computational times of C-IHA and C-MIP1 are 57.34 and 146.13 minutes (2.03 and 6.43 hours in worst case). Also, the average computational times of DVI Algorithm and Improved Heuristic are 13.59 and 13.02 minutes (15.48 and 14.41 minutes in the worst case), respectively.

We test the performance of the proposed algorithm in high distance instances and present the results in Table 10. The performances of the proposed algorithm in these instances compared to the original instances. The average upper bound gap values the C-IHA and C-MIP1 have, 3.63% and 2.26% (8.76% and 5.67% in the worst case), compared to the original instances. Also, they still outperform DVI Algorithm and Improved Heuristic with the average upper bound gap values of 6.37% and 7.04% (9.33% and 10.73% in the worst case), respectively. C-IHA manages to find a better solution in 26 out of 30 instances and C-MIP1 finds a better solution in all the instances compared to DVI Algorithm and Improved Heuristic. The average computational times of C-IHA and C-MIP1 are 29.54 and 64.03 minutes (66.48 and 159.07 minutes in the worst case), respectively. The average computational times of DVI Algorithm and Improved Heuristic are 16.55 and 15.44 minutes (17.38 and 16.52 minutes in the worst case), respectively.

Next, we decrease the numbers of available vehicles for each of the high distance instances and present the results in Table 11. Even though the average upper bound gap values the C-IHA and C-MIP1 have increased once again, 5.17% and 3.70% (10.57% and 7.99% in the worst case), they still outperform DVI Algorithm and Improved Heuristic with the average upper bound gap values of 7.75 and 8.28% (10.35% and 11.78% in the worst case), respectively. C-IHA manages to find a better solution in 26 out of 30 instances and C-MIP1 finds a better solution in all of the instances compared to DVI Algorithm and Improved Heuristic. The average computational times of C-IHA and C-MIP1 are 65.4 and 154.26 minutes (2.56 and 7.49 hours in the worst case), respectively. The average computational times of DVI Algorithm and Improved Heuristic are 16.34 and 14.42 minutes (17.56 and 16.23 minutes in the worst case), respectively.

Table 9: The performance of the algorithms on the varying initial inventory test instances with decreased number of vehicles

Instance	# of Disp. Sites	# of Vehicles	C-IHA	C-MIP1	DVI Alg.	Impr. Heurs.
1	30	5	0.00%	0.00%	5.75%	4.70%
2	30	5	3.03%	1.51%	2.95%	5.90%
3	30	5	0.00%	0.00%	3.89%	5.72%
4	30	5	3.70%	2.23%	1.74%	5.79%
5	30	5	2.70%	1.10%	4.61%	1.54%
6	30	5	0.00%	0.00%	6.07%	7.20%
7	30	5	0.00%	0.00%	5.80%	0.00%
8	30	5	0.00%	0.00%	0.00%	6.89%
9	30	5	0.00%	0.00%	0.34%	7.77%
10	30	5	2.31%	1.19%	2.86%	2.94%
11	40	6	0.00%	0.00%	2.02%	7.35%
12	40	6	0.00%	0.00%	1.59%	2.59%
13	40	6	0.00%	0.00%	2.72%	4.23%
14	40	6	0.00%	0.00%	0.00%	3.20%
15	40	6	2.49%	1.51%	2.56%	3.77%
16	40	6	3.70%	1.70%	3.10%	5.40%
17	40	6	2.99%	2.28%	3.83%	4.07%
18	40	6	0.00%	0.00%	0.59%	1.04%
19	40	6	3.84%	1.44%	2.59%	4.61%
20	40	6	0.00%	0.00%	2.75%	5.71%
21	50	7	0.00%	0.00%	4.00%	2.40%
22	50	7	2.51%	1.78%	3.11%	3.40%
23	50	7	1.38%	0.80%	1.96%	2.61%
24	50	7	2.54%	1.90%	2.33%	4.02%
25	50	7	2.87%	1.75%	4.31%	4.52%
26	50	7	0.00%	0.00%	0.87%	3.89%
27	50	7	2.10%	1.70%	3.29%	3.39%
28	50	7	2.01%	1.08%	2.51%	3.37%
29	50	7	0.00%	0.00%	2.53%	4.64%
30	50	7	2.75%	1.20%	3.18%	5.15%

Table 10: The performance of the algorithms on the high distance test instances

Instance	# of Disp. Sites	# of Vehicles	C-IHA	C-MIP1	DVI Alg.	Impr. Heurs.
1	30	6	0.45%	0.30%	5.22%	6.80%
2	30	6	5.06%	2.94%	7.29%	6.82%
3	30	6	3.88%	3.88%	6.79%	8.48%
4	30	6	6.05%	3.28%	3.49%	6.46%
5	30	6	4.65%	1.94%	4.46%	3.68%
6	30	6	1.23%	1.23%	8.11%	10.73%
7	30	6	4.43%	2.44%	6.69%	6.15%
8	30	6	4.46%	3.13%	6.65%	6.17%
9	30	6	2.99%	2.44%	7.80%	9.54%
10	30	6	4.54%	2.63%	7.53%	6.33%
11	40	8	3.11%	1.68%	5.64%	5.89%
12	40	8	1.35%	0.72%	4.76%	5.28%
13	40	8	0.64%	0.48%	8.64%	8.96%
14	40	8	2.23%	1.41%	6.70%	9.84%
15	40	8	3.01%	1.87%	4.15%	7.89%
16	40	8	5.98%	3.16%	7.91%	9.49%
17	40	8	8.76%	5.67%	7.22%	8.25%
18	40	8	4.23%	2.46%	5.12%	6.59%
19	40	8	3.67%	2.29%	9.33%	9.02%
20	40	8	0.48%	0.32%	6.99%	4.58%
21	50	10	1.03%	0.24%	5.07%	8.80%
22	50	10	2.19%	1.56%	6.15%	4.48%
23	50	10	2.70%	1.66%	4.26%	3.32%
24	50	10	5.36%	3.93%	7.32%	7.14%
25	50	10	4.87%	3.25%	9.03%	7.04%
26	50	10	3.37%	1.54%	4.23%	4.62%
27	50	10	4.74%	2.71%	7.28%	6.43%
28	50	10	4.32%	2.85%	4.64%	4.85%
29	50	10	3.13%	2.74%	7.91%	8.85%
30	50	10	6.05%	2.90%	4.67%	8.58%

Table 11: The performance of the algorithms on the high distance test instances with decreased number of vehicles

Instance	# of Disp. Sites	# of Vehicles	C-IHA	C-MIP1	DVI Alg.	Impr. Heurs.
1	30	5	0.68%	0.45%	6.88%	7.26%
2	30	5	6.00%	3.76%	9.18%	8.71%
3	30	5	4.68%	4.36%	7.84%	8.89%
4	30	5	6.36%	3.49%	5.13%	7.79%
5	30	5	4.65%	2.42%	5.04%	3.97%
6	30	5	3.52%	3.28%	9.17%	11.06%
7	30	5	4.88%	2.89%	7.32%	6.96%
8	30	5	5.22%	3.61%	7.03%	6.74%
9	30	5	5.44%	5.12%	9.38%	10.32%
10	30	5	6.21%	3.46%	8.36%	7.77%
11	40	6	5.89%	4.55%	10.35%	10.69%
12	40	6	1.86%	0.62%	5.07%	5.80%
13	40	6	2.64%	2.64%	9.20%	10.88%
14	40	6	3.23%	2.40%	7.03%	10.34%
15	40	6	4.88%	3.74%	6.96%	10.28%
16	40	6	8.61%	5.45%	9.67%	11.78%
17	40	6	10.57%	7.99%	9.02%	10.31%
18	40	6	5.81%	3.84%	7.09%	7.68%
19	40	6	5.35%	4.43%	9.79%	9.48%
20	40	6	4.34%	4.34%	8.19%	6.83%
21	50	7	4.52%	3.65%	8.64%	10.62%
22	50	7	3.54%	2.81%	8.44%	7.60%
23	50	7	3.53%	2.39%	7.27%	6.85%
24	50	7	6.96%	5.18%	7.86%	7.32%
25	50	7	7.58%	5.60%	9.39%	8.66%
26	50	7	4.81%	2.60%	4.81%	5.29%
27	50	7	5.58%	4.23%	8.12%	6.94%
28	50	7	5.17%	3.27%	7.17%	6.12%
29	50	7	4.86%	3.99%	6.03%	6.26%
30	50	7	7.69%	4.41%	7.19%	9.08%

Finally, we test the performance of the proposed algorithms on the varying delivery volume test instances and present the results in Table 12. The performances of the proposed algorithm in these instances compared to the original instances. The average upper bound gap values the C-IHA and C-MIP1 have, 2.83% and 1.56% (9.11% and 8.21% in the worst case), compared to the original instances. Also, they still outperform DVI Algorithm and Improved Heuristic with the average upper bound gap values of 3.30% and 3.67% (10.14% and 10.67% in the worst case), respectively. C-IHA manages to find a better solution in 17 out of 30 instances and C-MIP1 finds a better solution in all the instances compared to DVI Algorithm and Improved Heuristic. The average computational times of C-IHA and C-MIP1 are 21.03 and 32.54 minutes (58.14 and 65.18 minutes in the worst case), respectively. The average computational times of DVI Algorithm and Improved Heuristic are 11.40 and 11.08 minutes (16.52 and 15.11 minutes in the worst case), respectively.

We again decrease the numbers of available vehicles for each of the varying delivery volume instances and present the results in Table 13. Even though the average upper bound gap values the C-IHA and C-MIP1 have increased once again, 3.39% and 1.97% (9.66% and 8.70% in the worst case), they still outperform DVI Algorithm and Improved Heuristic with the average upper bound gap values of 3.75 and 4.48% (10.42% and 10.78% in the worst case), respectively. C-IHA manages to find a better solution in 17 out of 30 instances and C-MIP1 finds a better solution in 29 out of 30 instances compared to DVI Algorithm and Improved Heuristic. The average computational times of C-IHA and C-MIP1 are 59.23 and 148.39 minutes (2.02 and 6.58 hours in the worst case), respectively. The average computational times of DVI Algorithm and Improved Heuristic are 15.17 and 14.29 minutes (16.32 and 15.09 minutes in the worst case), respectively.

In summary, our proposed solution approaches outperform both DVI Algorithm and Improved Heuristic in all of the problem variants with different characteristics. The main reason is that we solve the routing and delivery schedule stages in an integrated manner as the clustering decisions given in the first stage significantly affect the decisions given in the routing stage and hence the solution quality.

Table 12: The performance of the algorithms on the varying delivery volume test instances

Instance	# of Disp. Sites	# of Vehicles	C-IHA	C-MIP1	DVI Alg.	Impr. Heurs.
1	30	6	0.87%	0.87%	2.86%	3.73%
2	30	6	3.56%	1.74%	3.62%	3.91%
3	30	6	4.71%	4.76%	5.08%	6.35%
4	30	6	4.33%	1.47%	1.96%	3.79%
5	30	6	3.28%	0.59%	0.97%	3.12%
6	30	6	1.56%	1.51%	2.88%	4.25%
7	30	6	4.12%	2.13%	6.70%	5.18%
8	30	6	9.11%	8.21%	10.14%	10.63%
9	30	6	1.37%	1.21%	2.58%	3.47%
10	30	6	4.04%	2.04%	3.72%	5.20%
11	40	8	2.45%	1.17%	2.01%	5.54%
12	40	8	2.21%	1.80%	4.95%	4.73%
13	40	8	2.02%	2.06%	3.03%	3.44%
14	40	8	0.89%	0.60%	1.92%	1.72%
15	40	8	1.65%	0.71%	1.58%	2.93%
16	40	8	3.42%	0.36%	1.15%	1.23%
17	40	8	3.62%	0.47%	3.05%	3.73%
18	40	8	2.22%	0.67%	2.38%	1.86%
19	40	8	1.16%	0.48%	1.93%	1.68%
20	40	8	1.24%	1.24%	2.40%	2.32%
21	50	10	2.92%	2.77%	5.54%	4.92%
22	50	10	1.77%	1.37%	4.70%	4.89%
23	50	10	1.75%	0.78%	2.05%	1.86%
24	50	10	5.28%	3.35%	5.95%	6.32%
25	50	10	2.81%	0.93%	2.70%	1.96%
26	50	10	3.06%	1.08%	2.76%	3.00%
27	50	10	2.16%	0.31%	2.75%	1.12%
28	50	10	2.08%	0.85%	3.81%	1.69%
29	50	10	1.16%	0.56%	1.42%	1.37%
30	50	10	4.09%	0.81%	2.31%	4.16%

Table 13: The performance of the algorithms on the varying delivery volume test instances with decreased number of vehicles

Instance	# of Disp. Sites	# of Vehicles	C-IHA	C-MIP1	DVI Alg.	Impr. Heurs.
1	30	5	2.34%	1.56%	3.30%	4.34%
2	30	5	4.05%	2.03%	4.92%	4.34%
3	30	5	5.08%	1.59%	7.94%	7.30%
4	30	5	4.65%	2.20%	2.32%	3.55%
5	30	5	3.71%	0.97%	3.05%	4.75%
6	30	5	2.19%	1.92%	3.56%	5.62%
7	30	5	4.57%	2.74%	7.31%	3.81%
8	30	5	9.66%	8.70%	10.42%	10.78%
9	30	5	2.18%	2.02%	2.99%	4.44%
10	30	5	4.28%	2.79%	6.69%	6.51%
11	40	6	3.02%	2.35%	4.03%	3.19%
12	40	6	3.60%	2.93%	3.83%	6.08%
13	40	6	2.61%	2.61%	4.23%	4.09%
14	40	6	1.26%	0.99%	1.12%	3.57%
15	40	6	2.06%	1.66%	1.50%	4.12%
16	40	6	3.65%	0.83%	1.27%	1.78%
17	40	6	4.25%	1.24%	1.37%	2.27%
18	40	6	2.59%	0.88%	2.90%	2.74%
19	40	6	1.81%	1.02%	1.62%	2.89%
20	40	6	1.90%	1.90%	3.89%	4.55%
21	50	8	4.00%	3.38%	6.90%	4.92%
22	50	8	2.35%	1.96%	4.96%	4.89%
23	50	8	2.05%	1.17%	2.05%	4.31%
24	50	8	5.95%	3.72%	5.43%	6.55%
25	50	8	3.17%	1.30%	3.82%	4.47%
26	50	8	3.36%	1.32%	3.12%	4.76%
27	50	8	2.48%	0.46%	1.05%	1.63%
28	50	8	2.68%	1.06%	1.91%	3.03%
29	50	8	1.57%	0.76%	1.72%	2.74%
30	50	8	4.62%	0.92%	3.24%	6.47%

CHAPTER VI

CONCLUSION

In this study, we focus on the distribution planning of the emergency relief supplies to the affected people who are in need. Our main objective is to maximize the minimum slack (similar to the safety stock in commercial supply chains) in the distribution process rather than to the minimizing the total cost (including related inventory and transportation cost) in the traditional IRP. The motivation of maximizing the minimum slack is to prevent a potential shortage at the dispensing sites, which might danger human lives.

We analyze the single-vehicle version and the multi-vehicle version of the ISRP. For the single-vehicle version of the ISRP, we both formulate the problem mathematically and solve this model using a commercial solver. We also develop a heuristic algorithm called Iterative Heuristic Algorithm (IHA). For the multi-vehicle version of the problem, we develop a two-phase solution approach. In the first phase, we implement a clustering algorithm, which classifies the dispensing sites into the clusters and solves a set partitioning problem to determine the best set of clusters among the all the generated ones based on the snapshot routing solutions of the algorithm designed for the second phase. For the second phase, we propose two different routing algorithms, namely Clustering based-Iterative Heuristic Algorithm (C-IHA) and Clustering based-MIP1 (C-MIP1) model, to determine the delivery routes and the corresponding visit schedule. Finally, we test the performance of our proposed algorithms on randomly generated instances with different characteristics and observe that our proposed approaches significantly outperform the algorithms proposed in the literature. In particular, Clustering based-MIP1 algorithm not only provides the lowest average gap from the upper bound but also identifies the best in 117 out of 120 instances in total.

Bibliography

[1] IFRC, "Haiti earthquake 2010 one-year progress report," 2010.

[2] W. J. Petak, "Emergency management: a challenge for public administration," *Public Administration Review*, vol. 45, no. 3, pp. 3–7, 1985.

[3] 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.

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

[5] 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, pp. 181–190, 1984.

[6] A. Campbell, L. Clarke, A. Kleywegt, and M. Savelsbergh, *Fleet management and logistics*, ch. The inventory routing problem, pp. 95–112. Kluwer Academic Publishers, 1998.

[7] 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.

[8] 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.

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

[10] 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.

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

[12] 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.

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

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

[15] 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.

[16] S. Anily and A. Federgruen, "One warehouse multiple retailer systems with vehicle routing costs," *Management Science*, vol. 36, no. 1, pp. 192–214, 1990.

[17] C. Archetti, L. Bertazzi, A. Hertz, and M. G. Speranza, "A hybrid heuristic for an inventory routing problem," *Journal on Computing*, vol. 24, no. 1, pp. 101–116, 2011.

[18] F. Niakan and M. Rahimi, "A multi-objective healthcare inventory routing problem; a fuzzy possibilistic approach," *Transportation Research*, vol. 80, no. 1, pp. 74–94, 2015.

[19] R. G. van Anholt, L. C. Coelho, G. Laporte, and I. F. A. Vis, "An inventory-routing problem with pickups and deliveries arising in the replenishment of automated teller machines," *Transportation Science*, vol. 50, no. 3, pp. 1071–1091, 2016.

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

[21] 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.

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

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

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

[25] 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.

[26] O. Solyali and H. Sural, "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.

[27] T. W. Chien, A. Balakrishnan, and R. T. Wong, "An integrated inventory allocation and vehicle routing problem," *Transportation Science*, vol. 23, no. 2, pp. 67–76, 1989.

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

[29] J. C. Goodson, B. W. Thomas, and J. W. Ohlmann, "Restocking-based rollout policies for the vehicle routing problem with stochastic demand and duration limits," *Transportation Science*, vol. 50, no. 2, pp. 591–607, 2015.

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

[31] D. B. 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.

[32] P. Trudeau and M. Dror, "Stochastic inventory routing: route design with stockouts and route failures," *Transportation Science*, vol. 26, no. 3, pp. 171–184, 1992.

[33] L. C. Coelho, J. F. Cordeau, and G. Laporte, "Consistency in multi-vehicle inventory-routing," *Transportation Science*, vol. 24, no. 3, pp. 270–281, 2012.

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

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

[36] M. Dror and M. Ball, "Inventory/routing: reduction from an annual to a short-period problem," *Naval Research Logistics*, vol. 34, no. 2, pp. 801–995, 1987.

[37] A. A. Juan, S. E. Grasman, J. C. Cruz, and T. Bektas, "A simheuristic algorithm for the single-period stochastic inventory-routing problem with stock-outs," *Simulation Modelling Practice and Theory*, vol. 46, no. 1, pp. 40–52, 2014.

[38] Y. Yu, H. Chen, and F. Chu, "A new model and hybrid approach for large scale inventory routing problems," *European Journal of Operational Research*, vol. 189, no. 2, pp. 1022–1040, 2008.

[39] E. Luis, I. S. Dolinskaya, and K. R. Smilowitz, "Disaster relief routing: integrating research and practice," *Socio-Economic Planning Sciences*, vol. 46, no. 1, pp. 88–97, 2012.

[40] R. P. Knott, "Vehicle scheduling for emergency relief management: a knowledge-based approach," *Disasters*, vol. 12, no. 4, pp. 285–293, 1988.

[41] G. Barbarosoglu, L. Ozdamar, and A. Cevik, "An interactive approach for hierarchical analysis of helicopter logistics in disaster relief operations," *European Journal of Operational Research*, vol. 140, no. 1, pp. 118–133, 2002.

[42] A. Haghani and S. C. Oh, "Formulation and solution of a multi-commodity, multi-modal network flow model for disaster relief operations," *Transportation Research Part A: Policy and Practice*, vol. 30, no. 3, pp. 231–250, 1996.

[43] G. Barbarosoglu and Y. Arda, "A two-stage stochastic programming framework for transportation planning in disaster response," *Journal of the Operational Research Society*, vol. 55, no. 1, pp. 43–53, 2004.

[44] L. Ozdamar, E. Ekinci, and B. Kucukyazici, "Emergency logistics planning in natural disasters," *Annals of Operations Research*, vol. 129, no. 1-4, pp. 217–245, 2004.

[45] B. Balcik, B. M. Beamon, and K. Smilowitz, "Last mile distribution in humanitarian relief," *Journal of Intelligent Transportation Systems*, vol. 12, no. 2, pp. 51–63, 2008.

[46] A. Montjoy, S. Brown, and J. W. Herrmann, "Solving the inventory slack routing problem for medication distribution planning," tech. rep., The institute for systems research, University of Maryland, 2009.

[47] Z. Yan, A. Montjoy, and J. W. Herrmann, "Variants of the adaptive large neighborhood search for the inventory slack routing problem," tech. rep., The institute for systems research, University of Maryland, 2011.

[48] X. Yang, "Solution to the multi-depot inventory slack routing problem at the planning stage," *Journal of Computing in Civil Engineering*, vol. 30, no. 1, pp. 544–561, 2014.

[49] X. Yang and L. Feng, "Routing and scheduling approach with the objective of slack maximization," *Transportation Research*, vol. 2378, no. 1, pp. 32–42, 2014.

[50] G. Clarke and J. W. Wright, "Scheduling of vehicles from a central depot to a number of delivery points," *Operations Research*, vol. 12, no. 4, pp. 568–581, 1964.

VITA

Emre Çankaya received the BSc degree in industrial engineering from Özyegin University, Istanbul, Turkey in 2013. In February, 2014, he entered Graduate School of Engineering at Özyegin University, in Istanbul. He works as a teaching and research assistant throughout his MSc degree. assisted in courses: Productions Systems Analysis, Operations Research II, Simulation Modelling and Analysis and Mathematical Modelling and Exact Methods