A Lagrangian Approach for The Airfreight Consolidation Problem Under Pivot-weight

by

Mohammad Alqssem Alzaeem

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Applied Science
in
Management Sciences

Waterloo, Ontario, Canada, 2018

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

International airfreight forwarders are faced with the problem of consolidating shipments for efficient transportation by airline carriers. The use of standard unit loading devices (ULDs) is a solution adopted by the airfreight industry to speed up cargo loading, increase safety, and protect cargo. We study the airfreight consolidation problem from the forwarders perspective where a decision on the number of ULDs used and the assignment of shipments to ULDs is optimized. The cost of using a ULD consists of a fixed charge and depends on the weight of the cargo it contains. A ULD is charged at an under-pivot rate if the total weight is below a threshold limit, called the pivot-weight. Additional weight is charged at the over-pivot rate. We propose a solution methodology based on Lagrangian relaxation that is capable of providing high quality solutions in reasonable computational times. Besides, a high-quality lower bound, we propose three heuristics to generate feasible solutions, all based on the solution of the subproblems. The first, takes the solution of one of the subproblems and solves a restricted version of the original problem (LagHeur). The other two heuristics are a heuristic based on solving two knapsack problems (2knap) and a best-fit greedy heuristic (bestfit). Problems with up to 100 ULDs and 1000 shipments are solved to within an average of 1%, 2%, 2% of optimality in less than 51.05s, 50.57s and 589.16s by bestfit, 2knap and LagHeur, respectively.

Acknowledgements

I would like to thank my supervisor Professor Elhedhli for his unlimited support and guidance who made this thesis possible.

Dedication

This thesis is dedicated to the one I love.

Table of Contents

List of Tables viii

List of Figures ix

1 Introduction 1

2 Literature Review 4

2.1 Importance of Air-cargo Transportation 4

2.2 Air-cargo Transportation 6

2.3 The Container Loading Problem 6

2.4 Airfreight Consolidation with Pivot Weight 8

2.5 Unit Load Device Types 9

2.6 Air Cargo Loading Problems 11

2.7 Lagrangian Relaxation Techniques 13

3 Mathematical Formulation **16**

3.1 Lagrangian Relaxation 18

3.2 Finding Feasible Solutions 20

3.2.1 Lagrangian Heuristic (LagHeur) 20

3.2.2 Greedy Heuristics 21

3.2.3 The Two-knapsack Heuristic (2knap) 22

3.2.4 The Bestfi Heuristic (bestfit) 24

4 Computational Analysis **26**

4.1 Numerical Testing 27

4.2 Numerical Results 29

4.2.1 Lagrangian Heuristic (LagHeur) 29

4.2.2 The Two-knapsack Heuristic (2knap) 36

4.2.3 The Bestfit Heuristic (bestfit) 40

4.3 Comparison of Heuristics Performance 43

4.4 Comparison of Heuristics Gaps 51

5 Conclusion **58**

References **60**

List of Tables

4.1 Parameters used for numerical testing 28

4.2 Performance of LagHeur for $c_j^E/c_j = 1.2$ and $U_j^E/U_j = 0.1$ 30

4.3 Performance of LagHeur for $c_j^E/c_j = 1.2$ and $U_j^E/U_j = 0.3$ 30

4.4 Performance of LagHeur for $c_j^E/c_j = 3$ and $U_j^E/U_j = 0.1$ 31

4.5 Performance of LagHeur for $c_j^E/c_j = 3$ and $U_j^E/U_j = 0.3$ 31

4.6 Performance of LagHeur0 for $c_j^E/c_j = 1.2, U_j^E/U_j = 0.1$ and $c_j^E/c_j = 1.2, U_j^E/U_j = 0.3$ 33

4.7 Performance of LagHeur0 for $c_j^E/c_j = 3, U_j^E/U_j = 0.1$ and $c_j^E/c_j = 3, U_j^E/U_j = 0.3$ 35

4.8 Performance of 2knap for $c_j^E/c_j = 1.2, U_j^E/U_j = 0.1$ and $c_j^E/c_j = 1.2, U_j^E/U_j = 0.3$ 37

4.9 Performance of 2knap for $c_j^E/c_j = 3, U_j^E/U_j = 0.1$ and $c_j^E/c_j = 3, U_j^E/U_j = 0.3$ 39

4.10 Performance of bestfit for $c_j^E/c_j = 1.2, U_j^E/U_j = 0.1$ and $c_j^E/c_j = 1.2, U_j^E/U_j = 0.3$ 41

4.11 Performance of bestfit for $c_j^E/c_j = 3, U_j^E/U_j = 0.1$ and $c_j^E/c_j = 3, U_j^E/U_j = 0.3$ 42

List of Figures

1.1 Airfreight transportation process. . . . 3

2.1 World airfreight growth. . . . 5

2.2 ULD Container Rohlig (2018) 10

2.3 ULD Pallet Rohlig (2018) 10

3.1 LagHeur Heuristic Flow 21

3.2 2knap Heuristic Flow 23

3.3 Bestfit Heuristic Flow 24

4.1 Heuristics Total Time Comparison for $c_j^E/c_j = 1.2, U_j^E/U_j = 0.1$. . . . 45

4.2 Heuristics Total Time Comparison for $c_j^E/c_j = 1.2, U_j^E/U_j = 0.3$. . . . 47

4.3 Heuristics Total Time Comparison for $c_j^E/c_j = 3, U_j^E/U_j = 0.1$. . . . 48

4.4 Heuristics Total Time Comparison for $c_j^E/c_j = 3, U_j^E/U_j = 0.3$. . . . 50

4.5 Heuristics Gaps Comparison for $c_j^E/c_j = 1.2, U_j^E/U_j = 0.1$. . . . 52

4.6 Heuristics Gaps Comparison for $c_j^E/c_j = 1.2, U_j^E/U_j = 0.3$. . . . 54

4.7 Heuristics Total Time Comparison for $c_j^E/c_j = 3, U_j^E/U_j = 0.1$. 55

4.8 Heuristics Gaps Comparison for $c_j^E/c_j = 3, U_j^E/U_j = 0.3$. 57

Chapter 1

Introduction

Although airfreight transportation accounts for less than 1% of world traded goods, it represents more than 36% of world trade value, since most of the goods transported by air are high in value and require fast and secure transportation (Shepherd et al., 2016). The revenue generated by air cargo transportation has outpaced passenger transportation by 1 to 2 % every year between 1991 and 2001 (Reynolds-Feighan, 2008). Furthermore, it is expected that global revenue generated by air cargo will increase from 223 billion Revenue Tonne-Kilometres (RTKs) in 2015 to 509 billion RTKs in 2035 (Boeing, 2016).

The demand for international air-cargo transportation is increasing due to multiple factors such as increasing global trading, suppliers seeking to lower their inventory, and customers increasing demand for timely delivery of items. An annual growth of 4.3% is expected in the international airfreight transportation industry that will eventually triple in its revenue by 2035 (Boeing, 2016).

International airfreight consolidation involves many players and complex operations (Huang and Chi, 2007). Airfreight forwarders play an important coordination role between shippers and airlines. Figure 1.1 depicts the airfreight transportation process. Shippers look for carriers that provide the cheapest and fastest service for shipment. Forwarders receive cargo and consolidate it into space previously reserved from airlines. Airlines provide space in two stages, the first stage is couple months before the flight date. Any space acquired by forwarders during this stage is called the allotted space. In the second stage, airlines offer space few days before the departure of the flight. Airfreight forwarders try to skillfully consolidate the received cargo into the reserved containers in order to satisfy shippers requirements, and at the same time minimize the total cost incurred by reserving or returning the containers.

Airfreight is shipped in Unit Load Devices (ULD) to standardize and speed-up the process of air cargo transportation. ULDs take two forms: containers and pallets, and are specified by the International Air Transport Association (IATA) depending on the type of aircraft. A ULD is a essentially an aluminum container while pallets are typically an aluminum base, on which cargo is placed and covered with a net to secure it in place (IATA, 2010). (IATA, 2010).

Air cargo forwarders rent different ULDs from cargo airlines. Instead of paying for the chargeable weight that small forwarders do, a large forwarder pays a fixed cost for the ULD and a per unit rate up to the pivot-weight, and a higher per unit charge above the pivot weight. Based on this pricing structure, forwarders are interested in finding the best consolidation plan in order to minimize their total cost.

An overview of the airfreight consolidation problem under the pivot weight scheme was studied in Li et al. (2009) where the authors propose a heuristic to find the best cargo consolidation plan that minimizes an airfreight forwarder's total cost. Bookbinder et al. (2015) propose four solution methodologies for the air cargo consolidation problem under pivot weight based on branch-and-price, best-fit decreasing, and local branching.

In this thesis, we provide an efficient solution approach for the problem based on an enhanced formulation and Lagrangian Relaxation. On top of the lower bound, three heuristics are provided. Problems with up to 100 ULDs and 1000 shipments are solved to within an average of 1% to 2% of optimality in less than 51s.

The remainder of this thesis is organized as follow. In Chapter 2, we review the relevant literature. In Chapter 3, we present the formal definition and mathematical formulations for the airfreight consolidation under pivot-weight problem. Computational analysis results are presented in Chapter 4. Finally, we conclude our work and highlight future works in Chapter 5.

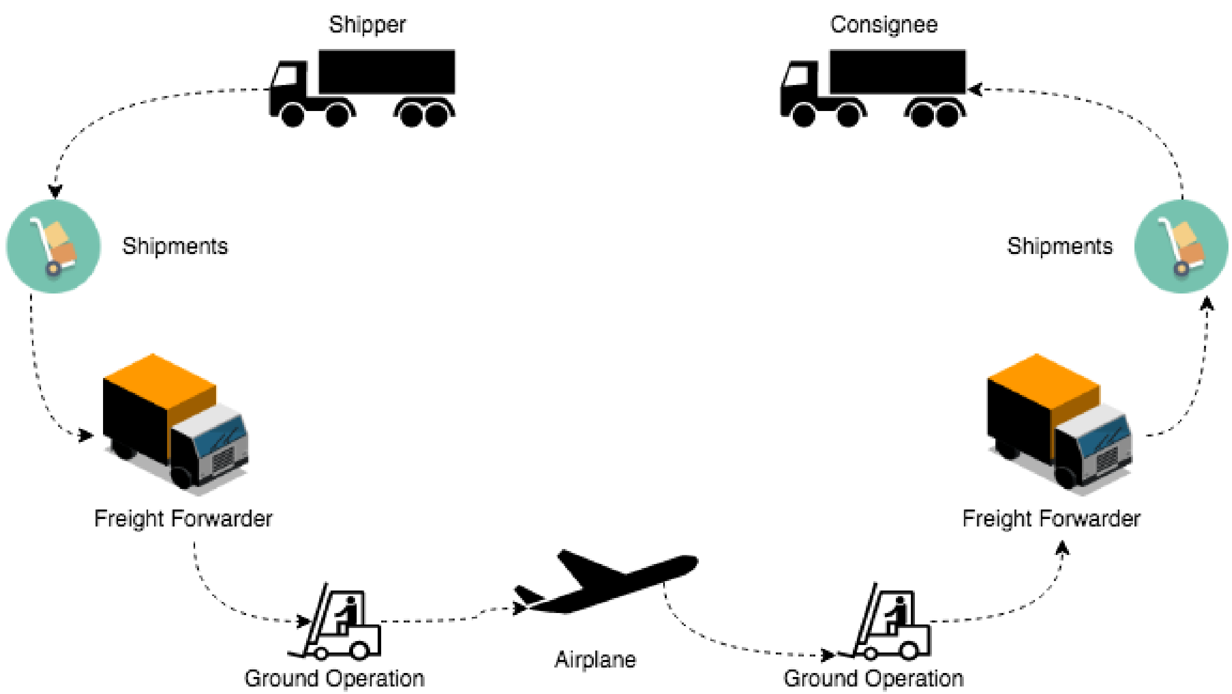


Figure 1.1: Airfreight transportation process.

Chapter 2

Literature Review

In this chapter, we review the literature on the airfreight consolidation problem under the pivot weight scheme. In Section 2.1 and Section 2.2, we address the importance of air cargo transportation in general, and discuss the cargo shipment process. Airfreight container loading is studied in Section 2.3. The most relevant literature to air cargo container loading under pivot-weight is presented in Section 2.4. In Section 2.5 we review the available types of unit load device (ULD) that are used in airfreight consolidation industry. General literature related to airfreight consolidation will be discussed in Section 2.6. Finally, solution methodologies used in this work are reviewed in Section 2.7.

2.1 Importance of Air-cargo Transportation

Airfreight transportation is crucial for many sectors. For instance, airfreight allows speed delivery for perishable goods and pharmaceutical products. The air-cargo transportation industry has played an important role in world trade, and since the 1970's it has doubled volume every 10 years (Chang et al., 2007). The estimated revenue of air cargo was US$3.25 trillion in 2005, and represented 36% of the value of all traded goods globally, estimated at US$9.14 trillion (IATA, 2006).

From 1995 to 2004, there was a significant growth of airfreight transportation compared to passenger transportation which was 50% faster by volume. As a consequence, many airlines transformed from passenger-only carriers to passenger-cargo carriers (Wong et al., 2009). A forecast by Boeing expects a growth of 4.3% annually and that will triple by 2035 (Boeing, 2016). Figure 2.1 shows an increase of 282 billion revenue ton kilometers (RTKs) by the year 2035.

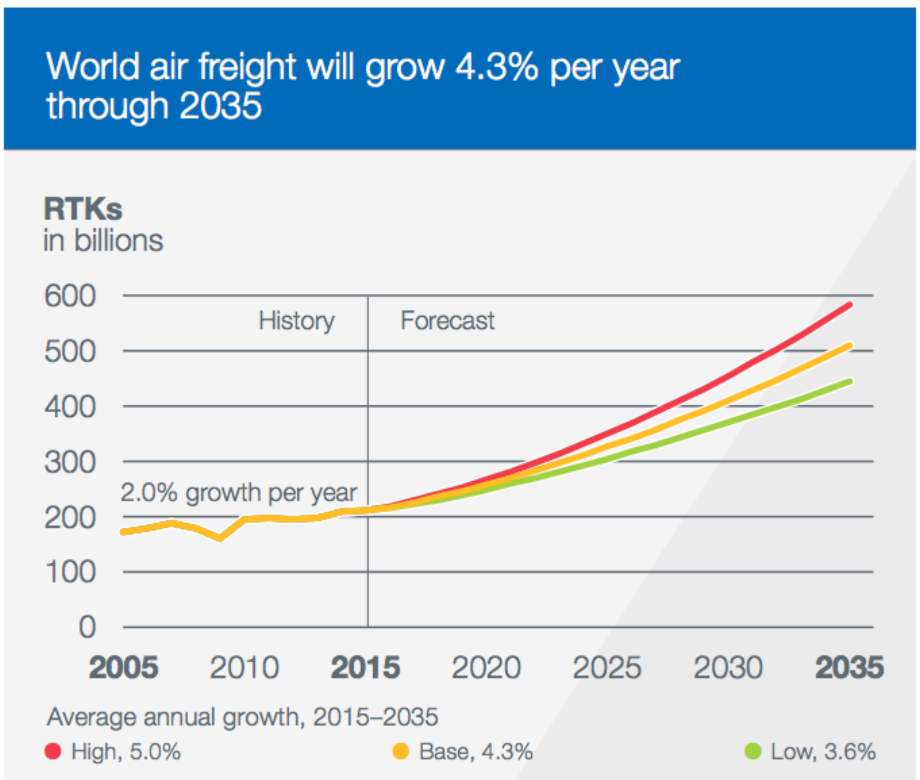


Figure 2.1: World airfreight growth.

2.2 Air-cargo Transportation

Air-cargo transportation includes multiple stages to move cargo from source to destination and involves shippers, forwarders, and airline carriers (Huang and Chi, 2007). There are three main players involved in air cargo transportation, shipper, forwarder, and airline carrier (Derigs et al., 2009). The shipper sends cargo to forwarders to be shipped anywhere in the world, expecting the lowest price and the shortest time of delivery. The latter is the most important criteria for shippers (Matear and Gray, 1993). The forwarder acts as the middle-man between shippers and airlines.

Forwards need to skillfully consolidate the cargo from shippers to satisfy their demands, at the same time minimizing the cost incurred by the airlines (Huang and Chi, 2007). The airline offers available cargo space in two stages. The first is few months before the shipment date, and the second is few days before the departure of the flight. The main role of the airlines consists of receiving the cargo from the forwarders, loading it into containers, storing the cargo, transporting it and unloading it.

2.3 The Container Loading Problem

Cargo loading into (ULDs) is exactly a container loading problem, i.e. putting shipped items into ULDs subject to weight and volume limits and in a predefined time frame. The container loading problem, however, accounts explicitly for geometric constraints whereas consolidation problems do not explicitly do so. Xue and Lai (1997) introduce a mixed integer programming model for container selection and cargo loading in order to minimize the total cost. Pisinger (2002) proposes a new heuristic based on wall building approach to maximize the volume of the loaded rectangular boxes in limited capacity rectangular container in ocean cargo transportation. Huang and Chi (2007) develop a recursive heuristic based on Lagrangian relaxation to minimize the total cost of consolidating shipper's items into airline containers and utilizing the quantity discount provided by the airlines, from the airfreight forwarder's perspective.

A practical problem faced by a logistic company in Hong Kong was used in the work of Wu (2008) to introduce a framework to help air-freight forwarders allocate the required number and type of cargo containers, and simultaneously finding the optimal way to load the shipments into the previously rented containers in order to minimize the total cost and satisfy customers demand. A two-stage approach for air cargo forwarders was introduced in Wu (2010) where in the first stage the types and numbers of cargo containers is determined based on a deterministic information usually one week before shipping. The second stage is for any actions taken on the day of shipping which contains defining the required types and numbers of containers, and loading the cargo into the rented containers.

Mostaghimi Ghomi et al. (2017) proposed a three dimensional bin-packing model for the container loading problem. A mixed integer linear programming model is developed for consolidating rectangular-shape boxes into a container where the volume of consolidated boxes is maximized, while satisfying other constraints like vertical stability and handling pre-placed boxes inside containers. A heuristic algorithm based on simulated annealing is used to solve the problem and get a good quality solution in a reasonable time.

Paquay et al. (2012) developed a mixed integer linear program for the three dimensions bin packing problem (3D-BPP) for air cargo loading into ULDs, and CPLEX was used for small version of the problem. The proposed model takes into account many realistic constraints such as stability or the fragility of the cargo, and distribution of cargo weight carried by aircraft.

Hai (2016) discuss the consolidation problem faced by international airfreight forwarders, where they need to pack different size, weight, and release date shipments into ULDs. A three dimensional bin-packing model is developed which considers cargo position, weight, and priority. Also, a relaxed version of the original model is presented to solve some unsolved cases that original model failed to solve. CPLEX is used to solve both models using random generated dataset.

2.4 Airfreight Consolidation with Pivot Weight

For safety reasons, airlines utilize the concept of pivot weight when forwarders load ULDs with cargo. The cost consists of a fixed cost for using the ULD, and two unit costs known as under-pivot rate and over-pivot rate. The cost of loading depend on a weight threshold where any weight that is below the threshold is charged at the under-pivot rate, while any weight that exceeds the threshold is charged at the over-pivot rate.

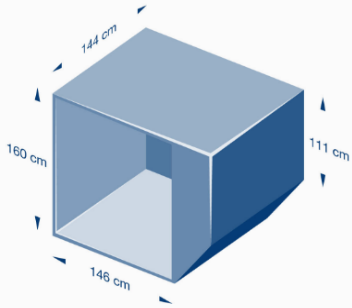
Based on this pricing scheme, forwarders are interested in minimizing their consolidation cost. Li et al. (2009) propose a large-scale neighborhood search heuristic to minimize total cost of airfreight forwarders, by finding the best plan for loading cargo into rented containers (ULDs). Another related work is the work of Bookbinder et al. (2015) in which four solution methodologies were proposed to solve the air cargo consolidation problem under pivot-weight scheme that are branch-and-price, best-fit decreasing heuristic, and two extensions of the local branching heuristic. The relaxation-induced neighborhood search in Bookbinder et al. (2015) was found to be the best among all proposed solution approaches. Solutions within 3.4% of optimality were obtained in under 20 minutes with up to 400 shipments and 80 containers.

2.5 Unit Load Device Types

ULDs were introduced to standardize and increase the efficiency off air cargo transportation. ULDs provides forwarders and shippers with many advantages for example:

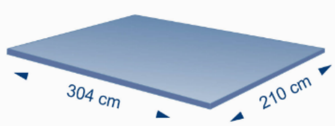
- Faster loading/unloading process
- Protect cargo and aircraft from damage
- Require less experienced personnel to fill

Pallets provide the advantages of allowing the shipper to load some cargo that does not fit into regular containers, and the feasibility in stocking the cargo because of the openness. ULD containers and pallets are designed in different shapes and sizes by IATA to fit on different aircrafts. Also, airfreight forwarders need to choose the required containers and pallets in order to be compatible with the carrier aircraft. Figures 2.2, and 2.3, show some standard ULD types, codes, dimensions, and aircraft's compatibility. In this thesis, the pivot weight scheme focuses on ULD containers rather than ULD pallets.



LD3 container AKE/AVE	
Type	LD3 container
Code	AKE / AVE
Base dim. outside	1,5 x 1,5 m
Height outside	1,6 m
Useable volume	3 m³
Dimensions inside	1,4 x 1,4 x 1,5 m
Acceptable for	A300, A310, A330, A340, B747-200F, B747-400, B767-400, B767(DE), MD11F

Figure 2.2: ULD Container Rohlig (2018)



Standard pallet	
Type	standard pallet
Code	PAG, PAJ
Dimensions	3,1 x 2,2 m
Useable loading area	3 x 2,1 m
Acceptable for	A300, A310, A330, A340, B747-200F, B747-400, MD11F

Figure 2.3: ULD Pallet Rohlig (2018)

2.6 Air Cargo Loading Problems

Roesener and Barnes (2016) investigate the dynamic airlift loading problem from a military perspective. The problem combines four interdependent sub-problems: 1-packing cargo into pallets, 2-partitioning pallets, 3-selecting aircraft, and 4-placing pallets in the allowable locations in order to minimize the total cost of moving all pallets and meeting all temporal constraints. They propose a Tabu Search that they call the dynamic airlift loading problem-tabu search (DALP-TS) algorithm to aid in the decision-making process. The DALP-TS algorithm utilizes three search neighborhoods that each is called under different scenarios to traverse both the feasible and infeasible solution space. The military airlift problem is different from it commercial counterpart in that the military airlift problem tries to satisfy the effectiveness first (i.e. on time delivery) and efficiency second (i.e. minimizing cost), while commercial airlift problem mainly aims to minimize cost and maximize revenue. There is no consideration of pivot-weight scheme, as in ACPW, when loading pallets into allowable locations in aircraft.

Limbourg et al. (2012) developed a mixed integer linear programming model for the aircraft cargo loading planning problem. The proposed model tries to find the optimal placement plan for ULDs in predefined positions and maintain aircraft stability by positioning the center of gravity as close as possible to the required center of gravity by the loadmaster. The model also considers the moment of inertia of the loaded cargo in order to reduce stress on the structure of the aircraft and to increase fuel economy. They designed a software that utilizes CPLEX for solving the proposed model to obtain an optimal loading plan for a set of ULDs into aircraft. Tests were conducted on a real-world data provided by an industrial partner CHAMP Cargosystems. The proposed problem is different from the ACPW problem because it does not consider optimizing the loading of freight inside

ULDs, instead it tries to find the optimal placement of ULDs inside an aircraft. Another difference from ACPW is the pivot-weight scheme that is enforced for safety reasons.

Feng et al. (2015b) present a detailed review of air cargo operations and show how air cargo is an operation-intensive industry that involve many important players. Also, they highlighted the gap between previous research efforts and real-world practice. One of the gaps related to container loading problem is the capacity allocation between long-term contract and the spot market for freight forwarders. Another commonly found gap in different air cargo problems is the model assumption deviation from reality.

Vancroonenburg et al. (2014) study the air cargo loading problem to maximize profit, by introducing a model that select the most profitable subset of available ULDs to load into aircraft while minimizing the deviation between the aircraft's center of gravity and the required center of gravity. The main objective of proposed mixed integer programming model is maximizing profit and minimize deviation of aircraft's center of gravity. The model is solved using Gurobi solver and tested using real-world data obtained from a commercial cargo carrier. The problem is different from ACPW problem since ACPW is concerned with airfreight loading in ULDs while Vancroonenburg et al. (2014) work addresses the air cargo loading into aircraft.

Lurkin and Schyns (2015) considered the air cargo loading problem in multiple airport context and aircraft with several doors to use in loading and unloading cargo. They proposed a mixed integer programming model to solve the problem of loading and unloading containers into and from cargo aircraft. The main objective of the proposed model is to minimize fuel consumption and handling operation cost of unloading and reloading task at stop-over airports. The proposed model is tested on a real-world data provided by (TNT Airways) using CPLEX solver that provided a near optimal solution in a short computational time. The problem has a different objective than ACPW as it minimize the cost of loaded cargo into ULDs.

Feng et al. (2015a) investigate the imbalance between demand and capacity on different flight routes. They suggested a bundling mechanism to balance the allocation of air cargo capacity between popular and underutilized flight routes to maximize profit. A nonlinear integer programming model was proposed, and a dynamic programming algorithm was developed to solve the problem using data from real-word forwarders and airlines.

Chao and Li (2017) discus the important aspects influencing air cargo revenue, and how the cargo charges and density affect revenues. They developed a set of mathematical models and used an actual flight data to show the effect of Density Ratio of Heavy cargo to Light cargo (DRHL) on the chargeable weight, and how the percentage of small cargo affect the revenue of air cargo.

2.7 Lagrangian Relaxation Techniques

Lagrangian relaxation was developed in the early 1970's by Held and Karp (1970). They applied it to the travelling salesman problem. Since then, it has been one of the most used method in optimization. Many view Lagrangian relaxation as a technique to convert hard problems into relatively easy problems by dualizing complicating constraints to get an easy-to-solve problems and a lower bound (in case of minimization) (Fisher, 2004). Also, the intuitive concept behind Lagrangian relaxation made it very easy to adopt and implement. The goal of applying Lagrangian relaxation to an optimization problem is to find a lower bound (in case of minimization problem) by eliminating a set of hard constraints and placing them in the objective function using Lagrangian multipliers. The Lagrangian multipliers are found using algorithms such as subgradient optimization and cutting plane methods.

Lagrangian Relaxation provides a lower bound for a minimization problem. To generate a feasible solution, Lagrangian heuristics use the solution of the subproblem and try to make it feasible for the original problem. The quality of the solution is measured by its relative proximity to the lower bound.

To illustrate the Lagrangian Relaxation methodology, consider the optimization problem:

$$\begin{aligned} \min \quad & c^T x \\ \text{s.t.} \quad & Ax = b \\ & Bx = d \\ & x_j \quad integer, \qquad j \in J \end{aligned}$$

Relaxing the first set of constraints with Lagrangian multipliers λ, leads to the subproblem:

$$\begin{aligned} \min \quad & (c - A^T\lambda)^T x + b^T\lambda \\ & Bx = d \\ & x_j \quad integer, \qquad j \in J \end{aligned}$$

The Lagrangian lower bound is given by:

$$\max \quad \{b^T\lambda + \min \quad \{x : Bx = d, \quad x_j \quad integer, \quad j \in J\}(c - A^T\lambda)^T x\}$$

Which is equivalent to the master problem:

$$\begin{aligned} \max \quad & b^T\lambda + \theta \\ & \theta + (Ax^h)^T\lambda \leq c^T x^h \quad h \in H \end{aligned}$$

Where H is the set of integer feasible points to

$$\{x : \quad Bx = d, \quad x_j \quad integer, \quad j \in J\}$$

which is assumed to be bounded. The Lagrangian multipliers can be alternatively updated using subgradient optimization (Fisher, 2004).

Chapter 3

Mathematical Formulation

Using indices $j \in J$ for ULDs and $i \in I$ for shipments, we model the ACPW using three decision variables x_{ij}, z_j, y_j^E where x_{ij} is binary variable that takes value 1 if shipment i is consolidated in ULD j, and 0 otherwise. z_j is also a binary variable that takes value 1 if ULD j is used, and 0 otherwise. y_j^E is a continuous variable to denote the additional capacity over the pivot weight. We also define the following parameters:

f_j: the fixed cost for using ULD j, $j = 1, ..., J$.

c_j: the under-pivot shipment cost for ULD j, $j = 1, ..., J$.

c_j^E: the over-pivot shipment cost for ULD j, $j = 1, ..., J$.

g_i: the weight of shipment i, $i = 1, ..., I$.

U_j: the under-pivot capacity for ULD j, $j = 1, ..., J$.

U_j^E: the over-pivot capacity for ULD j, $j = 1, ..., J$.

The ACPW is modeled as follows:

$$
\begin{aligned}
\min \quad & \sum_j f_j z_j + \sum_i \sum_j g_i c_j x_{ij} + \sum_j c_j^E y_j^E & & (3.1)\\
\text{s.t.} \quad & \sum_j x_{ij} = 1 & \forall i \in I \quad & (3.2)\\
& \sum_i g_i x_{ij} \leq U_j z_j + y_j^E & \forall j \in J \quad & (3.3)\\
& y_j^E \leq U_j^E z_j & \forall j \in J \quad & (3.4)\\
& x_{ij}, \quad z_j \in \{0,1\}, \quad y_j^E \geq 0 & \forall i \in I, j \in J. \quad & (3.5)
\end{aligned}
$$

The objective function (3.1) minimizes the fixed cost plus the under-pivot cost plus the over-pivot cost. Constraints (3.2) require that each shipment is consolidated exactly once over all available ULDs. Constraints (3.3) and (3.4) model the capacity for ULD j including the under-pivot and the over-pivot capacities. By defining y_j to be the fraction of over-pivot capacity U_j^E used, the ACPW can modeled as:

$$
\begin{aligned}
[ACPW]: \quad \min \quad & \sum_j f_j z_j + \sum_i \sum_j g_i c_j x_{ij} + \sum_j c_j^E U_j^E y_j & & (3.6)\\
\text{s.t.} \quad & \sum_j x_{ij} = 1 & \forall i \in I \quad & (3.7)\\
& \sum_i g_i x_{ij} \leq U_j z_j + U_j^E y_j & \forall j \in J \quad & (3.8)\\
& y_j \leq z_j & \forall j \in J \quad & (3.9)\\
& 0 \leq y_j \leq 1; \quad x_{ij}, \quad z_j \in \{0,1\} & \forall i \in I, j \in J. \quad & (3.10)
\end{aligned}
$$

3.1 Lagrangian Relaxation

By relaxing constraints (3.1) with Lagrangian multipliers $\lambda_j, j \in J$, we get the following subproblem:

$$
\begin{aligned}
[SP]: \quad & \min \quad \sum_j (f_j - \lambda_j U_j) z_j + \sum_i \sum_j (c_j + \lambda_j) g_i x_{ij} + \sum_j (c_j^E - \lambda_j) U_j^E y_j \\
& \text{s.t.} \quad \sum_j x_{ij} = 1 \qquad \forall i \in I \\
& y_j \le z_j \qquad \forall j \in J \\
& \sum_j U_j z_j + \sum_j U_j^E y_j \ge \sum_i g_i \qquad (3.11) \\
& 0 \le y_i \le 1; \quad x_{ij}, z_j \in \{0,1\}, \qquad \forall i \in I, j \in J.
\end{aligned}
$$

Note that we added the redundant constraint (3.11) to strengthen the subproblem. The constraint is redundant in [ACPW] but not in [SP]. It is obtained by summing (3.1) over all $j \in J$.

$[SP]$ is decomposed into two subproblems $[SP1]$ and $[SP2]$. The first subproblem $[SP1]$ is:

$$
\begin{aligned}
[SP1]: \quad v_1 = \; & \min \quad \sum_j (f_j - \lambda_j U_j) z_j + \sum_j (c_j^E - \lambda_j) U_j^E y_j \\
& \text{s.t.} \quad y_j \le z_j \qquad \forall j \in J \\
& \sum_j U_j z_j + \sum_j U_j^E y_j \ge \sum_i g_i \\
& z_j \in \{0,1\}, \quad 0 \le y_j \le 1 \qquad \forall j \in J.
\end{aligned}
$$

The second subproblem $[SP2]$ is decomposable into $|I|$ subproblems, one for each shipment i:

$$[SP2_i]: \qquad v_{2i} = \min \sum_j (c_j + \lambda_j) g_i x_{ij}$$
$$\text{s.t.} \sum_j x_{ij} = 1$$
$$x_{ij} \in \{0, 1\}.$$

The Lagrangian bound is $(v_1 + \sum_i v_{2i})$. The best bound is the solution of the Lagrangian master problem:

$$
\begin{aligned}
[MP]: \qquad v_{MP} = \max \quad & \theta_0 + \sum_i \theta_i \\
\text{s.t.} \quad & \theta_0 \leq \sum_j (f_j - \lambda_j U_j) z_j^h + \sum_j (c_j^E - \lambda_j) U_j^E y_j^h && h \in H_0 \\
& \theta_i \leq \sum_j (c_j + \lambda_j) g_i x_{ij}^h && h \in H_i, i \in I \\
& \theta_i, \quad \theta_0 \quad \textit{unrestricted}, \quad \lambda_j \geq 0 && i \in I, j \in J
\end{aligned}
$$

Where $(x_{ij}^h, z_j^h, y_j^{Eh})$ are the feasible solutions to $[SP1]$ and $[SP2_i]$, $i \in I$.

The Lagrangian algorithm is summarized below:

Initialize $\bar{H} \subseteq H$, $LRLB = -\infty, LRUB = +\infty$.

While $LRLB \neq LRUB$

Step 1: solve $[MP]$ corresponding to $\bar{H}$ to obtain λ_j. $LRUB = v_{MP}$

Step 2: solve subproblem $[SP2_i]$ and $[SP1]$ at λ_j to obtain $\bar{x}_{ij}^h$, $\bar{y}_j^h, \bar{z}_j^h$, respectively. $LRLB = \max(LRLB, (v_1 + \sum_i v_{2i}))$.

Step 3: Update $\bar{H}$ and add corresponding constraints to $[MP]$.

3.2 Finding Feasible Solutions

In this section, we propose three heuristics to find feasible solutions for ACPW based on the solutions of the subproblems. For all proposed heuristics, we run each heuristic when the Lagrangian lower and upper bounds are within 10% of optimal bound , i.e. $(LRUB - LRLB)/LRLB > .1$.

3.2.1 Lagrangian Heuristic (LagHeur)

Lagrangian heuristics use the solution of the subproblem and try to convert it into a feasible solution for the original problem by some adjustment. This feasible solution constitute an upper bound on the optimal solution to the problem. The key idea is that just as the solution value for relaxed version of the original problem gives us useful information (a lower-bound on the optimal integer solution value) also the solution structure of the relaxed version of the original problem (i.e. the value of the decision variables) may also provide useful information about the solution structure of the optimal integer solution.

Solving [SP1] will return $\bar{z}_j$ that represent the ULDs to be used, and $\bar{y}_j$ that determine the percentage of over-pivot capacity used in each used ULD. Using these values, we fix z and y in [ACPW] and solve for the assignments variable x_{ij} as shown in Figure 3.1.

$$
\begin{aligned}
[HP]: \qquad & \min \quad \sum_i \sum_j g_i c_j x_{ij} + \sum_j f_j \bar{z}_j + \sum_j c_j^E U_j^E \bar{y}_j \\
& \text{s.t.} \quad \sum_j x_{ij} = 1 && \forall i \in I \\
& \qquad\quad \sum_i g_i x_{ij} \leq U_j \bar{z}_j + U_j^E \bar{y}_j && \forall j \in J \\
& \qquad\quad x_{i,j} \in \{0,1\} && \forall i \in I, j \in J.
\end{aligned}
$$

If a solution is found, it is feasible to ACPW. The procedure can be executed every time [$SP1$] is solved or when the Lagrangian lower bound is close to the optimal one. The solution of [HP] could be stopped after a predetermined time limit or number of nodes.

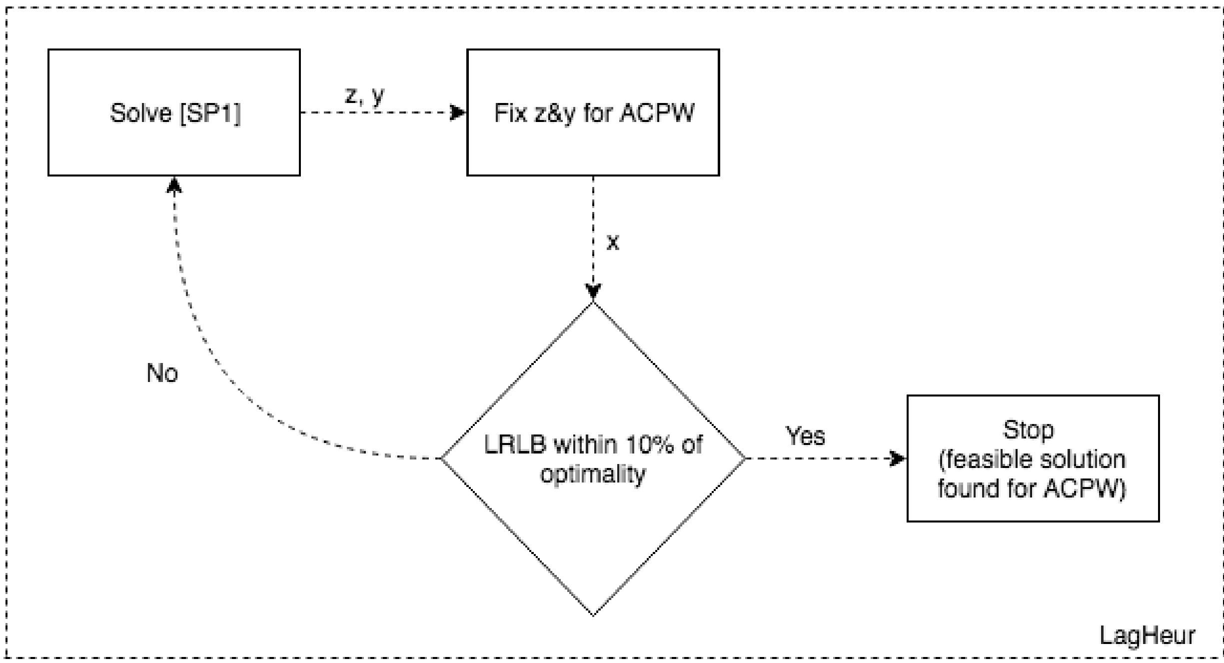


Figure 3.1: LagHeur Heuristic Flow

3.2.2 Greedy Heuristics

Greedy heuristics try to trade the solution optimality for speed. In many optimization problems greedy strategy does not produce an optimal solution, but it may find good quality solutions in a very short computational time. In this work, we try two different greedy heuristics: the two knapsacks approach and the bestfit approach.

3.2.3 The Two-knapsack Heuristic (2knap)

Given a set of items with different values and weights, the knapsack problem tries to select a combination of items with the maximum value where the total weight of items does not exceed a certain limit. Heuristics for the knapsack loading problem have been presented in Gehring et al. (1990). In this work, each ULD is treated as two knapsacks where the first is the under-pivot knapsack, and the second is the over-pivot knapsack. The problem is to choose a subset of shipments that fits into a single ULD so that total cost of using ULDs is minimized.

In this heuristic, the activated ULD based on subproblem SP1 solution will be treated as two knapsacks the first is the under-pivot knapsack, and the second is the over-pivot knapsack. The heuristic starts with ordering the selected ULDs in an ascending order of their total cost density, and the shipments in descending order of their chargeable weight. After ordering both ULDs and shipments, the heuristic starts by assigning shipments to the under-pivot knapsack of ULDs if it's possible. Next, if any shipments is left unassigned, the heuristic will assign shipments to the over-pivot knapsack if the capacity permits as shown in Figure 3.2. The heuristic steps are:

At every Lagrangian iteration, the solution of subproblem [SP1] yields the identity of the ULDs to be used as $\bar{z}_j$, and the percentage of the used over-pivot weight capacity as $\bar{y}_j$. Let $\bar{z}_j, \bar{y}_j$ be the solution of [SP1].

Step 1: Identify the ULDs to be used ($\bar{z}_j$) from [SP1] solution, and sort them in ascending order based on their under-pivot cost density ($\frac{f_j}{U_j} + c_j$).

Step 2: sort the shipments (g_i) based on their chargeable weight in a descending order.

Step 3: For each unassigned shipment (g_i). Let u_j be the available under-pivot capacity in ULD with $z_j = 1$.

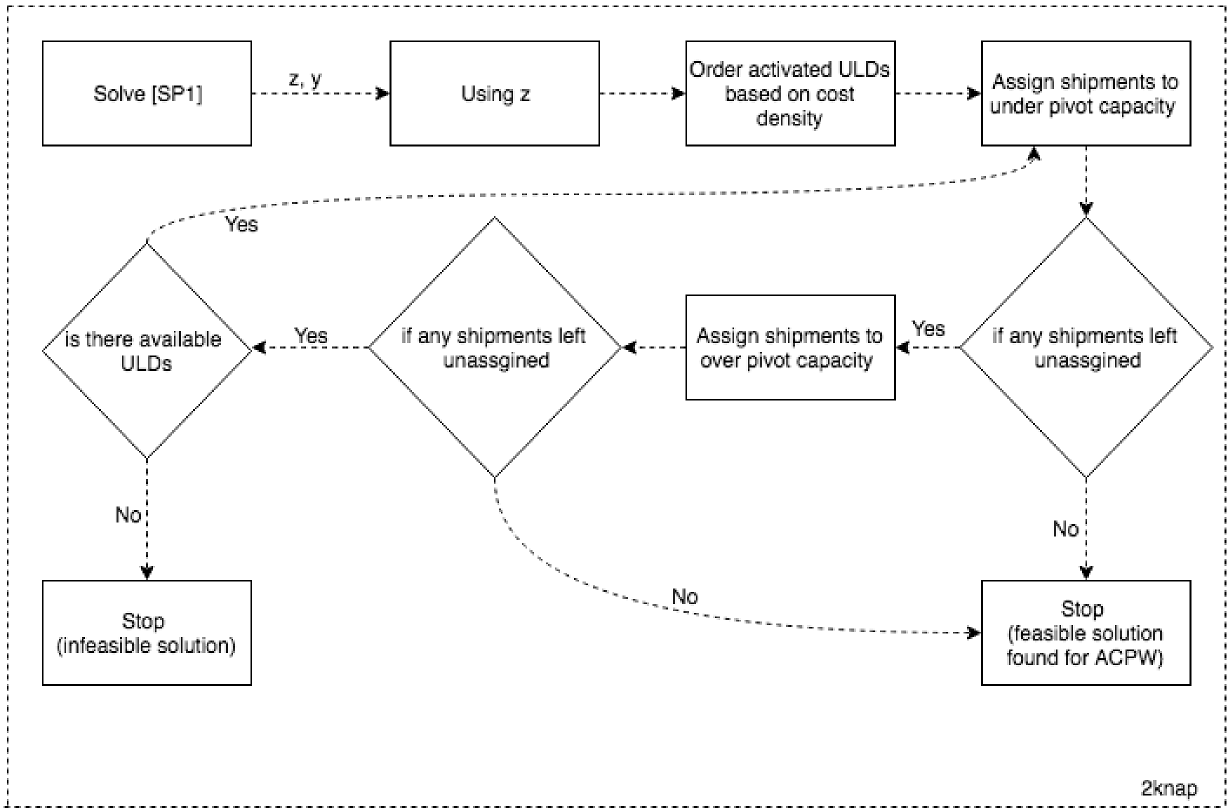


Figure 3.2: 2knap Heuristic Flow

If $(g_i \leq u_j)$ set $x_{ij} = 1$

Step 4: For all unassigned shipments (g_i). Let o_j be the available total capacity in ULD with $z_j = 1$ which include all over-pivot capacity and all unused under-pivot capacity.

If $(g_i \leq o_j)$ set $x_{ij} = 1$ and $y_j = \dfrac{\sum_i g_j x_{ij} - U_j}{UE_j}$

Step 5: The solution is infeasible if any shipments are left unassigned after Step 4, so by activating an empty ULD that has $(z_j = 0)$ and the least cost density $(\frac{f_j}{U_j} + c_j)$ and make it $z_j = 1$ we then go back to Step 3.

3.2.4 The Bestfi Heuristic (bestfit)

Each ULD has two types of capacities, the under-pivot and over-pivot capacities. In the Best fit approach, both capacities are merged into one. In this heuristic, the activated ULD based on [SP1] solution will be treated as one knapsack. After assigning all shipments to the available ULD, the heuristic calculates the value of the over-pivot usage y_j in each activated ULD as shown in Figure 3.3. The heuristic steps are as follow:

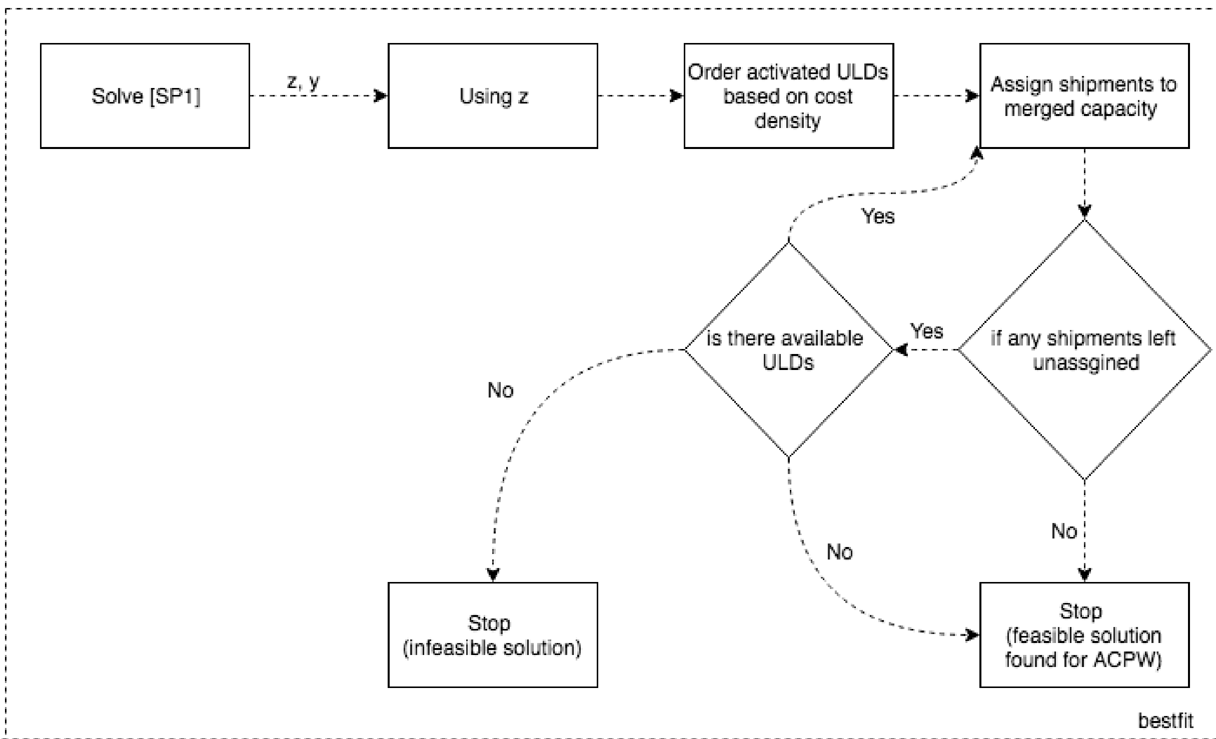


Figure 3.3: Bestfit Heuristic Flow

Let $\bar{z}_j, \bar{y}_j$ be the solution of [SP1].

Step 1: identify the ULDs to be used ($\bar{z}_j$) from [SP1] solution, and sort them in ascending order based on their total cost density ($\frac{(f_j + c_j U_j + c_j^E U_j^E)}{U_j + UE_j}$).

Step 2: sort the shipments (g_i) based on their chargeable weight in a descending order.

Step 3: For each unassigned shipment (g_i). Let t_j be the available total capacity (under-pivot capacity + over-pivot capacity) in ULD with $z_j = 1$.

If ($g_i \leq t_j$) set $x_{ij} = 1$

Step 4: Find the over-pivot usage for each used ULD.

$$y_j = \frac{\sum_i g_j x_{ij} - U_j}{UE_j}$$

Step 5: The solution is infeasible if some shipments are left unassigned after Step 4, so by activating an empty ULD that has ($z_j = 0$) and the least cost density ($\frac{(f_j + c_j U_j + c_j^E U_j^E)}{U_j + UE_j}$) and make it $z_j = 1$ we then go back to Step 3.

Chapter 4

Computational Analysis

In this section, we evaluate the computational performance of the proposed solution methodologies. The methods is coded in Matlab with CPLEX and Gurobi solvers on a laptop with 2.6 GHz Inter Core i7 and 8GB of memory. As real data from airfreight forwarders is hard to obtain since it is confidential, we generate our own test cases in order to compare the performance of the proposed methods. Each case is described by the number of ULDs $|J|$, the number of shipments $|I|$, the ratio by which the over-pivot capacity exceeds the pivot-weight, and the ratio of the over-pivot cost to the under-pivot cost. Similar to Bookbinder et al. (2015), the shipment weight g_i is uniformly distributed in the range [100, 300]. The percentage by which U_j^E exceeds U_j takes one of two values 10% or 30%. The over-pivot c_j^E is tested for 1.2 and 3.

4.1 Numerical Testing

We generate 80 instances with different combinations of parameter values as shown in Table 4.1. As in Bookbinder et al. (2015), the size of U_j is uniformly distributed between [300, 7500]. Both the fixed costs f_j and variable costs c_j are related to the size of ULD U_j. The fixed cost f_j is related to the size of ULD U_j as follow: $f_j = 4000 + 0.5(U_j - 1000)$. The variable cost c_j is related to the size of ULD U_j as follows:

if $U_j < 3000$ then c_j is uniformly distributed between [7, 8]

if $U_j \geq 3000$ then c_j is uniformly distributed between [6.5, 7.5]

Parameter	**Description**	**Value**
J	Set of ULDs	8, 16, 40, 80, 100
I	Set of shipments	Small instances 20, 40, 100, 150, and large instances 100, 150, 200, 300, 400, 600, 1000
U_j	Under-pivot capacity of ULD j	U_j is uniformly distributed between [300, 7500]
U_j^E	Over-pivot capacity of ULD j	Ratio of $U_j^E/U_j = 0.1$ or 0.3
c_j	Under-pivot cost for ULD j	Depends on the value of U_j if $U_j < 3000$ then c_j ∼U[7, 8]; if $U_j \geq 3000$ then c_j ∼U[6.5, 7.5]
c_j^E	Over-pivot cost for ULD j	Ratio of $c_j^E/c_j = 1.2$ or 3, In the airfreight industry, the c_j^E/c_j ratio is usually set between 1.2 and 2.
g_i	Weight of shipment i	g_i ∼U[100, 300]
f_j	Fixed cost for using ULD j	f_j value depend on the ULD capacity U_j where $f_j = 4000 + 0.5(U_j - 1000)$.

Table 4.1: Parameters used for numerical testing

The generated dataset could be divided into small and large based on the number of selected ULDs and shipments. In the small instances, the number of ULDs is 8 or 16, while in the large instance, the number of ULDs is 40, 80, or 100. Also, in the small instances, the number of shipments is set to 100, or 150, while in the large instances, the number of shipments is set to 200, 300, 400, 600, or 1000.

4.2 Numerical Results

4.2.1 Lagrangian Heuristic (LagHeur)

The 80 instances are solved using the Lagrangian approach and the three heuristics. A time limit of 600 seconds is set. We report the heuristic gap as 100 $\frac{UB-LB}{LB}$ (HeurGap). SP1Time, SP2Time, MPTime, HeurTime, TotalTime denote the time taken by Subproblem 1, Subproblem 2, Master problem, heuristic, and the entire approach, respectively. The number of iterations are displayed in the last column.

Tables 4.2, 4.3, 4.4, and 4.5 show results of Lagrangian heuristic. The first column of these tables indicates the number of ULD available, while the second column shows the number of shipments to be consolidated. The third and fourth columns show the value of c^E the cost for using under-pivot weight and U^E the over-pivot weight capacity respectively.

Tables 4.2, 4.3, 4.4, and 4.5 display statistics and computational performance for $c_j^E/c_j = 1.2$, $U_j^E/U_j = 0.1$, $c_j^E/c_j = 1.2$, $U_j^E/U_j = 0.3$, $c_j^E/c_j = 3$, $U_j^E/U_j = 0.1$, $c_j^E/c_j = 3$, and $U_j^E/U_j = 0.3$, respectively.

ULDs	Shipments	c_j^E/c_j	U_j^E/U_j	HeurGap	SP1Time	SP2Time	MPTime	HeurTime	TotalTime	Iterations
8	20	1.2	0.1	0.01	0.04	0.2	0.02	70.11	70.39	9
	40	1.2	0.1	0.01	0.08	0.39	0.02	15.54	16.1	9
	100	1.2	0.1	0.01	0.02	1.05	0.05	1.54	2.73	9
	150	1.2	0.1	0.01	0	1.5	0.07	0.87	2.58	9
16	20	1.2	0.1	0.23	0.2	0.45	0.07	69.49	70.26	18
	40	1.2	0.1	0.01	0.07	0.75	0.1	1352.8	1353.8	17
	100	1.2	0.1	0.01	0.12	1.93	0.1	1755.4	1757.7	17
	150	1.2	0.1	0.01	0.19	2.93	0.15	2987.7	2991.2	18
Min				0.01	0.00	0.20	0.02	0.87	2.58	9.00
Max				0.23	0.20	2.93	0.15	2987.70	2991.20	18.00
Avg				0.03	0.09	1.15	0.07	781.68	783.10	13.25

Table 4.2: Performance of LagHeur for $c_j^E/c_j = 1.2$ and $U_j^E/U_j = 0.1$

ULDs	Shipments	c_j^E/c_j	U_j^E/U_j	HeurGap	SP1Time	SP2Time	MPTime	HeurTime	TotalTime	Iterations
8	20	1.2	0.3	0.00	0.05	0.22	0	79.59	79.9	10
	40	1.2	0.3	0.65	0.03	0.43	0.03	33.27	33.81	9
	100	1.2	0.3	0.01	0.05	1.08	0.03	117.88	119.05	9
	150	1.2	0.3	0.01	0.04	1.48	0.06	4	5.66	9
16	20	1.2	0.3	0.04	0.21	0.44	0.07	318.56	319.28	17
	40	1.2	0.3	0.00	0.08	0.83	0.02	1191	1192	17
	100	1.2	0.3	0.01	0.09	2.11	0.13	3374.9	3377.4	18
	150	1.2	0.3	0.01	0.11	3	0.21	2669	2672.6	18
Min				0.00	0.03	0.22	0.00	4.00	5.66	9.00
Max				0.65	0.21	3.00	0.21	3374.90	3377.40	18.00
Avg				0.09	0.08	1.20	0.07	973.53	974.96	13.38

Table 4.3: Performance of LagHeur for $c_j^E/c_j = 1.2$ and $U_j^E/U_j = 0.3$

Tables 4.2 and 4.3 show that when the ratio of U_j^E/U_j increases to 0.3, the Lagrangian heuristic solution average gap increases relative to when $U_j^E/U_j = 0.1$. The maximum gap achieved when $U_j^E/U_j = 0.3$ is 0.65 in 33.81s compared to a maximum gap of 0.23 in 70.26s when $U_j^E/U_j = 0.1$.

ULDs	Shipments	c_j^E/c_j	U_j^E/U_j	HeurGap	SP1Time	SP2Time	MPTime	HeurTime	TotalTime	Iterations
8	20	3	0.1	0.00	0.06	0.21	0.04	194.51	194.85	9
	40	3	0.1	0.01	0.06	0.45	0.02	667.22	667.79	9
	100	3	0.1	0.01	0.03	1.03	0.05	39.03	40.2	9
	150	3	0.1	0.01	0.01	1.44	0.08	2.59	4.21	9
16	20	3	0.1	0.56	0.09	0.45	0.09	49.77	50.43	18
	40	3	0.1	0.01	0.08	0.79	0.04	175.38	176.35	16
	100	3	0.1	0.00	0.09	1.94	0.11	1228.3	1230.5	16
	150	3	0.1	0.01	0.12	3.04	0.22	854.01	857.58	18
Min				0.00	0.01	0.21	0.02	2.59	4.21	9.00
Max				0.56	0.12	3.04	0.22	1228.30	1230.50	18.00
Avg				0.08	0.07	1.17	0.08	401.35	402.74	13.00

Table 4.4: Performance of LagHeur for $c_j^E/c_j = 3$ and $U_j^E/U_j = 0.1$

ULDs	Shipments	c_j^E/c_j	U_j^E/U_j	HeurGap	SP1Time	SP2Time	MPTime	HeurTime	TotalTime	Iterations
8	20	3	0.3	0.00	0.03	0.27	0.01	42.76	43.11	9
	40	3	0.3	0.00	0.04	0.47	0.03	136.02	136.59	9
	100	3	0.3	0.01	0.02	1.02	0.06	41.94	43.13	9
	150	3	0.3	0.01	0.01	1.53	0.06	2.58	4.23	9
16	20	3	0.3	0.55	0.19	0.53	0.02	64.39	65.15	18
	40	3	0.3	0.01	0.06	0.79	0.06	1848.9	1849.9	16
	100	3	0.3	0.01	0.05	1.83	0.14	439.46	441.6	16
	150	3	0.3	0.01	0.11	3.01	0.16	2129.6	2133.1	18
Min				0.00	0.01	0.27	0.01	2.58	4.23	9.00
Max				0.55	0.19	3.01	0.16	2129.60	2133.10	18.00
Avg				0.07	0.06	1.18	0.07	588.21	589.60	13.00

Table 4.5: Performance of LagHeur for $c_j^E/c_j = 3$ and $U_j^E/U_j = 0.3$

The results in tables 4.2, 4.3, 4.4, and 4.5 show that the proposed Lagrangian heuristic achieves small gaps when the number of ULDs is small (8-16). However, when the number of ULDs increases the same is most probably true if the running time for the heuristic is increased. This clearly indicates that the Lagrangian bound is very tight. Also the results reveal that the performance of the Lagrangian heuristic does not differ considerably when the ratio of the over-pivot cost to the under-pivot cost c_j^E/c_j varies.

Give that Lagheur takes long to solve, we restrict its solution to just the first node and recuperate the feasible solution that Cplex generates in node 0. We refer to the heuristic as LagHeur0. The results are displayed in Tables 4.6 and 4.7. Table 4.6 displays statistics for $c_j^E/c_j = 1.2$, $U_j^E/U_j = 0.1$, and $c_j^E/c_j = 1.2$, $U_j^E/U_j = 0.3$. Table 4.7 displays statistics and computational performance for $c_j^E/c_j = 3$, $U_j^E/U_j = 0.1$, and $c_j^E/c_j = 3$, $U_j^E/U_j = 0.3$.

Compared to results in Section 4.2.1, the LagHeur0 is capable of solving the large instances and has a maximum gap of 6.36 in 37.61s. With respect to computational time, LagHeur0 takes more time compared to the 2knap and the bestfit heuristics.

		$c_j^E/c_j = 1.2, U_j^E/U_j = 0.1$			$c_j^E/c_j = 1.2, U_j^E/U_j = 0.3$		
ULDs	**Shipments**	**HeurGap**	**HeurTime**	**TotalTime**	**HeurGap**	**HeurTime**	**TotalTime**
8	20	0.05	0.54	0.82	0.12	0.42	0.71
	40	0.24	0.07	0.58	0.09	0.24	0.73
	100	0.01	1.46	2.47	0.12	1.2	2.21
	150	0.02	1.15	2.8	0.02	1.22	2.93
16	20	0.08	0.64	1.38	0.22	1.84	2.56
	40	0.13	2.55	3.96	1.44	2.55	3.56
	100	0.97	5.61	8.27	0.53	9.73	12.53
	150	0.42	8	11.94	0.32	9.85	13.79
40	100	0.72	92.05	100.69	0.81	80.5	88.57
	200	0.62	97.39	118.6	0.66	214.88	238.62
	300	0.63	22.83	42.55	0.81	37.02	56.94
	400	0.40	44.8	70.59	0.35	43.33	69.29
80	100	3.33	37.84	262.91	2.26	42.46	255.67
	200	0.93	429.85	462.12	0.85	448.52	482.53
	300	0.53	3835.9	4059.4	0.69	3197.2	3358.8
	400	0.48	1510.9	1606.5	0.56	1455.3	1544.2
100	150	2.76	43.02	483.93	0.18	265.62	487.92
	300	0.77	4702.3	4838.7	0.82	5091.8	5230.5
	600	0.52	475.63	638.01	0.82	467.52	632.16
	1000	0.63	637.31	937.49	0.68	568.13	886.11
Min		0.01	0.07	0.58	0.02	0.24	0.71
Max		3.33	4702.30	4838.70	2.26	5091.80	5230.50
Avg		0.71	597.49	682.69	0.62	596.97	668.52

Table 4.6: Performance of LagHeur0 for $c_j^E/c_j = 1.2, U_j^E/U_j = 0.1$ and $c_j^E/c_j = 1.2, U_j^E/U_j = 0.3$

Table 4.6 shows that when U_j^E/U_j increases to 0.3, LagHeur0 finds a better solution than when $U_j^E/U_j = 0.1$ in 7 cases, while the maximum gap achieved when $U_j^E/U_j = 0.3$ is 2.26 in 255.67s compared to a maximum gap of 3.33 in 262.91s when $U_j^E/U_j = 0.1$.

		$c_j^E/c_j = 3, U_j^E/U_j = 0.1$			$c_j^E/c_j = 3, U_j^E/U_j = 0.3$		
ULDs	**Shipments**	**HeurGap**	**HeurTime**	**TotalTime**	**HeurGap**	**HeurTime**	**TotalTime**
8	20	0.38	0.53	0.83	1.08	0.63	0.97
	40	1.48	0.08	0.58	0.46	0.67	1.26
	100	0.09	0.96	2.07	0.06	1.15	2.27
	150	0.03	1.15	2.78	0.02	1.32	3.01
16	20	2.14	0.87	1.53	1.80	0.8	1.41
	40	1.37	1.86	2.81	2.21	1.33	2.22
	100	1.61	4.23	6.77	1.34	4.69	6.96
	150	0.61	9.74	13.68	0.66	5.45	8.81
40	100	2.12	61.66	69.85	6.36	31.2	37.61
	200	1.82	82.54	102.54	1.56	94.46	113.87
	300	1.58	22.14	41.97	1.84	9.91	28.6
	400	1.40	39.53	65.5	1.07	41.88	67.3
80	100	3.09	223.5	246.66	4.39	166.15	187.44
	200	2.13	983.74	1033.8	2.41	1234	1292.6
	300	2.02	1032.3	1093.8	2.04	1060.6	1128.9
	400	1.41	946.34	1023.8	1.49	1000.4	1086.5
100	150	2.99	351.27	376.54	2.59	376.88	402.56
	300	1.81	4119.6	4234.8	2.18	4182.4	4299.9
	600	2.03	244.35	391.39	1.90	350.12	507.13
	1000	1.65	779.53	1109	1.47	788.65	1108.9
Min		0.03	0.08	0.58	0.02	0.63	0.97
Max		3.09	4119.60	4234.80	6.36	4182.40	4299.90
Avg		1.59	445.30	491.04	1.85	467.63	514.41

Table 4.7: Performance of LagHeur0 for $c_j^E/c_j = 3, U_j^E/U_j = 0.1$ and $c_j^E/c_j = 3, U_j^E/U_j = 0.3$

Table 4.7 shows that when U_j^E/U_j increases to 0.3, the LagHeur0 solution maximum gap increases more than when $U_j^E/U_j = 0.1$ in 8 cases, and the maximum gap achieved when $U_j^E/U_j = 0.3$ is 6.36 in 37.61s compared to a maximum gap of 3.09 in 246.66s when $U_j^E/U_j = 0.1$. Comparing Tables 4.6 and 4.7 when c_j^E/c_j increases to 3, an increase of the average heuristic gap and maximum gap is noticed. In Table 4.6, the average gap is 0.71 and when c_j^E/c_j increases to 3, the average gap is 1.59 in Table 4.7. Also, the average gap in Table 4.6 is 0.62 while the average gap decreases to 1.85 in Table 4.7.

4.2.2 The Two-knapsack Heuristic (2knap)

Tables 4.8 and 4.9 display statistics for $c_j^E/c_j = 1.2$, $U_j^E/U_j = 0.1$, and $c_j^E/c_j = 1.2$, $U_j^E/U_j = 0.3$, and for $c_j^E/c_j = 3$, $U_j^E/U_j = 0.1$, and $c_j^E/c_j = 3$, $U_j^E/U_j = 0.3$, respectively.

Tables 4.8 and 4.9 reveal that 2knaps achieves better quality solutions than LagHeur. For instances with more than 16 ULDs and 150 shipments, the 2knap heuristic achieves a maximum gap of 5.73 in less than 78.05s. With respect to solution time the 2knap heuristic can finish all computations within 200.95s.

		$c_j^E/c_j = 1.2, U_j^E/U_j = 0.1$			$c_j^E/c_j = 1.2, U_j^E/U_j = 0.3$		
ULDs	**Shipments**	**HeurGap**	**HeurTime**	**TotalTime**	**HeurGap**	**HeurTime**	**TotalTime**
8	20	3.61	0	0.32	1.23	0	0.29
	40	0.50	0	0.48	0.73	0.01	0.54
	100	0.30	0.01	1	0.73	0	1.07
	150	0.12	0.02	1.67	0.12	0.01	1.69
16	20	3.66	0.01	0.63	2.77	0.04	0.61
	40	0.41	0.01	1.01	0.57	0	1.01
	100	1.32	0.02	2.74	1.30	0.03	2.87
	150	0.67	0.03	3.97	0.65	0.03	4.02
40	100	1.06	0.4	8.93	1.25	0.22	8.18
	200	1.40	0.4	20.05	1.38	0.62	24.79
	300	1.03	0.12	19.8	1.11	0.12	19.89
	400	0.93	0.15	26	0.92	0.14	26.04
80	100	2.49	2.89	22.94	1.88	2.75	22.3
	200	0.46	1.08	33.36	0.52	0.84	35.13
	300	1.28	4.9	200.95	1.32	4.77	171.18
	400	1.35	1.99	88.82	0.88	2.49	93.05
100	150	2.41	2.83	29.8	0.65	1.05	27.97
	300	1.03	4.1	139.07	1.59	4.1	142.35
	600	1.35	1.68	164.4	1.70	1.74	169.4
	1000	0.87	2.94	303.02	0.86	2.94	328.56
Min		0.12	0.00	0.32	0.12	0.00	0.29
Max		3.66	4.90	303.02	2.77	4.77	328.56
Avg		1.31	1.18	53.45	1.11	1.10	54.05

Table 4.8: Performance of 2knap for $c_j^E/c_j = 1.2, U_j^E/U_j = 0.1$ and $c_j^E/c_j = 1.2, U_j^E/U_j = 0.3$

Tables 4.8 shows that when the ratio of U_j^E/U_j increases to 0.3, the 2knap heuristic finds better solutions than when the ratio of $U_j^E/U_j = 0.1$ in seven cases, while the maximum gap achieved when ratio of $U_j^E/U_j = 0.3$ is 2.77 in 0.61s compared to a maximum gap of 3.66 in 0.63s when ratio $U_j^E/U_j = 0.1$.

		$c_j^E/c_j = 3, U_j^E/U_j = 0.1$			$c_j^E/c_j = 3, U_j^E/U_j = 0.3$		
ULDs	**Shipments**	**HeurGap**	**HeurTime**	**TotalTime**	**HeurGap**	**HeurTime**	**TotalTime**
8	20	0.22	0	0.3	0.22	0.01	0.32
	40	1.51	0	0.55	0.99	0	0.56
	100	0.07	0.02	1.15	0.07	0	1.15
	150	0.12	0.02	1.64	0.12	0.01	1.71
16	20	5.73	0	0.67	2.66	0	0.56
	40	0.19	0.01	1	2.05	0	0.91
	100	3.32	0.01	2.55	1.38	0.01	2.22
	150	0.73	0.04	4.01	0.74	0.01	3.34
40	100	1.26	0.21	8.33	1.50	0.09	6.44
	200	2.20	0.39	20.45	3.35	0.24	18.39
	300	2.28	0.14	19.75	3.47	0.04	18.62
	400	2.64	0.15	26	2.37	0.16	25.51
80	100	2.31	1.21	22.69	3.77	0.4	21.23
	200	0.35	1.95	52.45	0.71	2.06	60.57
	300	4.03	1.38	62.45	3.57	1.63	68.8
	400	2.33	1.53	78.05	2.80	1.94	87.75
100	150	2.92	0.72	26.12	4.14	0.32	25.95
	300	2.02	3.48	122.86	2.03	3.53	125.97
	600	2.68	1.01	148.29	3.31	1.36	158.31
	1000	2.11	3.98	347	2.10	3.3	320.96
Min		0.07	0.00	0.30	0.07	0.00	0.32
Max		5.73	3.98	347.00	4.14	3.53	320.96
Avg		1.95	0.81	47.32	2.07	0.76	47.46

Table 4.9: Performance of 2knap for $c_j^E/c_j = 3, U_j^E/U_j = 0.1$ and $c_j^E/c_j = 3, U_j^E/U_j = 0.3$

Similarly, Table 4.9 shows that when the ratio of U_j^E/U_j increases to 0.3, the 2knap heuristic finds better solutions than when $U_j^E/U_j = 0.1$ in six cases, while the maximum gap achieved when $U_j^E/U_j = 0.3$ is 4.14 in 25.95s compared to a maximum gap of 5.73 in 0.67en $U_j^E/U_j = 0.1$. We notice that when c_j^E/c_j increases to 3, the average heuristic gap and the maximum gap is noticed. In Table 4.8, the average gap is 1.31 and when c_j^E/c_j increases to 3, the average gap is 1.95 in Table 4.9. Also, the average gap in Table 4.8 is 1.11, while the average gap increases to 2.07 in Table 4.9.

4.2.3 The Bestfit Heuristic (bestfit)

Tables 4.10 and 4.11 show results of the bestfit heuristic. Table 4.10 displays statistics for $c_j^E/c_j = 1.2$, $U_j^E/U_j = 0.1$, and $c_j^E/c_j = 1.2$, $U_j^E/U_j = 0.3$. Table 4.11 displays statistics and computational performance for $c_j^E/c_j = 3$, $U_j^E/U_j = 0.1$, and $c_j^E/c_j = 3$, $U_j^E/U_j = 0.3$.

Compared to the results in Section 4.2.1, the bestfit heuristic achieves better solution quality than the LagHeur, also for cases with more than 16 ULDs and 150 shipments, the bestfit heuristic has a maximum gap of 5.73 in very fast time at 0.66s. With respect to computational time, the bestfit heuristic terminates within 348.46s.

When comparing the 2knap heuristic and the bestfit heuristic both have a maximum gap of 5.73 in 0.67 and 0.66, respectively, but the 2knap heuristic is better in 45 out of 80 cases compared with the 35 cases where bestfit heuristic achieves better gaps.

		$c_j^E/c_j = 1.2, U_j^E/U_j = 0.1$			$c_j^E/c_j = 1.2, U_j^E/U_j = 0.3$		
ULDs	**Shipments**	**HeurGap**	**HeurTime**	**TotalTime**	**HeurGap**	**HeurTime**	**TotalTime**
8	20	0.13	0	0.32	1.21	0	0.3
	40	4.60	0.01	0.49	4.60	0	0.52
	100	2.35	0.02	1.15	2.38	0.01	1.02
	150	0.12	0.01	1.7	0.12	0.01	1.66
16	20	5.40	0	0.56	0.84	0.02	0.6
	40	2.71	0.04	1.05	3.31	0.01	1.03
	100	1.93	0.03	2.8	1.93	0.03	2.86
	150	1.30	0.05	3.99	1.30	0.03	4.01
40	100	1.30	0.31	8.78	1.33	0.31	8.22
	200	0.84	0.57	22.11	0.82	0.74	24.93
	300	0.75	0.14	19.98	0.75	0.15	19.96
	400	0.63	0.18	26.05	0.63	0.17	26.13
80	100	2.61	1.12	20.01	1.65	1.89	20.81
	200	1.07	0.89	33.22	1.19	1.19	35.24
	300	1.00	10.67	202.93	1.19	9.33	161.63
	400	0.93	3.8	95.98	0.88	4.01	93.04
100	150	3.34	2.23	28.87	0.36	1.96	29.04
	300	1.59	10.4	145.42	1.59	10.54	147.97
	600	0.88	3.18	165.8	0.96	3.3	170.64
	1000	0.35	3.82	304.35	0.35	3.05	320.95
Min		0.12	0.00	0.32	0.12	0.00	0.30
Max		5.40	10.67	304.35	4.60	10.54	320.95
Avg		1.69	1.87	54.28	1.37	1.84	53.53

Table 4.10: Performance of bestfit for $c_j^E/c_j = 1.2, U_j^E/U_j = 0.1$ and $c_j^E/c_j = 1.2, U_j^E/U_j = 0.3$

Table 4.10 shows that when the ratio of U_j^E/U_j increases to 0.3 while c_j^E/c_j is fixed at 1.2, the bestfit heuristic finds better solutions when the ratio of $U_j^E/U_j = 0.1$, and the maximum gap achieved is 4.6 in 0.52s.

		$c_j^E/c_j = 3, U_j^E/U_j = 0.1$			$c_j^E/c_j = 3, U_j^E/U_j = 0.3$		
ULDs	**Shipments**	**HeurGap**	**HeurTime**	**TotalTime**	**HeurGap**	**HeurTime**	**TotalTime**
8	20	2.51	0	0.29	0.22	0	0.33
	40	2.87	0	0.52	0.99	0	0.58
	100	0.07	0.02	1.13	0.07	0	1.13
	150	0.12	0.02	1.67	0.12	0.01	1.66
16	20	5.73	0	0.66	5.71	0	0.57
	40	0.19	0.02	1.04	3.43	0	0.95
	100	1.98	0.02	2.58	1.38	0.01	2.28
	150	0.73	0.03	3.96	0.74	0.02	3.33
40	100	1.35	0.26	8.29	1.43	0.11	6.43
	200	0.86	0.47	20.34	0.78	0.34	18.94
	300	0.75	0.17	19.8	0.75	0.05	18.74
	400	0.63	0.17	26.05	0.63	0.2	25.52
80	100	4.45	2.19	23.79	4.43	1.96	21.93
	200	0.74	2.53	53	0.71	3.14	61.92
	300	1.12	2.72	63.33	1.57	3.63	70.26
	400	0.87	2.53	79.01	0.87	3.35	85.56
100	150	2.93	0.65	25.9	3.06	0.64	26.08
	300	1.90	8.54	127.65	1.88	9.05	131.1
	600	1.00	1.82	148.88	0.97	2.54	160.58
	1000	0.36	5.39	348.46	0.35	4.32	333.59
Min		0.07	0.00	0.29	0.07	0.00	0.33
Max		5.73	8.54	348.46	5.71	9.05	333.59
Avg		1.56	1.38	47.82	1.50	1.47	48.57

Table 4.11: Performance of bestfit for $c_j^E/c_j = 3, U_j^E/U_j = 0.1$ and $c_j^E/c_j = 3, U_j^E/U_j = 0.3$

Table 4.11 shows that when the ratio of U_j^E/U_j increases to 0.3, the bestfit heuristic finds a better solution than when the ratio of $U_j^E/U_j = 0.1$ in 10 cases, while the maximum gap achieved when ratio of $U_j^E/U_j = 0.3$ is 5.71 in 0.57s compared to a maximum gap of 5.73 in 0.66s when $U_j^E/U_j = 0.1$. Comparing Tables 4.10 and 4.11 when the ratio c_j^E/c_j increases to 3, a decrease of the average heuristic gap and an increase of the maximum gap is noticed. In Table 4.10 the average gap is 1.69 and when the ratio c_j^E/c_j increases to 3, the average gap is 1.56 in Table 4.11. Also, the average gap in Table 4.10 is 1.37 while the average gap increases to 1.50 in Table 4.11.

In comparison to the 2knap heuristic, we notice that LagHeur0 achieves better maximum gap in almost every data set except in one when $c_j^E/c_j = 3$ and $U_j^E/U_j = 0.3$. Also, when comparing with the bestfit heuristic we observe that LagHeur0 dominates since it gives better maximum gap for every instance.

4.3 Comparison of Heuristics Performance

In this section, we compare the total computational time for each proposed heuristic by fixing the number of ULD and the four combinations of c_j^E/c_j and U_j^E/U_j ratios.

Figures 4.1, 4.2, 4.3 and 4.4 compare computational time achieved by the three heuristics, 2knap, bestift, and LagHeur0 for $c_j^E/c_j = 1.2$, $U_j^E/U_j = 0.1$, $c_j^E/c_j = 1.2$, $U_j^E/U_j = 0.3$, $c_j^E/c_j = 3$, $U_j^E/U_j = 0.1$, $c_j^E/c_j = 3$, and $U_j^E/U_j = 0.3$, respectively.

Figure 4.1 displays the total computational time for the three heuristics at $c_j^E/c_j = 1.2$ and $U_j^E/U_j = 0.1$. We observe that the 2knap and the bestfit heuristics have a better time performance than LagHeur0, especially when the number of shipments increases above 40, and 100 instances. In some cases, the 2knap heuristic achieves better performance than the bestfit heuristic. For example in Figure 4.1, and for 8 ULD and 100 shipments, we notice a slight time improvement in the performance of 2knap heuristic over the bestfit heuristic.

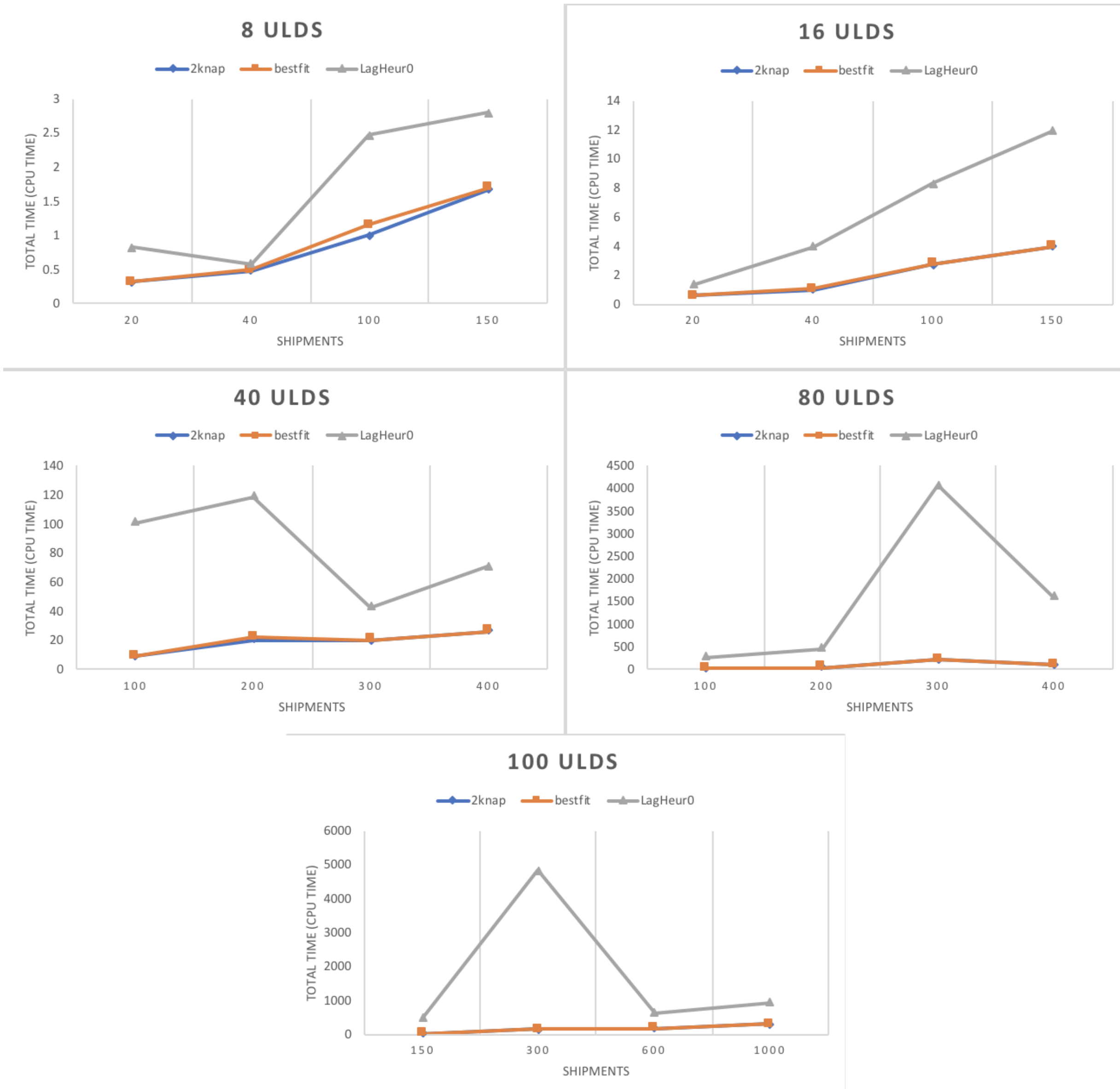


Figure 4.1: Heuristics Total Time Comparison for $c_j^E/c_j = 1.2, U_j^E/U_j = 0.1$.

Similar results can be realized in Figure 4.2 that shows the time performance comparison for all three heuristics for $c_j^E/c_j = 1.2$ and $U_j^E/U_j = 0.3$. Both the 2knap heuristic and the bestfit heuristic have better time performance than LagHeur0. Also, both the 2knap heuristic and the bestfit heuristic have nearly identical time performance.

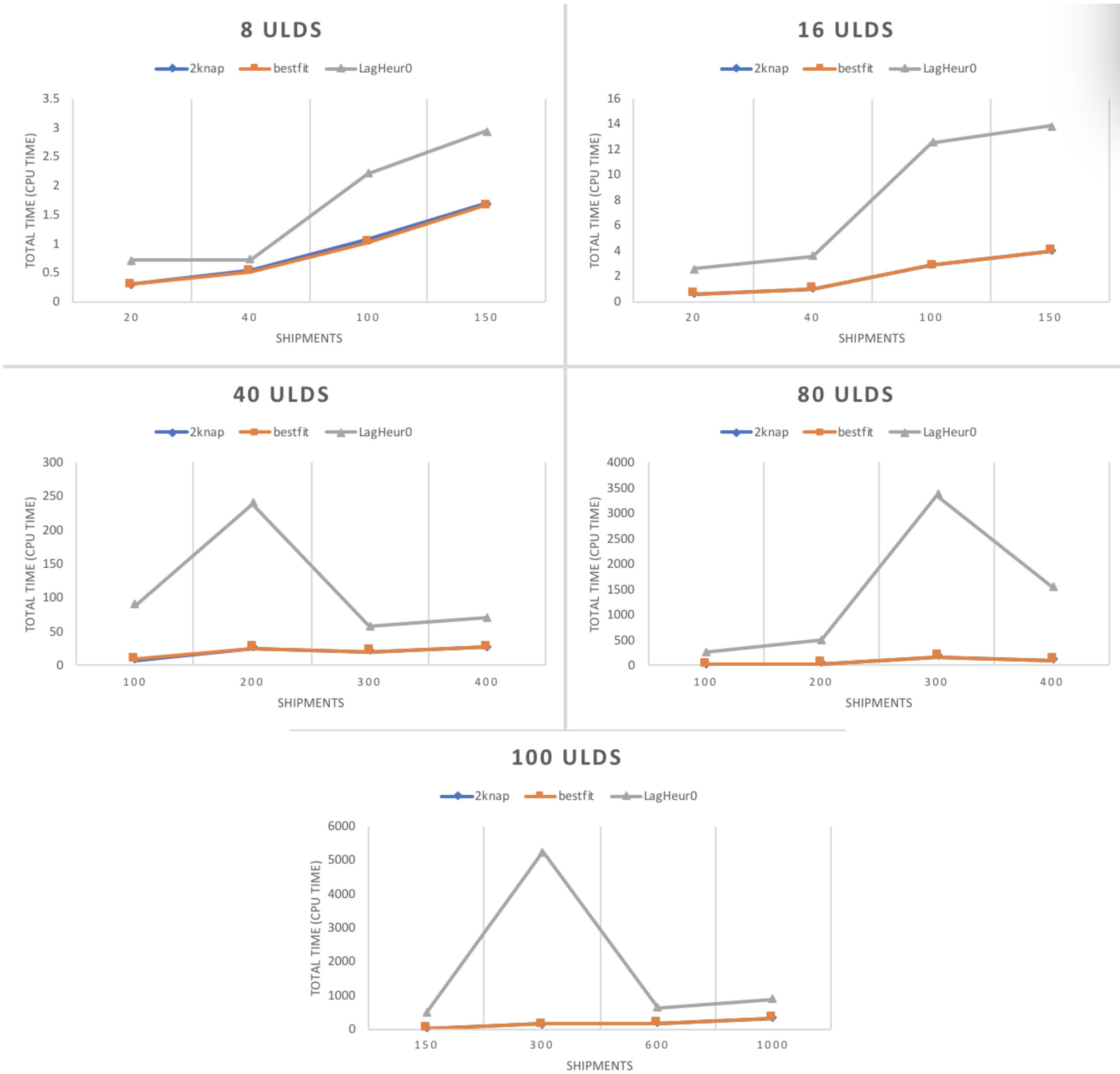


Figure 4.2: Heuristics Total Time Comparison for $c_j^E/c_j = 1.2, U_j^E/U_j = 0.3$.

Figure 4.3 displays the total computational time for the three heuristics and $c_j^E/c_j = 3$

and $U_j^E/U_j = 0.1$.

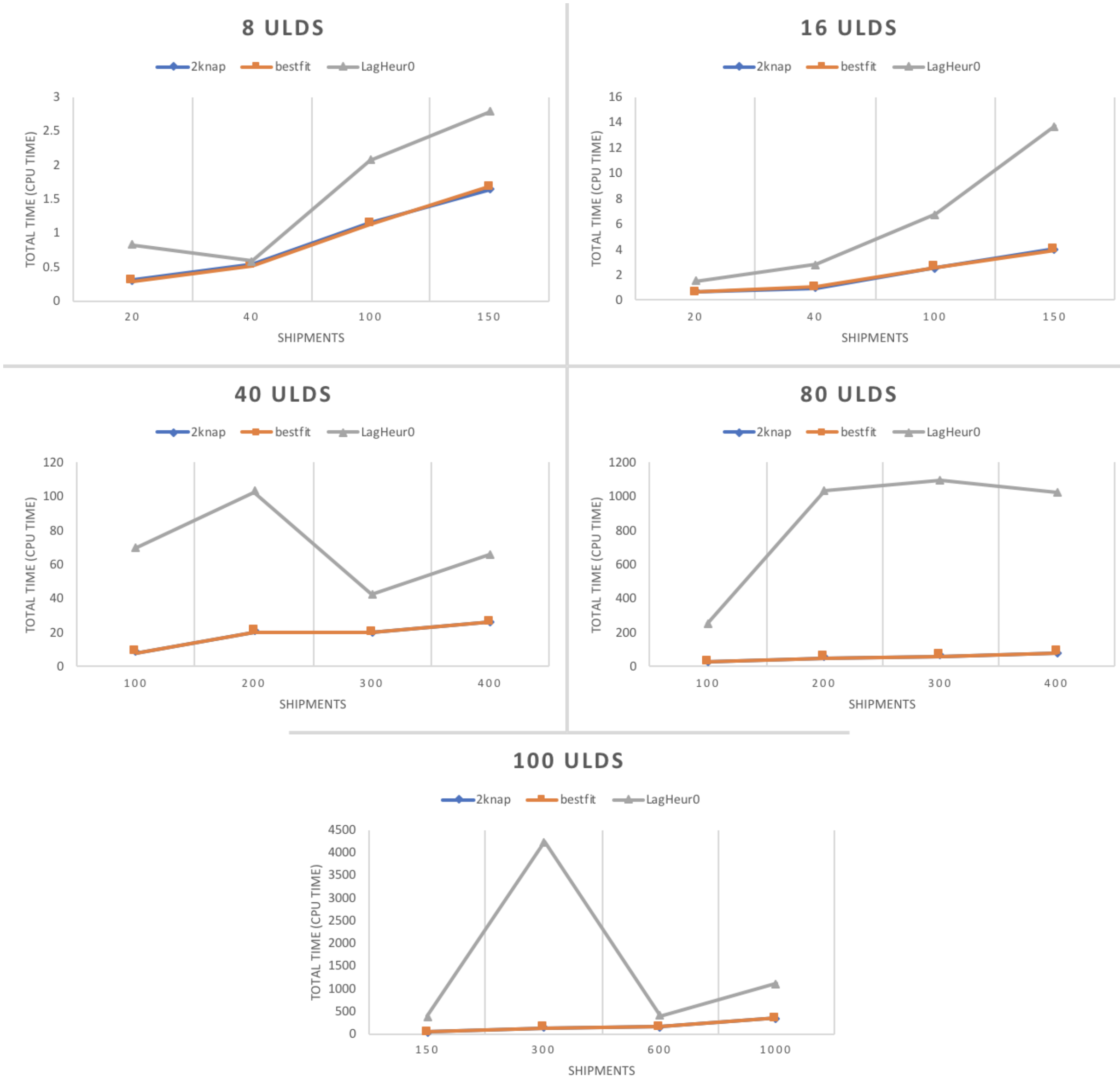


Figure 4.3: Heuristics Total Time Comparison for $c_j^E/c_j = 3, U_j^E/U_j = 0.1$.

Figure 4.4 displays the total computational time for the three heuristics and $c_j^E/c_j = 3$ and $U_j^E/U_j = 0.3$.

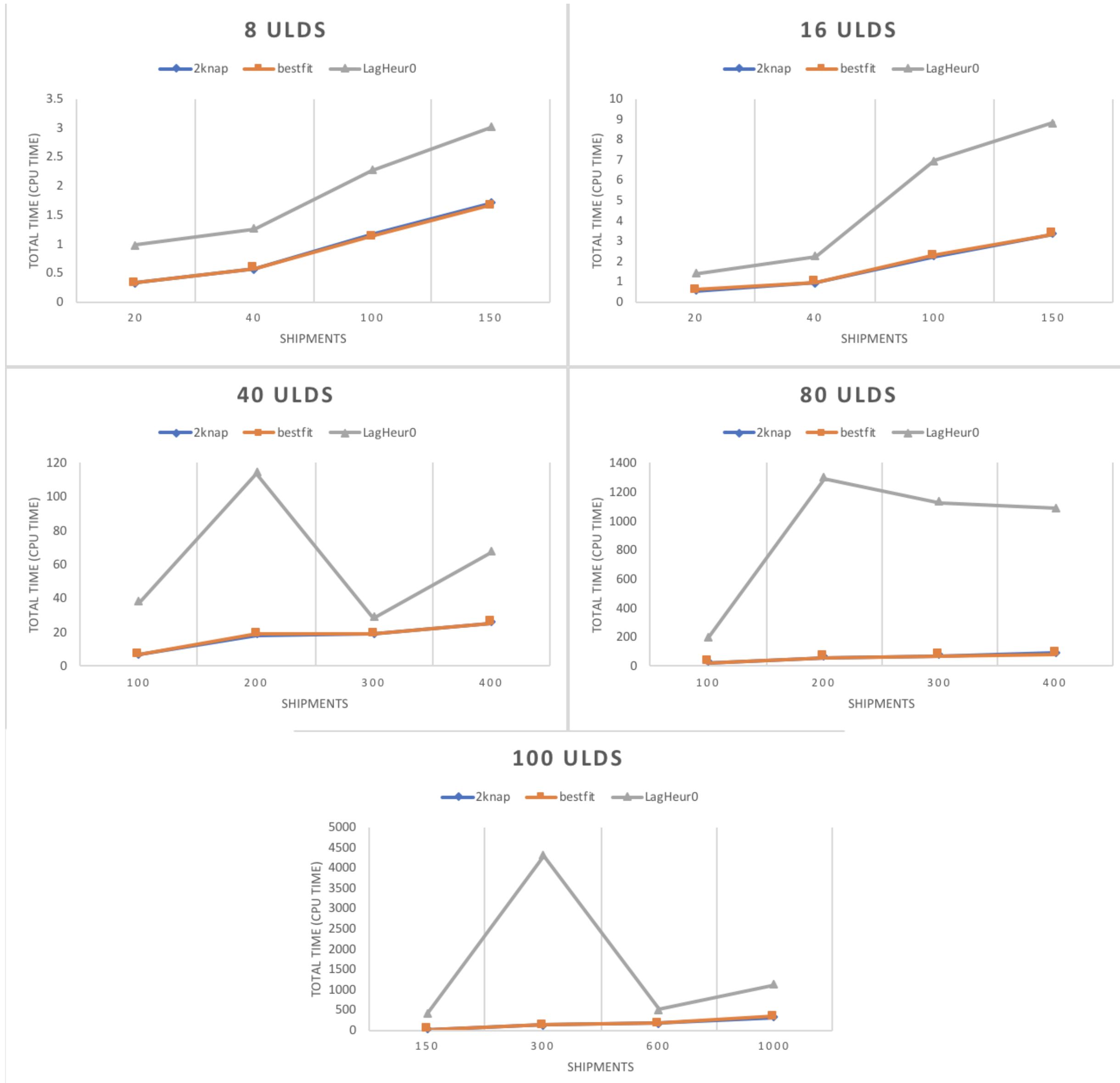


Figure 4.4: Heuristics Total Time Comparison for $c_j^E/c_j = 3, U_j^E/U_j = 0.3$.

Figures 4.1, 4.2, 4.3, and 4.4 show that both the 2knap heuristic and the bestfit heuristic have better time performance regardless of the change in over-pivot cost ratio c_j^E/c_j and over-pivot capacity ratio U_j^E/U_j than LagHeur0. However, the LagHeur0 produces a better gap in almost every test-case.

4.4 Comparison of Heuristics Gaps

In this section, we compare the optimality gaps for each proposed heuristic for for specific number of ULD and the four combinations of c_j^E/c_j and U_j^E/U_j ratios.

Figures 4.5, 4.6, 4.7 and 4.8 compare the average gaps achieved by the three heuristics, 2knap, bestift, and LagHeur0 for $c_j^E/c_j = 1.2$, $U_j^E/U_j = 0.1$, $c_j^E/c_j = 1.2$, $U_j^E/U_j = 0.3$, $c_j^E/c_j = 3$, $U_j^E/U_j = 0.1$, $c_j^E/c_j = 3$, and $U_j^E/U_j = 0.3$, respectively.

Figure 4.5 displays the gaps for the three heuristics at $c_j^E/c_j = 1.2$ and $U_j^E/U_j = 0.1$. We can observe a smaller gaps for LagHeur0 compared to both the 2knap heuristic and the bestfit heuristic, especially when the number of shipments increases above 40, and 100 instances. In some cases, the 2knap heuristic achieves better gap than the bestfit heuristic and the LagHeur0. For example in Figure 4.5, and for 80 ULD and 100 or 200 shipments, we notice a better gap achieved by the 2knap heuristic compared to bestfit and LagHeur0.

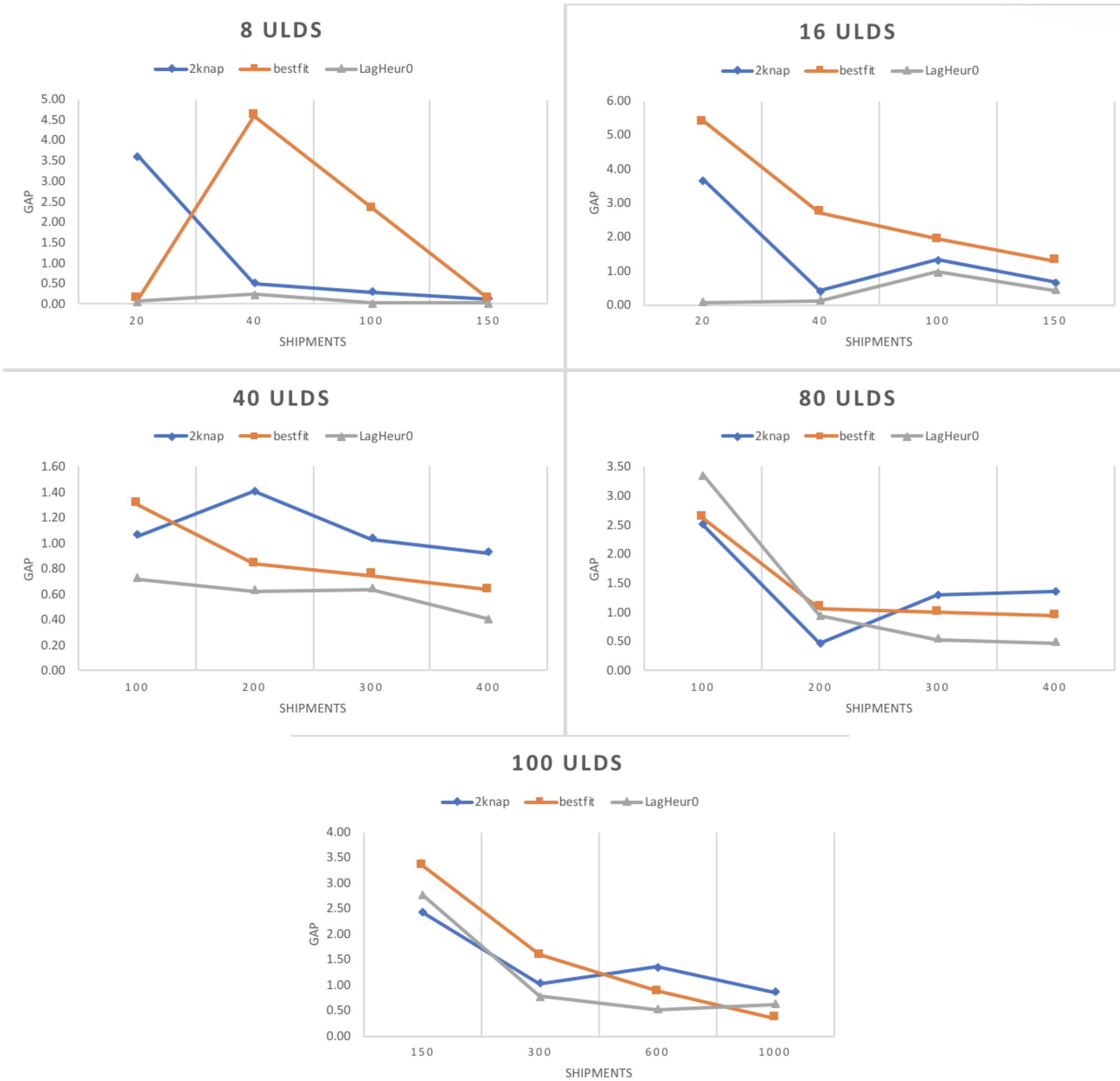


Figure 4.5: Heuristics Gaps Comparison for $c_j^E/c_j = 1.2, U_j^E/U_j = 0.1$.

Similar results can be observed in Figure 4.6. It shows the gaps for all three heuristics when using the ratios $c_j^E/c_j = 1.2$ and $U_j^E/U_j = 0.3$. LagHeur0 has better gaps compared to both the 2knap heuristic and the bestfit heuristic. Also, for most instances the 2knap heuristic achieves better gaps than the bestfit heuristic.

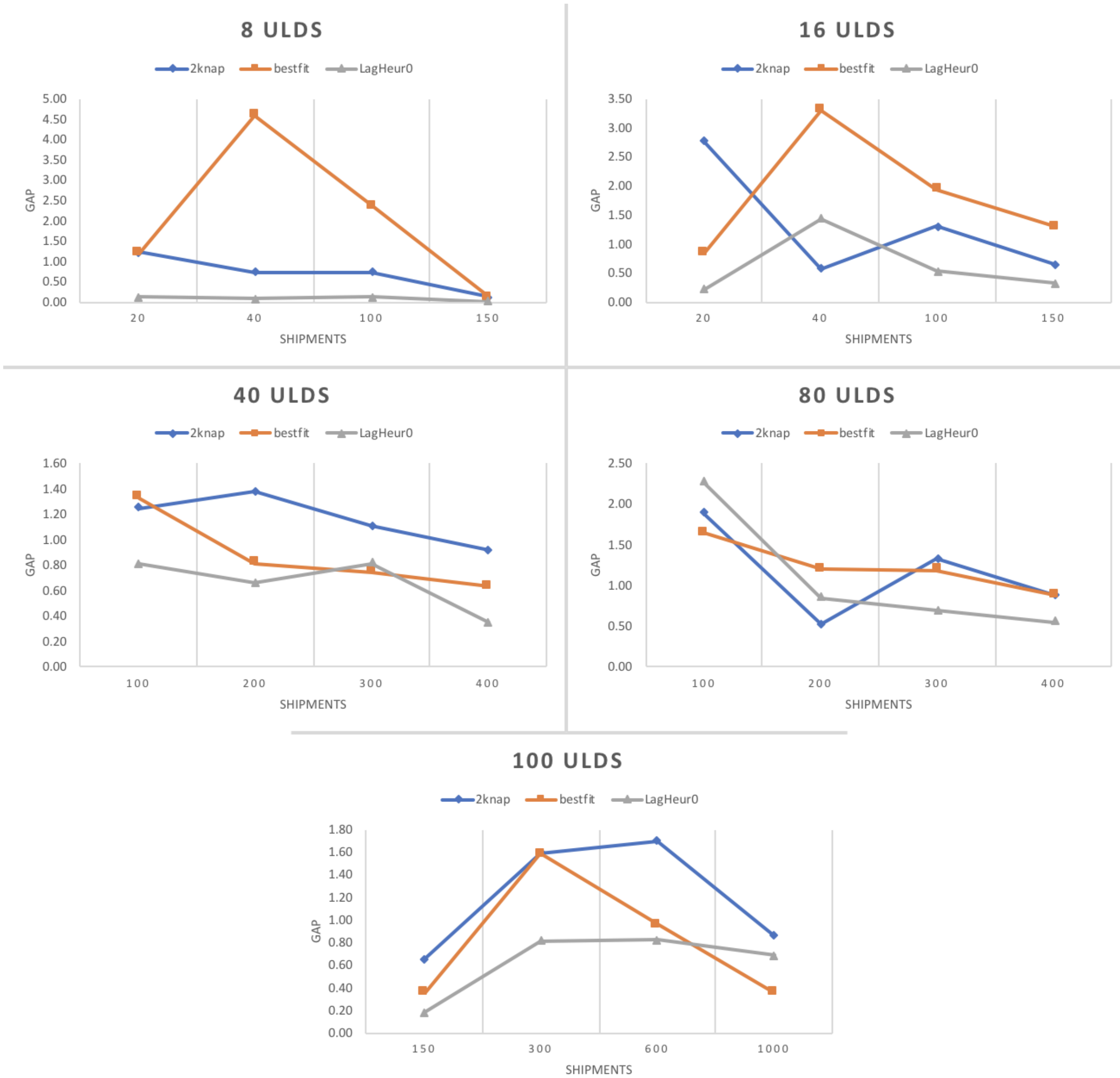


Figure 4.6: Heuristics Gaps Comparison for $c_j^E/c_j = 1.2, U_j^E/U_j = 0.3$.

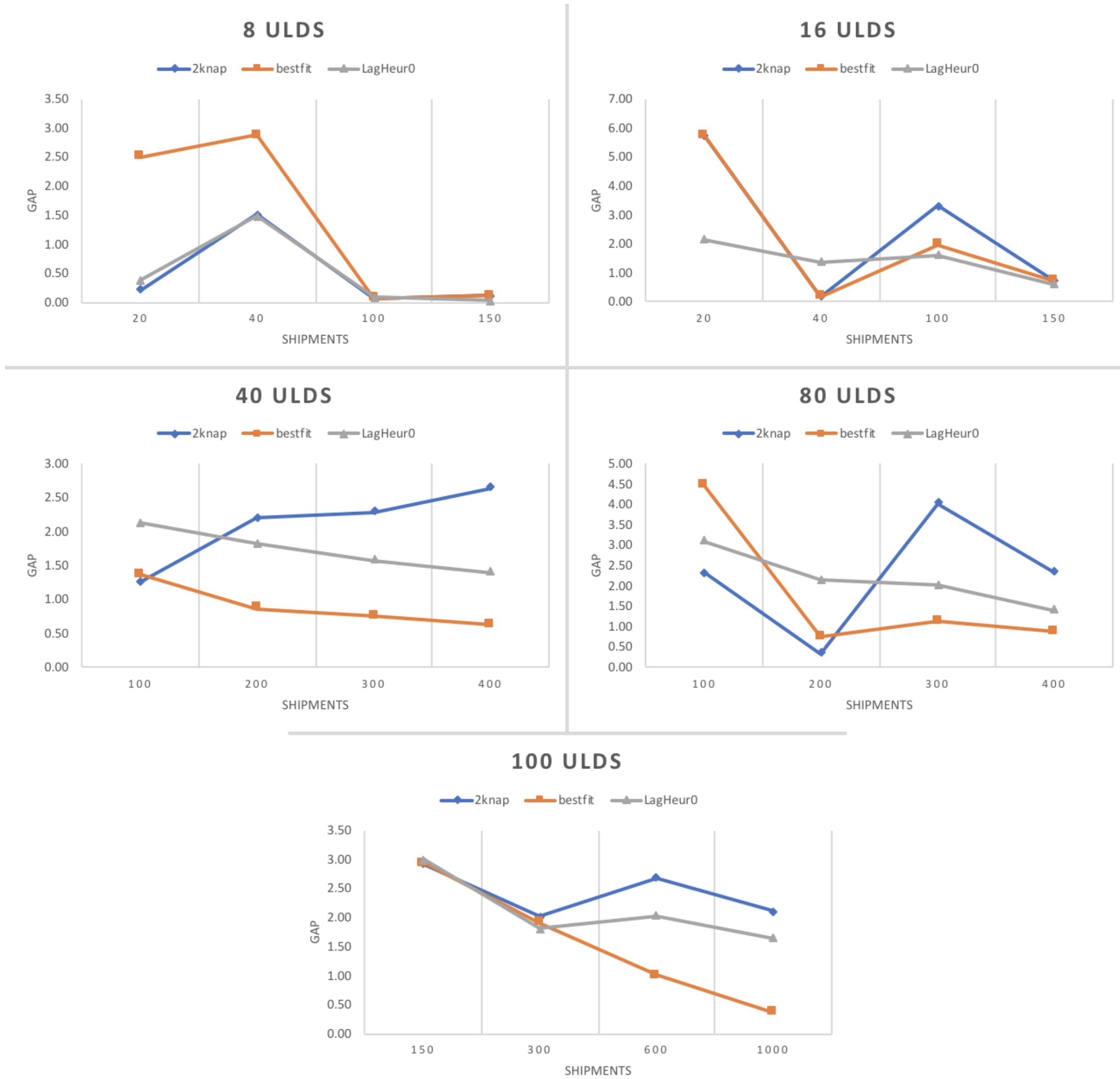


Figure 4.7: Heuristics Total Time Comparison for $c_j^E/c_j = 3, U_j^E/U_j = 0.1$.

Figures 4.5, 4.6, 4.7, and 4.8 show that the LagHeur0 produces a better gaps in almost every test-case.

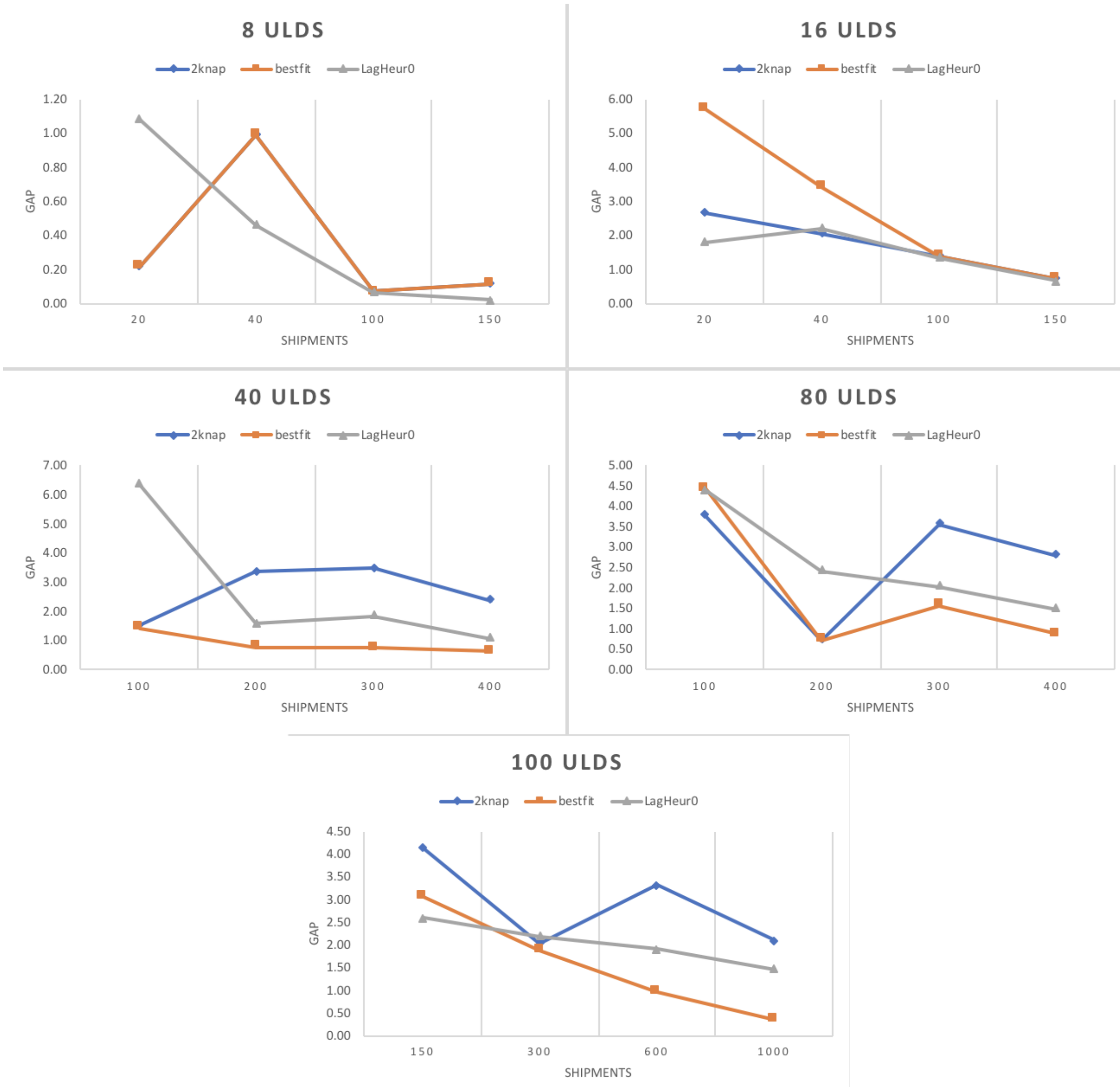


Figure 4.8: Heuristics Gaps Comparison for $c_j^E/c_j = 3, U_j^E/U_j = 0.3$.

Chapter 5

Conclusion

A standard practice in airfreight transportation is based on Unit Loading Devices (ULDs). Airlines provide a set of ULDs with different sizes, cost structure, and requirements for forwarders to rent. For reasons related to aircraft safety and logistics efficiency, the weight of ULDs is limited to a certain threshold, called the pivot-weight. The limit is not hard as forwarders are allowed to exceed it by paying a premium. Essentially, the cost of using a ULD consists of a fixed rental charge and a variable under-pivot rate if the total weight is below the pivot-weight. Weight beyond the pivot weight is charged at the higher over-pivot rate. Airfreight forwarders are interested in finding the optimal consolidation plan to minimize total cost.

Although the problem can be easily modelled as an integer program, the solution of realistic size instances is time consuming. In this work, we propose a Lagrangian approach that is capable of finding high quality lower bounds, often very close to the optimal, in very short times. We use Lagrangian relaxation approach and propose three heuristics based on the partial subproblem solutions. The first takes the solution of one of the subproblems and solves a restricted version of the original problem (LagHeur). The other two are two knapsack based heuristic (2knap) and a best-fit greedy heuristic (bestfit). Problems with up to 100 ULDs and 1000 shipments are solved to within an average of 1%, 2%, 2% of optimality in less than 51.05s, 50.57s and 589.16s by bestfit, 2knap and LagHeur, respectively.

Future research can focus on devising other heuristic approaches based on the partial subproblem solutions or add additional practical constraints. For example, it is assumed that cargo will fit in ULDs as long as the ULD volume is sufficient to cover the total volume of cargo. In practice, the shape of cargo will not allow this, necessitating the inclusion of three-dimensional packing constraints in the formulation.

References

Boeing (2016). World Air Cargo Forecast 2016–2017. `https://bit.ly/1Ok9KwL`, Accessed on 15/9/2017.

Bookbinder, J. H., Elhedhli, S., and Li, Z. (2015). The air-cargo consolidation problem with pivot weight: models and solution methods. *Computers & Operations Research*, 59:22–32.

Chang, Y.-H., Yeh, C.-H., and Wang, S.-Y. (2007). A survey and optimization-based evaluation of development strategies for the air cargo industry. *International Journal of Production Economics*, 106(2):550–562.

Chao, C.-C. and Li, R.-G. (2017). Effects of cargo types and load efficiency on airline cargo revenues. *Journal of Air Transport Management*, 61:26–33.

Derigs, U., Friederichs, S., and Schäfer, S. (2009). A new approach for air cargo network planning. *Transportation Science*, 43(3):370–380.

Feng, B., Li, Y., and Shen, H. (2015a). Tying mechanism for airlines' air cargo capacity allocation. *European Journal of Operational Research*, 244(1):322–330.

Feng, B., Li, Y., and Shen, Z.-J. M. (2015b). Air cargo operations: Literature review and comparison with practices. *Transportation Research Part C: Emerging Technologies*, 56:263–280.

Fisher, M. L. (1973). Optimal solution of scheduling problems using lagrange multipliers: Part i. *Operations Research*, 21(5):1114–1127.

Fisher, M. L. (2004). The Lagrangian relaxation method for solving integer programming problems. *Management Science*, 50(12):1861–1871.

Frangioni, A. (2005). About Lagrangian methods in integer optimization. *Annals of Operations Research*, 139(1):163–193.

Gehring, H., Menschner, K., and Meyer, M. (1990). A computer-based heuristic for packing pooled shipment containers. *European Journal of Operational Research*, 44(2):277–288.

Hai, T. H. H. (2016). Air cargo loading management systems for logistics forwarders.

Held, M. and Karp, R. M. (1970). The traveling-salesman problem and minimum spanning trees. *Operations Research*, 18(6):1138–1162.

Huang, K. and Chi, W. (2007). A Lagrangian relaxation based heuristic for the consolidation problem of airfreight forwarders. *Transportation Research Part C: Emerging Technologies*, 15(4):235–245.

IATA (2006). IATA economics briefing, air freight 2006 brighter skies ahead. *International Journal of Production Economics*.

IATA (2010). IATA ULD technical manual. International Air Transport Association, Montreal.

Li, Y., Tao, Y., and Wang, F. (2009). A compromised large-scale neighborhood search heuristic for capacitated air cargo loading planning. *European Journal of Operational Research*, 199(2):553–560.

Limbourg, S., Schyns, M., and Laporte, G. (2012). Automatic aircraft cargo load planning. *Journal of the Operational Research Society*, 63(9):1271–1283.

Lurkin, V. and Schyns, M. (2015). The airline container loading problem with pickup and delivery. *European Journal of Operational Research*, 244(3):955–965.

Matear, S. and Gray, R. (1993). Factors influencing freight service choice for shippers and freight suppliers. *International Journal of Physical Distribution & Logistics Management*, 23(2):25–35.

Mostaghimi Ghomi, H., St Amour, B. G., and Abdul-Kader, W. (2017). Three-dimensional container loading: A simulated annealing approach. *International Journal of Applied Engineering Research*, 12(7):1290.

Paquay, C., Schyns, M., and Limbourg, S. (2012). Three dimensional bin packing problem applied to air transport.

Pisinger, D. (2002). Heuristics for the container loading problem. *European Journal of Operational Research*, 141(2):382–392.

Reynolds-Feighan, A. J. (2008). Air-freight logistics. In *A.M Brewer et al (ed.). Handbook of Logistics and Supply-Chain Management*, pages 431–438. Emerald Group Publishing Limited.

Roesener, A. G. and Barnes, J. W. (2016). An advanced tabu search approach to the dynamic airlift loading problem. *Logistics Research*, 9(1):12.

Rohlig (2018). Air freight containers, pallets, unit load devices (uld).

Shepherd, B., Shingal, A., and Raj, A. (2016). Value of air cargo: Air transport and global value chains. *Montreal: The International Air Transport Association (IATA).*

Vancroonenburg, W., Verstichel, J., Tavernier, K., and Berghe, G. V. (2014). Automatic air cargo selection and weight balancing: a mixed integer programming approach. *Transportation Research Part E: Logistics and Transportation Review*, 65:70–83.

Wong, W. H., Zhang, A., Van Hui, Y., and Leung, L. C. (2009). Optimal baggage-limit policy: airline passenger and cargo allocation. *Transportation Science*, 43(3):355–369.

Wu, Y. (2008). Modelling containerisation of air cargo forwarding problems. *Production Planning and Control*, 19(1):2–11.

Wu, Y. (2010). A dual-response forwarding approach for containerizing air cargoes under uncertainty, based on stochastic mixed 0-1 programming. *European Journal of Operational Research*, 207(1):152–164.

Xue, J. and Lai, K. (1997). A study on cargo forwarding decisions. *Computers & Industrial Engineering*, 33(1):63–66.