Transportation Service Provider Collaboration Problem: Potential Benefits and Solution Approaches

Robert S. Roesch

Dissertation submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Industrial and Systems Engineering

Kimberly P. Ellis, Chair
Barbara M.P. Fraticelli
Charles P. Koelling
Gaylon D. Taylor

February 1, 2017
Blacksburg, Virginia

Keywords: Horizontal Collaboration, Freight Transportation, Pickup and Delivery Problems with Transshipment

Transportation Service Provider Collaboration Problem: Potential Benefits and Solution Approaches

Robert S. Roesch

ABSTRACT

Truck-based freight transportation continues to play a vital role in the delivery of goods in the United States. Despite its size and importance, the truck transportation industry continues to struggle with fulfilling transportation requests in an efficient and sustainable manner. One potential solution to alleviate many of the current truck industry problems is for transportation service providers (TSPs) to collaborate by sharing volume, resources, and facilities. This research introduces the Transportation Service Provider Collaboration Problem (TSP-CP) to demonstrate the benefits of using optimal freight routing and consolidation decisions for collaborating TSPs. A mathematical model for the TSP-CP is introduced to describe the problem in detail. Additionally, two separate adaptive large neighborhood search (ALNS) heuristics are developed to provide solutions to industry representative problem instances. Finally, the benefits and insights achieved by enabling collaboration between TSPs using the TSP-CP are identified using industry representative data sets.

The representative data sets were derived from actual freight data provided by a freight pooling company that manages collaboration among TSPs. Carriers were chosen from the industry data to evaluate collaborative partnerships and to gain insights on the effects of partnership characteristics on overall benefit as well as the benefits obtained by individual carriers. The computational results suggested collaboration among TSPs offers the potential for substantial reductions in the total distance required to deliver all loads, in the number of miles that were traveled completely empty, and the number of containers required for delivery compared to individual performance. Additionally, collaboration increased delivery resource capacity utilization as measured by the percentage of weighted full miles. Detailed analysis of the results from the TSP-CP revealed new insights into the collaboration between full truckload and less-than truckload carriers that have not been quantified or highlighted

in previous research. These insights included the effect that an individual carrier's type and size had on the amount of benefit received to each carrier. Finally, the results highlighted the importance of building collaborative partnerships that consider a carrier's geographic location.

Transportation Service Provider Collaboration Problem: Potential Benefits and Solution Approaches

Robert S. Roesch

GENERAL AUDIENCE ABSTRACT

Truck-based freight transportation continues to play a vital role in the delivery of goods in the United States by carrying nearly 70% of all freight tonnage. Despite its size and importance, the truck industry continues to struggle with transporting freight in an efficient, timely, and sustainable manner. One potential solution to alleviate many of the current truck industry problems is for transportation service providers (TSP) to collaborate by sharing resources, facilities, and freight volume. This research introduces the Transportation Service Provider Collaboration Problem (TSP-CP) to demonstrate the benefits of using optimal freight routing and consolidation decisions for collaborating TSPs.

The benefits and insights achieved by enabling collaboration between TSPs using the TSP-CP are identified using industry representative data sets. The representative data sets were derived from actual freight data provided by a freight pooling company that manages collaboration among TSPs. The computational results suggested collaboration among TSPs offers the potential for substantial reductions in the total distance required to deliver all freight, in the number of miles that were traveled by containers completely empty, and in the number of containers required for delivery compared to individual performance. Additionally, collaboration increased delivery resource capacity utilization. Detailed analysis of the results from the TSP-CP also revealed new insights into TSP collaboration. These insights included the effect that an individual carrier's type and size had on the amount of benefit received to each carrier. Finally, the results highlighted the importance of building collaborative partnerships that consider a TSP's geographic location.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Truck-based freight transportation carries 68% of all freight tonnage in the United States and is expected to continue to play a vital role in the delivery of goods in the U.S. [1]. Freight transportation by truck is made up of two primary modes: full truckload (FTL) and less-than-truckload (LTL). FTL carriers represent 52% of the TSP industry in the U.S., while LTL carriers make up an addtional 24% [39]. The remaining 24% includes a combination of different specialty carriers. Despite its size and importance, the truck transportation industry continues to struggle with fulfilling transportation requests in an efficient and sustainable manner. The focus of this research is to evaluate opportunities to reduce costs and increase sustainability enabled through collaboration among transportation service providers (TSPs). The following sections provide additional background information on FTL and LTL carriers and the motivation for this research on TSP collaboration.

1.1 Full Truckload Carriers

Full truckload (FTL) carriers handle large shipments by delivering transportation requests directly from origin to destination. A full truckload shipment is generally any shipment that is 10,000 lbs or larger. In order to make a delivery directly from origin to destination, an FTL carrier requires that when purchasing a full truckload shipment, the entire delivery resource capacity must be purchased. This enables FTL carriers to deliver transportation requests direct using a single delivery resource without any intermediate consolidation.

Although the restriction of purchasing a full delivery resources can limit customers, it allows FTL carriers to construct routes for delivery resources optimally each time they are used. This also allows for fast delivery times and limits the number of times that a transportation request is handled. However, this does ultimately result in lower delivery resource capacity utilization. An additional downside to the way the FTL industry operates is that shipments often require delivery over a great distance. This requires that FTL carriers operate in a significantly large geographic region, making it nearly impossible to build routes that allow drivers to remain close to their domicile. Due to this, an FTL driver can be away from their home for as long a two weeks at a time. As such, the FTL industry is plagued by problems with driver shortages, driver turnover, and shipping less than full trucks over long distances.

1.2 Less-than-truckload Carriers

Less-than-truckload (LTL) carriers handle smaller shipments by consolidating multiple transportation requests onto a single delivery resource at a consolidation hub. LTL shipments typically range in size from as small as 100 lbs up to as much as 10,000 lbs. An LTL shippers network consists of two different types of locations: end-line terminals and break-bulk terminals. There will normally be more end-line terminals than break-bulk terminals in a standard LTL carrier network. An end-line terminal acts as an origin or a destination for freight shipments while a break-bulk terminal is where consolidation happens. Many end-line terminals send frequent shipments to and receive shipments from a single break-bulk terminal typically located a short distance away. The break-bulk terminals consolidate shipments from end-line terminals and other break-bulk terminals to build full or nearly full delivery resources. Once enough volume has been accumulated, the delivery resources then travels to the next break-bulk terminal. Once shipments reach the next break-bulk terminal they either remain on the current delivery resource heading to another break-bulk terminal or they are moved to another delivery resources heading for their destination end-line terminal or to another break-bulk terminal where this process is repeated.

In order to handle the difficulty of allowing both intermediate consolidation and routing decisions, LTL carriers typically use predetermined load plans for fulfilling transportation requests in the manner descriped above. A load plan is a set of rules that stipulates how freight will flow through an LTL carrier's existing transportation network. This is the exact sequence of break-bulk terminals that a shipment will travel through given the terminal it is currently sitting in and the end-line terminal it is to be delivered to. Although shipment trends tend to change over time, load plans are often used for multiple planning periods. The benefit of the load plan is that it allows for quick and consistent decisions for each transportation request and delivery resource. However, due to the relatively static nature of the load plan, it can result in sub-optimal freight consolidation and routing decisions. Additional derlivery time is also required in order to consolidate enough LTL shipments for efficient delivery [39].

1.3 Motivation

Maintaining efficient and sustainable operations through freight consolidation and routing is essential for enabling transportation service providers to compete in a largely segmented industry. The current practices already highlighted by both segments of the industry create many of the problems that the industry is facing today. Estimates in 2010 found that approximately 25% of all miles are traveled with an empty delivery resource and the remaining 75% of miles are traveled at only 56.8% full on average. This results in the average delivery resource traveling the road at only 42.6% full [28]. In addition to being inefficient, truck transportation is also environmentally and socially unsustainable. Truck transportation was solely responsible for nearly 6% of the total U.S. greenhouse gas emissions in 2011 [16]. The industry is also plagued with driver shortages resulting from high turnover rates and extended time away for drivers. Currently there is a shortage of nearly 25,000 drivers and long-haul truck drivers have a turnover rate of around 100% [11] [5]. Industry fragementation also tends to exaserbated the problems the industry is already facing. In 2011 there were nearly half a million trucking companies on record with nearly 90% of those containing fewer than 10 trucks [12].

With the importance of truck transportation to the U.S. and global economy, new opportunities must be leveraged and new techniques must be developed to enable a more economically, socially, and environmentally sustainable industry moving forward. One potential solution is to create opportunities to reduce costs and increase sustainability through TSPs collaborating to sharing volume, resources, and facilities.

1.4 Collaboration in the Supply Chain

Collaboration has been identified as one of four capabilities that the material handling and logistics industry must develop in order to support the needs of the U.S. economy through 2025 [2]. With a supply chain being made up of different types of organizations such as suppliers, manufactures, transportation service providers, and retailers there are two separate forms of supply chain collaboration in use today. The more commonly seen vertical collaboration is collaboration that takes place between two different organization types such as a manufacturer and a retailer working together to manage inventory.

The less commonly seen form of supply chain collaboration is referred to as horizontal collaboration. Horizontal collaboration is collaboration that takes place between two similar, often competing, organization types that operate in the same level of the supply chain. The collaboration between TSPs stuidied in this research is a form of horizontal collaboration.

Although, horizontal collaboration is less common due to the issues in dealing with possible competitors, it is starting to gain traction as way for supply chain partners to reduce costs and increase sustainability. Several recent case studies highlight the growing interest and the potential benefits associated with horizontal collaboration between supply chain partners [7] [6] [25] [24].

1.5 Research Objectives

The objective of this research is to motivate horizontal collaboration in the supply chain through demonstrating the benefits of using optimal freight routing and consolidation decisions for collaborating TSPs. In order to fully realize the benefits of collaboration between FTL and LTL carriers, the limitations of current freight routing and consolidation practices must be overcome. To support the advancement of horizontal collaboration, the Transportation Service Provider Collaboration Problem (TSP-CP) is introduced and a mathematical model for is developed to fully describe the problem. With the TSP-CP mathematical model as a starting point, two separate heuristics are developed to provide solutions to industry representative problem instances. Finally, a case study is performed using industry representative data and insights are gained on the outcomes achieved by enabling collaboration between FTL and LTL carriers.

In accomplishing this objective, this research provides the following contributions:

1. Introduce a new mathematical model and modeling approach for the pickup and delivery problem with transshipment.

2. Develop the first optimization based adaptive large neighborhood search (ALNS) for the pickup and delivery problem with transshipment.

3. Implement a new application of an ALNS adapted to fit the needs required by the TSP-CP.

4. Provide new insights into the collaboration between FTL and LTL carriers using the TSP-CP.

1.6 Dissertation Structure

In the remainder of this dissertation, Chapter 2 describes the underlying problem motivating the TSP-CP and gives an overview of the TSP-CP itself. Chapter 3 provides an overview of the relevant literature. Chapter 4 presents the mathematical model and assumptions for the TSP-CP. The two new TSP-CP heuristics are developed in Chapter 5 while Chapter 6 tests the performance of the mathematical model and the two new heuristics on test data sets. Chapter 7 performs a case study by applying the TSP-CP to industry representative data sets to gather insights into TSP collaboration. Finally, Chapter 8 presents final conclusions and the areas of future research.

Chapter 2

Problem Description

This research introduces the Transportation Service Provider Collaboration Problem (TSP-CP), which provides optimal freight routing and consolidations decision for a set of collaborating TSPs that includes both FTL and LTL carriers. The TSP-CP is modeled as a pickup and delivery problem with time windows and transshipment (PDPTWT). In this model, the combined set of transportation requests are serviced using the interconnected network and pooled delivery resources created by the collaborating TSPs. The model prescribes the optimal routing decisions for the collaborating carriers. Additionally, the inclusion of transshipment allows for freight consolidation at intermediate locations by enabling the ability to move transportation requests between delivery resources.

2.1 The General Pickup and Delivery Problem

The underlying problem of the TSP-CP is the pickup and delivery problem (PDP). The PDP determines a set of delivery resource routes to satisfy transportation requests. The typical objective is to deliver all transportation requests utilizing the available resources while minimizing the travel distance of delivery resources. However, other objectives exist such as minimizing fleet size, minimizing request delivery duration, or maximizing profit. In the PDP, each transportation request must be transported by a single delivery resource from its origin to its destination without changing between resources. A description of the general PDP and the main variants can be found in Savelsbergh and Sol [37].

2.1.1 PDP Characteristics

Pickup and delivery problems typically have very similar characterisitics such as delivery resources, transportation requests, and locations. However, there are often extensions and additional considerations added to the PDP that make for a more difficult, but more realistic problem. A subset of these characteristics that are relevant to the TSP-CP are introduced in this section. A more exhaustive list of characteristics can be found again by Savelsbergh and Sol [37].

Delivery Resources and Transportation Requests

Delivery resources and transportation requests are integral parts of every PDP application. Delivery resources most commonly represent vehicles that used to deliver people or freight. Each vehicle has a capacity, a starting location, and an ending location. Starting and ending locations are either set as the same central depot or they are set as two preexisting locations. Limitations are often placed on vehicles to help mitigate the difficulty of the PDP. These include, but are not limited to restricting a vehicle to a single route or limiting the number of times a vehicle can enter any single location.

Transportation requests are the loads of freight, people, or groups of people that must be delivered by delivery resources. Each transportation request has a size, an origin location, and a destination location. The PDP can either be modeled to consider the specific assignment of transportation requests to delivery resources or not. When the specific assignment transportation requests to delivery resources is not considered, the only constraint is that enough delivery resource capacity travels an between locations to cover the combined capacity occupied by all transportation requests. When the specific assignment is considered, the capacity of a single delivery resource cannot be exceeded by the transportation requests that are carried on that delivery resource between locations. Transportation requests can be assigned to delivery resources either with or without split delivery. Split delivery allows a transportation request to be split across multiple delivery resources. For example, given that a transportation request consists of 10,000 lbs. In the case where split delivery is not allowed, all 10,000 lbs must be carried on the same delivery resources. However, in the case where split delivery is allowed, that 10,000 lbs may be broken down into one delivery resource carrying 2,000 lbs and a separate one carrying the remaining 8,000 lbs. Furthermore, it may be the case that the transportation request can be split across more than two delivery resources.

Time

Time is an essential addition to any realistic PDP application. However because of the additional complexity introduced by time, it is not considered in every application. When time is including in a PDP, it is most commonly modeled as continuous time. Continuous time is typically handled by an independent clock variable associated with each location and delivery resources. When time is handled in this way, each time a delivery resource enters a location, the variable is updated to the current system time and the variable is then used to handle any precedence constraints or time windows. In the situation where time is handled through the use of discrete time periods, a predetermined number of time periods are used based on a known time period length and a known time horizon. Once the time periods are established, the variables in the problem are explicitly defined for each time period.

Time Windows

A pickup and delivery problem that contains time windows is referred to as a pickup and delivery problem with time windows (PDPTW). Time windows are used to restrict when service can take place. Time windows are most commonly enforced on either locations or on transportation requests. A time window enforced on location indicates when all pick ups and deliveries to that location must take place. A time window enforced on a transportation request signifies the pick up and delivery time that must be met for that single transportation request. This allows for multiple time windows at a single location. In limited instances, time windows can also be enforced on delivery resources. A time window enforced on a delivery resource typically indicates when a delivery resource may begin their first route and and the time by which they must complete their last route.

Transshipment

Allowing for freight consolidation at intermediate locations requires that freight must be removed from a delivery resource, held, and reloaded back onto a different delivery resource. To facilitate these actions, transfers can be added to the general PDP. A transfer involves physically moving a transportation request from one delivery resource to another in order to obtain better resource utilization and lower overall costs. The pickup and delivery problem with transshipment (PDPT) is a less studied variant on the PDP that allows for transportation requests to be transferred between delivery resources at transshipment nodes. The literature on the exact and heuristic solution approaches for the PDPT is included in Chapter 3.

The complexity of the PDPT often limits transshipment to a limited number of transfer points or only allows for a single transfer. Another complication of transshipment is that precedence constraints are needed to ensure that the arrival and departure times of transportation requests on delivery resources match up for a feasible transfer. These constrains are handled either by explicitly defining constraints to handle them, thus making the problem more complex or by modeling the node that represents each transfer location as two split nodes. With two nodes, the first node is where the transportation request is unloaded from the delivery resources that brings it into the transshipment location. The travel between the two nodes then represents the transfer. The second node represents where the transportation request is loaded onto the delivery resource that it is being transferred too. Having this set up ensures that the departure of a transportation request on a delivery resource is always after it arrives and has been transferred without using complicating constraints. Finally, it may also be required that transfers are synchronized. At a minimum, a synchronized transfer requires that both delivery resources involved in the transfer arrive at exactly the same time.

2.2 The Transportation Service Provider Collaboration Problem

The TSP-CP is modeled as a time-expanded pickup and delivery problem with time windows and transfers (t-PDPTWT). The t-PDPTWT provides optimal consolidation and routing solutions for large scale transportation networks. Given a set of transportation requests, a set of delivery resources, and a transportation network consisting of nodes and arcs, the t-PDPTWT simultaneously determines the routes of transportation requests, the routes of delivery resources, and the assignment of transportation requests to delivery resources. The t-PDPTWT provides additional flexibility over the traditional pickup and delivery problem by allowing transportation requests to be transferred between delivery resources at transshipment nodes. Again, a transfer involves physically moving a transportation request from one delivery resource to another in order to obtain better resource utilization and lower overall costs. Although, each transportation request must be carried by a delivery resource, having independent routes for the transportation requests and the delivery resources allows decoupling the transportation requests from the delivery resources to achieve flexibility. The objective of the t-PDPTWT is to deliver all transportation requests while minimizing the transportation costs for delivery resources and the handling and holding costs of transportation requests. This section provides more information on the transportation network, the transportation requests, and the delivery resources.

2.2.1 Transportation Network: Nodes and Arcs

The transportation network is made up of nodes and arcs. Nodes represent the physical locations in the transportation network such as suppliers, customers, end-line terminals, break-bulk terminals, or other locations where transportation requests are picked up, transferred, or delivered. If it is possible to travel between two nodes in the transportation network, they are connected. However, there is no guarantee that two nodes are connected by an arc. Each arc has a distance and a traversal time associated with it. The distance of an arc is given simply as the distance (in miles) required to travel between the two nodes. The traversal time is based on the average speed that a delivery resource can travel across that arc and the distance of that arc. Figure 2.1 shows a graphical representation of a three node transportation network. The arcs are labeled with the miles between the two nodes on the left and the average speed of a delivery resource given in miles per hour on the right.

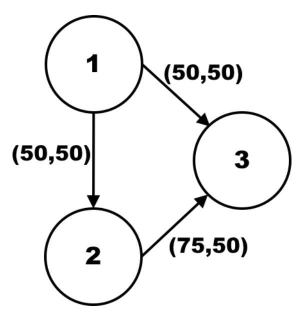

Figure 2.1: Standard graphical representation of a transportation network

Nodes are categorized into two distinct sets: transshipment nodes and non-transshipment nodes. The non-transshipment nodes are strictly locations where transportation requests can originate or terminate their routes. If a transportation request enters a non-transshipment node that is not its origin or destination, then the transportation request must remain on the delivery resource that carried it there. In contrast, transshipment nodes are the subset of nodes where transportation requests can be unloaded from their delivery resource so they can be transferred to a different delivery resource or held to leave on a delivery resource at a later time. Each time a transfer happens at this facility there is a per request handling cost charged each time a request is unloaded from or reloaded onto a delivery resource. Additionally, each transshipment node has a per time holding cost charged each time a transportation request is held there. There is no capacity associated with each transshipment node.

2.2.2 Transportation Network: Time

Rather than using continuous time, the TSP-CP captures time through the use of discrete time periods. Given the planning horizon and a known time period length, the number of discrete time periods is given as the ceiling of the time horizon divided by the time period length. Considering time in this manner requires the creation of a time-expanded network. In a time-expanded network, a node represents both a physical location and a time period. Essentially, the time-expanded network creates a copy of each location in every time period and appropriately connects locations with arcs. Figure 2.2 provides the time-expanded version of the network in Figure 2.1 for a planning horizon of five time periods. Two nodes are only connected if the delivery resources enters the second location in the time period that is equal to the ceiling of the time period in which the resource left the first node plus the traversal time between the two nodes.

A time-expanded network transforms the graph that represents the network into a directed acyclic graph and eliminates issues associated with sub-tours that may arise in the traditional network. This does not come without some inherent trade offs between using the discrete time approach over the traditional continuous time approach. The first and most obvious is the increase in the number of nodes and arcs in the time-expanded network. Additionally, there is a loss of granularity with discrete time periods which may increase costs. As long as

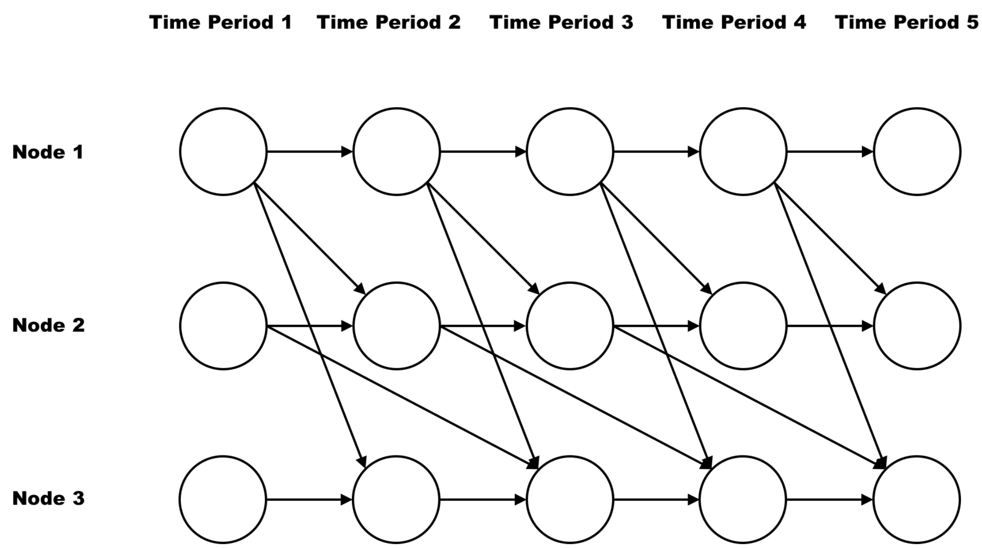

Figure 2.2: Time-expanded representation of the transportation network from Figure 2.1

the time period length is deemed to be sufficiently small, this is an acceptable trade off for the gains in simplifying the network. For the TSP-CP the intended time horizon is between eight and 24 time periods expressed in hours.

2.2.3 Transportation Requests: Loads

Loads of freight are the transportation requests required for delivery by the transportation service providers in the TSP-CP. Each load has an origin node where it must begin its route and a destination node where its route must terminate. Each load also has a pickup time window defined by an earliest pickup time period and a latest departure time period as well as a delivery time window defined by an earliest arrival time period and a latest delivery time period. While traveling between nodes, loads must always be carried by a delivery resource. However, when being held at a transshipment node, loads may be decoupled from a delivery resource and may wait at the transshipment node without any delivery resource present. Additionally, each load has a required space expressed as a percentage of a delivery resource. Required space for all loads is always less than or equal to the capacity of the delivery resource. This percentage allows for flexibility in measuring the required space of loads in terms of either weight, volume, or pallets. Each load is meant to be a single delivery unit that remains together throughout the entire journey from origin to destination. That is, if a load occupies 50% of a delivery resource, this 50% remains together at all times. Loads can be carried on or transferred between any delivery resource as long as the capacity of the delivery resource is not exceeded. Loads can also flow through any node as long as it follows the rules associated with that node being a transshipment or a non-transshipment node.

2.2.4 Delivery Resources: Containers

Containers are the delivery resources in the transportation network. In this case where drivers are not explicitly considered, the container is defined as the combination of the tractor, the trailer, and the driver. There are no restrictions on driver rules considered here, but each container has a capacity that must never be exceeded by the loads it is carrying. Although the capacity is never to be exceeded, it is assumed to be able to be fully utilized and can represent weight, volume, or pallets. In addition to the capacity, each container has an initial starting location where it must begin its route and a final ending location where it must complete its route. These two locations may be the same node or they may be different nodes and there are no restrictions placed on whether or not a container may enter these locations multiple times throughout the route. Furthermore, containers can be used on as many routes as necessary and there are no other limitations placed on how a container may move throughout the network. Additionally, containers are allowed to move throughout the network without any loads on them. Unlike loads, container may be held at all nodes without a cost. Finally, there is a per distance cost associated with each container that represents the variable cost of operation including fuel, depreciation, and driver wages

Chapter 3

Literature Review

The TSP-CP lies at the intersection of two emerging areas of research. The first of these areas is the horizontal collaboration that takes place in the supply chain area of transportation and logistics. The second area of research is the underlying problem of the TSP-CP that is the pickup and delivery problem with transfers. This chapter covers the relevant literature in both of these two emerging areas.

3.1 Horizontal Collaboration in Transportation

Limited research is currently avaliable on the collaboration between FTL and LTL carriers. However, research findings on FTL carrier collaboration shows the growing interest and the potential gains of horizontal collaboration in transportation. Cruijssen [13] covers a number of topics on horizontal collaboration in transportation and logistics including assessing the benefits associated with joint route planning. In this research, a vehicle routing problem with time windows is used to compare the distribution costs of scenario where a company plans routes individually to the case where a number of companies jointly plan their routes from a shared depot. Ergun, Kuyzu, and Savelsbergh [17] present the lane-covering problem

(LCP) for FTL carrier collaboration. The LCP focuses on finding repeatable, dedicated, continuous move tours with the objective of minimizing trailer repositioning. Krajewska et al. [23] consider finding routes for FTL carrier collaboration by solving a multi-depot pickup and delivery problem with time windows that has the objective of minimizing the cost required to service all requests. Additionally, they incorporate cooperative game theory to share the savings between the participating carriers.

Berger and Bierwirth [4] present a post-market optimization based collaboration mechanism for independent carriers. They maximize profit for a collaborative carrier network through an auction that requires minimal information sharing. Single or bundled transportation requests are exchanged to increase the collective profit by each carrier independently submitting requests for reassignment. Juan, Faulin, Prezer-Bernebea, and Jozfowiez [21] consider finding routes for a collaborative carrier network aimed at minimizing empty back-hauls using an approach based on the vehicle routing problem. Finally, Ozener, Ergu, and Savelsbergh [32] consider FTL carrier collaboration from a decentralized perspective by using simple lane exchange mechanisms. These mechanisms allow for two FTL carriers to bilaterally exchange lanes based on predefined protocols without forming long-term carrier alliances.

Although research on the horizontal collaboration between LTL carriers is limited, it is starting to gain attention. Wang and Kopfer [41] study LTL carriers collaborating through exchanging transportation requests. They propose a mechanism that exchanges transportation requests after an initial route generation phase based on a pickup and delivery problem with time windows (PDPTW). As a decentralized approach, they assume that each carrier solves their own PDPTW without any transshipment, thus reducing some of the potential benefits from collaboration. Bailey, Unnikrishnan, and Lin [3] present two models aimed at filling empty back-hauls. Given the last location of a previous truck delivery, the models determine if a collaborative freight request can be fulfilled instead of the truck returning empty. The first model simply pairs a truck with a collaborative node and then the truck will fulfil a shipment request to that node on the route back to its depot. The second model allows for multiple pickup and deliveries on the way back to the depot. If no opportunity is found, the truck simply returns to the depot empty in both models. Dia and Chen [14] adapt the LCP to fit LTL carriers in order to find routes based on the number of times a vehicle needs to travel an arc as well as the quantity of product that flows over that arc. This problem is similar to the VRP with pickup and delivery except that it allows for transfers, split deliveries, and vehicles may begin and end routes at any node. This approach, however does not consider time and does not consider individual vehicles.

Nadarajah and Bookbinder [30] focus on LTL carrier collaboration prior to planning intra-city routes from entry points to a city. They present a three-phased approach that starts by separately finding optimal routes for each city entrance by using a vehicle routing problem with time windows. The second phase clusters customer locations to find transshipment points based on the centroids of these clusters. Finally, the third phase finds collaborative routing opportunities between trucks entering the city at different entry points facilitated through transshipment.

Peeta and Hernandez [19] study the single carrier collaboration problem. This is collaboration from the perspective of a single LTL carrier and it is modeled as a network flow problem to determine which carriers transport shipments over arcs. In this case, all capacity requirements are volume based and individual trucks are not considered. Additionally, they assume that each carrier uses their available capacity first and then looks for collaborative opportunities. Peeta, Hernandez, and Kalaftas [20] extend the single carrier collaboration problem to the deterministic dynamic single carrier collaboration problem. This extension allows for time dependent collaborative capacities, pickup and delivery time windows, and holding. However, they assume that the collaborative capacity over each arc in every time period is known ahead of time.

3.2 Pickup and Delivery Problem with Transfers

3.2.1 Exact Solution Methods

Due to its difficulty, there have been few exact solution approaches proposed for the pickup and delivery problem with trasnfers (PDPT). The widely recognized first exact approach comes from Cortes, Matamala, and Contardo [10]. They present a mathematical model for the PDPT considering passenger transportation in which a transportation request contains multiple passengers. The objective of this model is to minimize the total ride time of vehicles, the waiting and ride times for passengers, the fleet size, and any time window violations. They only allow for a single transfer point that is located at a central depot where all vehicles must begin and end their one allowable route. They model precedence constraints at the transfer node through splitting the nodes into two separate nodes: the first node where unloading happens and the second node where reloading happens. Additionally, they handle time through the use of independent continuous time variables associated with each node and vehicle. Finally, they enforce time windows on locations rather than on requests. In order to obtain optimal solutions to their mathematical model, a branch-and-cut solution approach is introduced utilizing the combinatorial Benders cuts of Codato and Fischetti [9]. However, many of their constraints utilize the “big M” modeling technique which can lead to a weak linear relaxation. They apply this approach and compare the average solution times of their branch-and-cut approach to the traditional branch-and-bound approach for problem instances involving six requests, 13 nodes, two vehicles, and one transfer point.

Rais, Alevlos, and Carvolho [36] present a mathematical model for the PDPT in which they aim to minimize the cost of vehicle routes. They consider heterogeneous vehicles which may only enter any node one time and may only be used on a single route. However, they do allow for vehicles to have different starting and ending locations. They include multiple transfer locations, an unlimited number of transfers for requests, and enforce time windows on requests instead of locations. Time is modeled as a continuous variable represented by an independent clock associated with each vehicle. Additionally, they include split deliveries which allows for a load to be broken down and transported separately on multiple vehicles. This assumption causes their model to be a relaxation of the traditional PDPT where split deliveries are not allowed. They directly solve the mathematical model to assess the benefits of allowing transfers. They are able to provide solutions for problem instances of 14 nodes, seven vehicles, and seven requests.

Similarly, Kerivin, Lacroix, Mahjob, and Quilloit [22] propose two models for the pickup and delivery problem with transfers and split delivery. As they again consider split delivery, both models are a relaxation of the general PDPT. In both models, the objective is to minimize all vehicle related costs. They model time through the use of discrete time periods, but do not enforce time windows. The first model is a multi-commodity network flow based formulation that has one flow representing request routes and one flow representing vehicle routes. In this case they do not consider individual vehicles, rather they ensure that enough vehicles travel an arc such that their combined vehicle capacity is not exceeded. The second model is similar, but replaces the variables that represent request routes with metric constraints. In both cases, vehicles start and end their routes at a central depot with no restrictions placed on the number of times they may pass through a node. Both models are solved using a branch-and-cut solution approach with the goal of checking the efficiency of future heuristics by providing a lower bound. With this in mind, they are able to solve problem instances with 15 requests, 10 nodes, and seven vehicles.

As part of a larger problem focusing on postal and express shipments, Grunert and Sebastian [18] propose a mathematical model for the global area transportation network (GATN) problem. The GATN can be seen as an extension of the multi-commodity dynamic fixed charge network design problem in which requests are transported from origin to destination by a fleet of heterogeneous vehicles. As they are dealing with postal and express shipments, some additional constraints are considered such as shipments must be sorted each time they enter a node, there is a limited number of vehicles that can arrive at a node in each time period, and there are limitations on sorting and storage capacities. The objective of the model is to minimize the vehicle and sorting costs. Time in this situation is again handled using discrete time periods. However, in this case time windows are enforced on the nodes, the requests, and when the drivers may begin and end their routes. All locations are modeled as two nodes: one for loading and one for unloading. This enables the additional assumption that each time a vehicle enters a transshipment node it must be unloaded, sorted, and then reloaded. Furthermore, they do not model individual vehicles, rather they model vehicles types that can serve requests with different vehicle capacities again allowing for split deliveries. The model was used to highlight similarities and differences between postal and express shipment and other related models; as such, no computational results were provided.

Vornhusen, Wang, and Kopfer [40] present a mathematical model for LTL carrier collaboration using a pickup and delivery problem with transshipment and time windows. The objective of this model is to minimize the total distance traveled by all vehicles. In their model, the limitation is included that nodes can be visited exactly one time by a vehicle. This allows for a node to only be an origin to a single load, a destination to a single load, or a transshipment point for a single transfer. Additionally, vehicles may only be used for one route and must begin and end their route at the central depot that corresponds to their LTL carrier. At transshipment points, only a single transfer is allowed per load and all transfers must be synchronized such that the trucks involved in the transfer must arrive at the transshipment location at the same time and must have the same service time. Time is modeled through a continuous variable representing when a node is served by a vehicle. This requires that time windows are related to the nodes rather than the requests. The model is directly applied to compare the performance of the network with each LTL carrier working independently without transshipment to how the network performs with the LTL carriers collaborating both with and without transshipment. Although the detailed solution approach is not described, they provide results for problem instances consisting of two carriers (i.e. two vehicle hubs), nine requests, 20 nodes, and one transshipment location.

Dondo, Mendez, and Cerda [15] minimizes total routing costs using a mathematical model for the supply-chain pick-up and delivery problem with transshipment. In this model, transfers are modeled through splitting transshipment nodes into two nodes: a reception node and a delivery node. Additionally, this model requires that each non-transshipment node is visited exactly once by a vehicle and that all vehicles start and end their routes at a central depot. Continuous time is used, time windows are based on locations rather than requests, and a maximum route time constraint is placed on vehicles. They directly apply their mathematical model to an illustrative example in which two production facilities produce a single product for 20 customers. Demand at these customers may be serviced using four vehicles either directly from one of the production facilities or from one of two intermediate transshipment locations.

Masson, Ropke, Lehuede, and Peton [27] present a mathematical model solved by a branch-cut-and-price solution approach for the pickup and delivery problem with shuttle routes (PDPS). The PDPS minimizes the travel distance of vehicles required to transport all passengers. The PDPS is a version of the PDPT in which passengers are picked up at a large number of pickup locations and delivered to a few common destinations. In this case, destinations act as transfer points, all time windows are associated with locations, vehicles start and end all routes at a central depot, and vehicles can visit at most two delivery points each route. In order to utilize their branch-cut-and-price solution approach, they introduce two master problems based on set partitioning formulations. This requires splitting all routes into two independent parts. The first is a route in which a vehicle visits a pickup point and is emptied at a transfer point or at a delivery point called a pickup route. The second is a shuttle route which begins at a transfer point and delivers passengers directly to the next stop.

The first master problem does not consider the specific vehicles that are performing shuttle routes between delivery points. Instead, an integer variable representing the number of vehicles that travel between delivery points is used. In the second master problem, the vehicles performing shuttle routes are explicitly considered. In addition to the two master problems, two sub-problems are also introduced. The first sub-problem is dedicated to finding pickup routes while the second is dedicated to finding shuttle routes. Additionally, they introduce three families of valid inequalities and test their performance. The branch-cut-and-price solution approach for both master problems is tested using randomly generated problem instances. These problem instances consist of a large number of pickup locations delivering to a few destinations which also serve as possible transfer points. In the largest test cast, 75 requests must be delivered on a network of 50 nodes with three transfer points utilizing a single vehicle.

3.2.2 Heuristic Solution Methods

Mitrovic-Minic and Laporte [29] present a two-phased heuristic for the PDPT aimed at minimizing the distance required to deliver all requests while allowing for transshipment. A limitation of this heuristic is that vehicles capacities are not binding since no request capacity is considered. One transfer per request is allowed and is handled through splitting requests into two requests at the transshipment point. The two-phase heuristic consists of a construction phase followed by an improvement phase. In the construction phase, several solutions are constructed using different randomly generated initial orderings of requests. This is done sequentially through an insertion procedure in which each request is split for every transshipment point. The now split request is evaluated to see if there is a feasible insertion slot. If a feasible insertion slot is found, the least cost insertion slot is chosen and the next request is checked. If no feasible insertion slot is found, the non-split request is inserted into a new route without transshipment. The best solution from the construction phase is then used as the initial solution for the improvement phase. During the improvement phase, each requests is removed and reinserted back into the solution allowing for requests to be split and un-split several times throughout this phase. This process is repeated for a set number of iterations or until no improvement is found. The two-phased heuristic was used to quantify the benefits of transshipment on randomly generated problem instances of up to 100 requests, 200 nodes, and four transfer points.

Masson, Lehuede, and Peton [26] aim to efficiently explore the solution space of the model presented by Cortes, Matamala, and Contardo [10] by using an adaptive large neighborhood search (ALNS) heuristic. The ALNS tries to improve the objective each iteration by destroying and repairing existing solutions to create improved solutions. During each iteration of the ALNS, one destroy method and one repair method is chosen amongst several possibilities. The two methods chosen are based on a probability that changes throughout the search process based on how each method performs when it is chosen. There are three possible destroy methods for the ALNS. The first is the transfer point removal heuristic which removes all requests that use a given transfer point so they may be rerouted through another transfer point. The second is the pickup and delivery cluster removal heuristic which removes requests that can be delivered together by going through one transfer point on a single vehicle. The final destroy method is the history removal heuristic which removes requests that seem to be poorly placed based on a comparison to the best known solutions.

There are also three repair methods used by the ALNS. The first repair method is the best insertion with transfers heuristic. This heuristic splits requests into two legs: one leg going from the request's origin to a transfer point and the other leg going from a transfer point to request's destination. Then for each transfer point and request, the first leg is inserted into its best position on each route followed by the second leg being inserted into its best position. This insertion cost is then compared to inserting the non-split request into each route and the best insertion is chosen. The second repair method is the transfer first heuristic. This sequentially inserts requests into the solution forcing each request to use a transfer point. After all requests have been inserted in this way, each request is removed and reinserted without transfers to see if this improves the solution. The final repair method is the regret insertion with transfer heuristic. This simply inserts the request that has the largest difference between the cost of inserting the request that uses a transfer point to the cost of that request without using a transfer point. They apply their ALNS to two real-life problem instances corresponding to picking up passengers at many locations and delivering them to a few destinations. However, it is assumed that the number of vehicles is not binding. The first set of problem instances contains up to 193 requests at 193 nodes to be delivered to five destinations which also act as transfer points. The second set of problem instances contain up to 84 request at 84 nodes to be delivered to 21 destinations which also act as transfer points. Additionally they apply their ALNS to the problem instances introduced by Mitrovic-Minic and Laporte [29] which range in size from 50 to 100 requests, 100 to 200 nodes, and up to four transfer points.

Petersen and Ropke [33] present an ALNS that minimizes the total transportation cost for the pick-up and delivery problem with cross-docking opportunity. In this situation, a central cross-dock acts as a single transfer point and vehicle depot. Requests are picked up and delivered to locations which must be serviced with time windows that may not be violated. It is assumed that there is a specific location associated with each request, that only one transfer is allowed, and that service times at the cross-dock are dependent on the number of requests to be handled. In their ALNS, an initial solution is created using a construction heuristic in which requests are split into two requests if they can be cross-docked or left as one request if they cannot be cross-docked. After it is determined if all requests can be split, the initial solution is created by first inserting the split requests and then inserting the non-split requests based on a greedy criteria.

The ALNS then tries to improve upon this initial solution using two destroy methods and one repair method. The first destroy method randomly chooses a request to be removed from the solution and the second destroy method randomly selects a vehicle route and removes all requests from that route. The lone repair method reinserts requests based on the largest difference between the least cost and second least cost way of inserting a request. The ALNS is applied to industry instances containing up to 982 requests corresponding to 1964 nodes with a single transfer point. However, no further information is given on the problem instances.

Finally, Qu and Bard [35] describe a greedy randomized adaptive search procedure (GRASP) which they use to explore a new graphical representation for the PDPT. The primary objective is to minimize the number of vehicles required to service customer demand with a secondary objective of minimizing the travel distance of vehicles. The secondary objective only comes into play when an alternative optimal solution exists for the number of vehicles. In the new graphical representation, there is exactly one central depot for all vehicles and then for each customer there is one pickup node, one destination node, one pickup transshipment node, and one transshipment delivery node. Utilizing this representation, the GRASP consists of a two phases. Phase I creates a number of feasible solutions by constructing routes using a basic insertion operation and a feasibility checking operation. Phase II then looks to improve the construction solutions using an ALNS.

In Phase I, routes are sequentially created through randomly choosing a request to insert in the position that increases the cost of the new solution the least as measured by a greedy function. Insertion is done by either single route insertion or double route insertion. Single route insertion is when no transshipment happens and the movement of sending a request directly from origin to destination is inserted into an existing route. Double route insertion is when transshipment occurs and a request is split into two segments: one from the request's origin to the transshipment node and one from the transshipment node to the request's destination. These two new segments are then inserted into the same or different routes. Before any insertion is selected, a prorogation algorithm is used to update all routes and to check the feasibility of the insertions.

Phase II utilizes an ALNS to remove request and reinsert them elsewhere to create new, potentially better solutions. There are three request removal heuristics that are used. The first one randomly selects requests to remove, the second removal heuristic aims to remove requests that are similar to each other, and finally the third removal heuristic randomly selects an entire vehicle route and removes all requests from that route. After removal, requests are ordered and reinserted using one of four insertion heuristics. The first insertion heuristic is a greedy heuristic that evaluates all requests with respect to all possible insertion positions one at a time and chooses the best. The second is a regret heuristic that reinserts the request that would produce the biggest amount of regret if it were not inserted first. The third is a random heuristic that randomly reinserts a request and finally the fourth insertion heuristic determines the request reinsertion based on a weighted function of the distance between origin and destination, the time window, and the size of the request. The GRASP was applied with varying parameters to randomly generated instances with 25 requests requiring 100 nodes and one transshipment location. Since minimizing fleet size was an objective, there was an unlimited number of potential vehicles.

The remaining research on the PDPT explores the problem in ways other than providing a solution approach. Nakao and Nagamochi [31] find that the lower bound for the travel cost saved by introducing a single transshipment point and transfer to the PDP is proportional to the square root of the number of routes and requests in the optimal solution. Chou, Chen, and Chen [8] use the PDPT as a foundation to redesign operational policies in the transportation of medical materials between departments.

3.2.3 Summary of the PDPT Literature Review

The distinguishing characteristics found in the literature survey on the PDPT are summarized in Tables 3.1 and 3.2. Table 3.1 summarizes the characteristics about how each PDPT is modeled. This table provides if time is considered as continuous or discrete and how time windows are handled. Specifically of note is if time windows are enforced on transportation requests, locations, delivery resources or on all three. The table also provides any relevant notes on requests such as if split delivery is allowed and if the assignment of requests to individual vehicles is considered. Additionally, the table contains any notes on how transfers are handled including if only a single transfer is allowed or if a transfer point is modeled as two split nodes. Finally, the table also provides any notes on vehicles including if they must begin and end at a central depot. Table 3.2 presents the distinguishing characteristics about the solution approach. This table includes the type of solution approach and objective that each paper presents and the largest problem instance that is solved in terms of the number of requests, the number of nodes, the number of vehicles, and the number of transfer points.

There are three works from the literature that stand out as most closely related to the TSP-

CP, but they have some important differences. Kerivin, Lacroix, Mahjon, and Quillot [22] utilize discrete time periods with no restrictions on vehicle routes or the number of transfers. However, they employ a multi-commodity network flow based formulation in which individual vehicles are not distinguished. They also require that vehicles start and end their routes at a central depot, enforce no time windows on requests or vehicles, and allow for split delivery. Vornhusen, Wang, and Kopfer [40] investigate carrier collaboration utilizing a pickup and delivery problem with vehicle assignments, multiple vehicle depots, no split delivery, and time windows. However, they limit vehicles to one route in which they can only visit any location once, allow for a single transfer, and require transfers to be synchronized. Finally, Rais, Alevlos, and Carvolho [36] present a PDPTWT in which specific vehicle assignments are considered, time windows are present, multiple transfers are allowed, and vehicles have multiple depots. However, they allow for split deliveries and only allow for vehicles to be used on a single route.

In summary, the TSP-CP is the only known research to provide the flexibility of multiple depots for vehicles while placing no restrictions on the number of routes or location visits, to include time windows, and allow for multiple transfer opportunities while still considering the specific assignment of transportation requests to vehicles without split delivery. These advances allow the TSP-CP to be solved to find additional insights into the collaboration of FTL and LTL carriers above what is current available from the limited research.

Table 3.1: PDPT literature categorization information

Paper	Time	Time Windows	Notes on Requests	Notes on Transfers	Notes on Vehicles
Cortes, Matamala, and Contardo [3]	continuous	locations	no split delivery, vehicle assignments	transfer point is split	central depot, single route
Rais, Alevlos, and Carvolho [11]	continuous	requests	split delivery, vehicle assignments	multiple transfers, multiple locations	multiple depots, single route
Kerivin, Lacroix, Mahjon, and Quillot [6]	discrete	none	spit delivery, no vehicle assignments	multiple transfers, multiple locations	central depot, individual vehicles not considered
Grunert and Sebastian [5]	discrete	all three	split delivery, no vehicle assignments	transfer point is split	multiple depots, always unloaded when entering a node
Vornhusen, Wang, and Kopfer [12]	continuous	locations	no split delivery, vehicle assignments	single transfer, synchronized	central depot for each LTL carrier, single route
Dondo, Mendez, and Cerda [4]	continuous	locations	no split delivery, no vehicle assignments	transfer point is split	central depot, must visit nodes once
Masson, Ropke, Lehuede, and Peton [8]	continuous	locations	no split delivery, vehicle assignments	transfer point is split, located at destinations	central depot, at most two delivery stops each route
Mitrovic-Mini and Laporte [9]	continuous	locations	split delivery, no vehicle assignments	single transfer, requests split at transfer	no capacities, no individual vehicles
Masson, Lehuede, and Peton [7]	continuous	locations	no split delivery, vehicle assignments	transfer point is split, located at destinations	central depot, unlimited number of vehicles
Peterson and Ropke [2]	continuous	locations	no split delivery, vehicle assignments	single transfer	central depot
Qu and Bard [10]	continuous	locations	no split delivery, vehicle assignments	single transfer	central depot, unlimited number of vehicles

Table 3.2: PDPT literature categorization information

Paper	Approach	Objective (Minimize)	Requests	Nodes	Vehicles	TP
Cortes, Matamala, and Contardo [3]	branch-and-cut	ride time of vehicles, waiting and ride time of requests, fleet size, time window violations	6	13	2	1
Rais, Alevlos, and Carvolho [11]	mathematical model	cost of vehicle routes	7	14	7	14
Kerivin, Lacroix, Mahjon, and Quillot [6]	branch-and-cut	vehicle costs	15	10	7	10
Grunert and Sebastian [5]	none	vehicle and sorting costs	NA	NA	NA	NA
Vornhusen, Wang, and Kopfer [12]	mathematical model	total distance	9	20	2	1
Dondo, Mendez, and Cerda [4]	mathematical model	total routing costs	20	24	4	2
Masson, Ropke, Lehuede, and Peton [8]	branch-cut and price	total distance	75	53	1	3
Mitrovic-Mini and Laporte [9]	construction and improvement heuristic	total distance	100	200	NA	4
Masson, Lehuede, and Peton [7]	adaptive large neighborhood search	total distance	193	198	infinite	5
Peterson and Ropke [2]	adaptive large neighborhood search	transportation cost	982	1964	NA	1
Qu and Bard [10]	greedy randomized adaptive search procedure	fleet size, total distance	25	100	NA	1

Chapter 4

Integrated Mathematical Model

In this research, the TSP-CP is modeled as a time expanded pickup and delivery problem with time windows and transshipment (t-PDPTWT). The TSP-CP model includes the two independent, yet linked movements of the loads and the containers. Having separate variables that represent the movements of loads from the movements of containers allows the TSP-CP to decouple the load from the container. Decoupling the load from the container enables the model the ability to hold and transfer loads throughout the network. However, unlike previous research, this approach includes individual containers and specifies which loads containers are carrying at all times. Because split deliveries are not allowed, it is essential to track exactly which loads containers are carrying to ensure that no container capacity is ever exceeded. Furthermore, the use of discrete time periods and the ability to track which containers are carrying specific loads provides the TSP-CP with additional visibility that helps alleviate a potential barrier of collaboration. With this model, it is possible for TSPs to know exactly how their loads, containers, and facilities are being utilized at all times. The TSP-CP integrated mathematical model presented below simultaneously routes loads, assigns loads to specific containers, and routes containers without restriction while still allowing for load transfers that do not have to be synchronized.

4.1 Assumptions

The following assumptions are considered in modeling the TSP-CP.

- loads begin their routes by traveling from the depot to their origin node exactly once during their pickup time period and this occurs instantaneously (zero time periods elapse);
- loads end their routes by traveling from their destination node to the depot exactly once during their delivery time period and this occurs instantaneously (zero time periods elapse);
- a container route begins by traveling from the depot to its initial starting node and this occurs instantaneously (zero time periods elapse);
- a container route must end by traveling from its final ending node to the depot and this occurs instantaneously (zero time periods elapse);
- loads are not carried on containers to and from the depot;
- a container may only leave the depot one time, traveling to its initial node, and may only enter the depot one time, traveling from its final node;
- loads are only transported between different nodes on containers; when they are held at a node it is assumed that they are not assigned to a container; and
- pickup and delivery time windows for loads are hard time windows and may not be violated.

4.2 Model Formulation

The TSP-CP parameters, decision variables, objective function, and constraints are developed in this section.

Parameters

L	set of loads.	
N	set of nodes.	
H	set of transshipment nodes,	$H \subset N$.
NT	set of non-transshipment nodes,	$NT = N \setminus H$.
C	set of containers.	
T	set of time periods.	
T_{max}	the latest available time period.	
D	depot.	
τ	per load handling cost.	
η	per load holding cost.	
o_l	origin of load l,	$\forall l \in L$.
$[p_l^1, p_l^2]$	pickup time window for load l,	$\forall l \in L$.
f_l	destination of load l,	$\forall l \in L$.
$[d_l^1, d_l^2]$	delivery time window for load l,	$\forall l \in L$.
r_l	required space of load l,	$\forall l \in L$.
ω_c	per distance cost of using container c,	$\forall c \in C$.
a_c	initial node of container c,	$\forall c \in C$.
b_c	final node of container c,	$\forall c \in C$.
u_c	capacity of container c,	$\forall c \in C$.
m_{ij}	distance of arc (i, j),	$\forall i, j \in N$.
ϵ_{ij}	traversal time of arc (i, j) in time periods,	$\forall i, j \in N$.

Decision Variables

$$X^l_{itjq} \quad \begin{cases} 1 & \text{if load } l \text{ travels from node } i \text{ in time period } t \text{ to node } j \\ & \text{in time period } q = t + \epsilon_{ij} \leq T_{max}, \\ 0 & \text{otherwise.} \end{cases}$$

$$Z^{cl}_{itjq} \quad \begin{cases} 1 & \text{if load } l \text{ travels on container } c \text{ from node } i \text{ in time period } t \\ & \text{to node } j \neq i \text{ in time period } q = t + \epsilon_{ij} \leq T_{max}, \\ 0 & \text{otherwise.} \end{cases}$$

$$V^c_{itjq} \quad \begin{cases} 1 & \text{if container } c \text{ travels from node } i \text{ in time period } t \text{ to node } j \\ & \text{in time period } q = t + \epsilon_{ij} \leq T_{max}, \\ 0 & \text{otherwise.} \end{cases}$$

$$I_c \quad \begin{cases} 1 & \text{if container } c \text{ is used,} \\ 0 & \text{otherwise.} \end{cases}$$

$$\gamma^+_{lcit} \quad \begin{cases} 1 & \text{if load } l \text{ is loaded onto container } c \text{ at transshipment node } i \\ & \text{in time period } t, \\ 0 & \text{otherwise.} \end{cases}$$

$$\gamma^-_{lcit} \quad \begin{cases} 1 & \text{if load } l \text{ is unloaded from container } c \text{ at transshipment node } i \\ & \text{in time period } t, \\ 0 & \text{otherwise.} \end{cases}$$

Minimize

$$\sum_{c\in C}\sum_{i\in N}\sum_{t\in T}\sum_{j\in N}\sum_{q\in T}\omega_c m_{ij} V^c_{itjq} + \sum_{l\in L}\sum_{i\in H}\sum_{t\in T}\eta X^l_{itit+1} + \sum_{l\in L}\sum_{c\in C}\sum_{i\in H}\sum_{t\in T}\tau(\gamma^-_{lcit} + \gamma^+_{lcit}) \tag{4.1}$$

Subject to:

$$\sum_{t=p_l^1}^{p_l^2} X^l_{Dto_lt} = 1 \qquad \forall l \in L, \tag{4.2}$$

$$\sum_{t=d_l^1}^{d_l^2} X^l_{f_ltDt} = 1 \qquad \forall l \in L, \tag{4.3}$$

$$\sum_{j\in N}\sum_{q\in T} X^l_{itjq} - \sum_{j\in N}\sum_{q\in T} X^l_{jqit} = 0 \qquad \forall l \in L; i \in N; t \in T, \tag{4.4}$$

$$X^l_{itjq} = \sum_{c\in C} Z^{lc}_{itjq} \qquad \forall l \in L; i, j \in N : i \neq j, i \neq D, j \neq D; \; t, q \in T, \tag{4.5}$$

$$\sum_{l\in L} r_l Z^{lc}_{itjq} \leq u_c V^c_{itjq} \qquad \forall c \in C; i, j \in N : i \neq j; t, q \in T, \tag{4.6}$$

$$Z^{lc}_{itjq} \leq V^c_{itjq} \qquad \forall l \in L; c \in C; i, j \in N : i \neq j; t, q \in T, \tag{4.7}$$

$$I_c \geq V^c_{itjq} \qquad \forall c \in C; i, j \in N; t, q \in T, \tag{4.8}$$

$$\sum_{t\in T} V^c_{Dta_ct} = I_c \qquad \forall c \in C, \tag{4.9}$$

$$\sum_{t\in T} V^c_{b_ctDt} = I_c \qquad \forall c \in C, \tag{4.10}$$

$$\sum_{j\in N}\sum_{q\in T} V^c_{itjq} - \sum_{j\in N}\sum_{q\in T} V^c_{jqit} = 0 \qquad \forall c \in C; i \in N; t \in T, \tag{4.11}$$

$$\sum_{j\in N}\sum_{q\in T} Z^{lc}_{itjq} - \sum_{j\in N}\sum_{q\in T} Z^{lc}_{jqit} = 0 \qquad \forall l \in L; c \in C; i \in NT : i \neq o_l, i \neq f_l; \; t \in T, \tag{4.12}$$

$$\sum_{j\in N}\sum_{q\in T} Z^{lc}_{itjq} - \sum_{j\in N}\sum_{q\in T} Z^{lc}_{jqit} \leq \gamma^-_{lcit} \qquad \forall c \in C; l \in L; i \in H; t \in T, \tag{4.13}$$

$$\sum_{j\in N}\sum_{q\in T} Z^{lc}_{jqit} - \sum_{j\in N}\sum_{q\in T} Z^{lc}_{itjq} \leq \gamma^+_{lcit} \qquad \forall c \in C; l \in L; i \in H; t \in T, \tag{4.14}$$

$$X^l_{itjq} \in \{0,1\}, \text{ integer} \qquad \forall l \in L; i, j \in N; t, q \in T, \tag{4.15}$$

$$V^c_{itjq} \in \{0,1\}, \text{ integer} \qquad \forall c \in C; i, j \in N; t, q \in T, \tag{4.16}$$

$$Z^{lc}_{itjq} \in \{0,1\}, \text{ integer} \qquad \forall l \in L; c \in C; i, j \in N : i \neq j; t, q \in T. \tag{4.17}$$

$$I_c \in \{0,1\}, \text{ integer} \qquad \forall c \in C, \tag{4.18}$$

$$\gamma^-_{lcit} \in \{0,1\}, \text{ integer} \qquad \forall l \in L; c \in C; i \in H; t \in T, \tag{4.19}$$

$$\gamma^+_{lcit} \in \{0,1\}, \text{ integer} \qquad \forall l \in L; c \in C; i \in H; t \in T, \tag{4.20}$$

The objective function (4.1) minimizes the total transportation cost required to deliver all loads based on the distance traveled by containers and the cost for using each container. The per mile cost (ω_c) for each $c \in C$ is slightly altered by container in order to alleviates symmetry concerns. Additionally, the objective function includes the cost required to handle and hold loads at transshipment nodes. Constraint (4.2) enforces that each load must enter its origin node from the depot once during its pickup time window. Constraint (4.3) enforces that each load must enter the depot from its destination node once during its delivery time window. Constraint (4.4) maintains the flow-conservation of loads in the network. Constraint (4.5) requires that if a load travels an arc, a container must carry it. Constraint (4.6) ensures that a container's capacity is never exceeded. Constraint (4.7) is a valid inequality that ensures a load cannot travel an arc on a container unless a container also travels that arc. Constraint (4.8) sets an indicator variable to one if a container is used across any arc. Constraints (4.9) and (4.10) ensure that if a container is in use, it must leave from its initial node and return to its final node. Constraint (4.11) maintains the flow-conservation of containers in the network. Constraint (4.12) maintains the flow-conservation of loads on containers through non-transshipment nodes by ensuring that a load enters and leaves on the same container. Constraint (4.13) captures the number of times that a load is taken off a container at a transshipment node while, Constraint (4.14) captures the number of times a load is loaded onto a container at a transshipment node. Constraints (4.15)–(4.20) enforce the binary restrictions on the decision variables.

Chapter 5

Solution Approaches for the TSP-CP

Given the difficulty of finding an optimal solution to the TSP-CP using the integrated model, a heuristic solution approach is developed to provide quality solutions to industry representative problems. Two heuristic solution approaches are presented which aim to increase the potential applications of the TSP-CP. The two heuristics are the fixed-optimization adaptive large neighborhood search and the heuristic insertion adaptive large neighborhood search. The required background information and detailed descriptions of both heuristics are covered in this chapter.

5.1 Background Information

Both heuristics introduced in this research utilize an adaptive large neighborhood (ALNS) search framework to explore the solution space. The ALNS is an extension of the large neighborhood search algorithm that has shown promise in providing solutions for vehicle routing problems and pickup and delivery problems.

5.1.1 Large Neighborhood Search

The large neighborhood search (LNS) is a local search algorithm introduced by Paul Shaw for the vehicle routing problem [38]. A local search algorithm is a technique that iteratively explores the solution space by quickly moving between feasible solutions. This move is done every iteration by making changes to the current solution in order to create a neighboring solution. In order to maximize the number of solutions that are visited, the neighborhood move is often based on a simple operation, such as swapping the position of two stops on a container route. Due to this, local search algorithms are often reliant on meta-heuristics to escape local minima allowing the algorithm to explore large portions of the solution space. Even with the use of a meta-heuristic, local search algorithms can still struggle to explore promising areas of the solution space leading to less than satisfactory performance on difficult problems.

The LNS introduced by Shaw overcomes this limitation by incorporating a much more powerful neighborhood move operator in place of the small changes used by traditional local search algorithms. The move operator removes part of one solution, fixes the remaining parts, and then optimally reinserts the removed parts back in to create a neighboring solution. The move operator is characterized by a single destroy method and a single repair method. The destroy method is the procedure that selects parts of the solution for removal to create a partial solution. The repair method is the procedure which reinserts all removed portions to create the neighboring solution. Specifically, Shaw applies the LNS to the vehicle routing problem using a destroy method which select a set of customer visits to remove based on how similar they are to each other. After the destroy method has removed the customer visits, the repair method reinserts them in their optimal position to minimize cost using constraint programming techniques.

5.1.2 Adaptive Large Neighborhood Search

The adaptive large neighborhood search (ALNS) heuristic introduced by Pisinger and Ropke [34] is an extension of the LNS introduced by Shaw [38]. Instead of using the same destroy and repair method for the entire search, the ALNS uses several different destroy and repair methods. Although there are several possibilities, each iteration employs a single destroy method and a single repair method allowing for the selection of methods to evolve over time. The destroy and repair methods are chosen independently of each other for every iteration. A method in a given iteration is chosen by a probability that adapts based on performance as the search progresses. For example, given that there are m_d destroy methods, each destroy method, d, is assigned a weight w_d. This weight determines the probability of destroy method d being chosen in each iteration calculated by:

$$\frac{w_d}{\sum_{i \in m_d} w_i} \tag{5.1}$$

To guide the search, the weights of the methods are changed throughout but do not get updated every iteration. Instead they are updated at the end of every segment. A segment is simply a set number of iterations during which the probability of choosing a method remains constant. During every segment a score, π_m, is assigned to each method. This score is reset to zero at the beginning of every segment and is updated each time a method is chosen based on what happens as a result of applying the method in that iteration. A higher score indicates the method performed better during that segment. Additionally, the number of times, t_m, that a method is chosen in a segment is calculated as well. Again, t_m is set to zero at the start of each segment and is updated by one each time method m is chosen. When a segment ends, π_m and t_m are used to update the weight of each method, w_m, as follows:

$$w_m = w_m(1 - \alpha) + \alpha(\frac{\pi_m}{t_m}) \tag{5.2}$$

The reaction factor, α, is used to control how much the score of the current segment influences the weight of a method in the next segment. If α is set to zero, the score remains constant throughout the search. Conversely, if α is set to one, the weight in the next segment is solely based on the performance in the previous segment. Similarly, repair methods are chosen independently with their own set of weights and probabilities. After the weights are updated, the probability of choosing a method is recalculated with the new weights using Equation 5.1. This approach ensures that methods with a track record of performing well will be more likely to be chosen during the upcoming segments.

Although the ALNS is an extension of the LNS by Shaw [38], moving to neighboring solutions is done much differently and performs more like a traditional local search algorithm. The ALNS typically uses small moves and heuristic insertion methods. Due to this, it relies heavily on a meta-heuristic to escape local minima. The most commonly used meta-heuristic with the ALNS is simulated annealing. More information on simulated annealing is provided in Section 5.1.3.

5.1.3 Simulated Annealing

Simulated annealing is a meta-heuristic that strives to keep the search from becoming trapped in a local minimum by sometimes accepting a solution with an objective function value that is worse than the current solution objective function value. Making this seemingly "bad" move during the search allows for the ALNS to reach areas of the solution space that may otherwise be missed. The probability of accepting a worse solution x' given the search is at the current solution x is given as:

$$e^{-\frac{f(x')-f(x)}{T}} \tag{5.3}$$

where $f(x')$ is the objective function value of x', $f(x)$ is the objective function value of x, and T is the temperature. The temperature is a non-negative parameter that controls the search by determining how often a worse solution is accepted. As the search progresses, the temperature is reduced such that a worse solution is accepted with a smaller and smaller probability. The rate at which the temperature is reduced, the cooling rate, is a non-zero value less than one. Temperature begins at T_{start} and is updated by the cooling rate each subsequent iteration such that $T = Tc$ where c is the cooling rate. Although it is possible to set simply T_{start} as a large number, with vehicle routing or pick up and delivery problems the starting temperature is often based on the initial starting solution. As suggested by Pisinger and Ropke [34], T_{start} is set such that a solution with an objective function value $w\%$ worse than the starting solution objective function value has an acceptance probability of 0.5.

5.1.4 ALNS applications for the PDPT

As introduced in Chapter 3, there have been three implementations of an ALNS as part of a solution approach to the PDPT. The first of these comes from Masson, Lehuede, and Peton [26]. They apply an ALNS to the PDPT considering only a single transfer. Their application focuses on showing the improvements that introducing a single transfer can make by solving real-life instances motivated by the transportation of people with disabilities from their home to schools, social centers, and vocational rehabilitation centers. As such, the network contains many pick up points with a limited number of delivery and transfer points. They also introduce four methods to determine a subset of transfer points to consider each iteration; however, they limit this to a single transfer point in each case.

The second application of an ALNS to the PDPT is from Petersen and Ropke [33]. They consider only a single transfer and require that all containers routes begin and end at the central cross-dock. They apply their ALNS to make flower deliveries from greenhouses to florists and supermarkets. In the network they consider, pick ups from greenhouses can either be delivered direct or pick ups may return to the cross-dock before delivery where they can be consolidated with other shipments. They include extensions to consider only the 10 best feasible routes according to an increase in distance and provide the ability to return to the global best solution at the start of each iteration. They apply their ALNS to solve industry instances in order to compare the variation in results and the benefit of returning to the best solution throughout the search.

The final application of the ALNS to the PDPT comes as part of the greedy randomized adpative search procedure (GRASP) presented by Qu and Bard [35]. Their ALNS acts as an improvement phase to their GRASP procedure following a construction phase. They apply their ALNS to randomly generated instances containing a single transfer point and require that containers start and end their routes at a central depot. The resulting GRASP solutions on the randomly generated data sets are compared to the optimal values considering various parameter settings.

The ALNS is characterized by the destroy and repair methods that are implemented and Tables 5.1 and 5.2 summarize the destroy and repair methods previously introduced for the PDPT.

Table 5.1: PDPT Destroy Methods

Paper	Destroy Method	Description
Masson, Lehuede, and Peton [26]	Transfer Point Removal	Removes all requests that use a given transfer point
	Pickup/Delivery Cluster Removal	Removes requests that can be delivered together by going through one transfer point on a single vehicle
	History Removal	Removes requests that seem to be poorly placed based on the best known solutions
Petersen and Ropke [33]	Random Destroy	Randomly selects requests for removal
	Trip Destroy	Randomly selects a vehicle route and removes all requests on that route
Qu and Bard [35]	Shaw Removal	Removes requests based on the formula introduced by Shaw [38]
	Random Removal	Randomly selects requests for removal
	Route Random Sweep Removal	Randomly selects a vehicle route and removes all requests on that route but leaves the empty route in the solution

Table 5.2: PDPT Repair Methods

Paper	Repair Method	Description
Masson, Lehuede, and Peton [26]	Best Insertion with Transfer	Sequentially evaluates all possible insertion points and selects the best
	Transfer First	Gives priority to the use of transfer points, but evaluates the cost of insertion with transfers and without transfers
	Regret Insertion with Transfer	Evaluates the difference in cost of inserting a request with transfer and without transfer and the largest such difference is inserted in its best position
Petersen and Ropke [33]	Regret Measure	Inserts the request that has the largest difference between the least cost and the second least cost insertion point
Qu and Bard [35]	Greedy Insertion	Evaluates all possible insertion points in all routes
	Regret-k Insertion	Selects the request that would produce the greatest regret were it not inserted first
	Random Insertion	Randomly decides the order in which requests are inserted and then uses the greedy insertion method
	Most Constrained First	Determines the order of insertion for use with the greedy insertion method by considering a weighted function of the distance between origin and destination, time windows, and shipment size

5.2 ALNS for the TSP-CP

The application of the ALNS to the TSP-CP comes in the form of two separate ALNS frameworks that rely on different repair method ideologies. In both cases, the same construction heuristic is used to generate initial solutions and the same destroy methods are used to remove loads to create partial solutions for repair. The two heuristics differ in how the removed portions of the solution are reinserted to create neighboring solutions. The fixed-optimization ALNS for the TSP-CP relies on powerful, but computationally expensive repair methods to create new solutions and does not rely on a meta-heuristic to guide the search. Instead a large portion of the solution is removed by the destroy method and is then simultaneously reinserted by solving the integrated TSP-CP mathematical model with limited variables and additional constraints. The other heuristic introduced here is the heuristic insertion ALNS. In place of the computational expensive optimization based repair method, the heuristic insertion ALNS repairs solutions by reinserting removed portions of the solution using a sequential insertion procedure. This requires making smaller changes to the solution each iteration and therefore requires many more iterations to fully search the solution space. Furthermore, a simulated annealing meta-heuristic is required to keep the search from becoming stuck in local minima.

An overview of the the TSP-CP heuristics is shown in Figure 5.1. These heuristics require a unique framework as the searches must be guided differently. Sections 5.3 and 5.4 cover the two heuristics and their ALNS frameworks in more detail. Although two heuristics are introduced, when providing a solution to the TSP-CP, only one heuristic is applied for the entire search. Either the fixed-optimization based insertion procedure or the heuristic insertion procedure is used for one full application of the ALNS and they are never mixed. In addition, both heuristics utilize the same construction heuristic to provide the initial feasible solution.

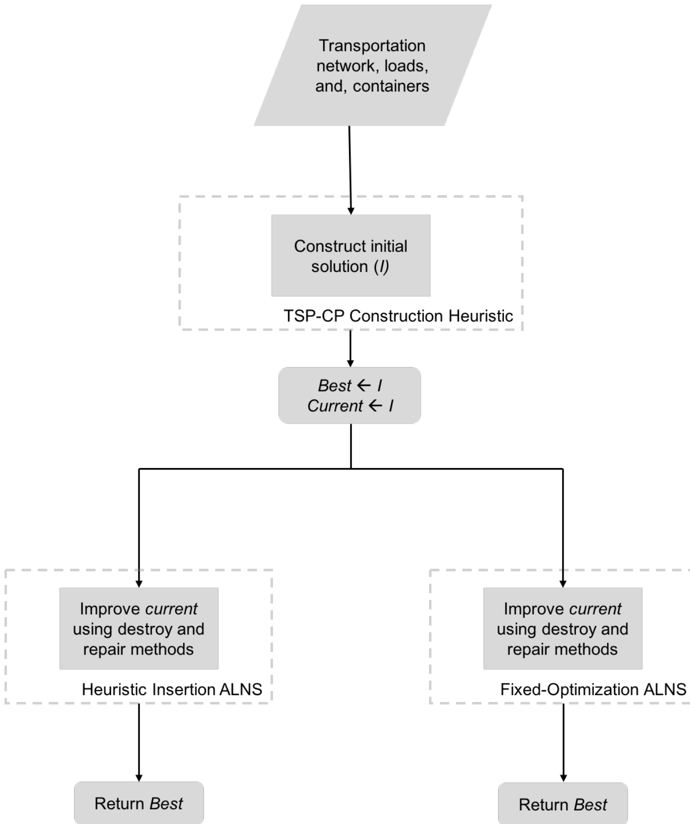

Figure 5.1: Overview of the TSP-CP Heuristics

5.2.1 Generating an Initial Solution: the TSP-CP Construction Heuristic

The ALNS requires an initial starting solution which is then partially destroyed and repaired to find the first neighboring solution. Typically, this solution is quickly generated using one of the preexisting repair methods. However, generating an initial solution in this manner can lead to a solution which requires a significant amount of time and computational effort to find adequate improvement. In the application of the ALNS to the TSP-CP, a decomposed optimization construction heuristic is used to find an initial feasible solution in an attempt to reduce this additional time and effort.

The construction heuristic decomposes the integrated TSP-CP mathematical model from Section 4.2 into three smaller optimization problems that are handled in different phases.

Figure 5.2 illustrates the iterative process that is used to find a solution using these three phases. Phase I creates load routes by finding the assignment of loads to arcs. In this phase, loads must be delivered from origin to destination and their pickup and delivery time windows must be satisfied. The load routes developed in Phase I serve as an input to Phase II. Phase II determines paths that will have to be covered by container routes in Phase III. The first part of Phase II assigns loads to containers across arcs by enforcing container capacity restrictions while not allowing for split delivery or for loads to change containers at non-transshipment nodes. Using these load to container assignments on individual arcs, the second part of Phase II develops paths that will have to be covered by a single container in Phase III. Finally, Phase III takes the paths from Phase II and creates routes for containers that start at their initial node and end at their final node.

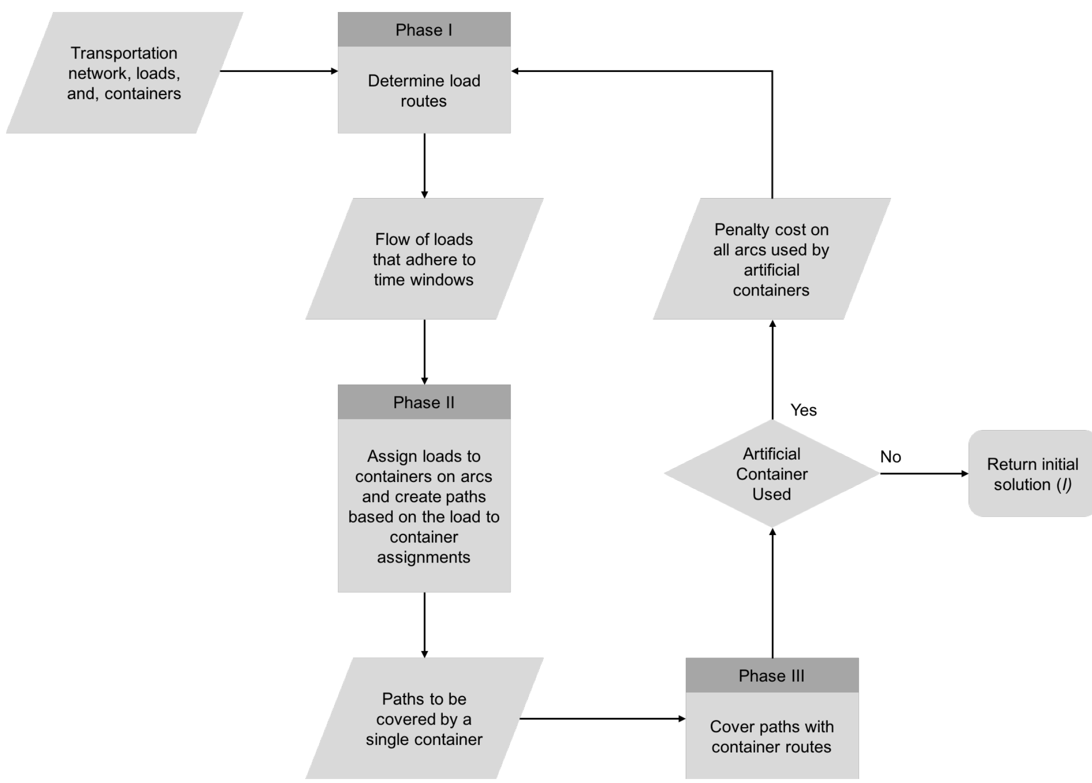

Figure 5.2: Flow chart of the construction heuristic

One drawback of the decomposition is that feasibility in Phase III is not guaranteed by the solutions from Phases I and II. As such, artificial containers are introduced to cover paths from Phase II that are not feasible using the original container set in Phase III. If an artificial container is used in the Phase III solution, a penalty cost is added to the Phase I objective function so the arcs making up the artificial container routes become less attractive to loads in the next iteration. At this point, all three phases are resolved and the process is repeated until no artificial containers are used in the Phase III solution.

The optimization models for each phase and the information passed between the phases are detailed in this section. The following global parameters are used throughout all three phases.

Global Parameters

L	set of loads.
N	set of nodes.
H	set of transshipment nodes.
NT	set of non-transshipment nodes.
C	set of containers.
T	set of time periods.
T_{max}	the latest available time period.
D	depot.
τ	per load handling cost.
η	per load holding cost.
m_{ij}	distance of arc (i, j).
ϵ_{ij}	traversal time of arc (i, j) in time periods.

Phase I of the Construction Heuristic

Phase I of the construction heuristic determines load routes by finding the assignment of loads to arcs. In this phase, individual containers are not considered and split deliveries are allowed. The objective is to minimize the travel distance required to deliver all loads from their origin to their destination while satisfying pickup and delivery time windows. The additional parameters, the decision variables, and the mathematical model are detailed below.

Additional Parameters

o_l	origin of load l.
$[p_l^1, p_l^2]$	pickup time window for load l.
f_l	destination of load l.
$[d_l^1, d_l^2]$	delivery time window for load l.
r_l	required space of load l.
p_{itjq}^l	penalty cost of load l traveling from node i in time t to node j in time q.
u	average container capacity of all containers in the original problem.
θ_i^l	$\begin{cases} 1 & \text{if node } i \text{ is the origin node for load } l, \\ 0 & \text{otherwise.} \end{cases}$

Decision Variables

X_{itjq}^l	$\begin{cases} 1 & \text{if load } l \text{ travels from node } i \text{ in time } t \text{ to node } j \text{ in time } q = t + \epsilon_{ij} \leq T_{max}, \\ 0 & \text{otherwise.} \end{cases}$
A_{itjq}	the number of containers that travel from node i in t to node j in $q = t + \epsilon_{ij} \leq T_{max.}$
O_{it}	the number of loads that leave node i as their origin node in time t.

Minimize

$$\sum_{i\in N}\sum_{t\in T}\sum_{j\in N}\sum_{q\in T} m_{ij}A_{itjq} + \sum_{l\in L}\sum_{i\in H}\sum_{t\in T}\eta X^l_{itit+1} + \sum_{l\in L}\sum_{i\in N}\sum_{t\in T}\sum_{j\in N}\sum_{q\in T} p^l_{itjq}X^l_{itjq} \tag{5.4}$$

Subject to:

$$\sum_{t=p_l^1}^{p_l^2} X^l_{Dto_lt} = 1 \qquad \forall l \in L, \tag{5.5}$$

$$\sum_{t=d_l^1}^{d_l^2} X^l_{f_ltDt} = 1 \qquad \forall l \in L, \tag{5.6}$$

$$\sum_{j\in N}\sum_{q\in T} X^l_{itjq} - \sum_{j\in N}\sum_{q\in T} X^l_{jqit} = 0 \qquad \forall l \in L; i \in N; t \in T, \tag{5.7}$$

$$\sum_{l\in L} r_l X^l_{itjq} \le uA_{itjq} \qquad \forall c \in C; i,j \in N; t \in T; q = t + \epsilon_{ij} \le T_{max}, \tag{5.8}$$

$$\sum_{l\in L}\sum_{j\in N}\sum_{q\in T} \theta^l_i X^l_{itjq} = O_{it} \qquad \forall i \in N; t \in T, \tag{5.9}$$

$$\sum_{j\in N}\sum_{q\in T} A_{itjq} \le \sum_{j\in N}\sum_{q\in T} A_{jqit} + O_{it} \qquad \forall i \in NT; t \in T, \tag{5.10}$$

$$X^l_{itjq} \in \{0,1\},\ \text{integer} \qquad \forall l \in L; i,j \in N; t \in T; q = t + \epsilon_{ij} \le T_{max}, \tag{5.11}$$

$$A_{itjq} \ge 0,\ \text{integer} \qquad \forall i,j \in N; t \in T; q = t + \epsilon_{ij} \le T_{max}, \tag{5.12}$$

$$O_{it} \ge 0,\ \text{integer} \qquad \forall i \in N; t \in T. \tag{5.13}$$

The objective function (5.4) minimizes the distance traveled by containers and the cost associated with holding loads. Additionally the objective function contains a penalty cost, p^l_{itjq}, for load l traveling from node i in time t to node j in time q. This penalty cost will be utilized after Phase III to ensure that arcs travelled by loads on artificial containers are less attractive. As such it is zero unless otherwise noted. Constraints (5.5)–(5.7) are Constraints (4.2)–(4.4) from the integrated model in Section 4.2. Constraint (5.8) requires that the total space requirement of all loads across an arc is covered by sufficient container capacity. Constraint (5.9) tracks the number of loads that leave node i as their origin node in time t. Constraint (5.10) ensures that the number of containers exiting a non-transshipment node is less than or equal to the number of containers that enter that node plus the number of loads that exit that node in time t given it is their origin node. This constraint aims to ensure that transfers do not occur at non-transshipment nodes by not allowing containers to leave from a non-transshipment node unless a container enters or the node acts as an origin for a load. Finally, Constraints (5.11)–(5.13) enforces the logical restrictions on the decision variables. The solution of this model provides the load routes utilized in Phase II.

Phase II of the Construction Heuristic

Using the load routes from Phase I, Phase II creates load to container assignments without using split delivery. Phase II utilizes the initial container set, but does not require that these containers start and end their route at any predefined node. However, Phase II does require the flow-conservation of containers at all nodes and does not allow for loads to be transferred between containers at non-transshipment nodes. The parameters, decision variables, objective function, and constraints are presented below for the Phase II optimization model.

Additional Parameters

u_c capacity of container c.

r_l required space of load l.

$$x^l_{itjq} \quad \begin{cases} 1 & \text{if load } l \text{ travels from node } i \text{ in time } t \text{ to node } j \text{ in time } q \text{ as found in Phase I,} \\ 0 & \text{otherwise.} \end{cases}$$

Decision Variables

$$Z^{cl}_{itjq} \quad \begin{cases} 1 & \text{if load } l \text{ travels on container } c \text{ from node } i \text{ in } t \text{ to node } j \neq i \text{ in } q = t + \epsilon_{ij} \leq T_{max}, \\ 0 & \text{otherwise.} \end{cases}$$

$$V^c_{itjq} \quad \begin{cases} 1 & \text{if container } c \text{ travels from node } i \text{ in time } t \text{ to node } j \text{ in time } q = t + \epsilon_{ij} \leq T_{max}, \\ 0 & \text{otherwise.} \end{cases}$$

$$\gamma^+_{lcit} \quad \begin{cases} 1 & \text{if load } l \text{ is loaded onto container } c \text{ at transshipment node } i \text{ in time period } t, \\ 0 & \text{otherwise.} \end{cases}$$

$$\gamma^-_{lcit} \quad \begin{cases} 1 & \text{if load } l \text{ is unloaded from container } c \text{ at transshipment node } i \text{ in timer period } t, \\ 0 & \text{otherwise.} \end{cases}$$

Minimize

$$\sum_{c\in C}\sum_{i\in N}\sum_{t\in T}\sum_{j\in N}\sum_{q\in T} m_{ij}V^c_{itjq} + \sum_{l\in L}\sum_{c\in C}\sum_{i\in H}\sum_{t\in T}\tau(\gamma^-_{lcit} + \gamma^+_{lcit}) \tag{5.14}$$

Subject to:

$$x^l_{itjq} = \sum_{c\in C} Z^{lc}_{itjq} \qquad \forall l \in L; i,j \in N : i \neq j, i \neq D, j \neq D; t,q \in T, \tag{5.15}$$

$$\sum_{l\in L} r_l Z^{lc}_{itjq} \leq u_c V^c_{itjq} \qquad \forall c \in C; i,j \in N : i \neq j; t,q \in T, \tag{5.16}$$

$$Z^{lc}_{itjq} \leq V^c_{itjq} \qquad \forall l \in L; c \in C; i,j \in N : i \neq j; t,q \in T, \tag{5.17}$$

$$\sum_{j\in N}\sum_{q\in T} V^c_{itjq} - \sum_{j\in N}\sum_{q\in T} V^c_{jqit} = 0 \qquad \forall c \in C; i \in N; t \in T, \tag{5.18}$$

$$\sum_{j\in N}\sum_{q\in T} Z^{lc}_{itjq} - \sum_{j\in N}\sum_{q\in T} Z^{lc}_{jqit} = 0 \qquad \forall l \in L; c \in C; i \in NT : i \neq o_l, i \neq f_l; t \in T, \tag{5.19}$$

$$\sum_{j\in N}\sum_{q\in T} Z^{lc}_{itjq} - \sum_{j\in N}\sum_{q\in T} Z^{lc}_{jqit} \leq \gamma^-_{lcit} \qquad \forall l \in L; c \in C; i \in H; t \in T, \tag{5.20}$$

$$\sum_{j\in N}\sum_{q\in T} Z^{lc}_{jqit} - \sum_{j\in N}\sum_{q\in T} Z^{lc}_{itjq} \leq \gamma^+_{lcit} \qquad \forall l \in L; c \in C; i \in H; t \in T, \tag{5.21}$$

$$V^c_{itjq} \in \{0,1\}, \text{ integer} \qquad \forall c \in C; i,j \in N; t,q \in T, \tag{5.22}$$

$$Z^{lc}_{itjq} \in \{0,1\}, \text{ integer} \qquad \forall l \in L; c \in C; i,j \in N : i \neq j; t,q \in T, \tag{5.23}$$

$$\gamma^-_{lcit} \in \{0,1\}, \text{ integer} \qquad \forall l \in L; c \in C \cup \Lambda; i \in H; t \in T, \tag{5.24}$$

$$\gamma^+_{lcit} \in \{0,1\}, \text{ integer} \qquad \forall l \in L; c \in C \cup \Lambda; i \in H; t \in T. \tag{5.25}$$

The objective function (5.14) minimizes the total transportation cost required to deliver all loads based on the distance traveled by containers and the cost required handle loads at transshipment nodes. Constraints (5.15)–(5.17) and Constraints (5.18)–(5.21) are Constraints (4.5)–(4.7) and Constraints (4.11)–(4.14) from the integrated model in Section 4.2. Constraints (5.22)–(5.25) enforce the binary restrictions on the variables. The result of this model is the assignment of loads to containers on arcs which will be converted into paths that will need to be covered in Phase III using the following procedure.

In order to provide paths for Phase III, an additional post processing step converts the Z variables from the Phase II solution into paths that will be covered by containers in Phase III. In order to do this, the following set and variables are introduced.

Set

L^c_{itjq} the set of loads that container c carries from node i in time t to node j in time q

Variables

$$z^{lp}_{itjq} \begin{cases} 1 & \text{if load } l \text{ travels on path } p \text{ from node } i \text{ in time } t \text{ to node } j \text{ in time } q = t + \epsilon_{ij} \leq T_{max}, \\ 0 & \text{otherwise.} \end{cases}$$

$$\delta^{p}_{itjq} \begin{cases} 1 & \text{if path } p \text{ travels from node } i \text{ in time } t \text{ to node } j \text{ in time } q = t + \epsilon_{ij} \leq T_{max}, \\ 0 & \text{otherwise.} \end{cases}$$

Algorithm 2, described in Figure 5.3, creates a sequence of paths that must be covered by a single container in Phase III. These paths are based on the load on container assignments from the Phase II solution. This additional step is required given that loads may only be transferred between containers at transshipment nodes. A load on container route found in Phase II may be split into a new path whenever the container exits a transshipment node or whenever all loads that are carried on the container into a node are unloaded from the container.

Algorithm 2 Phase II: Path Creation

procedure PATH CREATION
  $p = 0$
  for $c \in C$ **do**
    $p = p + 1$
    for $l \in L; i, j \in N; t, q \in T$ **do**
      if $Z^{lc}_{itjq} > 0$ **then**
        if $i \notin H$ & $L^c_{itjq} \cap L^c_{jqit} \neq \emptyset$ **then**
          $\delta^p_{itjq} = 1$
          $z^{lp}_{itjq} = 1$
        else
          $p = p + 1$
          $\delta^p_{itjq} = 1$
          $z^{lp}_{itjq} = 1$
  return $\delta^p_{itjq}, z^{lp}_{itjq}$

Figure 5.3: Phase II: Path Creation

Phase III of the Construction Heuristic

Phase III covers each path found in Phase II using the given container set making certain that containers start their route at their initial node and end their route at their final node. However, due to the time component considered in the TSP-CP, artificial containers are introduced to maintain feasibility. These artificial containers are not required to start or end their routes at predefined nodes. This allows the artificial containers to cover any paths that may not be covered using the original container set. In order to make these artificial containers unattractive, the cost of using an artificial container is much greater than that of using a regular container. If any artificial container is used in the optimal solution found in Phase III, an iterative solution process is utilized to remove the artificial containers in the final construction heuristic solution.

Additional Parameters

Λ	set of artificial conatiners.
$C \cup \Lambda$	set of all containers and artificial containers.
P	set of paths from Phase II: Path Creation.
ω_c	per distance cost of using container c.
a_c	initial node of container c.
b_c	final node of container c.
u_c	capacity of container c.
z_{itjq}^{lp}	$\begin{cases} 1 & \text{if load } l \text{ travels on path } p \text{ from node } i \text{ in time } t \text{ to node } j \text{ in time } q, \\ 0 & \text{otherwise.} \end{cases}$
δ_{itjq}^{p}	$\begin{cases} 1 & \text{if path } p \text{ travels from node } i \text{ in time } t \text{ to node } j \text{ in time } q, \\ 0 & \text{otherwise.} \end{cases}$

Decision Variables

$$\lambda_p^c \quad \begin{cases} 1 & \text{if path } p \text{ is covered by container } c, \\ 0 & \text{otherwise.} \end{cases}$$

$$I_c \quad \begin{cases} 1 & \text{if container } c \text{ is used,} \\ 0 & \text{otherwise.} \end{cases}$$

$$V_{itjq}^c \quad \begin{cases} 1 & \text{if container } c \text{ travels from node } i \text{ in time } t \text{ to node } j \text{ in time } q = t + \epsilon_{ij} \leq T_{max}, \\ 0 & \text{otherwise.} \end{cases}$$

$$\gamma_{lcit}^+ \quad \begin{cases} 1 & \text{if load } l \text{ is loaded onto container } c \text{ at transshipment node } i \text{ in time period } t, \\ 0 & \text{otherwise.} \end{cases}$$

$$\gamma_{lcit}^- \quad \begin{cases} 1 & \text{if load } l \text{ is unloaded from container } c \text{ at transshipment node } i \text{ in timer period } t, \\ 0 & \text{otherwise.} \end{cases}$$

Minimize

$$\sum_{c\in C}\sum_{i\in N}\sum_{t\in T}\sum_{j\in N}\sum_{q\in T}\sum_{p\in P}\omega_c m_{ij}\delta_{itjq}^p\lambda_p^c + \sum_{l\in L}\sum_{c\in C}\sum_{i\in H}\sum_{t\in T}\tau(\gamma_{lcit}^- + \gamma_{lcit}^+) \tag{5.26}$$

Subject to:

$$I_c \geq V_{itjq}^c \qquad \forall c \in C; i, j \in N; t, q \in T, \tag{5.27}$$

$$\sum_{t\in T} V_{Dta_ct}^c = I_c \qquad \forall c \in C, \tag{5.28}$$

$$\sum_{t\in T} V_{b_ctDt}^c = I_c \qquad \forall c \in C, \tag{5.29}$$

$$\sum_{j\in N}\sum_{q\in T} V_{itjq}^c - \sum_{j\in N}\sum_{q\in T} V_{jqit}^c = 0 \qquad \forall c \in C; i \in N; t \in T, \tag{5.30}$$

$$\sum_{j\in N}\sum_{q\in T}\sum_{p\in P} z_{itjq}^{lp}\lambda_c^p - \sum_{j\in N}\sum_{q\in T}\sum_{p\in P} z_{jqit}^{lp}\lambda_c^p \leq \gamma_{lcit}^- \qquad \forall l \in L; c \in C \cup \Lambda; i \in H; t \in T, \tag{5.31}$$

$$\sum_{j\in N}\sum_{q\in T}\sum_{p\in P} z_{jqit}^{lp}\lambda_c^p - \sum_{j\in N}\sum_{q\in T}\sum_{p\in P} z_{itjq}^{lp}\lambda_c^p \leq \gamma_{lcit}^+ \qquad \forall l \in L; c \in C \cup \Lambda; i \in H; t \in T, \tag{5.32}$$

$$V_{itjq}^c \geq \sum_{p\in P}\delta_{itjq}^p\lambda_p^c \qquad \forall c \in C \cup \Lambda; i, j \in N; t, q \in T, \tag{5.33}$$

$$\sum_{c\in C\cup\Lambda}\lambda_p^c = 1 \qquad \forall p \in P, \tag{5.34}$$

$$\lambda_p^c \in \{0,1\}, \text{ integer} \qquad \forall c \in C \cup \Lambda; p \in P, \tag{5.35}$$

$$V_{itjq}^c \in \{0,1\}, \text{ integer} \qquad \forall c \in C \cup \Lambda; i, j \in N; t, q \in T, \tag{5.36}$$

$$I_c \in \{0,1\}, \text{ integer} \qquad \forall c \in C, \tag{5.37}$$

$$\gamma_{lcit}^- \in \{0,1\}, \text{ integer} \qquad \forall l \in L; c \in C \cup \Lambda; i \in H; t \in T, \tag{5.38}$$

$$\gamma_{lcit}^+ \in \{0,1\}, \text{ integer} \qquad \forall l \in L; c \in C \cup \Lambda; i \in H; t \in T. \tag{5.39}$$

The objective function (5.26) minimizes the total transportation cost required to cover all paths based on the distance traveled by containers and the cost for using each container. Additionally, the objective function includes the cost required handle loads at transshipment nodes. Constraints (5.27)–(5.30) and Constraints (5.31)–(5.32) are Constraints (4.8)–(4.11) and Constraints (4.13)–(4.14) from the integrated model in Section 4.2. Constraint (5.33) requires that the container or artificial container assigned to cover a path travels all the arcs on that path. Constraint (5.34) ensures that each path is covered by either a single container or an artificial container. Constraints (5.35)–(5.39) enforce the binary restrictions on the decision variables.

Finding a solution using the construction heuristic

This section describes the procedure that the construction heuristic follows to find a feasible solution that does not include any artificial containers. This iterative procedure solves the three phases in succession until no artificial containers are used. In order to make the use of arcs on load routes that require an artificial container in Phase III unattractive, the Phase I penalty cost is updated for each load route the artificial container covers. This makes those load routes less attractive in Phase I and facilitates finding different solution while still maintaining feasibility. This process is described by Algorithm 3 in Figure 5.4.

Algorithm 3 TSP-CP Construction Heuristic

procedure CONSTRUCTION PROCEDURE
    set $a = 1$
    while $a > 0$ **do**
        Solve Phase I
            Phase II $x^l_{itjq} \leftarrow$ Phase I X^l_{itjq}
        Solve Phase II
            Run Phase II: Path Creation
                Phase III $\delta^r_{itjq} \leftarrow$ Phase II: Path Creation δ^r_{itjq}
                Phase III $z^{lr}_{itjq} \leftarrow$ Phase II: Path Creation z^{lr}_{itjq}
        Solve Phase III
        if $\sum_{r \in R} \sum_{\alpha \in \Lambda} \lambda^\alpha_r > 0$ **then**
            $p^l_{itjq} = p^l_{itjq} + M(\sum_{r \in R} z^{lr}_{jqit} \lambda^r_\alpha)$.
        else
            $a = 0$
    return $X^l_{itjq}; Z^{lc}_{itjq}; V^c_{itjq}; \gamma^-_{lcit}; \gamma^+_{lcit}$

Figure 5.4: TSP-CP Construction Heuristic

5.2.2 Destroy Method Overview

This section introduces the four destroy methods that are used by both the fixed-optimization ALNS and the heuristic insertion ALNS. A destroy method for the TSP-CP removes r loads each iteration based on the characteristics of the destroy method that is chosen. The value of r is a randomly selected number of loads chosen each iteration between a maximum and a minimum limit. These limits are set in the beginning of the search and remain constant throughout. The four destroy methods used by the heuristics include: the random load removal destroy method, the random container removal destroy method, the Shaw removal destroy method, and the penalty removal destroy method.

Random Load Removal Destroy Method

The random load removal destroy method randomly selects loads for removal in an effort to explore areas of the solution space that may not otherwise be explored. With this destroy method, loads are randomly selected one at a time until r loads have been removed. In the application of the random load removal destroy method, each load is equally likely to be selected for removal.

Random Container Removal Destroy Method

Similar to the random load removal destroy method, the random container removal destroy method randomly selects a container that is carrying loads in the current solution and removes all the loads that are carried on that container. Each container is equally likely to be chosen for removal. Containers are selected and their loads are removed one at a time until r is exceeded. Once all loads are removed from a container, the container route is removed as well.

Shaw Removal Destroy Method

The Shaw removal destroy method aims to remove loads that are similar to each other with the idea that similar loads may be served by a single container. This removal procedure was introduced by Shaw [38] and is commonly applied to pickup and delivery applications of the ALNS. Each load has a score with all other loads and a lower score indicates the loads are similar. The removal score is based on three components: distance, time windows, and size. The distance component between two loads is based on the distance between their two origin nodes and the distance between their two destination nodes. The time window component is based on the difference between the two pick up and delivery time windows of the loads. Finally, the size component is based on the sum of the required container space occupied by the two loads. The removal score between load i and load j is calculated as follows:

$$\zeta^R(m_{o_i o_j} + m_{f_i f_j}) + \beta^R(|p_i^1 - p_j^1| + |p_i^2 - p_j^2| + |d_i^1 - d_j^1| + |d_i^2 - d_j^2|) + \kappa^R(r_i + r_j) \quad (5.40)$$

Parameters ζ^R, β^R, and κ^R represent the weight of the distance, time window, and size components respectively.

Each time the Shaw removal destroy method is chosen, a starting load is first randomly selected for removal. Additional loads are then chosen for removal based on the removal score associated with the randomly chosen starting load. Loads are chosen one at a time until r has been met.

Penalty Removal Destroy Method

The penalty removal destroy method aims to remove loads that are poorly placed in their current position. Poor placement for a load is determined by the score calculated with Equation 5.41 and is updated each time a new solution is accepted. The score consists of a component based on the total distance cost associated with delivering a load from origin to destination determined by its route and container assignment. Also includes is a component based on the average container space utilization of all the containers that carry the load.

$$\zeta^P(\sum_{c\in C}\sum_{i\in N}\sum_{t\in T}\sum_{j\in N}\sum_{q\in T}\omega_c m_{ij} Z^{lc}_{itjq}) + \kappa^P R_l \tag{5.41}$$

Parameter R_l is the average container space utilization across all containers that carry load l, ζ^P is the weight of the cost component, and κ^P is the weight of the size component.

The R_l penalty component is critical due to the inclusion of transfers. Traditional pickup and delivery applications often only look at the distance based cost of a particular load placement to determine if it is poorly placed or not. When transfers are included, the distance based cost alone can be misleading. With transfer opportunities, it can be beneficial for loads to take routes that may seem undesirable based solely on distance cost. These more expensive individual routes can lead to better overall solutions because loads are able to capitalize on consolidation opportunities. Due to this, the most heavily weighted term of the penalty formula is the size component. The higher the penalty, the more poorly placed the load is assumed. Each time the penalty removal procedure is chosen, r loads are removed such that in the end the sum of the penalty score of all removed loads is maximized.

5.3 The Fixed-Optimization ALNS for the TSP-CP

The fixed-optimization ALNS relies on a powerful, but computationally expensive, move operator to explore large areas of the solution space without the use of a meta-heuristic for escaping local minima. The fixed-optimization ALNS involves removing large portions of the current solutions for optimal reinsertion. Portions of the current solution are removed by the destroy methods presented in Section 5.2.2. In the case of the fixed-optimization ALNS, the r removed loads can represent up to as much as 70% of all loads in certain cases.

Following the application of the destroy method, the removed portions of the solution are simultaneously reinserted using one of two fixed-optimization repair methods. These repair methods operate by fixing or limiting a subset of variables from the integrated TSP-CP model in Section 4.2. The fixed variables represent the portion of the solution that was not removed by the destroy method. The values of the remaining non-fixed variables are determined by solving a mathematical model based on the integrated model with a subset of the original variables and any additional constraints or new parameters needed to fix the required portions of the solution. Figure 5.5 provides a flow chart of the fixed-optimization ALNS.

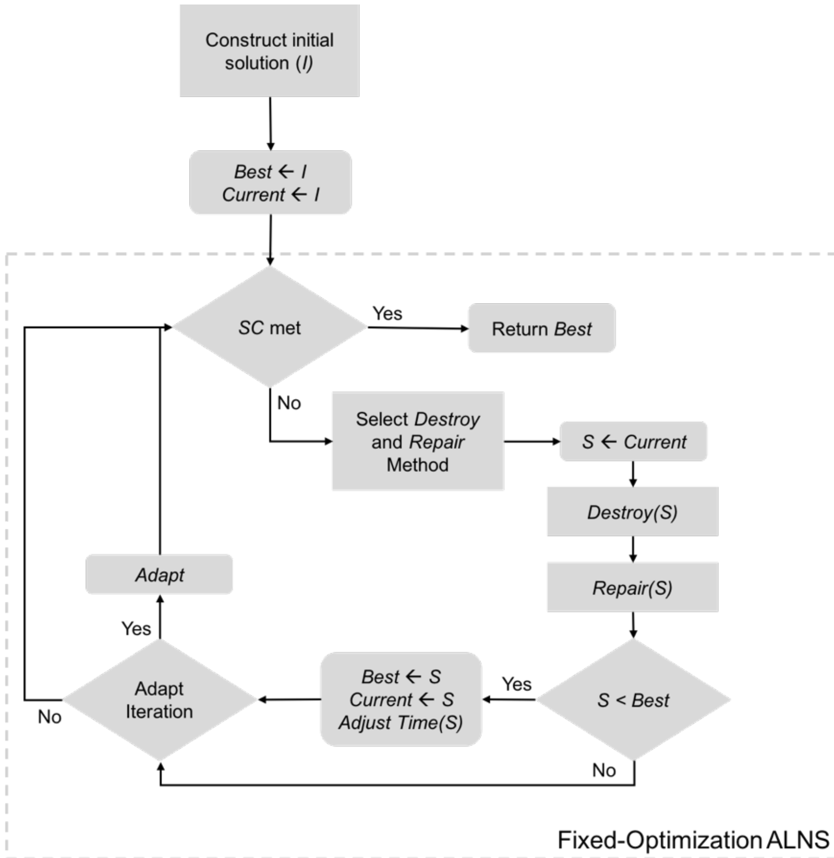

Figure 5.5: Flow chart of the Fixed-Optimization ALNS

5.3.1 Fixed-Optimization Repair Methods

The fixed-optimization ALNS uses the following two repair methods optimally create a new solution from the partial solution created by the application of the destroy method. These two repair methods are the fixed load route repair method and the fixed load and container route repair method. The following global parameters are required by both repair methods. The additional parameters required by a specific repair method are introduced in the section dedicated to that method.

Global Parameters

L	set of loads.	
N	set of nodes.	
H	set of transshipment nodes,	$H \subset N$.
NT	set of non-transshipment nodes,	$NT = N \setminus H$.
C	set of containers.	
T	set of time periods.	
T_{max}	the latest available time period.	
D	depot.	
τ	per load handling cost.	
η	per load holding cost.	
o_l	origin of load l,	$\forall l \in L$.
$[p_l^1, p_l^2]$	pickup time window for load l,	$\forall l \in L$.
f_l	destination of load l,	$\forall l \in L$.
$[d_l^1, d_l^2]$	delivery time window for load l,	$\forall l \in L$.
r_l	required space of load l,	$\forall l \in L$.
ω_c	per distance cost of using container c,	$\forall c \in C$.
a_c	initial node of container c,	$\forall c \in C$.
b_c	final node of container c,	$\forall c \in C$.
u_c	capacity of container c,	$\forall c \in C$.
m_{ij}	distance of arc (i, j),	$\forall i, j \in N$.
ϵ_{ij}	traversal time of arc (i, j) in time periods,	$\forall i, j \in N$.

Fixed Load Route Repair Method

In order to optimally insert the removed loads, the fixed load route repair method solves the integrated TSP-CP model with the additional constraint that the load routes for the non-removed loads are fixed. The values of the X variables for the non-removed loads from the current solution are fixed in the new solution. With these X variables fixed, the following model simultaneously determines the routes for removed loads, the routes for all containers, and the assignment of all loads to containers.

Additional Parameters

L^F set of non-removed loads.

L^R set of removed loads.

L set of all loads: $L = L^F \cup L^R$.

$$x^l_{itjq} \begin{cases} 1 & \text{if non-removed load } l \text{ travels from node } i \text{ in time period } t \text{ to node } j \\ & \text{in time period } q = t + \epsilon_{ij} \leq T_{max}, \\ 0 & \text{otherwise.} \end{cases}$$

Decision Variables

$$X^l_{itjq} \begin{cases} 1 & \text{if removed load } l \text{ travels from node } i \text{ in time period } t \text{ to node } j \\ & \text{in time period } q = t + \epsilon_{ij} \leq T_{max}, \\ 0 & \text{otherwise.} \end{cases}$$

$$Z^{cl}_{itjq} \begin{cases} 1 & \text{if load } l \text{ travels on container } c \text{ from node } i \text{ in time period } t \\ & \text{to node } j \neq i \text{ in time period } q = t + \epsilon_{ij} \leq T_{max}, \\ 0 & \text{otherwise.} \end{cases}$$

$$V^c_{itjq} \begin{cases} 1 & \text{if container } c \text{ travels from node } i \text{ in time period } t \text{ to node } j \\ & \text{in time period } q = t + \epsilon_{ij} \leq T_{max}, \\ 0 & \text{otherwise.} \end{cases}$$

$$I_c \begin{cases} 1 & \text{if container } c \text{ is used,} \\ 0 & \text{otherwise.} \end{cases}$$

$$\gamma^+_{lcit} \begin{cases} 1 & \text{if load } l \text{ is loaded onto container } c \text{ at transshipment node } i \\ & \text{in time period } t, \\ 0 & \text{otherwise.} \end{cases}$$

$$\gamma^-_{lcit} \begin{cases} 1 & \text{if load } l \text{ is unloaded from container } c \text{ at transshipment node } i \\ & \text{in time period } t, \\ 0 & \text{otherwise.} \end{cases}$$

Minimize

$$\sum_{c\in C}\sum_{i\in N}\sum_{t\in T}\sum_{j\in N}\sum_{q\in T}\omega_c m_{ij} V^c_{itjq} + \sum_{l\in L^R}\sum_{i\in H}\sum_{t\in T}\eta X^l_{itit+1} + \sum_{l\in L}\sum_{c\in C}\sum_{i\in H}\sum_{t\in T}\tau(\gamma^-_{lcit} + \gamma^+_{lcit}) \tag{5.42}$$

Subject to:

$$\sum_{t=p_l^1}^{p_l^2} X^l_{Dto_lt} = 1 \qquad \forall l \in L^R, \tag{5.43}$$

$$\sum_{t=d_l^1}^{d_l^2} X^l_{f_ltDt} = 1 \qquad \forall l \in L^R, \tag{5.44}$$

$$\sum_{j\in N}\sum_{q\in T} X^l_{itjq} - \sum_{j\in N}\sum_{q\in T} X^l_{jqit} = 0 \qquad \forall l \in L^R; i \in N; t \in T, \tag{5.45}$$

$$X^l_{itjq} = \sum_{c\in C} Z^{lc}_{itjq} \qquad \forall l \in L^R; i,j \in N : i \neq j, i \neq D, j \neq D; \; t,q \in T, \tag{5.46}$$

$$x^l_{itjq} = \sum_{c\in C} Z^{lc}_{itjq} \qquad \forall l \in L^F; i,j \in N : i \neq j, i \neq D, j \neq D; \; t,q \in T, \tag{5.47}$$

$$\sum_{l\in L} r_l Z^{lc}_{itjq} \leq u_c V^c_{itjq} \qquad \forall c \in C; i,j \in N : i \neq j; t,q \in T, \tag{5.48}$$

$$Z^{lc}_{itjq} \leq V^c_{itjq} \qquad \forall l \in L; c \in C; i,j \in N : i \neq j; t,q \in T, \tag{5.49}$$

$$I_c \geq V^c_{itjq} \qquad \forall c \in C; i,j \in N; t,q \in T, \tag{5.50}$$

$$\sum_{t\in T} V^c_{Dta_ct} = I_c \qquad \forall c \in C, \tag{5.51}$$

$$\sum_{t\in T} V^c_{b_ctDt} = I_c \qquad \forall c \in C, \tag{5.52}$$

$$\sum_{j\in N}\sum_{q\in T} V^c_{itjq} - \sum_{j\in N}\sum_{q\in T} V^c_{jqit} = 0 \qquad \forall c \in C; i \in N; t \in T, \tag{5.53}$$

$$\sum_{j\in N}\sum_{q\in T} Z^{lc}_{itjq} - \sum_{j\in N}\sum_{q\in T} Z^{lc}_{jqit} = 0 \qquad \forall l \in L; c \in C; i \in NT : i \neq o_l, i \neq f_l; \; t \in T, \tag{5.54}$$

$$\sum_{j\in N}\sum_{q\in T} Z^{lc}_{itjq} - \sum_{j\in N}\sum_{q\in T} Z^{lc}_{jqit} \leq \gamma^-_{lcit} \qquad \forall c \in C; l \in L; i \in H; t \in T, \tag{5.55}$$

$$\sum_{j\in N}\sum_{q\in T} Z^{lc}_{jqit} - \sum_{j\in N}\sum_{q\in T} Z^{lc}_{itjq} \leq \gamma^+_{lcit} \qquad \forall c \in C; l \in L; i \in H; t \in T, \tag{5.56}$$

$$X^l_{itjq} \in \{0,1\}, \text{ integer} \qquad \forall l \in L^R; i,j \in N; t,q \in T, \tag{5.57}$$

$$V^c_{itjq} \in \{0,1\}, \text{ integer} \qquad \forall c \in C; i,j \in N; t,q \in T, \tag{5.58}$$

$$Z^{lc}_{itjq} \in \{0,1\}, \text{ integer} \qquad \forall l \in L; c \in C; i,j \in N : i \neq j; t,q \in T. \tag{5.59}$$

$$I_c \in \{0,1\}, \text{ integer} \qquad \forall c \in C, \tag{5.60}$$

$$\gamma^-_{lcit} \in \{0,1\}, \text{ integer} \qquad \forall l \in L; c \in C; i \in H; t \in T, \tag{5.61}$$

$$\gamma^+_{lcit} \in \{0,1\}, \text{ integer} \qquad \forall l \in L; c \in C; i \in H; t \in T, \tag{5.62}$$

The objective function (5.42) minimizes the total transportation cost required to deliver the removed loads based on the distance traveled by containers and the cost for using each container. Additionally, the objective function includes the cost required to hold removed loads as well as the cost to handle all loads at transshipment nodes. Constraint (5.43) enforces that each removed load must enter its origin node from the depot once during its pickup time window. Constraint (5.44) enforces that each removed load must enter the depot from its destination node once during its delivery time window. Constraint (5.45) maintains the flow-conservation of removed loads through the network. Constraints (5.46) and (5.47) require that if a load travels an arc, a container must carry it. Constraints (5.48)-(5.62) remain the same as Constraints (4.6)-(4.20) from the integrated model presented in Section 4.2.

Fixed Load and Container Route Repair Method

The fixed load and container route repair method solves the integrated TSP-CP model with the additional constraints that the loads routes of the non-removed loads are fixed as well as the assignment of non-removed loads to containers. Furthermore, the routes for any containers that carry a non-removed load may not be changed. However, it is possible for a removed load to be picked up as part of a preexisting route for one of these containers. With these limitations included, the fixed load and container route repair model simultaneously determines the routes for removed loads, the routes for any additional containers that are needed, and the assignment of removed loads to containers.

Additional Parameters

L^F	set of non-removed loads.
L^R	set of removed loads.
L	set of all loads: $L = L^F \cup L^R$.
C^F	set of fixed containers.
C^R	set of non-fixed containers.
C	set of all containers: $C = C^F \cup C^R$

$$x^l_{itjq} \quad \begin{cases} 1 & \text{if non-removed load } l \text{ travels from node } i \text{ in time period } t \text{ to node } j \\ & \text{in time period } q = t + \epsilon_{ij} \leq T_{max}, \\ 0 & \text{otherwise.} \end{cases}$$

$$z^{cl}_{itjq} \quad \begin{cases} 1 & \text{if non-removed load } l \text{ travels on container } c \text{ from node } i \text{ in time period } t \\ & \text{to node } j \neq i \text{ in time period } q = t + \epsilon_{ij} \leq T_{max}, \\ 0 & \text{otherwise.} \end{cases}$$

$$v^c_{itjq} \quad \begin{cases} 1 & \text{if fixed container } c \text{ travels from node } i \text{ in time period } t \text{ to node } j \\ & \text{in time period } q = t + \epsilon_{ij} \leq T_{max}, \\ 0 & \text{otherwise.} \end{cases}$$

Decision Variables

$$X^l_{itjq} \quad \begin{cases} 1 & \text{if removed load } l \text{ travels from node } i \text{ in time period } t \text{ to node } j \\ & \text{in time period } q = t + \epsilon_{ij} \leq T_{max}, \\ 0 & \text{otherwise.} \end{cases}$$

$$Z^{cl}_{itjq} \quad \begin{cases} 1 & \text{if removed load } l \text{ travels on container } c \text{ from node } i \text{ in time period } t \\ & \text{to node } j \neq i \text{ in time period } q = t + \epsilon_{ij} \leq T_{max}, \\ 0 & \text{otherwise.} \end{cases}$$

$$V^c_{itjq} \quad \begin{cases} 1 & \text{if container } c \text{ travels from node } i \text{ in time period } t \text{ to node } j \\ & \text{in time period } q = t + \epsilon_{ij} \leq T_{max}, \\ 0 & \text{otherwise.} \end{cases}$$

$$I_c \quad \begin{cases} 1 & \text{if container } c \text{ is used,} \\ 0 & \text{otherwise.} \end{cases}$$

$$\gamma^+_{lcit} \quad \begin{cases} 1 & \text{if removed load } l \text{ is loaded onto container } c \text{ at transshipment node } i \\ & \text{in time period } t, \\ 0 & \text{otherwise.} \end{cases}$$

$$\gamma^-_{lcit} \quad \begin{cases} 1 & \text{if removed load } l \text{ is unloaded from container } c \text{ at transshipment node } i \\ & \text{in time period } t, \\ 0 & \text{otherwise.} \end{cases}$$

Minimize

$$\sum_{c \in C} \sum_{i \in N} \sum_{t \in T} \sum_{j \in N} \sum_{q \in T} \omega_c m_{ij} V^c_{itjq} + \sum_{l \in L^R} \sum_{i \in H} \sum_{t \in T} \eta X^l_{itit+1} + \sum_{l \in L^R} \sum_{c \in C} \sum_{i \in H} \sum_{t \in T} \tau(\gamma^-_{lcit} + \gamma^+_{lcit}) \tag{5.63}$$

Subject to:

$$\sum_{t=p_l^1}^{p_l^2} X^l_{Dto_lt} = 1 \qquad \forall l \in L^R, \tag{5.64}$$

$$\sum_{t=d_l^1}^{d_l^2} X^l_{f_ltDt} = 1 \qquad \forall l \in L^R, \tag{5.65}$$

$$\sum_{j\in N}\sum_{q\in T} X^l_{itjq} - \sum_{j\in N}\sum_{q\in T} X^l_{jqit} = 0 \qquad \forall l \in L^R; i \in N; t \in T, \tag{5.66}$$

$$X^l_{itjq} = \sum_{c\in C} Z^{lc}_{itjq} \qquad \forall l \in L^R; i,j \in N : i \neq j, i \neq D, j \neq D; \; t,q \in T, \tag{5.67}$$

$$\sum_{l\in L^R} r_l Z^{lc}_{itjq} + \sum_{l\in L^F} r_l z^{lc}_{itjq} \leq u_c V^c_{itjq} \qquad \forall c \in C; i,j \in N : i \neq j; t,q \in T, \tag{5.68}$$

$$Z^{lc}_{itjq} \leq V^c_{itjq} \qquad \forall l \in L^R; c \in C; i,j \in N : i \neq j; t,q \in T, \tag{5.69}$$

$$z^{lc}_{itjq} \leq V^c_{itjq} \qquad \forall l \in L^F; c \in C; i,j \in N : i \neq j; t,q \in T, \tag{5.70}$$

$$v^c_{itjq} = V^c_{itjq} \qquad \forall c \in C^F; i,j \in N : i \neq j; t,q \in T, \tag{5.71}$$

$$I_c \geq V^c_{itjq} \qquad \forall c \in C^R; i,j \in N; t,q \in T, \tag{5.72}$$

$$\sum_{t\in T} V^c_{Dta_ct} = I_c \qquad \forall c \in C^R, \tag{5.73}$$

$$\sum_{t\in T} V^c_{b_ctDt} = I_c \qquad \forall c \in C^R, \tag{5.74}$$

$$\sum_{j\in N}\sum_{q\in T} V^c_{itjq} - \sum_{j\in N}\sum_{q\in T} V^c_{jqit} = 0 \qquad \forall c \in C^R; i \in N; t \in T, \tag{5.75}$$

$$\sum_{j\in N}\sum_{q\in T} Z^{lc}_{itjq} - \sum_{j\in N}\sum_{q\in T} Z^{lc}_{jqit} = 0 \qquad \forall l \in L^R; c \in C; i \in NT : i \neq o_l, i \neq f_l; \; t \in T, \tag{5.76}$$

$$\sum_{j\in N}\sum_{q\in T} Z^{lc}_{itjq} - \sum_{j\in N}\sum_{q\in T} Z^{lc}_{jqit} \leq \gamma^-_{lcit} \qquad \forall c \in C; l \in L^R; i \in H; t \in T, \tag{5.77}$$

$$\sum_{j\in N}\sum_{q\in T} Z^{lc}_{jqit} - \sum_{j\in N}\sum_{q\in T} Z^{lc}_{itjq} \leq \gamma^+_{lcit} \qquad \forall c \in C; l \in L^R; i \in H; t \in T, \tag{5.78}$$

$$X^l_{itjq} \in \{0,1\}, \text{ integer} \qquad \forall l \in L^R; i,j \in N; t,q \in T, \tag{5.79}$$

$$V^c_{itjq} \in \{0,1\}, \text{ integer} \qquad \forall c \in C; i,j \in N; t,q \in T, \tag{5.80}$$

$$Z^{lc}_{itjq} \in \{0,1\}, \text{ integer} \qquad \forall l \in L^R; c \in C; i,j \in N : i \neq j; t,q \in T. \tag{5.81}$$

$$I_c \in \{0,1\}, \text{ integer} \qquad \forall c \in C, \tag{5.82}$$

$$\gamma^-_{lcit} \in \{0,1\}, \text{ integer} \qquad \forall l \in L^R; c \in C; i \in H; t \in T, \tag{5.83}$$

$$\gamma^+_{lcit} \in \{0,1\}, \text{ integer} \qquad \forall l \in L^R; c \in C; i \in H; t \in T, \tag{5.84}$$

The objective function (5.63) minimizes the total transportation cost required to deliver removed loads based on the distance traveled by containers and the cost for using each container. Additionally, the objective function includes the cost required handle and hold removed loads at transshipment nodes. Constraint (5.64) enforces that each removed load must enter its origin node from the depot once during its pickup time window. Constraint (5.65) enforces that each removed load must enter the depot from its destination node once during its delivery time window. Constraint (5.66) maintains the flow-conservation of removed loads through the network. Constraint (5.67) requires that if a removed load travels an arc, a container must carry it. Constraint (5.68) ensures that a container's capacity is never exceeded. Constraints (5.69) and (5.70) are valid inequalities that ensures a load cannot travel an arc on a container unless a container also travels that arc. Constraint (5.71) sets the routes for all fixed containers. Constraint (5.72) sets an indicator variable to one if a non-fixed container is used across an arc. Constraints (5.73) and (5.74) ensure that if a non-fixed container is in use, it must leave from its initial node and return to its final node. Constraint (5.75) maintains the flow-conservation of non-fixed containers through the network. Constraint (5.76) maintains the flow-conservation of removed loads on containers through non-transshipment nodes by ensuring that a load enters and leaves on the same container. Constraint (5.77) captures the number of times that a removed load is taken off a container at a transshipment node while, Constraint (5.78) captures the number of times a removed load is loaded onto a container at a transshipment node. Constraints (5.79)–(5.84) enforce the binary restrictions on the decision variables.

5.3.2 Fixed Load Route Time Adjustment Method

One major limitation associated with fixing part of the solution before optimization is that transfer opportunities can missed due to the fixed arrival and departure times for non-removed loads. To overcome this limitation, it can be beneficial to adjust the arrival and departure times of loads at nodes in order to find transfer opportunities that may have been missed. The fixed load route time adjustment method is an addition to the fixed-optimization ALNS that does just that. In order to accomplish this, the parameter n^l_{ij} is introduced for use in the following mathematical model. For each load l, n^l_{ij} specifies the set of locations i, j and the order in which those locations are visited. Since the fixed load route time adjustment method does not require loads to be removed for reinsertion, it is run each time a new solution is accepted to see if time adjustments can be beneficial. The following model provides optimal time adjustments for load routes, the routes for containers, and the assignment of loads to containers.

Additional Parameters

n^l_{ij} number of times load l travels arc (i,j) $\forall l \in L; i,j \in N$.

Decision Variables

$$X^l_{itjq} \quad \begin{cases} 1 & \text{if load } l \text{ travels from node } i \text{ in time period } t \text{ to node } j \\ & \text{in time period } q = t + \epsilon_{ij} \leq T_{max}, \\ 0 & \text{otherwise.} \end{cases}$$

$$Z^{cl}_{itjq} \quad \begin{cases} 1 & \text{if load } l \text{ travels on container } c \text{ from node } i \text{ in time period } t \\ & \text{to node } j \neq i \text{ in time period } q = t + \epsilon_{ij} \leq T_{max}, \\ 0 & \text{otherwise.} \end{cases}$$

$$V^c_{itjq} \quad \begin{cases} 1 & \text{if container } c \text{ travels from node } i \text{ in time period } t \text{ to node } j \\ & \text{in time period } q = t + \epsilon_{ij} \leq T_{max}, \\ 0 & \text{otherwise.} \end{cases}$$

$$I_c \quad \begin{cases} 1 & \text{if container } c \text{ is used,} \\ 0 & \text{otherwise.} \end{cases}$$

$$\gamma^+_{lcit} \quad \begin{cases} 1 & \text{if load } l \text{ is loaded onto container } c \text{ at transshipment node } i \\ & \text{in time period } t, \\ 0 & \text{otherwise.} \end{cases}$$

$$\gamma^-_{lcit} \quad \begin{cases} 1 & \text{if load } l \text{ is unloaded from container } c \text{ at transshipment node } i \\ & \text{in time period } t, \\ 0 & \text{otherwise.} \end{cases}$$

Minimize

$$\sum_{c\in C}\sum_{i\in N}\sum_{t\in T}\sum_{j\in N}\sum_{q\in T}\omega_c m_{ij}V^c_{itjq}+\sum_{l\in L}\sum_{i\in H}\sum_{t\in T}\eta X^l_{itit+1}$$
$$+\sum_{l\in L}\sum_{c\in C}\sum_{i\in H}\sum_{t\in T}\tau(\gamma^-_{lcit}+\gamma^+_{lcit}) \tag{5.85}$$

Subject to:

$$\sum_{t\in T}X^l_{itjt+\epsilon_{ij}}=n^l_{ij} \qquad \forall i,j\in N, \tag{5.86}$$

$$\sum_{t=p^1_l}^{p^2_l}X^l_{Dto_lt}=1 \qquad \forall l\in L, \tag{5.87}$$

$$\sum_{t=d^1_l}^{d^2_l}X^l_{f_ltDt}=1 \qquad \forall l\in L, \tag{5.88}$$

$$\sum_{j\in N}\sum_{q\in T}X^l_{itjq}-\sum_{j\in N}\sum_{q\in T}X^l_{jqit}=0 \qquad \forall l\in L;i\in N;t\in T, \tag{5.89}$$

$$X^l_{itjq}=\sum_{c\in C}Z^{lc}_{itjq} \qquad \forall l\in L;i,j\in N:i\neq j,i\neq D,j\neq D;\ t,q\in T, \tag{5.90}$$

$$\sum_{l\in L}r_lZ^{lc}_{itjq}\leq u_cV^c_{itjq} \qquad \forall c\in C;i,j\in N:i\neq j;t,q\in T, \tag{5.91}$$

$$Z^{lc}_{itjq}\leq V^c_{itjq} \qquad \forall l\in L;c\in C;i,j\in N:i\neq j;t,q\in T, \tag{5.92}$$

$$I_c\geq V^c_{itjq} \qquad \forall c\in C;i,j\in N;t,q\in T, \tag{5.93}$$

$$\sum_{t\in T}V^c_{Dta_ct}=I_c \qquad \forall c\in C, \tag{5.94}$$

$$\sum_{t\in T}V^c_{b_ctDt}=I_c \qquad \forall c\in C, \tag{5.95}$$

$$\sum_{j\in N}\sum_{q\in T}V^c_{itjq}-\sum_{j\in N}\sum_{q\in T}V^c_{jqit}=0 \qquad \forall c\in C;i\in N;t\in T, \tag{5.96}$$

$$\sum_{j\in N}\sum_{q\in T}Z^{lc}_{itjq}-\sum_{j\in N}\sum_{q\in T}Z^{lc}_{jqit}=0 \qquad \forall l\in L;c\in C;i\in NT:i\neq o_l,i\neq f_l;\ t\in T, \tag{5.97}$$

$$\sum_{j\in N}\sum_{q\in T}Z^{lc}_{itjq}-\sum_{j\in N}\sum_{q\in T}Z^{lc}_{jqit}\leq\gamma^-_{lcit} \qquad \forall c\in C;l\in L;i\in H;t\in T, \tag{5.98}$$

$$\sum_{j\in N}\sum_{q\in T}Z^{lc}_{jqit}-\sum_{j\in N}\sum_{q\in T}Z^{lc}_{itjq}\leq\gamma^+_{lcit} \qquad \forall c\in C;l\in L;i\in H;t\in T, \tag{5.99}$$

$$X^l_{itjq} \in \{0,1\}, \text{ integer} \quad \forall l \in L; i,j \in N; t,q \in T, \tag{5.100}$$
$$V^c_{itjq} \in \{0,1\}, \text{ integer} \quad \forall c \in C; i,j \in N; t,q \in T, \tag{5.101}$$
$$Z^{lc}_{itjq} \in \{0,1\}, \text{ integer} \quad \forall l \in L; c \in C; i,j \in N : i \neq j; t,q \in T. \tag{5.102}$$
$$I_c \in \{0,1\}, \text{ integer} \quad \forall c \in C, \tag{5.103}$$
$$\gamma^-_{lcit} \in \{0,1\}, \text{ integer} \quad \forall l \in L; c \in C; i \in H; t \in T, \tag{5.104}$$
$$\gamma^+_{lcit} \in \{0,1\}, \text{ integer} \quad \forall l \in L; c \in C; i \in H; t \in T. \tag{5.105}$$

The objective function (5.85) minimizes the total transportation cost required to deliver all loads based on the distance traveled by containers and the cost for using each container. Additionally, the objective function includes the cost required handle and hold loads at transshipment nodes. Constraint (5.86) ensures that arc (i, j) is traveled the required number of times. Constraints (5.87)-(5.105) remain the same as Constraints (4.2)-(4.20) from the integrated model in Section 4.2.

5.3.3 Acceptance, Adaptive Score Update, and Stopping Criteria

Because the fixed-optimization ALNS does not rely on any meta-heuristic, a new solution is only accepted if it is better than the current solution. This determination is based on the objective function value of each repair method optimization model. This is the same as the objective function from the integrated model in Section 4.2 and considers the cost of the distance traveled by containers as well as the costs to handle and hold loads.

As the search progresses, the score that is used to determine the probability of a method being chosen in a given iteration is updated to allow the search to learn from the previous iterations. Scores are updated each iteration if a new solution is accepted as a result of applying the destroy method and the repair method. If a new globally best solution is found, the score of each applied method is updated by σ^O_1. The score of each method is set to zero at the start of each segment and is used to recalculated the probability of each method at the end of each segment. The fixed-optimization ALNS continues finding neighboring solutions in this manner for a set number of iterations or until a set time limit has been reached.

5.4 The Heuristic Insertion ALNS for the TSP-CP

The heuristic insertion ALNS takes an opposing repair ideology to the fixed-optimization ALNS. With the heuristic insertion ALNS small changes are made to the current solution using a sequential insertion procedure for repair. Because fewer loads are removed and a less powerful move operator is employed, the heuristic insertion ALNS relies heavily on a simulated annealing meta-heuristic to search the solution space. This section introduces the sequential insertion procedures applied by the heuristic insertion ALNS as well as the repair methods and any additional features added for the heuristic insertion ALNS framework. A flow chart representing the heuristic insertion ALNS is shown in Figure 5.6.

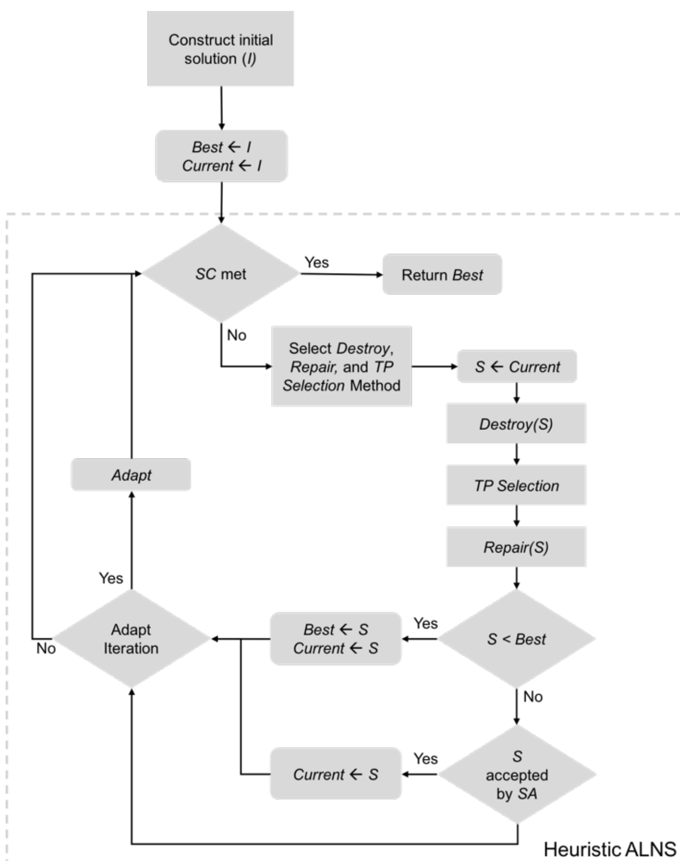

Figure 5.6: Flow chart of the heuristic insertion ALNS

5.4.1 Heuristic Insertion Procedures

Repairing a solution in the heuristic insertion ALNS is done through the use of a sequential insertion procedure. A sequential insertion procedure builds a feasible solution by selecting removed loads one at a time and inserting them into the partial solution until all loads have been inserted. An insertion involves finding a single container to cover the movements of a load between a pick up point and a drop off point. A load can either be picked up from its origin node or from a transfer point where it has been previously dropped off by a different container. Likewise, a load can either be dropped off at its destination or it can be dropped at a transfer point for transfer to another container. Feasible insertion points are found by searching all container routes to see if it is possible for a container to handle the pick up and drop off of the load. The best insertion point for a load is selected after evaluating all feasible insertion positions.

For a given pick up point, ρ, and drop off point, ς, insertion is completed in the following manner. Define a container route as the sequence of N stops that a container visits between its initial node a to its final node b. To fully describe the container route three characteristics are needed; the nodes that are visited, the arrival time at these nodes, and the amount of container space occupied when moving between nodes. For example, the route for container c is given as $I_c = (i_0^c, .., i_n^c, .., i_N^c)$ where i_n^c is the node associated with stop n. Additionally, let $AT_c = (at_0^c, .., at_n^c, .., at_N^c)$ where at_n^c is the arrival time of container c at stop n and $U_c = (u_0^c, .., u_n^c, .., u_N^c)$ where u_n^c is the utilized space of container c following all pick ups and drop offs at stop n. This route will always begin with $i_0^c = a_c; at_0^c = 0; u_0^c = 0$ and end with $i_N^c = b_c; at_N^c \leq T_{max}; u_N^c = 0$. All other values will depend on the sequence of nodes that the container travels between a_c and b_c and the loads that are picked up and dropped off along the way.

Evaluating the insertion of load l with pick up point ρ and drop off point ς into the route for container c is done by first evaluating the insertion of the pick up point ρ as stop n_ρ where $n_\rho - 1 \geq 0$ and $n_\rho + 1 \leq N$. Because inserting ρ can require traveling to a completely new location, a possible two extra travel legs can be added to the route: the travel from $i^c_{n_\rho - 1}$ to ρ and the travel back from ρ to $i^c_{n_\rho + 1}$. As such, the arrival time at stop n_ρ is given as $at^c_{n_\rho} = at^c_{n_\rho - 1} + \epsilon_{i^c_{n_\rho - 1}, \rho}$. For this to be a feasible insertion, $at^c_{n_\rho}$ must occur before the end of the planning horizon, T_{max}. Additionally, if ρ is the origin of load l, then $at^c_{n_\rho}$ must occur within its pick up time window or if ρ represents picking up load l at a transfer point from container c_1, then $at^c_{n_\rho}$ must be greater than or equal to the time that l was dropped off at ρ by container c_1.

If the insertion remains feasible, the arrival time at each stop following n_ρ will be updated as follows: $at^c_n = at^c_n + \epsilon_{i^c_{n_\rho - 1}, \rho} + \epsilon_{\rho, i^c_{n_\rho + 1}}$ $\forall n > n^c_\rho$. In order for the insertion to remain feasible, the new arrival time at each stop must be less than T_{max}. Furthermore, if one of these stop picks up a load from its origin or delivers a load to its destination, the newly updated arrival time must still not violate any pick up or delivery time windows. Finally, if any stop with an updated arrival time requires the drop off of a load to be transferred to a container c_2, the new arrival time must still be before the departure of c_2 from the transfer node. However, if it is possible to delay the departure of c_2 to accommodate picking up loads from container c with the new arrival time, then c_2 is delayed the required amount and the insertion remains feasible. If all the previous checks are met, then a feasible insertion point for ρ as stop n_p on container c has been found and all values of I_c and AT_c must be temporarily updated. Additionally, information associated with any other container routes that must be delayed to accommodate transfers from container c must be temporarily updated as well.

Using the newly updated routes and arrival times, a similar process is used to evaluate the insertion of the drop off node ς into the route of container c as stop n_ς where $n_\varsigma - 1 \geq n_\rho$ and $n_\varsigma + 1 \leq N$. Again, adding this stop can require traveling to a new location thus adding a possible two additional legs. As with the insertion of the pick up point, the insertion of the drop off point also requires the same arrival time updates and feasibility checks: $at^c_{n_\varsigma}$ must be less than or equal to T_{max}; if ς acts as the destination of load l, the arrival must satisfy the delivery time window; and if ς acts as a transfer point where load l is dropped off for container c_3 then container c must arrive before the departure plus any possible delay of container c_3. Moreover, accommodating the additional time added to the stops of the route following n_ς^c must not violate any of the pick up or delivery time windows of loads or any conditions to make a successful transfer as descried above. Finally, container c must be able to handle the extra space occupied by the addition of load l. This means that $u_n^c + r_l \leq u_c$ for all stops between n_ρ^c and n_ς^c.

If all this is possible, then a feasible insertion point for load l has been found as a part of the route for container c with pick up point ρ as stop n_ρ and drop off point ς as stop n_ς. The increase in the objective function value of this feasible insertion is then recorded. This procedure is repeated for each container $c \in C$ and all the possible values of n_ρ and n_ς on the route of container c. The best insertion point for load l given pick up point ρ and drop off point ς based on some predefined evaluation criteria is then selected. At this time, all temporary updates to container routes associated with the selected insertion are made final and the solution is updated as required. The next load from the removed load list is then selected and inserted in the same manner until all loads have been inserted.

Greedy without Transfer Insertion Procedure

In the greedy without transfer insertion procedure, any intermediate consolidation is not considered and only a single container may carry a load from origin to destination. This insertion procedure operates by evaluating the insertion of each removed load l with $\rho = o_l$ and $\varsigma = f_l$ where o_l is the origin of the removed load l and f_l is the destination of removed load l. After all possible insertion points for $\rho = o_l$ and $\varsigma = f_l$ have been evaluated, the delivery of removed load l without transfer is added the the container route that increases the overall objective function value the least. This process is repeated until all removed loads have been reinserted without transfer or an infeasible insertion is found.

Greedy with Transfer Insertion Procedure

The greedy with transfer insertion procedure reinserts removed loads requiring that at least one transfer takes place in route from origin to destination. With this requirement, at least two containers will carry the load. However, the heuristic insertion procedure only allows for a single container to carry a load from pick up point to drop off point. To overcome this restriction, inserting a load with a transfer requires splitting a load into multiple insertions each with its own pick up and drop off point. In the case where a single transfer is utilized, two insertions must be completed. Insertion 1 carries the load from its origin to the transfer point with $\rho_1 = o_l$ and $\varsigma_1 = h$ and insertion 2 carries the load from the transfer point to its destination with $\rho_2 = h$ and $\varsigma_2 = f_l$ where h is the chosen transfer point. The insertions are completed independently using the heuristic insertion procedure with the added restrictions that the arrival time at ς_1 is less than or equal to the departure time at ρ_2. In addition, the container that covers insertion 1 must be different than the container that covers insertion 2. Since sequential insertion is used, it may be beneficial to evaluate insertion 1 followed by insertion 2 as well as insertion 2 followed by insertion 1.

As the number of transfers in this research is not limited to one, multiple transfer opportunities must be evaluated as well. When evaluating the insertion of load l with two transfers at transfer node h_1 followed by transfer node h_2 where $h_1, h_2 \in H : h_1 \neq h_2$, load l is split into three insertions: insertion 1 with $\rho_1 = o_1; \varsigma_1 = h_1$, insertion 2 with $\rho_2 = h_1; \varsigma_2 = h_2$, and insertion 3 with $\rho_3 = h_2; \varsigma_3 = f_l$. Similarly, it follows that when evaluating the insertion of load l with x transfers using transfer nodes $h_1, h_2, ..., h_x \in H : h_1 \neq h_2, ..., h_{x-1} \neq h_x$, load l is split into $x+1$ insertions: insertion 1 with $\rho_1 = o_1; \varsigma_1 = h_1$, insertion 2 with $\rho_2 = h_1; \varsigma_2 = h_2$, ...,insertion x with $\rho_x = h_{x-1}; \varsigma_1 = h_x$, and finally insertion $x+1$ with $\rho_{x+1} = h_x; \varsigma_{x+1} = f_l$.

The transfer point that is chosen is never set in advance. Instead, during the first insertion each transfer point $h \in H$ must be evaluated to find the transfer point that increases the objective function value the least. Once this point is found, it becomes the transfer point used for the subsequent insertion. When multiple transfers are considered, the drop off transfer point is never set in advance. Instead, it is determined through evaluating all possible options and then choosing the one that increases the objective function the least based on the pick up transfer point set from the previous insertion. Furthermore, the number of transfers is not set in advance either. Instead, up to a maximum allowable number of transfers for each load is evaluated and the number of transfers that increases the objective function value the least overall is chosen. Inserting loads with transfer in this manner does significantly increase the solution time as the heuristic insertion procedure has to be applied multiple times to evaluate multiple insertions and the multiple possible transfer points; however it provides a level of flexibility not available in other ALNS applications that include transfers.

5.4.2 Heuristic Repair Methods

The heuristic insertion ALNS repair methods apply the greedy with and without transfer insertion procedures to create feasible solutions following the application of a destroy method from Section 5.2.2. Depending on the heuristic or combination of heuristics applied by the repair method, different types of load routes can be forced into the solution to try and obtain improvements. The heuristic insertion ALNS uses the following three repair methods.

With and Without Transfer Repair Methods

The with transfer repair repair method and the without transfer repair method inserts a load with a decided route type. A decided route type requires either forcing a load to be inserted with at least one transfers or to be inserted without allowing any transfers. The without transfer repair method inserts a load without allowing any transfers. As such, this repair method relies only on the without transfer insertion procedure.

In contrast, the with transfer repair method forces a load to be inserted with at least one transfer. This repair method relies only on the greedy insertion with transfer insertion procedure. As previously introduced, the order of the multiple insertions defined by the greedy with transfer insertion procedure can have an impact on solution quality. In order to find the best possible insertion point for a load with transfers, two different orders of insertions are evaluated. The insertion procedure is first evaluated with the insertion representing the move from the origin to transfer point done first followed by all other insertions concluding with the insertion representing the move from the final transfer point to the destination. Next, the procedure is applied again by first inserting the move from the final transfer point to the destination followed by all remaining insertions concluding with inserting the move from the origin to the first transfers point. The overall best insertion after both of these insertions have been completed is selected.

Full Evaluation Repair Method

The full evaluation repair method finds the best possible insertion point by considering the insertion of a load with transfers and without transfers. This method applies both the with and without transfer repair methods in full and then chooses the insertion that increases the overall objective function value the least. The full evaluation repair method is the most time consuming repair method as it includes both of the other two repaid methods. However, it provides the most exhaustive search of the possible options for insertion.

5.4.3 Transfer Point Section Methods

Due to the exhaustive nature of the heuristic insertion procedures, evaluating all possible transfer points may be time prohibitive. In order to reduce solution times, only a subset of transfer nodes are considered during each insertion. During the insertion for removed load l, $|H'| : H' \subseteq H$ transfer points are chosen for evaluation using one of the following three transfer point selection methods. One transfer point selection method is chosen each iteration based on a probability that adapts based on performance in the same manner as the destroy and repair methods. Each transfer point selection method aims to provide a different subset of transfer points for consideration that will allow the search to evaluate all transfer points, but focuses on providing the most likely to be used.

In-Use Transfer Point Selection

Because containers may already be traveling to certain transfer nodes, it is often the case that an opportunity for improvement may be found by utilizing a transfer node where consolidation is already happening. The in-use transfer point section method strives to leverage these opportunities by considering the subset of available transfer nodes as the transfer nodes that are already in use. If more than $|H'|$ transfer points are in use, transfers points are selected randomly from the set of in-use transfer nodes to determine the $|H'|$ transfer points that are considered. If no transfer points are in use, $|H'|$ transfer points are randomly selected from the full set of transfer nodes.

Minimum Distance Transfer Point Selection

Consolidation may be more likely to occur if traveling to a transfer node only interrupts an existing container route by a small distance. The minimum distance transfer point selection method selects the $|H'|$ transfer nodes that minimize the distance between the origin of removed load l and the transfer points in H' plus the distance between the destination node of removed load l and the transfer points in H'.

Random Transfer Point Selection

Ensuring diversity in the $|H'|$ transfer nodes evaluated by the search is vital in exploring the solution space. Although it might not immediately seem advantageous to use a particular transfer node based on the previous methods, there is a possibility it can provide improvements in the current or future iterations. The random transfer point selection method provides this much needed diversity to the search by randomly selecting the $|H'|$ transfer nodes to be considered.

5.4.4 Sorting Loads Prior to Insertion

When the sequential insertion procedure is used for repair, the order in which the removed loads are reinserted into the partial solution may have a noticeable effect on solution quality. Sorting the removed load list prior to insertion in certain instances can help find improvements that may otherwise be missed. In the heuristic insertion ALNS, each iteration there is a $s\%$ chance that loads will be sorted based on the values from Equation 5.106. Each removed load l is assigned the following sorting score.

$$\zeta^S(m_{o_l,f_l}) + \beta^S(p_l^2 - p_l^1 + d_l^2 - d_l^1) + \kappa^S(r_l) \tag{5.106}$$

Parameters ζ^S, β^S and κ^S are the weights of the distance, time window, and space terms respectively. The higher the sorting score, the more difficult the insertion of load l is perceived to be. Accordingly, the removed load list is sorted from largest to smallest score and is inserted in this order. If the removed load list is not sorted prior to insertion, then the list is inserted in the order that it stands coming out of the destroy method.

5.4.5 Return to the Best Solution

An addtional considertion added to the ALNS by Petersen and Ropke [33] is the ability to return to the global best solution found thus far. Because this showed promise in their application to the pickup and delivery problem with cross docking, the extension is also utilized here. However returning to the global best solution is done with a much smaller probability each iteration. In this case, at the beginning of an iteration before any destroy method is applied, there is a small probability, $g\%$, that the search will return to the global best solution. Returning to this solution is an effort to enable an exhaustive search around the most promising areas of the solution space.

5.4.6 Acceptance, Adaptive Score Update, and Stopping Criteria

Unlike the fixed-optimization ALNS where only a solution with an improved objective function value is accepted, the heuristic insertion ALNS accepts a new solution in two cases. The first case is the same as before; a solution with a better objective function value than the current solution objective function value is accepted with a probability of 1. However, since simulated annealing is controlling the search it is also possible to accept a solution with a worse objective function value given that it meets the simulated annealing acceptance probability. As such, the probability of accepting a new solution x' given the search is currently at solution x is given as:

$$P(x') = \begin{cases} e^{-\frac{f(x')-f(x)}{T}} & \text{if } f(x') > f(x) \\ 1 & \text{if } f(x') < f(x) \end{cases} \tag{5.107}$$

Again $f(x')$ is the objective function value of solution x', $f(x)$ is the objective function value of solution x, and T is the temperature. Due to the sequential nature of the repair methods, sometimes infeasbile insertions are found. In the case that a removed load insertion cannot be made, the iteration is stopped and the search returns to the previous feasible solution.

The score of each method in the heuristic insertion ALNS is updated in two cases based on if a new solution is accepted as a result of applying the destroy method, the repair method, and the transfer point selection method. If a new global best solution is found the score of each applied method is updated by σ_1^H and if a solution better than the current solution, but not the global best solution is found then the score of each applied method is updated by σ_2^H. The heuristic insertion ALNS again is complete after a set number of iterations or after a given amount of time has elapsed.

Chapter 6

Solution Approach Performance

This chapter provides computational results and performance comparisons for the integrated TSP-CP mathematical model introduced in Section 4.2, the fixed-optimization ALNS introduced in Section 5.3, and the heuristic insertion ALNS introduced in Section 5.4. The results and comparisons are based on the application of the three solution approaches to a set of test instances created to capture the complexities of the TSP-CP network. The results were generated using the same Windows 8.1 computer using an Intel Xeon E5-2620 2.4 GHz processor and 32 GB of RAM. The models and heuristics were implemented using the Java programming language through the Eclipse IDE, and all mathematical models were defined with ILOG concert technology and solved using CPLEX version 12.6.

Computational results provide insights into the ability of each approach to provide a solution to the routing and consolidation needs for a TSP or a group of collaborating TSPs. Results include the total distance required to deliver all loads measured in miles, the number of those miles that are traveled by containers completely empty, and the utilization of the available container space. Container space utilization is captured through the use of weighted full miles. Weighted full miles consider both the container space utilization across an arc as well as the distance of the arc. Weighted full miles are calculated with Equation 6.1 where m_{ij} is the distance in miles between node i and node j, r_l is the required container space occupied by load l, and Z^{lc}_{itjq} is the indicator variable that is set to one if load l travels from node i in time period t to node j in time periods q on container c and zero otherwise.

$$\sum_{l\in L}\sum_{c\in C}\sum_{i\in N}\sum_{t\in T}\sum_{j\in N}\sum_{q\in T} m_{ij}(r_l Z^{lc}_{itjq}) \tag{6.1}$$

Weighted full miles are used because with collaboration it may be beneficial to make short distance moves empty or nearly empty to leverage consolidation opportunities for the moves with a longer distance.

Solutions times are provided for the integrated model on the various problem sizes as well as the ability to find collaborative benefit when comparing the combination of two single test instances to a combined collaborative test instance. Additionally, the objective function values obtained by both heuristic solution approaches are compared to the optimal objective function value for each of the test instances. Finally, the total distance, the empty miles, the weighted full miles, and the collaborative benefits obtained by the heuristic solutions approaches are compared to the integrated model.

6.1 Randomly Generated Data Sets

Due to the lack of established research on the pickup and delivery problem with transshipment, adequate benchmark data sets are not available for comparison. Benchmark data sets for the general pickup and delivery problem do not contain enough information to capture the complexities of the TSP-CP network that make it unique and challenging. To overcome this limitation, randomly generated test instances for the TSP-CP were created to incorporate the required network characteristics, information concerning containers and loads, and the required time information.

Two types of data sets were created to test all the required functions of the TSP-CP solution approaches: single test instances and collaborative test instances. A single test instance represents a generic transportation service provider and related information pertaining to loads, nodes, and containers. A collaborative instance combines two or more single problem instances. A collaborative instance was used to both test the solution approaches on larger problems, but also to evaluate the ability of each solution approach to capture the benefits of collaboration.

A single problem instance requires the following information. First, a set of nodes containing both transshipment and non-transshipment nodes was generated. Non-transshipment nodes were created by randomly generating (X,Y) coordinates in a 150 mile by 150 mile square. Additionally, at least three transshipment nodes were generated for each single instance. One transshipment node was placed in the center of the 150 mile by 150 mile square, one transshipment node was placed at the average (X,Y) coordinates of the non-transshipment nodes, and the rest were randomly generated in the same way as the non-transshipment nodes. Distances between nodes was measured using the Euclidean distance between the (X,Y) coordinates.

Next, all load information was generated. Load information includes origin and destination nodes, required container space occupied by the load, and pickup and delivery time windows. Origin nodes and destination nodes for loads were randomly selected from the set of non-transshipment nodes. The required space for each load was randomly selected as either 25%, 50%, 75% or 100% of a container. For time windows, a random pickup and delivery time window within the time horizon was generated for each load.

Finally, the the initial nodes and final nodes for the containers were generated. Both initial and final nodes were randomly selected from the set of all nodes. All container capacities were one and containers were assumed to travel all arcs at 50 miles per hour. The time period length was one hour and a planning time horizon of eight time periods was used in each scenario.

After all single instances were generated, collaborative instances were created by combining the loads, nodes, and containers of two or more single instances. Recall that each single instance has one transshipment node in the center of the 150 mile by 150 mile square. Accordingly, it was assumed that there exists only one center transshipment node in the collaborative instances along with the union of the other transshipment nodes from the combined single instances.

Two data set sizes for single instances were created: one data set size contains 10 loads, 10 containers, and 10 nodes with three transshipment points. For this data set size, a total of 10 instances were generated. The second data set size contains 15 loads, 15 containers, and 15 nodes with four transshipment points. An additional five single instances of size 15 loads, 15 nodes, and 15 containers were generated as well. Finally, five collaborative instances were created by combining single instances to create collaborative instances containing 20 loads, 20 containers, and 20 nodes with five transshipment points. A summary of the data sets is provided in Table 6.1. Table 6.2 shows the multiple single instances that make up each collaborative instance.

Table 6.1: Test instance data set detailed information

Name	Type	Instances	Nodes	Loads	Containers	Transfer Points
TSP-CP 10	Single	10	10	10	10	3
TSP-CP 15	Single	5	15	15	15	4
TSP-CP 20	Collaborative	5	19	20	20	5

Table 6.2: Collaborative data set overview

Collaborative Set	Combined Single Instances
TSP-CP 20-01	TSP-CP 10-01 and TSP-CP 10-02
TSP-CP 20-02	TSP-CP 10-03 and TSP-CP 10-04
TSP-CP 20-03	TSP-CP 10-05 and TSP-CP 10-06
TSP-CP 20-04	TSP-CP 10-07 and TSP-CP 10-10
TSP-CP 20-05	TSP-CP 10-09 and TSP-CP 10-10

6.2 Integrated Model Performance

The computational results for each test instance are summarized in Tables 6.3, 6.4, and 6.5. These results were obtained by solving the integrated mathematical model from Section

4.2 to optimality using CPLEX 12.6. The model was implemented with Java using concert technology and uses default CPLEX settings with a 86,400 second (24 hour) computational time limit. The tables include the objection function value, the required solution time in seconds, the values of the total distance required to deliver all loads measured in miles, the number of those miles that are completely empty, and the number of weighted full miles (WFM) for each test instance.

Table 6.3: TSP-CP integrated model results for the TSP-CP 10 test instances

Name	Objective	Time	Distance	Empty	WFM
TSP-CP 10-01	2964.3	3	975	355	450.3
TSP-CP 10-02	2675.5	4	867	124	480.3
TSP-CP 10-03	2255.0	4	741	62	517.5
TSP-CP 10-04	3412.4	3	1130	304	491.8
TSP-CP 10-05	2846.8	6	942	279	494.8
TSP-CP 10-06	2070.5	76	674	93	334.3
TSP-CP 10-07	2139.6	136	690	56	431.5
TSP-CP 10-08	1906.6	112	613	31	467.5
TSP-CP 10-09	2768.2	42	897	221	537.0
TSP-CP 10-10	2454.2	6	793	138	516.3

Table 6.4: TSP-CP integrated model results for the TSP-CP 15 test instances

Name	Objective	Time	Distance	Empty	WFM
TSP-CP 15-01	3516.7	1587	1124	138	610.5
TSP-CP 15-02	5155.5	1011	1018	206	493.0
TSP-CP 15-03	2804.4	756	897	68	603.0
TSP-CP 15-04	3654.4	1021	1164	99	665.5
TSP-CP 15-05	3206.3	83	1038	181	641.8

Table 6.5: TSP-CP integrated model results for the TSP-CP 20 test instances

Name	Objective	Time	Distance	Empty	WFM
TSP-CP 20-01	4866.2	412	1574	279	996.5
TSP-CP 20-02	4520.5	2579	1474	154	1036.8
TSP-CP 20-03	4647.8	22043	1514	323	873.3
TSP-CP 20-04	4279.6	4771	1365	109	978.3
TSP-CP 20-05	4518.3	58699	1461	193	1076.3

As shown in Table 6.3, the integrated TSP-CP model solved the small TSP-CP 10 test instances with an average solution time of 39.2 seconds and a maximum solution time of 136 seconds. However, when the problem size increased to the moderately sized TSP-CP 15 test instances, solution times drastically increased. The average solution time increased to nearly 891 seconds while the maximum solution time increased to 1,587 seconds. When comparing the TSP-CP 15 test instances to the larger TSP-CP 20 test instances, solution times increased by nearly 20 times on average with the TSP-CP 20 test instances requiring 17,700 seconds on average. Additionally, the maximum solution time for a TSP-CP 20 test instance was nearly 59,000 seconds. Unfortunately, this indicates that a small increase in problem size can lead to an exponential increase in solution times.

An additional consideration to note with the solution times for the integrated TSP-CP model was that they tended to vary greatly depending on the specific instance. For example, solution times for the TSP-CP 10 instances ranged from 3 seconds to 136 seconds, times for the TSP-CP 15 instances ranged from 83 to 1,587 seconds, and times for the TSP-CP 20 instances ranged from as low as 412 seconds to as high as 58,699 seconds. This fluctuation in solution time for the same sized problem highlights what an impact the input data and underlying network can have on solution time. This unpredictability can lead to issues with estimating the ability of the integrated model to provide an optimal solution for any larger problem instance.

Another important feature of the TSP-CP is its ability to capture the amount of time that containers travel completely empty as well as how full containers are when traveling between nodes. Traveling empty is captured through the percentage of total miles that are traveled completely empty. The fullness of containers traveling between locations was evaluated using the percentage of total miles that were weighted full miles. These two values help quantify the ability of a TSP or a group of collaborating TSPs to utilize consolidation. Table 6.6 indicates the percentage of empty miles from the integrated model solutions while Table 6.7 shows the percentage of miles that were weighted full miles.

Table 6.6: Percentage of total miles that are empty for the integrated mathematical model

	TSP-CP 10	TSP-CP 15	TSP-CP 20
Average	18.5%	13.2%	14.1%
Minimum	5.1%	7.6%	8.0%
Maximum	36.4%	20.2%	21.3%

Table 6.7: Percentage of total miles that are WFM for the integrated mathematical model

	TSP-CP 10	TSP-CP 15	TSP-CP 20
Average	58.1%	57.8%	67.0%
Minimum	43.5%	48.4%	57.7%
Maximum	76.3%	73.7%	73.7%

The largest percentage of empty miles came from the small TSP-CP 10 test instances where 18.5% of all miles were empty on average. The TSP-CP 10 test instances also included the largest spread in percentage of empty miles, ranging from as low as 5% of miles being empty in an instance to an instance having as many as 36.4% of miles being empty. This can be attributed to the random placement of nodes within the 150 mile x 150 mile square. With only 10 locations, it was possible for the locations to be bunched together or distributed widely within the 150 mile x 150 mile region. As the density of the network increased with more locations, the percentage of empty miles decreased to 13.2% on average for the TSP-CP 15 instances and 14.1% on average for the TSP-CP 20 instances. The spread between maximum and minimum values also reduced as the number of locations increased. The percentage of empty miles ranged from 7.6% to 20.2% for the TSP-CP 15 test instances while the TSP-CP 20 test instances ranged from 8% to 21.3%.

The increased density of locations also led to the TSP-CP 20 test instances having the highest percentage of weighted full miles. The TSP-CP 20 test instances resulted in an average of 67% of all miles being weighted full miles. These instances also resulted in the least variance in the percentage of weighted full miles, ranging between 57.7% and 73.7%. Similar to empty miles, the TSP-CP 10 test instances had the largest range of weighted full mile percentages with values falling between 43.5% and 76.4%. Finally, the TSP-CP 15 test instances fell between the other two data set sizes with the percentage of weighted full miles ranging from

48.4% and 73.7%. However unlike empty miles, the smaller TPS-CP 10 and TSP-CP 15 instances had approximately the same percentage of weighted full miles, with 58.1% and 57.8% respectively.

6.3 Heuristic Solution Approach Performance

Because solution times tend to grow exponentially as problem size increases and solution times also vary greatly within problem size, heuristic solution approaches are explored to provide solutions to any industry representative data sets. The performance of each heuristic is measured by comparing the objective function value, the total distance measured in miles, the empty miles, and the weighted full miles of the solution obtained by each heuristic to the values obtained by the integrated model. Due to the random nature of the ALNS, all heuristic results presented are an average of five runs for each TSP-CP test instance. In order to compare the two TSP-CP heuristics, each ALNS was run for a predetermined amount of time rather than a predetermined number of iterations. Table 6.8 provides the time limit in seconds placed on each test instance size.

Table 6.8: TSP-CP Time Limits

Name	Time Limit (Seconds)
TSP-CP 10	500
TSP-CP 15	1500
TSP-CP 20	3000

The performance of the two heuristics is best captured through their ability to return solutions close to the optimal solution based on their objective function value. This is determined by the solution gap which is measured as the percentage difference between the optimal objective function value and the objective function value obtained by the heuristic. The solution gap was calculated by $\frac{f(x)-f(x^*)}{f(x^*)}$ where $f(x)$ is the heuristic objective function value and $f(x^*)$ is the optimal objective function value. In addition to objective function value performance, the ability of the heuristics to capture total distance, empty miles, and weight full miles is evaluated as well.

6.3.1 Fixed-optimization ALNS

For the fixed-optimization ALNS to be considered a viable replacement for the TSP-CP integrated model, the heuristic must provide solutions that are close in objective function value to the optimal solution within a reasonable computation time. The fixed-optimization ALNS solution gaps for the TSP-CP test instances are presented in Figure 6.1.

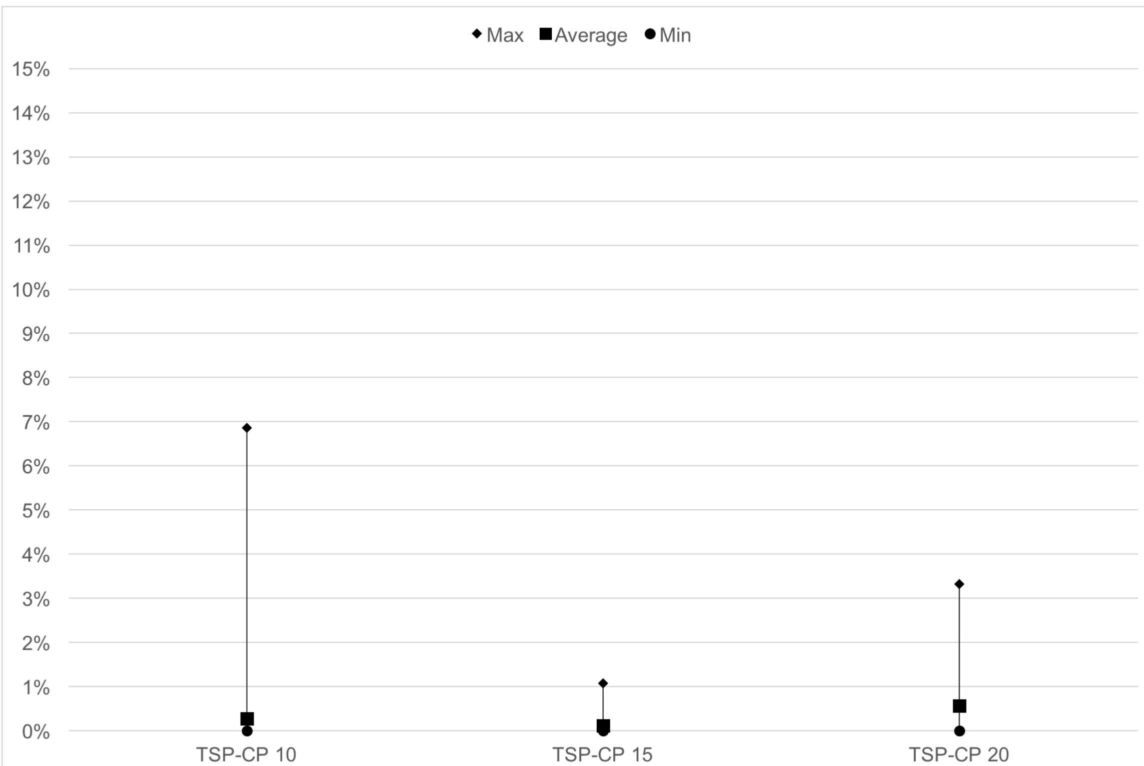

Figure 6.1: Solution gap compared to optimal for the fixed-optimization ALNS

In this figure, the minimum and maximum solution gap values for each TSP-CP test instance size were taken as the overall smallest and largest single solution found among all runs. For example, the maximum solution gap for the TSP-CP 10 instances was generated by the TSP-CP 10-08 instance on run #1. Conservatively, this value is reported as the maximum solution gap, rather than the average value of the solution gap across all five runs for the

TSP-CP 10-08 instance. The fixed-optimization ALNS found the optimal solution for all but one test instance, which still had a minimum solution gap of 0.67%. On average, the fixed-optimization ALNS was able provide solutions that were within 0.31% of optimal. The TSP-CP 10 instances were within 0.27% on average, the TSP-CP 15 test instances were within 0.12% on average, and the TSP-CP 20 test instances were within 0.56% on average. The maximum solution gap from a single solution was a TSP-CP 10 instance with a gap of 6.87%. However, the maximum TSP-CP 15 and TSP-CP 20 instance solution gaps were only 1.07% and 3.32% respectively.

Tables 6.9, 6.10, and 6.11 present the average objective function value, the average solution gap, the average total distance required to deliver all loads, the average number of miles that are empty, and the average weighted full miles for containers based on the five runs for each test instance as a result of applying the fixed-optimization ALNS.

Table 6.9: Fixed-optimization ALNS results for TSP-CP 10 test instances

Name	Avg Objective	Avg Gap	Avg Distance	Avg Empty	Avg WFM
TSP-CP 10-01	2964.3	0.00%	975	355	450.3
TSP-CP 10-02	2675.5	0.00%	867	124	480.3
TSP-CP 10-03	2255.0	0.00%	741	62	517.5
TSP-CP 10-04	3412.4	0.00%	1130	304	491.8
TSP-CP 10-05	2846.8	0.00%	942	279	494.8
TSP-CP 10-06	2070.5	0.00%	674	93	334.3
TSP-CP 10-07	2139.6	0.00%	690	56	431.5
TSP-CP 10-08	1959.0	2.75%	631	37	447.5
TSP-CP 10-09	2768.2	0.00%	897	221	537.0
TSP-CP 10-10	2454.2	0.00%	793	138	516.3

Table 6.10: Fixed-optimization ALNS results for TSP-CP 15 test instances

Name	Avg Objective	Avg Gap	Avg Distance	Avg Empty	Avg WFM
TSP-CP 15-01	3516.7	0.00%	1124	138	610.5
TSP-CP 15-02	3162.0	0.21%	1020	206	493.9
TSP-CP 15-03	2810.4	0.22%	897	59	611.1
TSP-CP 15-04	3660.6	0.17%	1166	115	660.0
TSP-CP 15-05	3206.3	0.00%	1038	181	641.8

Table 6.11: Fixed-optimization ALNS results for TSP-CP 20 test instances

Name	Avg Objective	Avg Gap	Avg Distance	Avg Empty	Avg WFM
TSP-CP 20-01	4907.4	0.85%	1589	262	980.3
TSP-CP 20-02	4528.6	0.18%	1477	155	1037.9
TSP-CP 20-03	4666.1	0.39%	1522	330	873.7
TSP-CP 20-04	4291.7	0.28%	1370	107	984.9
TSP-CP 20-05	4568.6	1.11%	1468	176	1055.0

The fixed-optimization ALNS finds similar values in comparison to the optimal solution for both the percentage empty miles and the percentage of weighted full miles. Table 6.12 shows the average, minimum, and maximum values for the percentage of empty miles found by the fixed-optimization ALNS. The average percentage of empty miles for the fixed-optimization ALNS compared to the integrated model for the TSP-CP 10, TSP-CP 15, and TSP-CP 20 test instances were all within less than 0.5%. Interestingly, it can be seen that the percentage of empty miles for certain instances actually decreased over the integrated model values. For example, for the TSP-CP 20 instances the average percentage of empty miles decreased by -0.45%. This happens because there are a number of alternative optimal solutions due to the symmetry in the time expanded network combined with the fact that empty miles are not explicitly included in the objective function. Considering the effect of adding empty miles to the objective function is an area that needs to be evaluated in future research. Table 6.13 includes the average, minimum, and maximum values for the percentage of weighted full miles for the fixed-optimization ALNS. The average percentage of weighted full miles were similar to empty miles all being within 0.5% of the optimal values.

Table 6.12: Percentage of total miles that are empty for the fixed-optimization ALNS

	TSP-CP 10	TSP-CP 15	TSP-CP 20
Average	18.5%	13.3%	13.7%
Minimum	5.9%	6.6%	7.8%
Maximum	36.4%	20.2%	21.7%

Table 6.13: Percentage of total miles that are WFM for the fixed-optimization ALNS

	TSP-CP 10	TSP-CP 15	TSP-CP 20
Average	57.5%	58.0%	66.6%
Minimum	43.5%	48.4%	57.4%
Maximum	70.9%	68.2%	71.9%

6.3.2 Heuristic Insertion ALNS

As with the fixed-optimization ALNS, the heuristic insertion ALNS must provide optimal or near optimal solutions to the TSP-CP test instances in a reasonable computation time. Figure 6.2 presents the solutions gaps for the various TSP-CP test instances.

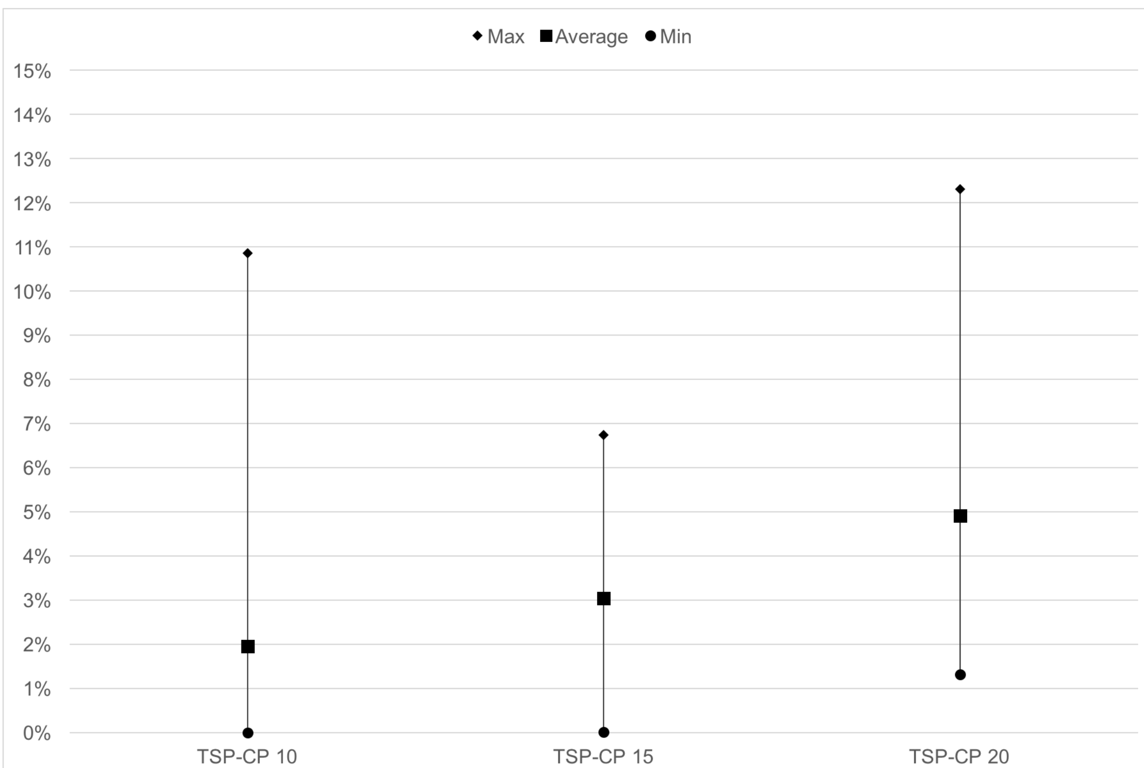

Figure 6.2: Solution gap compared to optimal for the heuristic insertion ALNS

Again, the minimum and maximum solution gaps were based on the worst and best single instance, while the average considers all instances within a given data set size. Overall, the average solution gap provided by the heuristic insertion ALNS was 2.96% with values ranging from optimal up to 12.30% in one case. The application of the heuristic insertion ALNS to the TSP-CP 10 test instances provided the lowest average solution gap at 1.95%, with solution gaps ranging from optimal to 10.85% of optimal on one test instance run. The TSP-CP 15 test instances had a slightly higher average solution gap of 3.04%, but had a considerably smaller range with solution gaps ranging from 0.01% to 6.75%. Finally, the highest on average solution gap and largest range came from the TSP-CP 20 test instances where the average solution gap was 4.91% of optimal and all instances ranged between 1.32% and 12.30%.

Tables 6.14, 6.15, and 6.16 provide the resulting average objective function value, the average solution gap, the average total distance, the average empty miles, and the average weighted full miles from the application of the heuristic insertion ALNS to TSP-CP test instances. Additionally, Table 6.17 provides the percentage of empty miles for the three test instance sizes for the heuristic insertion ALNS solutions. Again as empty miles were not specifically considered in the objective function, these tended to be further off from optimal as compared to the percentage of weighted full miles. As such, the largest difference between the integrated solution value and the heuristic insertion ALNS value was 3.7% with the integrated model finding 13.2% of miles empty for the TSP-CP 15 test instances while the heuristic insertion ALNS found 16.9% of miles empty. The percentage of empty miles for the TSP-CP 10 and TSP-CP 20 instances were much closer with 18.4% and 14.1% of miles being empty in the integrated model and 19.4% and 16.7% being empty with the heuristic insertion ALNS. Table 6.18 provides the percentage of weighted full miles resulting from the heuristic insertion ALNS. As with the fixed-optimization ALNS, these values were much closer to optimal all being within 2.5% of the optimal values.

Table 6.14: Heuristic insertion ALNS results for TSP-CP 10 test instances

Name	Avg Objective	Avg Gap	Avg Distance	Avg Empty	Avg WFM
TSP-CP 10-01	3130.2	5.60%	1030	320	442.4
TSP-CP 10-02	2837.7	6.06%	924	138	537.3
TSP-CP 10-03	2291.3	1.61%	753	78	518.3
TSP-CP 10-04	3412.4	0.00%	1130	304	491.8
TSP-CP 10-05	2846.8	0.00%	942	279	494.8
TSP-CP 10-06	2096.0	1.23%	682	135	353.6
TSP-CP 10-07	2165.0	1.19%	699	67	438.0
TSP-CP 10-08	1942.9	1.90%	626	46	477.4
TSP-CP 10-09	2782.3	0.51%	905	229	537.0
TSP-CP 10-10	2487.6	1.36%	806	151	516.3

Table 6.15: Heuristic insertion ALNS results for TSP-CP 15 test instances

Name	Avg Objective	Avg Gap	Avg Distance	Avg Empty	Avg WFM
TSP-CP 15-01	3673.7	4.47%	1180	234	609.8
TSP-CP 15-02	3197.5	1.34%	1034	218	496.2
TSP-CP 15-03	2824.8	0.73%	916	86	594.8
TSP-CP 15-04	3835.0	4.95%	1239	213	711.1
TSP-CP 15-05	3324.7	3.69%	1077	181	650.5

Table 6.16: Heuristic insertion ALNS results for TSP-CP 20 test instances

Name	Avg Objective	Avg Gap	Avg Distance	Avg Empty	Avg WFM
TSP-CP 20-01	5286.5	8.64%	1725	357	994.1
TSP-CP 20-02	4242.2	4.90%	1556	249	1065.2
TSP-CP 20-03	4771.8	2.67%	1558	370	870.4
TSP-CP 20-04	4472.7	4.51%	1437	125	977.8
TSP-CP 20-05	4691.8	3.84%	1504	218	1086.2

Table 6.17: Percentage of total miles that are empty for the heuristic insertion ALNS

	TSP-CP 10	TSP-CP 15	TSP-CP 20
Average	19.4%	16.9%	16.7%
Minimum	7.4%	9.4%	8.7%
Maximum	31.0%	21.1%	23.8%

Table 6.18: Percentage of total miles that are WFM for the heuristic insertion ALNS

	TSP-CP 10	TSP-CP 15	TSP-CP 20
Average	58.0%	56.5%	64.4%
Minimum	42.9%	48.0%	55.9%
Maximum	76.2%	64.9%	72.2%

6.4 Quantifying the Benefits of Collaboration

A critical aspect of a TSP-CP solution approach is the ability to find and quantify the benefits of collaboration. This capability was evaluated by comparing the results of two single test instances operating independently to the scenario where the two instances shared all containers and nodes to deliver the set of combined loads. For the TSP-CP test instances, the results from the TSP-CP 20 test instances were compared to the combined costs and miles of the two TSP-CP 10 test instances that were combined to make the TSP-CP 20 test instance. These combinations can be found in Section 6.1 in Table 6.2. Table 6.19 provides the results of the two TSP-CP single problem instances that makes up each TSP-CP 20 instance, the results of the TSP-CP 20 test instance, and the comparison of the non-collaborative scenario to the collaborative scenario with respect to total distance traveled, empty miles, and weighted full miles. Table 6.20 and Table 6.21 provide the same for the fixed-optimization ALNS and the heuristic insertion ALNS respectively.

Table 6.19: Integrated model: Benefits of collaboration

Name	Objective	Distance	Empty	WFM
TSP-CP 10-01	2964.3	975	355	450.3
TSP-CP 10-02	2675.5	867	124	480.3
Combined: No Collaboration	5639.8	1842	479	930.5
TSP-CP 20-01: With Collaboration	4866.2	1574	279	996.5
Difference	-773.6	-268	-200	36.0

Name	Objective	Distance	Empty	WFM
TSP-CP 10-03	2255.0	741	62	517.5
TSP-CP 10-04	3412.4	1130	304	491.8
Combined: No Collaboration	5667.4	1871	366	1009.3
TSP-CP 20-02: With Collaboration	4520.5	1474	154	1036.8
Difference	-1146.9	-397	-212	27.5

Name	Objective	Distance	Empty	WFM
TSP-CP 10-05	2846.8	942	279	494.8
TSP-CP 10-06	2070.5	674	93	334.3
Combined: No Collaboration	4917.3	1616	372	829.0
TSP-CP 20-03: With Collaboration	4647.8	1514	323	873.3
Difference	-269.5	-102	-49	44.3

Name	Objective	Distance	Empty	WFM
TSP-CP 10-07	2139.6	690	56	431.5
TSP-CP 10-10	2454.2	793	138	516.3
Combined: No Collaboration	4593.8	1483	194	947.8
TSP-CP 20-04: With Collaboration	4279.6	1365	109	978.3
Difference	-314.2	-118	-85	30.5

Name	Objective	Distance	Empty	WFM
TSP-CP 10-09	2768.2	897	221	537.0
TSP-CP 10-10	2454.2	793	138	516.3
Combined: No Collaboration	5222.4	1690	359	1053.3
TSP-CP 20-05: With Collaboration	4518.3	1461	193	1076.3
Difference	-704.1	-229	-166	23.0

Table 6.20: Fixed-optimization ALNS: Benefits of collaboration

Name	Objective	Distance	Empty	WFM
TSP-CP 10-01	2964.3	975	355	450.3
TSP-CP 10-02	2675.5	867	124	480.3
Combined: No Collaboration	5639.8	1842	479	930.5
TSP-CP 20-01: With Collaboration	4907.4	1589	262	980.3
Difference	-732.4	-253	-217	49.8

Name	Objective	Distance	Empty	WFM
TSP-CP 10-03	2255.0	741	62	517.5
TSP-CP 10-04	3412.4	1130	304	491.8
Combined: No Collaboration	5667.4	1871	366	1009.3
TSP-CP 20-02: With Collaboration	4528.6	1477	155	1037.9
Difference	-1138.8	-394	-211	28.6

Name	Objective	Distance	Empty	WFM
TSP-CP 10-05	2846.8	942	279	494.8
TSP-CP 10-06	2070.5	674	93	334.3
Combined: No Collaboration	4917.3	1616	372	829.0
TSP-CP 20-03: With Collaboration	4666.1	1522	330	873.7
Difference	-251.2	-94	-42	44.7

Name	Objective	Distance	Empty	WFM
TSP-CP 10-07	2139.6	690	56	431.5
TSP-CP 10-10	2454.2	793	138	516.3
Combined: No Collaboration	4593.8	1483	194	947.8
TSP-CP 20-04: With Collaboration	4291.7	1370	107	984.9
Difference	-302.1	-113	-87	37.2

Name	Objective	Distance	Empty	WFM
TSP-CP 10-09	2768.2	897	221	537.0
TSP-CP 10-10	2454.2	793	138	516.3
Combined: No Collaboration	5222.4	1690	359	1053.3
TSP-CP 20-05: With Collaboration	4568.6	1468	176	1055.0
Difference	-653.7	-222	-183	1.7

Table 6.21: Heuristic insertion ALNS: Benefits of collaboration

Name	Objective	Distance	Empty	WFM
TSP-CP 10-01	3130.2	1030	320	442.4
TSP-CP 10-02	2837.7	924	138	537.3
Combined: No Collaboration	5967.9	1954	458	979.7
TSP-CP 20-01: With Collaboration	5286.5	1725	357	994.1
Difference	-681.4	-229	-100	14.4

Name	Objective	Distance	Empty	WFM
TSP-CP 10-03	2291.3	753	78	518.3
TSP-CP 10-04	3412.4	1130	304	491.8
Combined: No Collaboration	5703.7	1883	382	1010.1
TSP-CP 20-02: With Collaboration	4242.2	1556	249	1065.2
Difference	-961.6	-327	-133	55.1

Name	Objective	Distance	Empty	WFM
TSP-CP 10-05	2846.8	942	279	494.8
TSP-CP 10-06	2096.0	682	135	353.6
Combined: No Collaboration	4942.8	1623.8	414	848.3
TSP-CP 20-03: With Collaboration	4771.8	1558	370	870.4
Difference	-171.0	-66	-44	22.1

Name	Objective	Distance	Empty	WFM
TSP-CP 10-07	2165.0	699	67	438.0
TSP-CP 10-10	2487.6	806	151	516.3
Combined: No Collaboration	4652.7	1505	218	954.3
TSP-CP 20-04: With Collaboration	4472.7	1437	125	977.8
Difference	-180.0	-67	-94	23.6

Name	Objective	Distance	Empty	WFM
TSP-CP 10-09	2782.3	905	229	537.0
TSP-CP 10-10	2487.6	806	151	516.3
Combined: No Collaboration	5269.9	1711	380	1053.3
TSP-CP 20-05: With Collaboration	4691.8	1504	218	1086.2
Difference	-578.1	-207	-162	33.0

Figure 6.3 summarizes the benefits of collaboration found by each solution approach as the percentage of savings in total distance and empty miles as well as the percentage increase in the capacity utilization measured by weighted full miles. As expected, the integrated mathematical model identified the most savings in terms of total distance by identifying a 13% reduction in the total distance required to deliver all loads on average across all five of the collaborative instances evaluated. Although the integrated model was able to identify the most benefit, the fixed-optimization ALNS and the heuristic insertion ALNS were able to identify reductions of 12% and 10% in total distance respectively. With empty miles, the fixed-optimization ALNS identified a slightly larger reduction at 42% on average over the 41% identified by the integrated model. The heuristic insertion ALNS identified a substantially smaller reduction in empty miles, but still provided a reduction of 31%.

The percentage of miles that are weighted full miles provides an indication of the ability to better fill containers across arcs. Figure 6.3 highlights the increase in the percentage of weighted full miles from allowing collaboration. As with total distance, the largest increase in weighed full miles came from the integrated model with an increase of 19%. The fixed-optimization ALNS provided a similar increase at 18% while the heuristic insertion ALNS provided a 15% increase in weighted full miles. Overall, each TSP-CP solution approach was able to identify substantial savings resulting from the collaboration of just two test TSPs represented by the randomly generated single instances.

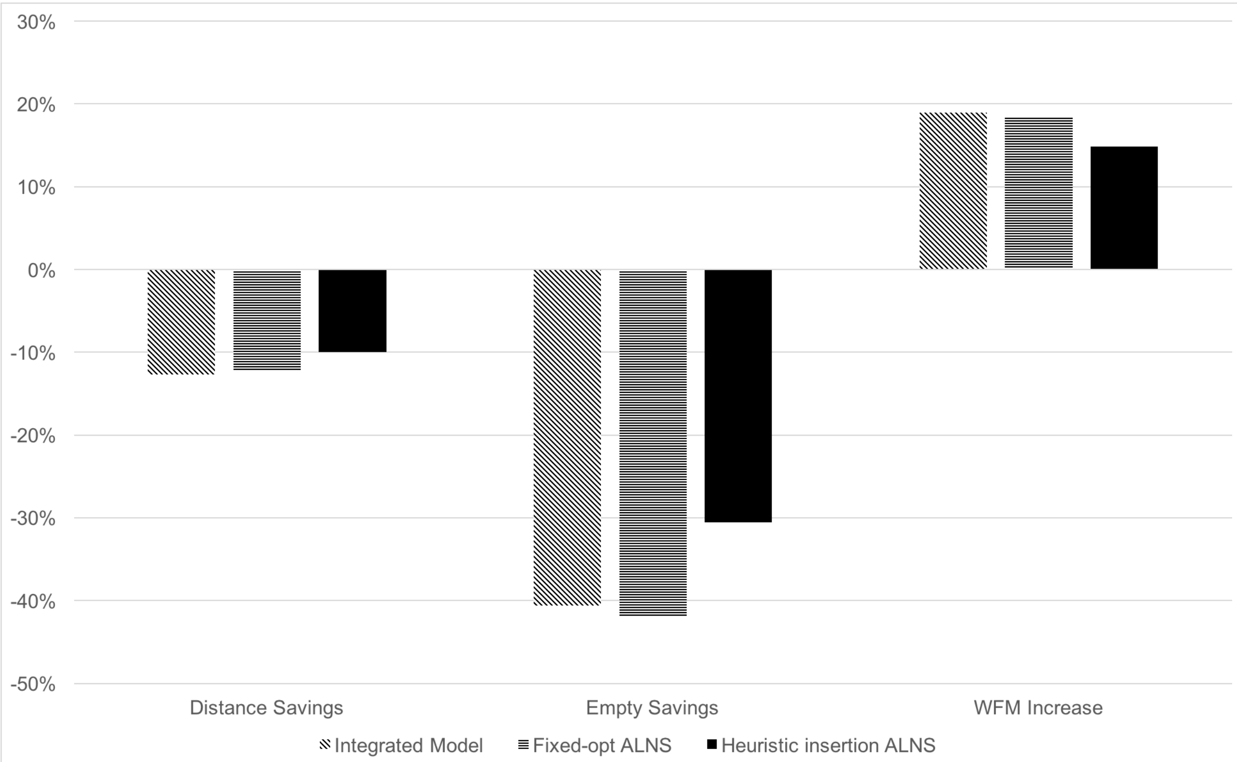

Figure 6.3: Percentage of savings introduced by collaboration quantified by each solution approach

6.5 Summary and Conclusions

This chapter demonstrated the ability of each TSP-CP solution approach to provide solutions and quantify the benefits of collaboration on randomly generated test instances. Two types of test instances were generated, including single instances that represent a single transportation service provider and collaborative instances made of two single instances. As expected, the integrated mathematical model presented in Section 4.2 preformed the best across most of the required criteria. Across all instances the integrated mathematical model provided solutions resulting in 16% of all miles being empty and 60% of miles being weighted full miles. The integrated mathematical model was also able to find the most collaborative benefit identifying an average reduction of 13% in total distance, a 41% reduction in empty miles, and a 19% increase in weighted full miles. Unfortunately, the solution times for the integrated model increased exponentially as the problem sizes increased. Average solution time for the TSP-CP 10 test instances was only 39 seconds, but average solution times increased to 891 seconds for the TSP-CP 15 test instances and to 17,700 seconds for the TSP-CP 20 test instances.

When comparing the two heuristic solution approaches to the optimal solution across all test instances, the fixed-optimization ALNS was able to provide solutions closest to the integrated model with an average solution gap of just 0.31%. Furthermore, the average percentages of empty miles and weighted full miles were 16% and 60% respectively. Collaborative benefits were also very similar to the integrated model finding an average distance savings of 12%, an average empty savings of 42%, and an average weighted full miles increase of 18%. The average solution gap across all instances for the heuristics insertion ALNS came in under 3% at 2.96% of optimal. Similarly, the heuristic insertion ALNS found slightly higher values for empty miles with 18% of miles being empty and slightly lower capacity utilization with 59% of miles being weighted full miles. Collaborative benefits follow a similar pattern with the heuristic insertion ALNS finding a 10% reduction in total distance, a 31% reduction in empty miles, and a 15% increase in weighted full miles.

A major consideration is that the repair methods for the fixed-optimization ALNS involve solving an optimization problem using CPLEX in the same manner as the integrated mathematical model. Because of this, as problems grow in size, fixed-optimization repair method solution times grow as well. As the solution time required for each iteration grows, the number of iterations that can be completed in a given amount of time falls. Overall the fixed-optimization ALNS already completes far fewer iterations in a given time period as compared to the heuristic insertion ALNS. Table 6.22 shows the average number of iterations completed on the TSP-CP test instances to obtain the results previously presented.

Table 6.22: Iterations completed in the imposed time limits

Name	Fixed-optimization	Heuristic insertion
TSP-CP 10	529	118069
TSP-CP 15	180	78586
TSP-CP 20	89	23566

An additional limitation of solving a mathematical model each iteration using CPLEX is that running out of memory is a concern. As the mathematical models grow in size, so does the amount of memory that is used. All of these limitations lead to concerns that the fixed-optimization ALNS will not be able to complete enough iterations to provide adequate improvement as problem sizes keep increasing. Fortunately, the heuristic insertion ALNS provides the ability to consistently perform more iterations and the power to eliminate any memory concerns associated with the mathematical models and CPLEX. Industry representative problem instances grow well beyond the TSP-CP test instances causing the required number of iterations for the fixed-optimization ALNS to increase and for memory concerns to rise. For this reason, although the fixed-optimization ALNS tends to out perform the heuristic insertion ALNS in comparison to the integrated mathematical model, the heuristic insertion ALNS is the preferred solution method to quantify the benefits of collaboration on industry representative data sets.

Chapter 7

TSP Collaboration: Benefits and Insights

The potential benefits of horizontal collaboration among transportation service providers (TSPs) are investigated in this chapter using the heuristic insertion ALNS for the TSP-CP. In this research, the specific area of interest is the efficiency gains possible for full truckload (FTL) and less than truckload (LTL) carriers enabled by collaborating to delivery their freight. These potential benefit are quantified using industry representative daily delivery data sets derived from actual freight information from both FTL and LTL carriers. The TSP-CP was utilized to optimize freight consolidation and delivery for the daily deliveries.

The benefits of collaboration were identified by comparing non-collaborative scenarios to fully collaborative scenarios. A non-collaborative scenario represented a single TSP operating independently to delivery only their freight. In each scenario without collaboration, carriers optimized their own freight with the TSP-CP using only their own facilities and resources. These scenarios essentially represented the best-case scenario on how a TSP would currently operate. In reality, each TSP would have a transportation plan based on their own freight which may or may not be optimal. In the fully collaborative scenarios, the freight of two or more TSPs was pooled for delivery by the combined set of resources using all available facilities. All decisions for the collective group are assumed to be made under centralized control, meaning that a single entity made all the decisions for the group with access to all required information, freight, resources, and facilities. Unfortunately, a large amount of trust and information sharing is required by centralized control. For this reason, in practice, centralized control may not be the most realistic control mechanism, but it does provide a best case scenario when assessing the potential benefits of collaboration. Further research would be needed to extend the TSP-CP to a decentralized setting. Currently, research into decentralized control mechanisms for horizontal collaborations are still in its infancy.

Each TSP was represented by a single non-collaborative scenario. Using the single carriers, collaborative scenarios were created by combining up to four carriers. The impact of collaboration was quantified by evaluating the differences that exist between the combined independently found, TSP-CP solutions to the non-collaborative scenarios that make up each collaborative scenario to the TSP-CP solution of the accompanying collaborative scenario. In addition to quantifying the benefits of collaboration, this research also provides additional insights into the horizontal collaboration of FTL and LTL carriers. These insights were gathered by further examining the non-collaborative and collaborative scenarios to determine the effects of collaboration on individual carriers. Furthermore, TSP partnership composition was examined in various ways to determine the impact of each collaborative network.

Solutions were provided for each scenario using the heuristic insertion ALNS for the TSP-CP implemented using Java through the Eclipse IDE. All solutions were generated using a Windows 8.1 computer with an Intel Xeon E5-2620 2.4 GHz processor and 32 GB of RAM. The remainder of this chapter covers the industry data sets in more detail as well as the benefits and insights of TSP collaboration.

7.1 Data Overview

The industry representative data sets were derived from actual transportation service provider data. The data was provided by a freight pooling company that specializes in operating horizontal collaborative partnerships and included the yearly freight movements from a number of independent LTL and FTL carriers. From the set of carriers, 68 were chosen to make collaborative partnerships based on their freight, their customers, their network, and their size. In total, 16 of the 68 carriers chosen were FTL carriers while the remaining 42 were LTL carriers.

Daily loads between origin locations and destination locations were extracted from the data based on the yearly freight movements. To complete each carrier network, the origin and destination locations required by loads acted as the set of nodes. Additionally, a set of available containers capable of delivering all loads was included as well. A few extra assumptions were made to include all the necessary information needed to provide a consolidation and routing solution for each carrier using the TSP-CP. In the data, a detailed description of each location was not available. Instead, locations were simply represented by a name, a longitude, and a latitude. Because the capability of each location was not known, transshipment was only permitted at an LTL break-bulk terminal. This assumption also differentiated the operation of an LTL carrier from an FTL carrier. If no LTL carrier was present in a given scenario, no intermediate consolidation node was available. A break-bulk terminal was easily identified in the data as the LTL location or locations where the majority of freight originated or terminated. The distance between locations was calculated in miles based on the longitude and latitude values.

Load sizes were determined based on the volume traveling between the origin and destination locations in the industry data. However, multiple volume units were used for loads in the data. To provide equal comparisons, the size of each load was converted to a fraction of a container. Although load sizes were included, any information about time windows was absent. Because loads were daily, the simple assumption was made that all pick up time windows occur in the first half of the time horizon while all delivery time windows occur in the second half of the time horizon. This was to represent pick ups taking place in the morning and deliveries taking place in the afternoon. Because load volumes were introduced as a fraction of a container, all container capacities were assumed to be one. Finally, starting and ending locations for containers were selected. The containers of an LTL carrier were assumed start and end at the break-bulk terminal. If multiple break-bulk terminals were present, a subset of containers was randomly assigned to each break-bulk terminal. Starting and ending locations for FTL carrier containers were randomly spread throughout the FTL carrier's network.

Table 7.1 provides an overview of the average size of the LTL and FTL carriers including the average number of loads, nodes, and containers. Table 7.2 shows the name and size for each of the 16 FTL carriers, while Table 7.3 shows same for the 42 LTL carriers.

Table 7.1: Industry Representative Data: TSP Overview

Type	Count	Avg Loads	Avg Nodes	Avg Containers
FTL	16	11	14	10
LTL	42	13	14	10

Table 7.2: Industry Representative Data: FTL Overview

Name	Loads	Nodes	Containers	Name	Loads	Nodes	Containers
FTL-01	27	29	20	FTL-09	9	15	12
FTL-02	21	30	25	FTL-10	9	14	8
FTL-03	20	25	20	FTL-11	8	11	10
FTL-04	18	20	20	FTL-12	7	12	6
FTL-05	15	18	20	FTL-13	7	10	8
FTL-06	14	19	15	FTL-14	8	10	6
FTL-07	11	18	10	FTL-15	5	6	5
FTL-08	14	14	10	FTL-16	2	2	1

Table 7.3: Industry Representative Data: LTL Overview

Name	Loads	Nodes	Containers	Name	Loads	Nodes	Containers
LTL-01	33	34	25	LTL-22	10	11	10
LTL-02	33	34	20	LTL-23	10	11	8
LTL-03	31	33	20	LTL-24	10	11	8
LTL-04	20	23	15	LTL-25	10	11	8
LTL-05	18	19	15	LTL-26	9	10	6
LTL-06	18	19	15	LTL-27	8	9	8
LTL-07	18	19	12	LTL-28	8	9	7
LTL-08	18	19	12	LTL-29	8	9	6
LTL-09	17	18	12	LTL-30	7	9	5
LTL-10	17	18	10	LTL-31	7	8	5
LTL-11	17	18	10	LTL-32	7	8	5
LTL-12	15	16	12	LTL-33	6	7	5
LTL-13	16	17	10	LTL-34	6	7	5
LTL-14	16	17	10	LTL-35	5	8	3
LTL-15	16	17	8	LTL-36	5	6	5
LTL-16	15	16	10	LTL-37	4	5	6
LTL-17	15	16	10	LTL-38	4	7	3
LTL-18	12	13	12	LTL-39	5	6	3
LTL-19	14	15	8	LTL-40	5	6	3
LTL-20	13	14	10	LTL-41	3	5	3
LTL-21	12	13	10	LTL-42	3	4	3

7.1.1 Collaborative Partnerships

Using the 16 FTL carriers and the 42 LTL carriers, 67 collaborative partnerships were created for evaluation. Collaborative partnerships of up to four different carriers were investigated. The 67 collaborative partnerships included 35 two TSP partnerships (2-TSP), 17 three TSP partnerships (3-TSP), and 15 four TSP partnerships (4-TSP). Table 7.4 details the name, number, and average size of the collaborative partnerships.

Table 7.4: Industry Representative Data: Collaboration Partnerships Overview

Partners	Name	Count	Avg Loads	Avg Nodes	Avg Containers
Two	2-TSP	35	24	28	19
Three	3-TSP	17	31	37	26
Four	4-TSP	15	32	39	30
Total		67	30	36	26

The 67 collaborative partnerships were constructed in an effort to gather insights and quantifiable benefits based on the different partnership characteristics. As illustrated by Tables 7.2 and 7.3, there was a large amount of variety in the sizes of the FTL and LTL carriers considered. The combination of having both FTL and LTL carriers as well as having different sized carriers for each carrier type allowed flexibility in building interesting collaborative partnerships. For example, it was possible to build a partnerships between an FTL carrier and an LTL carrier where the FTL carrier was the larger carrier, a partnership where the LTL carrier was the larger carrier, and a partnership where both carriers were roughly the same size.

In addition to partnerships based on carrier type and size, partnerships were also created by examining each carrier's customers, network, and geographic location. Partnerships were built on the basis that carriers with shared customers and complimentary networks in the same geographical region would be more likely to find benefit from collaboration. Tables 7.5, 7.6, and 7.7 show the details of each collaborative partnership containing two, three, and four carriers respectively.

Table 7.5: Industry Representative Data: 2-TSP Overview

Name	Partner 1 (P_1)	Partner 2 (P_2)	Loads	Nodes	Containers
2-TSP-01	FTL-14	LTL-04	28	32	21
2-TSP-02	FTL-03	LTL-30	27	24	25
2-TSP-03	LTL-21	LTL-05	30	33	30
2-TSP-04	LTL-16	LTL-15	21	33	18
2-TSP-05	FTL-12	LTL-26	16	21	12
2-TSP-06	LTL-28	FTL-12	15	21	12
2-TSP-07	FTL-08	LTL-02	47	48	20
2-TSP-08	FTL-08	LTL-31	21	21	15
2-TSP-09	FTL-08	LTL-15	30	31	18
2-TSP-10	FTL-08	FTL-12	21	26	16
2-TSP-11	FTL-01	FTL-12	34	41	26
2-TSP-12	FTL-04	LTL-07	36	39	32
2-TSP-13	FTL-07	FTL-05	26	36	30
2-TSP-14	LTL-37	LTL-15	20	21	14
2-TSP-15	LTL-32	FTL-03	27	33	25
2-TSP-16	FTL-02	FTL-10	30	44	33
2-TSP-17	LTL-17	FTL-08	29	30	20
2-TSP-18	LTL-19	LTL-16	29	31	18
2-TSP-19	LTL-19	LTL-10	21	33	18
2-TSP-20	FTL-15	LTL-16	20	22	15
2-TSP-21	FTL-15	FTL-12	12	18	11
2-TSP-22	FTL-15	LTL-28	13	15	12
2-TSP-23	LTL-39	LTL-07	23	25	15
2-TSP-24	LTL-40	FTL-12	12	18	9
2-TSP-25	LTL-09	LTL-15	33	35	29
2-TSP-26	LTL-09	FTL-12	24	29	18
2-TSP-27	LTL-15	LTL-15	21	23	13
2-TSP-28	LTL-37	LTL-36	9	11	11
2-TSP-29	FTL-16	LTL-28	13	15	12
2-TSP-30	FTL-15	LTL-17	20	22	15
2-TSP-31	LTL-38	LTL-35	9	7	6
2-TSP-32	LTL-39	LTL-35	10	12	6
2-TSP-33	LTL-39	LTL-38	9	11	6
2-TSP-34	LTL-19	LTL-15	30	32	16
2-TSP-35	LTL-19	LTL-07	32	34	20

Table 7.6: Industry Representative Data: 3-TSP Overview

Name	P_1	P_2	P_3	Loads	Nodes	Containers
3-TSP-01	FTL-16	LTL-37	LTL-15	22	22	15
3-TSP-02	LTL-39	LTL-07	LTL-10	40	42	25
3-TSP-03	LTL-38	LTL-35	LTL-07	27	26	18
3-TSP-04	LTL-25	LTL-11	LTL-13	43	45	28
3-TSP-05	LTL-39	LTL-37	FTL-08	23	25	19
3-TSP-06	LTL-34	FTL-11	LTL-06	32	37	30
3-TSP-07	FTL-09	FTL-13	LTL-12	31	41	32
3-TSP-08	FTL-01	FTL-08	FTL-12	48	55	36
3-TSP-09	LTL-39	LTL-17	FTL-08	34	36	23
3-TSP-10	FTL-12	LTL-15	LTL-31	30	36	19
3-TSP-11	LTL-38	LTL-19	LTL-07	36	39	23
3-TSP-12	LTL-14	LTL-29	LTL-27	32	34	24
3-TSP-13	LTL-41	LTL-22	LTL-08	31	34	25
3-TSP-14	LTL-39	FTL-11	FTL-06	27	36	28
3-TSP-15	FTL-15	LTL-36	FTL-04	28	31	30
3-TSP-16	FTL-15	FTL-11	FTL-06	27	36	30
3-TSP-17	FTL-15	FTL-12	FTL-06	26	37	26

Table 7.7: Industry Representative Data: 4-TSP Overview

Name	P_1	P_2	P_3	P_4	Loads	Nodes	Containers
4-TSP-01	LTL-38	FTL-15	LTL-36	LTL-35	19	18	16
4-TSP-02	FTL-16	LTL-37	FTL-08	LTL-15	36	36	25
4-TSP-03	LTL-38	LTL-36	LTL-35	LTL-07	32	32	23
4-TSP-04	LTL-39	FTL-16	LTL-37	FTL-08	25	26	20
4-TSP-05	FTL-16	LTL-37	LTL-35	FTL-08	25	26	20
4-TSP-06	LTL-24	LTL-42	LTL-33	FTL-06	33	41	31
4-TSP-07	LTL-39	LTL-38	LTL-24	LTL-23	29	33	22
4-TSP-08	FTL-15	FTL-07	FTL-12	FTL-06	37	55	36
4-TSP-09	FTL-15	FTL-11	FTL-12	FTL-06	34	48	36
4-TSP-10	FTL-15	FTL-07	FTL-11	FTL-06	38	54	40
4-TSP-11	FTL-15	LTL-24	FTL-11	FTL-06	37	47	38
4-TSP-12	LTL-39	FTL-15	FTL-11	FTL-06	32	42	33
4-TSP-13	LTL-20	FTL-11	FTL-12	FTL-14	36	47	32
4-TSP-14	LTL-34	FTL-11	LTL-22	LTL-18	36	42	37
4-TSP-15	LTL-25	LTL-34	LTL-22	LTL-18	38	42	35

7.2 The Benefits of TSP Collaboration

The performance metrics of interest for this research include the change in the total number of miles required to deliver all loads, the change in the number of miles that were traveled completely empty, and the change in container space utilization again measured by the percentage of miles that were weighted full miles. Miles were used to quantify benefit as a surrogate for cost because the evaluation of cost or the importance of certain measures may vary greatly by carrier. It was assumed that each carrier owned its own containers so the impact on the number of containers in use was also evaluated.

Figure 7.1 summarizes the potential benefits introduced by collaboration. Benefits were determined by calculating the percentage difference between the combined number of miles, empty miles, percentage of weighted full miles, and containers of all carriers in a collaborative partnership operating independently to the collective number found when all carriers collaborated. Figure 7.1 indicates the percentage difference in miles, empty miles, weighted full miles, and containers that was found between the values of the non-collaborative scenario as compared to the collaborative scenario.

The results suggest that as more partners become involved in collaboration, the benefits increase. The 4-TSP partnerships provided substantially more savings in each of the four performance metrics including using 18% fewer containers to find an average 22% reduction in miles, a 44% reduction in empty miles, and a 29% increase in the percentage of miles that were weighted full miles. More partners again lead to more savings with the 3-TSP partnerships having the second best performance. The 3-TSP partnerships provided an average 16% reduction in total miles, a 31% reduction in empty miles, a 15% increase in the percentage of weighted full miles, and a 10% decrease in the number of containers used. Finally, the 2-TSP partnerships provided a 12% decrease in total miles, a 24% decrease in empty miles, a 14% increase in the percentage of weighted full miles, and used 10% fewer containers.

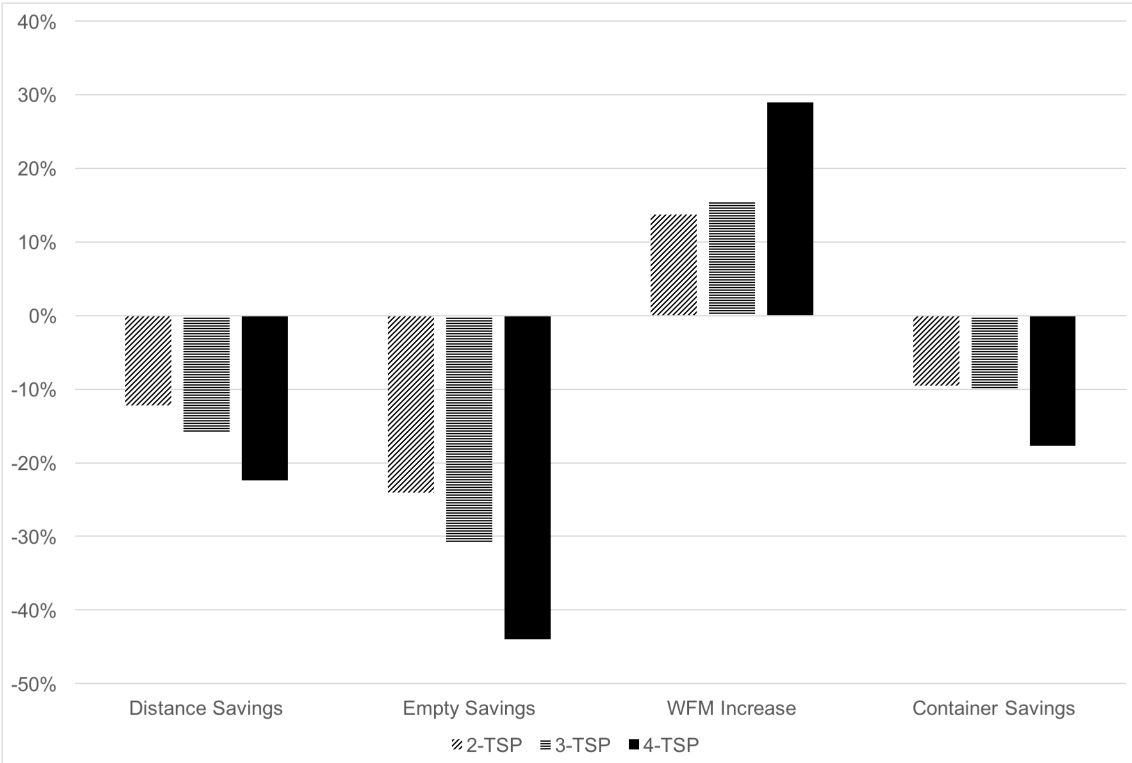

Figure 7.1: Benefits of TSP Collaboration for the various sized partnerships

Out of the 67 total partnerships, only two partnerships provided no change or an increase to the percentage of total miles. Both of these partnerships contained only two carriers. These 2-TSP partnerships as well as one other 2-TSP partnership resulted in an overall increase or found no change in the number of empty miles. No 3-TSP or 4-TSP partnership resulted in an increase in miles or empty miles. A total of six partnerships resulted in a decrease or found no increase in the percentage of weighted full miles. Four of these partnerships had only two carriers, one had three carriers, and one had four carriers. Finally, no partnerships resulted in an increase in the number of containers required for delivery.

7.3 TSP Collaboration Insights

In addition to quantifying the benefits of collaboration, the diversity in the set of TSPs allowed for additional insights to be gained about what happens to individual carriers in a collaborative setting. This research examines the impact in the following areas: the impact of carrier type, the impact of carrier size, the impact of the amount of each network provided by carrier type, the impact of relative carrier size, and the impact of the change in average network distance. This section explores each of these impacts in detail.

7.3.1 Impact of Carrier Type

Examining the different benefits of collaboration observed by a partnership based on the type of carriers involved in that partnership provides additional insights into TSP collaboration. For example, an easy way to break down a partnerships was by the number of LTL carriers and the number of FTL carriers that collaborate in each partnership. This section provides the analysis of collaborative benefit based on the types of carriers involved in each partnership. Also, the benefit to an individual carrier based on whether it was an FTL or an LTL carrier is examined as well.

2-TSP Partnerships

Out of the 35 2-TSP partnerships, 16 partnerships consisted of one FTL carrier and one LTL carrier, 14 were with two LTL carriers, and five were with two FTL carriers. Despite the type of carriers involved in a particular partnership, each different combination of carrier types was able to identify an average benefit on all four metrics. Table 7.8 provides these benefits. As shown in the table, a partnership consisting of only FTL carriers found the most benefit across all four metrics. A partnership between an FTL carrier and an LTL carrier provided slightly less benefit than two FTL carriers, but more than a partnership between two LTL carriers.

Table 7.8: Benefit by Collaborative Makeup: 2-TSP

Partnership Type	Number	Total Miles	Empty Miles	WFM	Containers
Two LTL	14	-7%	-14%	11%	-5%
FTL and LTL	16	-15%	-30%	13%	-12%
Two FTL	5	-17%	-34%	24%	-12%

When examining the average benefit to an individual carrier based on its partners, an FTL carrier benefited substantially more from collaboration than an LTL carrier as indicated in Table 7.9. Across all cases with two partners, an FTL carrier benefited from collaboration with an average 18% reduction in miles, a 26% reduction in empty miles, a 36% increase in the percentage of weighted full miles, and a 5% reduction in the number of containers used. When collaborating with a single LTL carrier, an individual FTL carrier found even more benefit with a 21% reduction in miles, a 29% reduction in empty miles, a 40% increase in weighted full miles, and a 19% reduction in the number of containers needed.

Table 7.9: Benefit by Carrier Type: 2-TSP

Carrier Type	Total Miles	Empty Miles	WFM	Containers
LTL	1%	1%	8%	-5%
FTL	-18%	-26%	36%	-15%

Although Table 7.8 indicates that an LTL carrier found benefit when collaborating with another LTL carrier, the results indicated that with just two collaborative partners an LTL carrier actually increased their total miles and empty miles by 1% each. This was due to the fact that when collaborating with an FTL carrier, an LTL carrier had an average increase in miles of 17% and empty miles of 31%. However, overall an LTL carrier was still able to benefit from collaboration with an 8% increase in the percentage of weighted full miles and a 5% reduction in the number of containers used. The average benefit to an individual carrier based on its type and the type of its partner is provided in Table A.1 in Appendix A.

3-TSP Collaboration

The 17 3-TSP partnerships included five partnerships with two LTL carriers and one FTL carrier, six partnerships with three LTL carriers, three partnerships with one LTL carrier and two FTL carriers, and three partnerships with three FTL carriers. Table 7.10 summarizes the collaborative benefits identified for the four possible mixes of FTL and LTL carriers for the 3-TSP partnerships.

Table 7.10: Benefit by Collaborative Makeup: 3-TSP

Partnership Type	Number	Total Miles	Empty Miles	WFM	Containers
Three LTL	6	-9%	-11%	5%	-9%
FTL and Two LTL	5	-19%	-37%	19%	-10%
Two FTL and LTL	3	-17%	-39%	16%	-5%
Three FTL	3	-24%	-50%	30%	-16%

As shown in Table 7.10 with three partners, the collaboration between only FTL carriers provided the most benefit while the collaboration between only LTL carriers provided the least benefit. The second largest reduction in miles, empty miles, and containers was provided by the collaboration between two LTL carriers and a single FTL carrier. The third largest reduction came from the collaboration between two FTL carriers and a single LTL carrier. When evaluating the difference in the percentage of weighted full miles, this order was reversed with two LTL carriers and one FTL carrier finding a larger increase over the partnership of one LTL carrier and two FTL carriers.

As shown in Table 7.11, when a third carrier was added to the partnerships, the gap between the benefits obtained by FTL carriers and LTL carriers closed substantially. With three partners, FTL carriers still benefited slightly more than LTL carriers in the reduction of total miles and empty miles as well as in the increase in the percentage in weighted full miles. However, LTL carriers received a larger reduction in the number of containers used.

The increase in LTL benefit came from the drastic impact that an additional partner had on the collaboration between FTL and LTL carriers. When collaborating in a partnership with two FTL carriers and one LTL carrier, an FTL carrier still benefited slightly more on

Table 7.11: Benefit by Carrier Type: 3-TSP

Carrier Type	Total Miles	Empty Miles	WFM	Containers
LTL	-16%	-28%	13%	-11%
FTL	-17%	-33%	26%	-10%

average with a 24% reduction in miles and a 27% increase in weighted full miles as compared to the 20% reduction in miles and 0% increase in weighted full miles for the LTL carrier. However, in this case the LTL carrier benefited more in the reduction of empty miles and the number of containers used with a 51% reduction in empty miles and a 16% decrease in the number of containers used. An FTL carrier still benefited with a 29% reduction in empty miles and slight 6% decrease in the number of containers used.

A major change was also identified in the benefits of a partnership consisting of two LTL carriers and one FTL carrier. In this situation, an LTL carrier found substantially more benefit in the reduction of miles, empty miles, and containers over an FTL carrier. Here, collaboration provided an LTL carrier an average reduction of 25% in miles, 46% in empty miles, and 13% in the number of containers used. In comparison, an FTL carrier had a 4% reduction in miles, a 19% reduction in empty miles, and a 2% reduction in the number of containers. Collaboration still provided an FTL carrier with a larger increase in the percentage of weighted full miles at 31%, compared to the 14% increase provided to an LTL carrier. A full table of the benefits to an individual carrier based on its type and the type of its partners is provided in Table A.2 Appendix A.

4-TSP Collaboration

The 15 4-TSP partnerships broke down into three partnerships with four LTL carriers, three partnerships with three LTL carriers and one FTL carrier, three partnerships with two LTL carriers and two FTL carriers, three partnerships with one LTL carrier and three FTL carriers, and three partnerships with four FTL carriers. Table 7.12 summarizes the collaborative benefits identified for the five possible mixes of FTL and LTL carriers for the

4-TSP partnerships.

Table 7.12: Benefit by Collaborative Makeup: 4-TSP

Partnership Type	Number	Total Miles	Empty Miles	WFM	Containers
Four LTL	3	- 16%	-24%	14%	-16%
FTL and Three LTL	3	-27%	-46%	32%	-24%
Two FTL and Two LTL	3	-17%	-45%	24%	-15%
Three FTL and LTL	3	-24%	-50%	29%	-14%
Four FTL	3	-28%	-55%	45%	-21%

Similar to the collaboration of two carriers and three carriers, a partnership containing four FTL carriers provided the most benefit on average and the partnership of four LTL carriers provided the least overall benefit. However, in this situation a partnership consisting of two LTL carriers and two FTL carriers provided very similar benefit to the situation with four LTL carriers. The collaborative partnerships with three FTL carriers and one LTL carrier and the partnerships with one FTL carrier and three LTL carriers fell in the middle, but still provided very similar benefits to each other.

As indicated in Table 7.13, in the presence of four carriers the results suggest that an individual LTL carrier in fact benefited more on average in the reduction of miles, empty miles, and the number of containers used over an individual FTL carrier. However, FTL carriers still received a larger increase in the percentage of weighted full miles.

Table 7.13: Benefit by Carrier Type: 4-TSP

Carrier Type	Total Miles	Empty Miles	WFM	Containers
LTL	-24%	-40%	29%	-19%
FTL	-12%	-29%	39%	-11%

Overall in a four partner setting, an individual FTL carrier benefited the most from collaborating only with other FTL carriers. However, benefits were still found for an individual FTL carrier in a partnership with three FTL carriers and one LTL carrier and in a partnership with two FTL carriers and two LTL carriers. When collaborating in a partnership containing three FTL carriers and one LTL carrier, an individual FTL carrier reduced miles by 18%, reduced empty miles by 39%, increased the percentage of weighted full miles by

44%, and reduced the number of containers used by 5%. Similarly, when an individual FTL carrier was in a partnership containing two FTL carriers and two LTL carriers, an FTL carrier saw a 21% reduction in miles, a 35% reduction in empty miles, a 29% increase in its percentage of weighted full miles, and a 21% reduction in the number of containers it used. Limited benefit was found for a single FTL carrier when collaborating in a partnership with three LTL carriers. In this situation, the only benefit to the FTL carrier was a 27% increase in its percentage of weighted full miles. Even though the percentage of weighted full miles increased, the FTL carrier received a substantial 40% increase in miles, a 43% increase in empty miles, and an 8% increase in the number of containers used.

As expected, an individual LTL carrier benefited the most when the FTL carrier benefited the least. In a collaborative partnership containing three LTL carriers and one FTL carrier, on average an LTL carrier was able to reduce miles by 32%, empty miles by 48%, and increase the percentage of weighted full miles by 31%. This was all done while using 21% fewer containers. An LTL carrier benefited similarly in the remaining 4-TSP partnership types except in the area of weighted full miles. In a partnership with two FTL carriers and two LTL carriers, an LTL carrier was able to increase its percentage of weighted full miles by 50% while when in a partnership with three FTL carriers, the lone LTL carrier only increased its percentage of weighted full miles by 11%. However, these two partnership constructions still provided an LTL carrier with approximately a 20% reduction in miles, a 45% reduction in empty miles, and a 15% reduction in the number of containers used. A full summary of these values can be found in Table A.3 in Appendix A.

7.3.2 Impact of Individual Carrier Size

In addition to the impact that the type of carrier had on the benefits received from collaboration, the size of a particular carrier or set of carriers affected performance as well. Each carrier was classified as either an average sized carrier, a large sized carrier, or a small sized carrier. This classification was determined by the sum of a carrier's loads, nodes, and containers in comparison to the all other carriers. Out of the 42 LTL carriers, 11 were determined to be large sized carriers, 18 were average sized carriers, and 13 were small sized carriers. For FTL carriers, four were large sized carriers, seven were average sized carriers, and five were small sized carriers. Tables 7.14, 7.15, 7.16 summarize the benefits by carrier size for the 2-TSP, 3-TSP, and 4-TSP partnerships respectively.

2-TSP Collaboration

The results suggest that in a two partner collaborative setting, a small FTL carrier received the largest percentage reduction in miles, empty miles, and the number of containers used. Additionally, a small FTL carrier increased its percentage of weighted full miles the most. Although a small FTL carrier benefited the most, benefits across the board were achieved by an FTL regardless of its size.

When examining the benefit to an LTL carrier based on its size, the opposite was found. Overall a large LTL carrier was the only LTL carrier that saw a reduction in miles as a result of a 2-TSP partnership. This was also accompanied by the highest reduction in empty miles. For an average or small sized LTL carrier, collaboration increased their miles and had minimal impact on their empty miles. However, collaboration still allowed for both to increase their percentage of weighted full miles and provided a reduction in the number of containers that they used.

Table 7.14: Benefit by Carrier Size: 2-TSP Overview

Carrier Type	Size	Total Miles	Empty Miles	WFM	Containers
LTL	Small	3%	11%	20%	-11%
LTL	Average	4%	-2%	2%	-1%
LTL	Large	-8%	-12%	0%	-1%

Carrier Type	Size	Total Miles	Empty Miles	WFM	Containers
FTL	Small	-28%	-40%	69%	-31%
FTL	Average	-11%	-17%	8%	-1%
FTL	Large	-3%	11%	20%	-11%

3-TSP Collaboration

Similar to the two partner collaborative setting, with three partners a small FTL carrier benefited the most and a large FTL still found improvements on all four metrics. However, this time, an average sized FTL carrier actually increased their miles and number of containers used, but collaboration still reduced their empty miles and increased their percentage of weighted full miles.

With the addition of a third collaborative partner, benefits were found by all three sizes of LTL carriers. In fact, each LTL carrier was able to find additional benefit over an LTL carrier in a 2-TSP partnership including a substantial increase in the benefits for a small sized LTL carrier.

Table 7.15: Benefit by Carrier Size: 3-TSP Overview

Carrier Type	Size	Total Miles	Empty Miles	WFM	Containers
LTL	Small	-27%	-48%	22%	-17%
LTL	Average	-3%	-11%	6%	-7%
LTL	Large	-15%	-18%	4%	-7%

Carrier Type	Size	Total Miles	Empty Miles	WFM	Containers
FTL	Small	-50%	-62%	48%	-41%
FTL	Average	9%	-12%	15%	17%
FTL	Large	-19%	-35%	4%	-14%

4-TSP Collaboration

When increasing the number of partners to four, the results suggested that an average LTL carrier received the largest reduction in miles and empty miles. However, a small FTL carrier still increased weighted full miles and reduced the number of containers used the most. Additionally, an average FTL carrier received benefit in all categories except a slight increase in the number of containers it used. An average sized LTL carrier again received more benefit as a result of an additional partner. However, both a large sized LTL carrier and a small sized LTL carrier suffered in all areas except that a small LTL carrier was able to find an increase in its percentage of weighted full miles. Unfortunately, no large FTL carrier was used in any of the 4-TSP partnerships studied here.

Table 7.16: Benefit by Carrier Size:4-TSP Overview

Carrier Type	Size	Total Miles	Empty Miles	WFM	Containers
LTL	Small	-20%	-39%	34%	-17%
LTL	Average	-33%	-45%	25%	-25%
LTL	Large	5%	13%	-6%	0%

Carrier Type	Size	Total Miles	Empty Miles	WFM	Containers
FTL	Small	-21%	-24%	63%	-31%
FTL	Average	-5%	-32%	20%	4%

7.3.3 Impact of the Network Provided by Carrier Type

In addition to an individual carrier's size, the amount of the network that comes from a particular carrier type had an impact on the overall collaborative benefit. A collaborative partnership that contained both FTL and LTL carriers was defined as being either an LTL dominant partnership, an FTL dominant partnership, or an equal partnership. In an LTL dominant partnership, the collaborative network was comprised of predominately LTL loads, nodes, and containers. In the case of two partners, this would mean the LTL carrier was much larger than the FTL carrier. In the case of two or more partners, it would indicate that the sum of all loads, nodes, and containers provided by LTL carriers substantially out numbered the sum of the loads, nodes, and containers provided by FTL carriers. When a partnership only contained a single carrier type it was designated as an equal partnership or a dominant partnership. In order to be an equal partnership, all involved carriers had to be approximately the same size. In a dominant partnership, at least one partner was substantially larger than the other involved carriers. Tables 7.17, 7.18, and 7.19 provide the results of each partnership type and network make up based on carrier type for the 2-TSP, 3-TSP, and 4-TSP partnerships.

2-TSP Collaboration

Consistent with earlier results, on average, the most substantial benefit was provided when two equal sized FTL carriers were collaborating. Again as previously found, the partnerships with two LTL carriers provided the least substantial benefits. The smallest reduction in miles and empty miles was provided by a dominant LTL carrier partnership while the smallest increase in weighted full miles and the smallest reduction in containers used was provided by an equal LTL carrier partnership. Overall, more benefit was obtained from the presence of an FTL carrier as the predominate or equal provider of loads, nodes, and containers for a two carrier partnership.

Table 7.17: Benefit by Carrier Type Size: 2-TSP Overview

Partnership Type	Description	Total Miles	Empty Miles	WFM	Containers
FTL and LTL	LTL Dominant	-9%	-13%	7%	-14%
FTL and LTL	FTL Dominant	-15%	-30%	15%	-7%
FTL and LTL	Equal	-20%	-42%	16%	-14%
Two LTL	Equal	-6%	-15%	11%	-5%
Two LTL	Dominant	-8%	-11%	9%	-8%
Two FTL	Equal	-22%	-40%	35%	-13%
Two FTL	Dominant	-14%	-30%	17%	-12%

3-TSP Collaboration

The results again suggest that the addition of a third partner helped the presence of an LTL carrier to provide more benefit overall in each scenario it was involved in. This was best observed by a network that was LTL dominant providing the most benefit to a three carrier partnership that contained both an FTL carrier and an LTL carrier. Previously, the most benefit to a partnership containing both an FTL and an LTL carrier was provided by an FTL dominate network. Despite the increase in benefit seen by the presence of an LTL carrier, the most benefit overall was still obtained by an FTL carrier only network where each carrier provided roughly the same number of loads, nodes, and containers.

Table 7.18: Benefit by Available Carrier Type Size: 3-TSP Overview

Partnership Type	Description	Total Miles	Empty Miles	WFM	Containers
FTL and Two LTL	LTL Dominant	-20%	-39%	28%	-9%
FTL and Two LTL	Equal	-19%	-35%	6%	-12%
Two FTL and LTL	FTL Dominant	-18%	-36%	18%	-16%
Two FTL and LTL	Equal	-15%	-45%	13%	0%
Three LTL	Equal	-8%	-12%	6%	-15%
Three LTL	Dominant	-9%	-11%	4%	-6%
Three FTL	Equal	-24%	-50%	30%	-16%

4-TSP Collaboration

The results suggest that a four FTL carrier network still provided the most benefit. However, the gap between a network being FTL dominate versus LTL dominate decreased yet again. A four carrier partnership containing at least one FTL carrier and at least one LTL carrier now provided roughly the same benefit as an FTL carrier only network even if it was LTL dominate. Furthermore, the benefits possible from a partnership containing only LTL carriers rose enough to nearly equal the benefits provided by a 4-TSP partnership that contained both FTL and LTL carriers.

Table 7.19: Benefit by Available Carrier Type Size: 4-TSP Overview

Partnership Type	Description	Total Miles	Empty Miles	WFM	Containers
FTL and Three LTL	LTL Dominant	-27%	-42%	29%	-26%
FTL and Three LTL	Equal	-27%	-54%	39%	-19%
Two FTL and Two LTL	Equal	-17%	-45%	24%	-15%
Three FTL and LTL	FTL Dominant	-24%	-50%	29%	-14%
Four LTL	Equal	-16%	-29%	17%	-14%
Four LTL	Dominant	-16%	-14%	8%	-19%
Four FTL	Equal	-28%	-55%	45%	-21%

7.3.4 Impact of Relative Carrier Size

Studying the impact that the relative size of an individual carrier had on collaborative benefit provides additional insight into the collaboration between FTL and LTL carriers.

The relative size of a carrier in a partnership was determined by comparing an individual carrier to each of the other carriers it collaborated with in that particular partnership. For example, although an FTL carrier was of average size overall, when it collaborated as the largest of all carriers in a partnership, different benefits were observed from the same average sized FTL carrier operating in a partnership where it was smaller than all other carriers. The effect of these differences were evaluated for both LTL carriers and FTL carriers in the 2-TSP, 3-TSP, and 4-TSP partnerships separately.

FTL Carriers in a 2-TSP Partnership

In the situation where an FTL carrier was larger than its partner in a 2-TSP partnership, the larger FTL carrier only saw benefit when collaborating with a smaller LTL carrier. When an FTL carrier collaborated with another smaller FTL, the larger FTL actually increased miles, empty miles, and containers while reducing its percentage of weighted full miles. In the situation where, an FTL carrier was roughly the same size as its partner, benefits were found for the FTL carrier by collaborating with both a similar sized LTL carrier and an similar sized FTL carrier. However, collaborating with an LTL carrier consistently provided the FTL carrier with more benefit. When the FTL carrier was the smaller of the two partners in a 2-TSP partnership, the most benefit in all four metrics was found by collaborating with a larger FTL carrier. In this case, the smaller FTL carrier reduced miles by 36%, reduced empty miles by 56%, increased its percentage of weighted full miles by 48%, and reduced the number of containers it used by 36%. Despite the benefits being less, an FTL carrier collaborating with a larger LTL carrier reduced miles by 18%, empty miles by 26%, and containers by 11% while still increasing its percentage of weighted full miles by 17%.

LTL Carriers in a 2-TSP Partnership

In situations where an LTL carrier was the largest carrier, very little benefit was seen by the larger LTL when working with a smaller LTL carrier. When collaborating with a smaller

FTL carrier, the LTL carrier actually increased miles and empty miles while decreasing its percentage of weighted full miles. The only benefit seen by the larger LTL carrier was an under 10% reduction in the number of containers used. In the situation where an LTL carrier collaborated with another carrier of similar size, minimal benefit was again seen by the LTL carrier. When working with another LTL carrier of similar size, an LTL carrier performed slightly worse on each metric and while working with a similar sized FTL carrier, an LTL carrier performed slightly better on each metric.

In the situation where an LTL carrier was the smallest carrier, drastically different benefits were observed from collaborating with a larger LTL carrier versus a larger FTL carrier. When collaborating with a larger LTL carrier, the small LTL carrier obtained the most benefit of an LTL carrier in any situation by reducing miles by 13%, empty miles by 31% and increasing the percentage of weighted full miles by 17%. However, when collaborating with a larger FTL carrier, the smaller LTL carrier had substantially worse performance. In this case, the smaller LTL carrier increased miles by 50% and empty miles by 124% while providing no change in the percentage of weighted full miles. The only benefit was that the LTL carrier was able to use 27% fewer containers overall.

FTL Carriers in a 3-TSP Partnership

In a three partner setting when the largest carrier was an FTL carrier, the largest FTL carrier received the most benefit when working with a slightly smaller FTL carrier and a much smaller LTL carrier. In this partnership type, the largest FTL carrier reduced miles by 34%, empty miles by 43%, and containers by 33% while maintaining the same percentage of weighted full miles. When an FTL carrier was the largest carrier collaborating with a slightly smaller FTL carrier and a much smaller FTL carrier or with a much smaller FTL and a much smaller LTL carrier, the largest FTL carrier found a slight reduction or no change in miles and containers and a slight increase in the percentage of weighted full miles. However, in both of these cases the largest FTL carrier was able to reduce empty miles by around 30%. When an FTL carrier was the largest carrier collaborating with two much smaller LTL carriers, the large FTL carrier did not find any benefit. In this case, the FTL carrier increased miles by 14%, empty miles by 41%, and containers by 20% while reducing weighted full miles by 8%.

In a situation where an FTL carrier was neither the smaller nor the largest carrier in a three partner setting, collaborative results varied based on the other partners. Substantial reductions in miles, empty miles, and containers were found for an FTL carrier in three cases. These cases were when an FTL carrier was collaborating with a similar sized LTL carrier and a smaller LTL carrier, with a similar sized LTL carrier and a smaller FTL carrier, or with a similar sized LTL carrier and a much larger FTL carrier. The results indicated an average 30% reduction in miles, a 40% reduction in empty miles, a 59% increase in weighted full miles, and a 19% reduction in the number of containers used.

Conversely, a slight increase in miles and containers and a slight decreased in the percentage of weighted full miles was found for an FTL carrier when collaborating with a larger FTL carrier and a smaller FTL carrier. However, the FTL carrier in this scenario was still able to reduce empty miles by 16%. Similarly, when an FTL carrier collaborated with a larger FTL carrier and a smaller LTL carrier, the middle FTL carrier found a slight increase in miles, empty miles, and containers, but still benefited with a 40% increase in its percentage of weighted full miles.

Substantial benefit was provided to an FTL carrier when it operated as the smallest carrier in a partnership. When collaborating with two larger FTL carriers, the smallest FTL carrier had huge benefit by reducing miles by 51%, empty miles by 72%, and containers by 51%. Additionally, the percentage of weighted full miles increased by 111%. When operating with a larger FTL carrier and a larger LTL carrier, similar values were found for the small FTL carrier in the reduction of miles, empty miles, and containers. However, the small FTL carrier in this case reduced their overall percentage of weighted full miles by nearly 10%.

LTL Carriers in a 3-TSP Partnership

In a three partner setting where an LTL carrier was the largest carrier, substantial benefit for the LTL carrier was only found when collaborating with a smaller FTL carrier and a smaller LTL carrier. In this case, the large LTL carrier reduced miles by 59%, empty miles by 73%, increased its percentage of weighted full miles by 30%, and reduced the number of containers used by 29%. No major difference in any of the four metrics was found for an LTL carrier in all other 3-TSP partnerships where the LTL carrier was the largest carrier.

In the situation where an LTL was neither the largest carrier nor the smallest carrier, the benefit to the LTL carrier tended to vary largely based on the other partners. Substantial benefit was identified for an LTL carrier when they collaborated in following four cases. The cases were when an LTL carrier collaborated with either a larger LTL carrier and a smaller FTL carrier, a larger FTL carrier and a similar sized FTL carrier, a larger FTL carrier and a similar sized LTL carrier, or a larger LTL carrier and a similar sized LTL carrier. In these cases, miles were reduced by between 33% and 61%, empty miles were reduced by between 55% and 78%, and containers were reduced by between 0% and 42%. Weighted full miles were less steady with an LTL carrier collaborating with a larger FTL and a similar sized LTL carrier decreasing the percentage of weighted full miles by 5%. However, the remaining three partnerships still provided the LTL carrier between a 13% and 63% increase in the percentage of miles that were weighted full miles.

Not all partnerships with the LTL carrier being neither the largest carrier nor the smaller carrier provided the LTL carrier substantial benefit. Minimal changes were seen in four scenarios: when an LTL carrier collaborated with a larger LTL carrier and a smaller LTL carrier, a similar sized FTL carrier and a smaller LTL carrier, two similar sized LTL carriers, or a similar sized LTL carrier and a smaller LTL carrier. All four scenarios provided a 10% or less reduction in miles, a 1% increase to 27% decrease in empty miles, a 6% decrease to an 11% increase in the percentage of weighted full miles, and a 0% change to 17% reduction in the number of containers used. In the final case of an LTL carrier collaborating with a similar sized FTL carrier and a smaller FTL carrier, the LTL carrier increased miles by 12% and containers by 20% while reducing empty miles by 24% and the percentage of weighted full miles by 2%.

There were three distinct outcomes for an LTL carrier being the smallest partner in a 3-TSP partnership. First, the small LTL carrier received a less than 10% reduction in miles and a less than 20% reduction in empty miles while increasing its percentage of weighted full miles by up to 37%. This happened when a small LTL carrier collaborated with two larger LTL carriers and resulted in no change in the number of containers used. The second outcome was provided by the scenario where the smallest LTL carrier collaborated with a larger LTL carrier and a larger FTL carrier. In this scenario, the smaller LTL carrier saw a 19% increase in miles, a 20% increase in the number of containers used, and a 2% reduction in the percentage of weighted full miles. However, the carrier still reduced empty miles by 24%. Finally, two scenarios provided a reduction of up to 47% of miles, 75% of empty miles, and 33% of containers. This was when the smallest LTL carrier collaborated with much larger FTL carrier and a slightly larger LTL carrier or with a much larger FTL carrier and a sightly larger FTL carrier. However, the carrier's percentage of weighted full miles fell between 4% and 10%.

FTL Carriers in a 4-TSP Partnership

In the situation where an FTL carrier was the largest carrier involved in a 4-TSP partnership, a decrease in empty miles and an increase in the percentage of weighted full miles was always found. These situations provided at least a 23% reduction in empty miles and at least a 10% increase in the percentage of weighted full miles. However, when an FTL carrier was the largest carrier in a setting with one much smaller FTL carrier, one much smaller LTL carrier, and one slightly smaller LTL carrier, the large FTL carrier was able to additionally reduce total miles by 57% and containers by 40%. In contrast when the largest carrier was an FTL carrier collaborating with two slightly smaller LTL carriers and one much smaller

FTL carrier, the large FTL carrier actually had to increase total miles by 6% and could not reduce the number of containers used at all.

The results suggest very similar benefits for an FTL carrier collaborating with the following partners: an FTL carrier with one similar sized FTL carrier and two smaller FTL carriers, an FTL carrier with two similar sized FTL carriers and one smaller FTL carrier, an FTL carrier with one similar sized LTL carrier, one smaller LTL carrier, and one smaller FTL carrier, an FTL carrier with two similar sized FTL carriers and one similar sized LTL carrier, and finally an FTL carrier with three similar sized LTL carriers. In these cases, an average reduction of 20% in total miles, a 35% reduction in empty miles, a 35% increase in the percentage of weighted full miles, and a 14% reduction in the number of containers used was identified for the FTL carrier. Huge benefit was also found for an FTL carrier when collaborating with two larger FTL carriers and either one similar sized FTL or LTL carrier. The results suggest a minimum reduction of 34% in miles, a 57% reduction in empty miles, a 27% reduction of the number of containers used, and a 105% increase in the percentage of weighted full miles for the FTL carrier. Furthermore, very similar results were found for the scenario where an FTL carrier collaborated with a similar sized FTL carrier, a similar sized LTL carrier, and a smaller FTL carrier and the scenario where an FTL carrier collaborated with a similar sized FTL carrier, a smaller FTL carrier, and a smaller LTL carrier. For these partnerships, the results suggest an approximate 5% decrease in miles, a 40% decrease in empty miles, a 30% increase in the percentage of weighted full miles, and a 10% increase in the number of containers used by the FTL carrier.

Similar trends were identified in the benefits to an FTL carrier when collaborating in the following three scenarios. These three scenarios were when an FTL carrier collaborated with a similar sized FTL carrier and two smaller LTL carriers, with two similar sized LTL carriers and a smaller LTL carrier, or with a larger LTL carrier, a larger FTL carrier, and a similar sized LTL carrier. In all three cases, an increase in miles of at least 22% was seen for the FTL carrier as well as either no change or an increase of 75% in the number of containers used. However, in all three cases the percentage of weighted full miles increased by a minimum of

16% and a maximum of 75%. The impact on empty miles was split with the FTL carrier increasing empty miles by nearly 50% when working with two similar sized LTL carriers and one smaller LTL carrier, the FTL carrier seeing no change in empty miles when working with a larger LTL carrier, a larger FTL carrier, and a similar sized LTL carrier, and the FTL carrier seeing a 35% decrease in empty miles when working with one similar sized FTL and two smaller LTL carriers.

Finally, the benefit to an FTL carrier when they were the smallest carrier in a 4-TSP partnership was found to be dependent on the other carriers in the partnership. In a situation where the smallest FTL carrier collaborated with one larger LTL carrier and two larger FTL carriers or with three larger FTL carriers, the smallest FTL carrier found a substantial amount of benefit with a minimum reduction of 27% in miles, 45% in empty miles, and 33% in the number of containers used while increasing the percentage of weighted full miles by at least 38%. However, when collaborating with a larger FTL carrier and two larger LTL carriers, the smaller FTL carrier increased miles by 61% and empty miles by 126% while slightly decreasing the percentage of weighted full miles.

LTL Carriers in a 4-TSP Partnership

The only situation where an LTL carrier was the largest carrier in a 4-TSP partnership was when it was collaborating with three smaller LTL carriers. In this case, the largest LTL carrier increased miles by 5% and empty miles by 13% while the number of containers used remained unchanged and the percentage of weighted full miles fell by 6%.

There were many 4-TSP partnerships where an LTL carrier was neither the largest carrier nor the smallest carrier involved. Many of these partnerships resulted in similar benefits to the LTL carrier. These partnerships included where an LTL carrier collaborated with the following partners:

- one similar sized FTL carrier and two larger FTL carriers,
- one similar sized FTL carrier and two smaller LTL carriers,

- one similar sized FTL carrier, one similar sized LTL carrier, and one smaller LTL carrier,
- two similar sized FTL carriers and a smaller FTL carrier,
- two similar sized LTL carriers and one similar sized FTL carrier,
- one larger FTL carrier, one similar sized LTL carrier, and one smaller LTL carrier,
- one similar sized LTL carrier and two smaller LTL carriers,
- one similar sized FTL carrier, one smaller LTL carrier, and one smaller FTL carrier,
- one similar sized LTL carrier, a larger FTL carrier, and a larger LTL carrier,
- one larger LTL carrier and two similar sized LTL carriers,
- two similar sized LTL carriers and one smaller LTL carrier, and
- one larger FTL carrier, one larger LTL carrier, and one similar sized FTL carrier.

Across all of these partnerships the LTL carrier reduced overall miles by 37%, empty miles by 58%, and containers by 27% while they increased their percentage of weighted full miles by 36%.

Another set of partnerships that resulted in similar benefits for an LTL carrier came from the LTL carrier collaborating with one larger FTL carrier, one smaller FTL carrier, and one smaller LTL carrier, two larger LTL carriers and one similar sized LTL carrier, one larger FTL carrier, one larger LTL carrier, and one similar sized FTL carrier, or one larger FTL carrier and two larger LTL carriers. However, this time the benefits were far less. Overall, there was an average increase of 5% in miles, an average decrease of 19% in empty miles, an average increase of 24% in the percentage of weighted full miles, and no change to the number of containers used for the LTL carrier.

The remaining collaborative scenario for a 4-TSP partnership was when an LTL carrier was the smallest involved carrier. This was a partnership with the LTL carrier and three larger LTL carriers. In this scenario, the performance of the smallest LTL carrier was substantially decreased. Here the small LTL carrier increased miles by 60%, empty miles by 66% and the number of containers it used by 50% all while decreasing its percentage of weighted full miles by 2%.

7.3.5 Impact of Changes in Average Network Distance

The final insights into collaboration were provided by examining what happens to the average distance between locations for each carrier in their own network versus the network created by all available locations in the collaborative network. For this research, average network distance was defined as the average distance between locations or nodes in a network. For an individual carrier, the average network distance was the average number of miles between each of their locations. In a collaborative setting, the average network distance was the average number of miles between the combined locations of all involved carriers. Thus, in a collaborative setting the average network distance for each involved carrier was the same.

The change in the average network distance of an individual carrier compared to the average network distance of each collaborative network was used to highlight the impact of carriers collaborating based on their geographic location. When carriers in the same geographic area were collaborating, the change in the average network distance of a carrier compared to the average network distance of the collaborative partnership tended to either stay very similar or decrease. On the contrary, when carriers tried to collaborate with other carriers that were from different geographic regions, the change in average network distance tended to be much greater. The change in average network distance for an individual carrier in each collaborative partnership was measured as the percentage change in miles that resulted from subtracting the average network distance of the collaborative setting from the carrier's starting average network distance.

Tables 7.20, 7.21, and 7.22 show the benefit that an individual carrier received from a collaborative partnership based on the difference between their original average network distance and the collaborative partnership's average network distance. This change in average network distance was lumped into five scenarios based on when the change in average network distance from collaboration resulted in:

- a 25% or more decrease in the average network distance for the carrier,
- between a 25% decrease and a 10% decrease in the average network distance for the carrier,
- between a 10% increase and a 10% decrease in the average network distance for the carrier,
- between a 10% increase and a 25% increase in the average network distance for the carrier,
- or a 25% or more increase in the average network distance for the carrier.

Table 7.20: Benefit by Change in Average Network Distance: 2-TSP Overview

Change in Avg Network Distance	Total Miles	Empty Miles	WFM	Containers
-25% or more	-36%	-58%	62%	-42%
-10% to -25%	-25%	-38%	46%	-13%
10% to -10%	-15%	-29%	16%	-13%
10% to 25%	2%	-4%	-11%	4%
25% or more	33%	63%	-4%	3%

Table 7.21: Benefit by Change in Average Network Distance: 3-TSP Overview

Change in Avg Network Distance	Total Miles	Empty Miles	WFM	Containers
-25% or more	-62%	-81%	42%	-58%
-10% to -25%	-32%	-47%	26%	-25%
10% to -10%	-18%	-30%	19%	-14%
10% to 25%	-7%	-23%	14%	4%
25% or more	1%	-13%	8%	9%

Table 7.22: Benefit by Change in Average Network Distance: 4-TSP Overview

Change in Avg Network Distance	Total Miles	Empty Miles	WFM	Containers
-25% or more	-67%	-81%	83%	-67%
-10% to -25%	-32%	-54%	66%	-31%
10% to -10%	-15%	-28%	34%	-16%
10% to 25%	-21%	-44%	17%	-13%
25% or more	-1%	-12%	13%	9%

Regardless of the number of partners involved in the collaboration, the results suggest that if a partnership reduces the average network distance for a carrier by any amount a substantial reduction in miles, empty miles, and containers and a substantial increase in the percentage of weighted full miles was possible. Additionally, the results suggest that the more a partnership reduced the average network distance for a carrier, the more benefit that was possible across the board.

This was only not the case in two specific instances. Instance 1 was in a 2-TSP partnership the number of containers used by a TSP stayed approximately the same if their average network distance increased by between 10% and 25% or if their average network distance changed by 25% or more. Instance 2 was with 4-TSP partnerships seeing an increase in empty miles between carriers with an average network change of between 10% and -10% and carriers with an average network change between 10% and 25%. Both of these instances can be attributed to neither containers nor empty miles being explicitly considered in the objective function.

7.3.6 Summary and Conclusions

In this chapter, the benefits of TSP collaboration were quantified using the TSP-CP to solve industry representative data sets. The representative data sets were derived from actual freight data provided by a freight pooling company that manages collaboration between TSPs. For this research, 68 of the carriers included in the industry data were used to build 67 collaborative partnerships of two, three, or four carriers. Out of the 68 carriers, 16 were FTL carriers and 42 were LTL carriers. These carriers were chosen to create collaborative partnerships in order to gain additional insights into the effect that partnership construction had on the overall benefit of collaboration as well as the benefits obtained by individual carriers.

Overall, collaboration was able to substantially reduce the number of total miles required to delivery all loads, the number of those miles that were completely empty, and the number of containers needed to deliver all loads. Additionally, there was a substantial increase in the percentage of miles that were weighted full miles. Although benefit was found for partnerships containing two, three, or four carriers, the results suggested that as more partners became involved in collaboration, the greater the benefit that was possible. Across all partnerships, it was found that collaboration between only FTL carriers provided the most benefit while the collaboration between only LTL carriers provided the least benefit. When FTL and LTL carriers collaborated together, the benefit fell somewhere in between. Regardless of the number of partners involved, an FTL dominant network tended to provide more benefit. However, as the number partners increased, the benefit provided by an LTL dominant network approached the benefit provided by an FTL dominant network.

Collaborative insights were also gained by examining the benefits that individual carriers received based on the partnerships they were collaborating in. The results first suggested that whenever a collaborative partnership reduced the average network distance of an individual carrier, the carrier achieved a substantial benefit from collaboration. Additionally, the more that the average network distance was decreased, the more benefit the carrier was likely to see. The results also suggested that in the 2-TSP partnerships, an FTL carrier received substantially more benefit than an LTL carrier. However, as the the number of partners increased in the 3-TSP partnerships, FTL and LTL carriers received roughly the same benefit. When adding the fourth partner to create a 4-TSP partnership, the benefits shifted and an LTL carrier benefited substantially more than an FTL carrier. When including the size of each carrier into the analysis, a small FTL carrier received the most benefit from collaboration regardless of the number of partners involved. When examining the benefits to an LTL carrier based on its size, the results suggested that in a 2-TSP partnership a large LTL carrier benefited the most. However, as the number of partners increased to the 3-TSP and 4-TSP partnerships, the results suggested that a smaller LTL carrier benefited more.

Regardless of its independent size, an FTL carrier saw the most benefit in a 2-TSP partnership by collaborating with either a similar sized LTL carrier or a larger FTL carrier and the least benefit by collaborating with a smaller FTL carrier. In a 3-TSP partnership, an FTL carrier benefited the most when collaborating with two larger FTL carriers or with a larger FTL carrier and a larger LTL carrier. The FTL carrier was worst off when collaborating with a larger LTL carrier and a smaller LTL carrier. Finally, in a 4-TSP partnership, an FTL carrier by far received the most benefit by collaborating with a one similar sized FTL carrier and two larger FTL carriers. However, there were two partnerships which substantially hurt the performance metrics for an FTL carrier. These two cases were when the FTL carrier collaborated with a similar sized FTL carrier, a larger FTL carrier, and a larger LTL carrier and when the FTL carrier collaborated with a two similar sized LTL carriers and one smaller FTL carrier.

An LTL carrier saw the most benefit in a 2-TSP partnership when collaborating with a larger LTL carrier and the least benefit when working with a larger FTL carrier. In a 3-TSP partnership, an LTL carrier was provided the most benefit from working with a similar sized LTL carrier and a larger FTL or LTL carrier and the least benefit from working with a larger LTL carrier and a larger FTL carrier. Finally, in a 4-TSP partnership an LTL carrier received the most benefit from working with two smaller LTL carriers and one similar sized FTL carrier and the least benefit from working with three larger LTL carriers.

Chapter 8

Conclusions and Areas of Future Research

Truck-based freight transportation in the United States is expected to remain an integral part of the economy for the foreseeable future. Despite its continued importance, the industry remains socially, economically, and environmentally unsustainable. This research investigated collaboration between transportation service providers (TSP) as a potential approach for moving towards a more sustainable trucking industry in the future. Specifically, this research studied the collaboration among full truckload (FTL) and less-than-truckload (LTL) carriers by introducing the Transportation Service Provider Collaboration Problem (TSP-CP). The TSP-CP provides optimal freight routing and consolidation decisions for the combined freight of a set of collaborating FTL and LTL carriers utilizing their combined delivery resources and delivery network. The TSP-CP was modeled as a time-expanded pickup and delivery problem with transshipment (t-PDPTWT) that includes multiple transfer opportunities and transfer locations while not restricting delivery resources to a single route or a limited number of visits to any particular location.

8.1 Conclusions

An integrated mathematical model was developed for the TSP-CP, however because that the integrated model became intractable for large problem instances, two adaptive large neighborhood search (ALNS) heuristics were introduced as well: the fixed-optimization ALNS and the heuristic insertion ALNS. The same framework was used for the heuristics, but they differed in their parameters, implementation, and repair method ideology. The fixed-optimization ALNS employs powerful and computationally expensive simultaneous repair methods by solving an optimization model. Because large moves are made, the fixed-optimization ALNS does not rely on a meta-heuristic to guide the search. In contrast, the heuristic insertion ALNS relies heavily on a simulated annealing meta-heuristic to keep the search from becoming stuck in local minima. The heuristic insertion ALNS employs less powerful and less computationally expensive repair methods based on simple sequential insertions heuristics.

The performance of the proposed solution approaches was tested on randomly generated test instances as benchmark data instances were not readily available. The integrated mathematical model, the fixed-optimization ALNS, and the heuristic insertion ALNS all provided freight consolidation and routing solutions for a set of collaborating TSPs. The results identified a substantial reduction in the number of total and empty miles traveled by containers as well as a substantial increase in the percentages of miles that were weighted full miles. When comparing the solution gap of each ALNS, the fixed-optimization ALNS had an average solution gap of 0.31% compared to the heuristic insertion ALNS which had an average solution gap of 2.96%. Although it had a slightly higher solution gap on average, the heuristic insertion ALNS was selected as the preferred solution approach for use with industry representative data as it most reliably provided TSP-CP solutions. Based on the extensive computational studies, the integrated mathematical model grew exponentially as problem size increased and the fixed-optimization ALNS, which relied heavily on solving mathematical models during each iteration, had difficulty in completing a satisfactory number of iterations due to time and memory constraints.

In order demonstrate the impact that collaboration can have on the efficiency of FTL and LTL carriers, daily data sets for individual FTL and LTL carriers were derived from industry data sets provided by a freight pooling company. Using these data sets, collaborative partnerships of up to four carriers were created based on the included carriers size, type, and geographic location. Freight routing and consolidation solutions to these partnerships as well as the individual carriers were provided by the heuristic insertion ALNS and were evaluated to determine the benefits of collaboration and to gather additional insights into the collaboration between FTL and LTL carriers.

The computational results suggested that the TSP-CP provided substantial reductions in the total distance required to deliver all loads, in the number miles that were traveled completely empty, and to the number of containers required for delivery to a group of collaborating TSPs over what they could do individually. Additionally, the TPS-CP increased the percentage of weighted full miles on average. Although the results showed that these improvements were possible with just two collaboration TSPs, the more partners that became involved in the collaborative effort, the greater the benefit was across the board. By analyzing the solutions from the heuristic insertion ALNS, many new insights into the collaboration between FTL and LTL carriers have been quantified and highlighted. These insights included the effect that an individual carrier's type and size had on the amount of benefit received to each carrier. Finally, the results highlighted the importance of building collaborative partnerships by considering the geographic location of each carrier.

8.2 Areas of Future Research

One natural area of future research in relation to the TSP-CP modeling approach would be to explore the effects of different objective function components, such as explicitly considering the number of containers used or the number of empty miles. This extension may help to provide additional insights into collaboration and may potentially enhance solution approach performance.

Related to modeling, it may be beneficial to compare the discrete time model used here to a continuous time modeling approach. A continuous time model would increase the granularity lost with discrete time periods and may serve as a step in transitioning the TSP-CP towards a more operational model. However, a continuous time model may increase computational requirements due to inherent sub-tour issues with containers and loads that were avoided with the discrete time approach. Thus, this comparison warrants additional research.

Another potential extension of this research is to improve the performance of the fixed-optimization ALNS repair methods. The underling mathematical models for the current fixed-optimization ALNS could be explored for additional improvements or additional models and methods could be introduced based on the general fixed-optimization approach.

Properly allocating the benefits obtained as a result of using the TSP-CP is an area of significant interest. Ideas such as cooperative game theory, contracts, and incentives could be the key to collaboration becoming more attractive to TSP by allocating benefits fairly, such that all involved parties benefit an appropriate amount. Finally, in this research it was assumed that all partnerships operated under centralize control. However, as previously stated, this may not be the most realistic assumption for collaboration in practice due the amount of trust and information sharing required. As the research into decentralized control mechanisms for a set of collaborating TSP progresses, determining the impact that these mechanisms have on the benefits provided by the TSP-CP would be of interest.

References

[1] The logistics and transportation industry in the united states. http://selectusa.commerce.gov/industry-snapshots/logistics-and-transportation-industry-united-states.html. Accessed: May 12, 2015.

[2] MHI (2014). The U.S. roadmap for material handling and logistics.

[3] Bailey, E., A. Unnikrishnan, and D. Lin (2011). Models for minimizing backhual costs through freight collaboration. *Transportation Research Record: Journal of the Transportation Research Board*, 2224(1):51–60.

[4] Berger, S. and C. Bierwirth (2010). Solutions to the request reassignment problem in collaborative carrier networks. *Transportation Research Part E: Logistics and Transportation Review*, 46(5):627–638.

[5] Berman, J. (2013). Driver shortage issues continue to raise questions but not provide enough answers. Logistics Management.

[6] Bjornfot, A. and L. Torjussen (2012). Extent and effect of horizontal supply chain collaboration among construction SME. *Journal of Engineering, Project, and Production Management*, 2(1):47–55.

[7] Cao, M. and Q. Zhang (2011). Supply chain collaboration: Impact on collaborative advantage and firm performance. *Journal of Operations Management*, 29(3):163–180.

[8] Chou, Y.C., Y.H. Chen, and H. Chen (2014). Pickup and delivery routing with hub transshipment across flexible time periods for improving dual objectives on workload and waiting time. *Transportation Research Part E: Logistics and Transportation Review*, 61:98–114.

[9] Codato, G. and M. Fischetti (2004). Combinatorial benders cuts. *Integer Programming and Combinatorial Optimization*, pages 178–195. Springer Berlin Heidelberg.

[10] Cortes, C.E., M. Matamala, and C. Contardo (2010). The pickup and delivery problem with transfers: formulation and a branch-and-cut solution method. *European Journal of Operations Research*, 200(3):711–724.

[11] Costello, B. (2013). Reports, trends, and statistics: driver shortage. American Trucking Association (ATA).

[12] Costello, B. (2013). Reports, trends, and statistics: industry data. American Trucking Association (ATA).

[13] Cruijssen, F.C.A.M. (2006). Horizontal cooperation in transport and logistics. Tilburg: CentER, Center for Economic Research.

[14] Dai, B. and H. Chen (2009). Mathematical modeling and solution approach for collaborative logistics in less-than-truckload (LTL) transportation. *Computers and Industrial Engineering International Conference*, pages 767–772.

[15] Dondo, R., C.A. Mendez, and J. Cerda (2009). The supply-chain pick-up and delivery problem with transshipment. *Computer Aided Process Engineering*, 26:1009–1014.

[16] Environmental Protection Agency (EPA) (2013). Inventory of u.s. greenhouse gas emissions and sinks: 1990-2011.

[17] Ergun, O., G. Kuyzu, and M. Savelsbergh (2007). Reducing truckload transportation costs through collaboration. *Transportation Science*, 41(2):206–221.

[18] Grunert, T. and H. Sebastian (2000). Planning models for long-haul operations of postal and express shipment companies. *European Journal of Operations Research*, 122(2):289–309.

[19] Hernandez, S. and S. Peeta (2010). Less-than-truckload static single-carrier collaboration problem. Transportation research Board 89th Annual Meeting (No. 10-2574).

[20] Hernandez, S. S., Peeta, and G. Kalafatas (2011). A less-than-truckload carrier collaboration planning problem under dynamic capacities. *Transportation Research Part E: Logistics and Transportation Review*, 47(6):933–946.

[21] Juan, A.A., J. Faulin, E. Perez-Bernabeu, and N. Jozefowiez (2014). Horizontal cooperation in vehicle routing problems with backhualing and environmental criteria. *Procedia-Social and Behavioral Sciences*, 111:1133–1141.

[22] Kerivin, H.L.M., M. Lacroix, A.R. Mahjoub, and A. Quilliot (2008). The splittable pickup and delivery problem with reloads. *European Journal of Industrial Engineering*, 2(2):112–133.

[23] Krajewska, M.A., H. Kopfer, G. Laporte, S. Ropke, and G. Zaccour (2008). Horizontal cooperation among freight carriers: request allocation and profit sharing. *Journal of the Operational Research Society*, 59(11):1483–1491.

[24] Lehox, N., J.F. Audy, D.A. Sophie, and M. Ronnqvist (2009). Issues and experiences in logistics collaboration. *Leveraging Knowledge for Innovation in Collaborative Networks*, pages 69–76. Springer Berlin Heidelberg.

[25] Mason, R., C. Lalwani, and R. Boughton (2007). Combining vertical and horizontal collaboration for transport optimisation. *Supply Chain Management*, 12(3):187–199.

[26] Masson, R., F. Lehuede, and O. Peton (2013). An adaptive large neighborhood search for the pickup and delivery problem with transfers. *Transportation Science*, 47(3):344–355.

[27] Masson, R., S. Ropke, F. Lehuede, and O. Peton (2014). A branch-and-cut-and-price approach for the pickup and delivery problem with shuttle routes. *European Journal of Operational Research*, 236(3):849–862.

[28] McKinnon, A. (2010). European freight transport statistics: limitations, misinterpretations, and aspirations.

[29] Mitrovic-Minic, S. and G. Laporte (2006). The pickup and delivery problem with time windows and transshipment. *Information Systems and Operational Research*, 44(3):217–228.

[30] Nadarajah, S. and J.H. Bookbinder (2013). Less-than-truckload carrier collaboration problem: modeling framework and solution approach. *Journal of Heuristics*, 19(6):917–942.

[31] Nakao, Y. and H Nagamochi (2008). Worst case analysis for pickup and delivery problems with transfer. *IEICE Transactions on Fundamentals of Electronics, Communication, and Computer Sciences*, 91(9):2328–2334.

[32] Ozener, O.O., O. Ergun, and M. Savelsbergh (2011). Lane-exchange mechanisms for truckload carrier collaboration. *Transportation Science*, 45(1):1–17.

[33] Petersen H.L. and S. Ropke. The pickup and delivery problem with cross-docking opportunity. *Computational Logistics*, pages 101–113. Springer Berlin Heidelberg.

[34] Pisinger, D. and S. Ropke (2007). A general heuristic for vehicle routing problems. *Computers and Operations Research*, 34(8):2403–2435.

[35] Qu, Y. and J.F. Bard (2012). A GRASP with adaptive large neighborhood search for pickup and delivery problems with transshipment. *Computers and Operations Research*, 39(10):2439–2456.

[36] Rais, A., F. Alvelos, and S.M. Carvalho (2014). New mixed-integer programming model for the pickup-and-delivery problem with transshipment. *European Journal of Operations Research*, 235(3):530–539.

[37] Savelsbergh, M. and M. Sol (1995). The general pickup and delivery problem. *Transportation Science*, 29(1):17–29.

[38] Shaw, P. (1997). A new local search algorithm providing high quality solutions to vehicle routing problems. *APES Group, Dept of Computer Science, University of Strathclyde, Glasgow, Scotland, UK.*

[39] Torrey, W.F. and D. Murray (2014). An analysis of the operational costs of trucking: 2014 update.

[40] Vornhusen, B., X. Wang, and H. Kopfer (2014). Vehicle routing under consideration of transshipment in horizontal coalitions of freight carriers. *Procedia CIRP*, 19:117–122.

[41] Wang, X. and H. Kopfer (2014). Collaborative transportation planning for less-than-truckload freight. *OR Spectrum*, 36(2):357–380.

Appendix A

Additional Results

A.1 Additional Results for the 2-TSP Partnerships

This section provides any additional results for the 2-TSP partnerships from Chapter 7. Table A.1 provides the benefit to an individual carrier based on its type and the type of its partners.

Table A.1: 2-TSP Additional Results

Type	Partner	Miles	Empty Miles	WFM	Containers
FTL	FTL	-13%	-21%	30%	-10%
	LTL	-21%	-29%	40%	-19%
LTL	FTL	17%	31%	-3%	-3%
	LTL	-7%	-16%	13%	-5%

A.2 Additional Results for the 3-TSP Partnerships

This section provides any additional results for the 3-TSP partnerships from Chapter 7. Table A.2 provides the benefit to an individual carrier based on its type and the type of its partners.

Table A.2: 3-TSP Additional Results

Type	Partners	Miles	Empty Miles	WFM	Containers
FTL	FTL,FTL	-20%	-42%	22%	-16%
	FTL,LTL	-24%	-29%	27%	-6%
	LTL,LTL	-4%	-19%	31%	-2%
LTL	FTL,FTL	-20%	-51%	0%	-16%
	FTL,LTL	-25%	-46%	14%	-13%
	LTL,LTL	-10%	-14%	14%	-9%

A.3 Additional Results for the 4-TSP Partnerships

This section provides any additional results for the 4-TSP partnerships from Chapter 7. Table A.3 provides the benefit to an individual carrier based on its type and the type of its partners.

Table A.3: 4-TSP Additional Results

Type	Partners	Miles	Empty Miles	WFM	Containers
FTL	FTL,FTL,FTL	-25%	-49%	44%	-19%
	FTL,FTL,LTL	-18%	-39%	44%	-5%
	FTL,LTL,LTL	-21%	-35%	29%	-21%
	LTL,LTL,LTL	40%	43%	27%	8%
LTL	FTL,FTL,FTL	-20%	-47%	11%	-13%
	FTL,FTL,LTL	-23%	-44%	50%	-22%
	FTL,LTL,LTL	-32%	-48%	31%	-21%
	LTL,LTL,LTL	-23%	-32%	25%	-19%

Appendix B

Parameters

B.1 Global Parameters

The following parameters were used by all mathematical models and heuristics for the TSP-CP in this research.

Table B.1: General Model Parameters

Parameter	Description	Value
T	number of time periods	8 time periods
T_{max}	latest available time period	8
τ	per load handling cost	\$5 per load per time period
η	per load holding cost	\$2 per load per time period
ω_c	per mile cost	\$1.5 per mile per container
u_c	capacity of container c	1

B.2 General ALNS Parameters

The following parameters were used by both the fixed-optimization ALNS and the heuristic insertion ALNS.

Table B.2: General ALNS Parameters

Parameter	Description	Value
w_{d_1}	starting score of the random load destroy method	25
w_{d_2}	starting score of the random container destroy method	25
w_{d_3}	starting score of the Shaw destroy method	25
w_{d_4}	starting score of the penalty destroy method	25
ζ^R	weight of the distance component in the Shaw removal score	10
β^R	weight of the time window component in the Shaw removal score	1
κ^R	weight of the size component in the Shaw removal score	250
ζ^P	weight of the cost component in the Penalty removal score	1
κ^P	weight of the size component in the Penalty removal score	1000
α	adaptive score reaction factor	0.1

B.3 Fixed-Optimization ALNS Parameters

The following parameters were used by the fixed-optimization ALNS.

Table B.3: Fixed-Optimization ALNS Parameters

Parameter	Description	Value
w_{r_1}	starting score of fixed load route repair method	60
w_{r_2}	starting score of fixed load and container route repair method	40
r_{min}	minimum percentage of total loads removed by a destroy method	10%
r_{max}	maximum percentage of total loads removed by a destroy method	70%
σ_1^O	score update if new global best solution is found	10
s_{length}	adaptive segment length (iterations)	10

B.4 Heuristic Insertion ALNS Parameters

The following parameters were used by the heuristic insertion ALNS.

Table B.4: Heuristic Insertion ALNS Parameters

Parameter	Description	Value
w_{r_1}	starting score of the with transfer repair method	25
w_{r_2}	starting score of the without transfer repair method	25
w_{r_3}	starting score of the full evaluation repair method	50
w_{tps_1}	starting score of the random transfer point selection method	33
w_{tps_2}	starting score of the min distance transfer point selection method	33
w_{tps_3}	starting score of the in-use transfer point selection method	33
r_{min}	minimum percentage of total loads removed by a destroy method	10%
r_{max}	maximum percentage of total loads removed by a destroy method	20%
σ_1^H	score update if new global best solution is found	6
σ_2^H	score update if new current solution is found	4
p_{sort}	probability the removed load list is sorted by score	0.15
ζ^S	weight of the distance component in the sorting score	10
β^S	weight of the time window in the sorting score	1
κ^S	weight of the time window in the sorting score	250
c	cooling rate	0.99975
$w\%$	starting solution limit for setting T_{start}	0.15
p_{return}	probability search returns to the global best solution	0.005
$\|H'\|$	number of transfer points considered each iteration	2
s_{length}	adaptive segment length (iterations)	100