A Relax-and-Fix based Approach for Solving Heterogenous Fleet Vehicle Routing Problem with Three-Dimensional Loading Constraints

Gerardo Guedes Saraiva de Menezes

Master's Dissertation

Supervisors: Prof. Elsa Marília da Costa Silva
Dr. Khadija Hadj Salem

U. PORTO
FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

Integrated Master in Mechanical Engineering

2021-09-20

Abstract

Many companies has to deliver products to their clients, which is performed by either internal or external logistics. Because this transport is paramount and also a big share of companies' operational costs, the implementation of routing methods and optimization is necessary as a cost saving method.

Motivated by this, the work in this dissertation was developed as a means to try to offer an alternative to more traditional routing methods. At first, a mathematical model was developed in the form of a MILP. This model provided the possibility to optimize the routes of vehicles belonging to a depot by introducing realistic constraints relative to the 3D-loading of the vehicles. Having this, the decision was made to develop and implement a Relax-and-Fix heuristic that reduces solving times and improves solution quality for the cases where optimality is not reached. The heuristic was then thoroughly tested against the MILP and the influence of two important constraints was also evaluated.

It was observed that the heuristic generally provides better-quality solutions in considerably less time than the MILP is able to find with a 3600 seconds time limit. Additionally, it was possible to establish that using a distance-focused objective function is more efficient for routing optimization than a cost-focused one.

Key Words: Vehicle Routing Problem with 3D loading constraints, Heterogeneous Fleet, Full support, Fragility, Relax-and-Fix, Mixed Integer Linear Programming.

Resumo

Uma grande parte das empresas tem que entregar produtos aos seus clientes, o que é feito tanto por logística interna como externa. Visto que o transporte é fundamental e que representa uma grande parte dos custos operacionais das organizações, a implementação de métodos de definição e otimização de rotas torna-se necessária como forma de redução de custos.

Motivado por isso, o presente projeto foi desenvolvido na tentativa de se oferecer uma alternativa aos métodos de definição de rotas mais tradicionais. Inicialmente, desenvolveu-se um modelo matemático sob a forma de um MILP. Este modelo possibilitou a otimização do planeamento de rotas realizadas por veículos que partem de um depósito, ao implementarem-se restrições realistas relativas ao carregamento tridimensional. Perante isto, optou-se por desenvolver e implementar uma heurística Relax-and-Fix que reduz os tempos de resolução e que melhora a qualidade da solução. A heurística foi, então, exaustivamente testada em relação ao MILP e a influência de duas restrições importantes também foi avaliada.

Observou-se que a heurística geralmente fornece, em muito menos tempo, soluções de melhor qualidade que as alcançadas para o MILP quando o limite de tempo de 3600 segundos é atingido pelo *solver*. Além disso, concluiu-se que para o MILP e para a heurística, o uso de uma função objetivo focada na distância é mais eficiente na otimização da definição de rotas do que uma função focada nos custos.

Palavras-chave: Problema de Roteamento de Veículos com Restrições de Carregamento Tridimensionais, Frota Heterogénea, Suporte Completo, Fragilidade, Relax-and-Fix, Programação Linear Inteira

Acknowledgements

This dissertation is the result of several months of work. For this, I thank everyone that, in one way or another, helped and accompanied me in this journey.

To my supervisors, Dr. Elsa Marília da Costa Silva and Dr. Khadija Hadj Salem, I thank for all of their support and guidance throughout the project. Their dedication helped me to overcome most of the hurdles presented by this challenge.

To the Faculty of Engineering of the University of Porto and to the Professors that contributed to my path in this university, I thank for the quality education provided, for the care and knowledge provided.

To my friends, I thank for all of their support and companionship, that have allowed me to stay motivated and push through adversities. Without them, the journey would be far more difficult and less pleasant.

To my family, in special to my parents and grandparents, I thank for never doubting my abilities and for providing me with all of the possibilities and tools to evolve in my personal and professional life. Their dedication and love has allowed me fight for my dreams. To my sister, I thank for being a constant presence, even when she was over 2000 kilometers away, for the support she gives me and for her unfiltered honesty.

"Do not follow where the path may lead. Go instead where there is no path and leave a trail."

Ralph Waldo Emerson

Contents

Acronyms and Symbols

MILP	Mixed-Integer Linear Programming
VRP	Vehicle Routing Problem
CLP	Container Loading Problem
3DL-VRP	Vehicle Routing Problem with 3-dimmensional loading constraints
TSP	Travelling Salesman Problem
CVRP	Capacitated Vehicle Routing Problem
BPP	Bin Packing Problem
1D-BPP	1-dimmensional Bin Packing Problem
1D-CLP, 3D-CLP	1- and 3-dimmensional Container Loading Problem
2L-CVRP, 3L-CVRP	2- and 3-dimmensional Capacitated Vehicle Routing Problem
LIFO	Last In First Out
3L-HFVRP	Heterogeneous Fleet Vehicle Routing Problem with 3-dimmensional Loading Problem
OF1, OF2	Objective Function 1 and 2
fg	fragility
fs	full support

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

In today's industrial and commercial landscape, companies find themselves in very competitive markets that require innovation and research to improve profits in very tight markets. Along with this search for improvement, governments and clients have been pushing towards a more ecological, sustainable and green future. All these factors contribute to the implementation of new processes and methods of production and management.

A large part of the overall expenses of most companies is spent on logistics, including warehousing, handling, depreciation, personnel, and, mainly, transport. According to Tseng et al. (2005), between one and two thirds of a company's total logistics costs are related to the transportation of goods. Naturally, some companies began to invest in the development of methods to reduce the high transport costs. This has been achieved by implementing smaller and squarer packaging to improve the amount of product in a single vehicle or container, allowing for better transportation optimization.

The main types of cargo transportation used are air, water, railway and road transport. Each has advantages and disadvantages, but the most broadly used transport for industrial application is road transport Eurostat (2019). This is because road transport is the only type of transport capable of providing point-to-point transport for small and medium distances. Additionally, road transportation is more widely available, faster for short distances, and usually necessary to provide access to the other types of transport because most companies are not situated directly next to water or a rail- or airway connection. For this reason, optimization of transportation is considered a great potential source of savings for many companies. Some of this potential derives from optimizing routes to reduce distances, running costs and other factors. Another part of the potential comes from the optimization of the loading of vehicles or containers to transport the same amount of product with fewer resources, reducing the cost of transportation. Both steps have usually been tackled independently and in a case-by-case philosophy.

Routing optimization was considered some time ago by researchers as a Travelling Salesman Problem (TSP) or the Vehicle Routing Problem (VRP). It generally consists of a group of clients that must be visited by different vehicles belonging to a depot, with the objective of reducing the total distance or transportation costs. However, many different types of VRP have been developed with different constraints and objectives.

Many researchers have also studied the optimization of the loading of containers. It is generally treated as a Cutting and Packing (C&P) problem, more particularly as a Bin Packing Problem (BPP). This problem is comprised of the creation of a loading scheme for a 1D, 2D or 3D container, to fit the maximum number of objects possible into an enclosed space or maximizing volumetric occupation.

More recently, researchers have been tackling both problems as a single, integrated problem called 3-dimensional Loading and Vehicle Routing Problem (3DL-VRP). With this, companies would have the ability to input their clients' orders and receive an optimal loading scheme and route for their vehicles. However, this is a challenging problem to solve, and there is still not much literature on the subject.

An even more specific problem introduced since then was the 3dL-VRP for a fleet of heterogeneous vehicles. Having various vehicles with different weight and volume capacities gives the mathematical models greater applicability in the real world, as most companies do not have a fleet of identical vehicles.

1.2 Objectives and Methodology

The main goal of the work developed in this dissertation was to develop an algorithm that utilizes a Relax and Fix Heuristic to ease the solving of a Vehicle Routing Problem with 3-dimensional loading constraints.

At first, thorough research for relevant literature was done, and, based on certain papers found, an exact mathematical algorithm for the problem was proposed. This model was then implemented using Python to establish baselines for the performance of the heuristic at a future point. For this purpose, the model was run using the same instances as Wei et al. (2014). Additionally, some of the constraints were relaxed to quantify their impacts on the solving process.

Having this, the model was then adapted to the heuristic for the elaboration of an algorithm. This algorithm was then implemented in the same way as the exact model, and its abilities to solve the problem were then evaluated.

1.3 Structure

This dissertation is composed of six chapters.

In the first chapter, a global introduction to the topic is made by presenting the motivation for our work in the current industrial and commercial landscape.

In the second chapter, relevant literature is reviewed. In this context, a global overview of the evolution of Vehicle Routing Problems is given and the additional problems imposed by the implementation of 3D loading constraints are presented.

In chapter 3, the problem to be considered is fully described and fully defined. This is done by establishing essential constraints regarding the routing and loading parts.

In the fourth chapter, two solutions for the problem are proposed. A mathematical model is firstly established based on the literature, and a Relax-and-Fix Heuristic is then developed to improve the solving of the model.

In chapter 5, the results obtained by solving the mathematical model and the heuristic are presented. Some analysis is made relative to the instances' characteristics and their influence on solving the model. Additionally, the results of the two methods are compared to evaluate the effectiveness of the heuristic.

In the last chapter, the main conclusions of the dissertation are presented with some potential ideas for further research and development in this area.

Chapter 2

State of the art

In this chapter, a review of the relevant literature consulted for the elaboration of this dissertation can be found. The review mainly focuses on the two constituting problems of *Vehicle Routing Problem* and *Container Loading Problem*. A short analysis regarding the characteristics of the presented papers is also made. After approaching the two independent problems, a further review is made of the literature regarding the integrated problem of *Vehicle Routing Problem with Loading Constraints*.

2.1 Routing of Vehicles

As mentioned before, the first topic of literature to review was the routing of the vehicles, as it would then define the cargo each vehicle would have to be loaded with.

2.1.1 Vehicle Routing Problem (VRP)

The *Vehicle Routing Problem* or VRP was first introduced by Dantzig and Ramser (1959) as a variant of a very well-known problem, namely the *Travelling Salesman Problem (TSP)* .

The VRP was the first more industrially applicable variant of the TSP and typically consists of a depot with one or multiple vehicles that must distribute the depot's cargo to its intended clients and then return to the depot. The VRP has 3 frequent objectives: to minimize the total distance travelled by the vehicles, to minimize the cost associated with those travels, or to maximize the occupation of the vehicles used.

A more comprehensive review of some mathematical models for simple TSP and VRP problems can be found in Kulkarni and Bhave (1985).

2.1.2 Capacitated VRP (CVRP)

Because the VRP was still too simple for implementation on a more industrial or commercial landscape, a further variant was developed to include the capacities of the vehicles. This inclusion was achieved by introducing constraints limiting the amount of cargo attributed to a vehicle to its total weight carrying capacity.

Alternatively, on some models, the capacity is given in pallets or volume. This allows for an easy implementation in some industries that work with a lot of palletized loads that are not very heavy and therefore would not surpass the vehicles' carrying capacity.

This variant of VRP was introduced by Toth and Vigo (2002) with a fixed fleet, but has since been developed by many other researchers.

2.2 Loading of Vehicles - Container Loading Problems (CLP)

Having reviewed the literature related to the routing of the vehicles, some research was made into how the loading of vehicles or containers is treated in the literature.

The packing or loading of vehicles or containers is generally referenced under 2 main designations, Container Loading Problems (CLP) or Bin Packing Problems(BPP).

The first stage of these problems is similar to the capacity portion of the CVRP as it considers objects to be one-dimensional (weight, pallets, volume, etc..). It consists of the attribution of objects to a certain container or bin that has a capacity that must not be surpassed.

Because the 1D-CLP or 1D-BPP is relatively simple, it was first developed into 2-dimensional problems and then 3-dimensional problems.

These problems tackle the loading of containers considering various constraints of mainly geometrical origin, like dimensions of the objects and containers, but also some constraints of non-overlapping and non-over-packing.

Non-overlapping constraints consist of equations that guarantee that, within the container, no space is attributed to more than one object, meaning that they cannot intersect each other.

Non-over-packing constraints ensure that the objects attributed to a certain container, are completely contained within its boundaries or walls.

Chen et al. (1995) approaches the 3d-CLP in a mathematical model which Silva et al. (2019) then mentions in his comparative study of several CLP models like Chen et al. (1995), Tsai et al. (2015), Junqueira et al. (2012) and others. Another such comparative study and summary of CLP problems can be found in the work of Kurpel et al. (2020).

2.3 Integrated Problem

A very significant step for the evolution of VRP problems was the integration of some loading constraints typical of CLP problems. This would allow for a more realistic scenario for the problem.

2.3.1 CVRP with 2-dimensional loading constraints (2L-CVRP)

The first type of integrated problem developed considers only two dimensions for the constraints related to the loading of the vehicle.

This implementation allowed for a better control of the loading of the vehicles, introducing also the concepts of non-overlapping and non-over-packing into the problem.

Something also introduced by this development was the possibility of implementation for Last In First Out (LIFO) constraints. LIFO is a real-world constraint relevant for vehicles that only have one door for unloading. This constraint means that every time a box is loaded behind another, the box in front must be delivered first.

However, even though this was a big step for VRP problems, it combines two independently very complex problems. It is itself a complex problem. Consequently, the loading of this kind of problem would take considerable time. In order to reduce the time to achieve good or optimal solutions using current computational equipment, heuristics have been introduced. The 2L-CVRP is approached by Leung et al. (2013) utilising a meta-heuristic and by Escobar-Falcón et al. (2021) with the implementation of a Genetic Algorithm (GA).

2.3.2 CVRP with 3-dimensional loading constraints (3L-CVRP)

Like in the previous variant, the 3L-CVRP consists of introducing one further dimension of loading constraints, height, to the 2L-CVRP.

This implementation of height also introduces some other concepts integral to the real-world applicability. These are the stackability of objects, fragility and support surfaces.

Stackability is the concept that objects can be placed on top of each other, creating a multilevel packing plan, which was not possible in the 2L-CVRP.

The fragility of an object dictates if that object can support other objects on its top surface. A fragile object has to be placed on the floor or on top of a stack of objects.

The need for support surfaces is another real-world constraint that was never needed before this integration. It defines that, to place an object in a certain place, it needs to have underneath it, a supporting surface. The size of this supporting surface has to be defined by the model and can be defined as “full support”, where full surface of the top object needs to be supported by an underlying surface, or a "partial support", where only a certain percentage of the top object needs to be supported.

A mathematical model for this was developed by Junqueira et al. (2013) and a variant of the 3L-CVRP is also approached by Yi and Bortfeldt (2018) and Bortfeldt and Yi (2020). These two later ones introduce the concept of split delivery, where a clients’ load can be split into 2 or more vehicles instead of having to be delivered in just one.

2.3.3 3L-CVRP with Heterogeneous Fleet (3L-HFVRP)

For further applicability of 3L-CVRP in real-life scenarios, another variant of VRP was considered, which is the 3L-HFVRP. Because in reality most companies do not have a fleet composed of exactly identical vehicles, this variant of the problem implements the possibility for the depot to have a fleet composed of vehicles that have different characteristics between themselves. This integration allows greater flexibility in routing and loading as it includes the differentiation between the cost of vehicles and the other costs

A mathematical model that approaches the pure 3L-HFVRP could not be found in the literature, but a variant including the constraints of Time Windows was presented by Moura (2019).

Time Windows are another real-life constraint that may be important in some applications of this model, as it introduces the time variable and the fact that some clients may only be available for delivery at certain hours of the day.

Additionally, one work relevant for this dissertation was Wei et al. (2014) where the author used an Adaptive Variable Neighborhood Search algorithm to solve a very similar problem to the one proposed in this dissertation. The instances used for the computational experiments were also extracted from here. In this article, the author also implemented a Last In First Out (LIFO) constraint and considered an unlimited fleet of vehicles.

Table 2.1 presents some of the papers reviewed during this work.

Table 2.1: Summary table of articles, their models and respective constraints

References	Focus	one visit	time windows	backhaul	max driving time	non-overpacking	non-overlapping	lifo	support	vehicle stability	rotations	multi trips	fragility
Moura (2019)	3L-CVRP	x	x			x	x				x		
Kurpel et al. (2020)	CLP								x	x	x		
Silva et al. (2019)	clp,comparative					x	x		x	x	x		
Society (2018)	3l-hfVRPTW				x	x				x		x	
Wei et al. (2014)	3l-hfVRPTW	x						x	x		x		x
Yi and Bortfeldt (2018)	3l-cvrpsd					x	x		x		x		x
Leung et al. (2013)	2l-hfvrp	x				x	x						
Bortfeldt and Yi (2020)	3l-cvrpsd					x		x	x		x	x	x
Pace et al. (2015)	3l-hfvrptw	x	x			x			x	x		x	
Escobar-Falcón et al. (2021)	2l-hfvrp	x				x	x	x	x				
Junqueira et al. (2013)	3l-vrp	x				x	x		x				x
Kulkarni and Bhave (1985)	summary	-	-	-	-	-	-	-	-	-	-	-	-
Iori and Martello (2010)	summary	-	-	-	-	-	-	-	-	-	-	-	-
This work	3l-hfvrp	x	-	-	-	x	x	-	x	-	x	-	x

Chapter 3

Problem Description

This chapter will describe the 3L-HFVRP at hand, along with the constraints and assumptions considered to solve it.

The problem at hand is, at its core, a VRP where the routes of a vehicle fleet located at a depot must be planned to deliver loads to clients.

In this specific case, the fleet of nv vehicles is heterogeneous, meaning that not all vehicles have the same weight capacity or internal carrying dimensions. A vehicle k of the set of vehicles $V = 1,\dots,nv$ has attributes of weight capacity Q_k, and length, width and height of their loading spaces L_k, W_k, H_k. Every vehicle k also has an associated fixed cost fc_k and variable cost vc_k. The first is a cost supported every time the vehicle is used, and the other, a cost per unit of distance travelled by that vehicle.

A set of nodes $N = 0,\dots,nc$ contains the depot (node 0) and all the nc clients to visit. Every client i has a total weight demand wd_i, and an amount of boxes to be delivered to it , d_i.
All the boxes to transport are contained in set $B = 1,\dots,nb$ and their allocation to clients is given by $\psi_{i,b}$, a 0-1-matrix that defines to which client a box belongs. Every box b also has its own length, width and height (l_b, w_b, h_b). These are assumed to be rectangular in shape in order to ease the loading and stability of the boxes. Additionally, boxes have an attribute of fragility f_b which indicates if a certain box is fragile and may not support others placed on top of it.

As is usual in VRP problems, constraints must be added to a problem for imposing certain limitations. In this case, the routing part of the problem is ensured by the following constraints.

Vehicles used must leave from the depot at the beginning of their routes and return to it at the end. Additionally, vehicles cannot visit the depot in the middle of their routes. An example of how a routing diagram for a depot could be drawn up is shown in Figure 3.1.

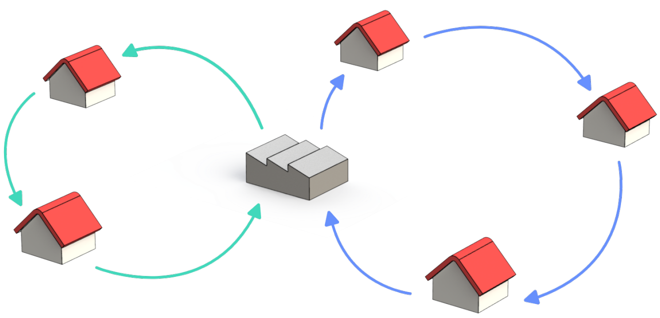

Figure 3.1: Visualization of routing possibilities

Each client must be visited exactly once and by one vehicle and it is assumed that the load of a single client can be loaded into a single vehicle in terms of space and weight. Clients must also receive the totality of their demands and consequently, all boxes must also be delivered to their respective clients.

Subtours are to be discouraged as it is assumed that the shortest distance between two nodes is the direct route from one to the other.

This problem integrates the typical VRP with the vehicle or container loading problem, some constraints related to the second part are also paramount.

Firstly, and most importantly, boxes loaded into a vehicle may not overlap each other. Additionally, all boxes must be completely contained within their respective vehicles.

To ease the loading of the boxes, it is assumed that these can only be placed with their edges parallel to the interior edges of the vehicle. However, rotation of the boxes along all axis are allowed to make better use of available space. Because boxes are rectangular, they can have 6 possible orientations, as can be viewed in Figure 3.2.

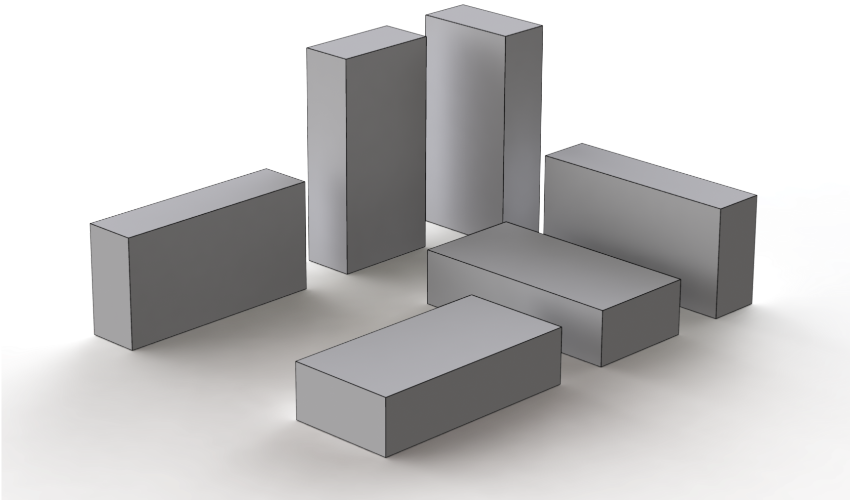

Figure 3.2: Orientations allowed for a box

Stability constraints are also necessary for the loading of a vehicle, and, in this case, full support is required to stack boxes on top of each other. What this means is that a box placed on top of another must have its base placed within the top surface of the underlying box. A feasible loading diagram respecting these types of constraints is presented in Figure 3.3.

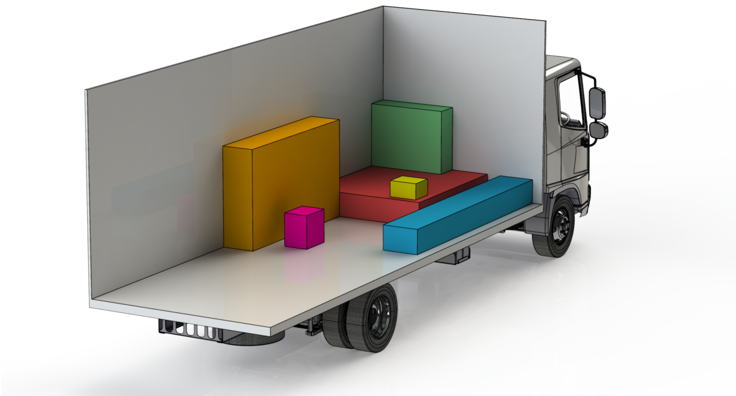

Figure 3.3: Example of vehicle loading plan

Boxes that are marked as fragile must not have other boxes placed on top of them. One possible way to stack a fragile object is the yellow box in figure 3.3.

Chapter 4

Methodology

Before implementing a Relax-and-Fix Heuristic, a mathematical model on which to base it must be developed. The model shown here-under was achieved by joining the two composing problems and their constraints.

4.1 Notation

The parameters used and listed below are the relevant parameters for the problem resolution. These can be acquired directly or must be derived/calculated from the information present in each instance.

$dist_{i,j}$ -	Distance between client i and client j
$\psi_{i,b}$ -	Is equal to 1 if box b goes to client i
d_i, wd_i -	Demand of client i in boxes and weight
Q_k -	Weight capacity of vehicle k
l_b, w_b, h_b -	Length, width and height of box b
L_k, W_k, H_k -	Length, width and height of vehicle k
fc_k, vc_k -	Fixed and variable cost of vehicle k
M -	Big number

4.2 Mixed-Integer Linear Programming (MILP) Model

As mentioned before, the integrated model was composed by joining the two composing problems of *Vehicle Routing* and *3D Loading*. For this, some constraints from models established by other authors were implemented here.

The main part of the routing was derived from the Model of Moura (2019) and one constraint from Kulkarni and Bhave (1985). To establish the loading portion of the model, some equations were taken from the model of Chen et al. (1995) and the constraints regarding full support and fragility were specifically designed.

4.2.1 Decision Variables:

The decision variables required for the MILP model are the following:

$\alpha_{i,j,k}$ -	Binary variable equal to 1 if vehicle k travels from client i to client j
$\gamma_{i,k}$ -	Binary variable equal to 1 if vehicle k visits client i
$\beta_{k,i,b}$ -	Binary variable equal to 1 if box b belongs to client i and is transported by vehicle k
$s_{b,k}$ -	Binary variable equal to 1 if box b is transported by vehicle k
kul_i -	Continuous number for Kulkarni and Bhave (1985) sub-tour constraint
x_b, y_b, z_b -	Indicates the position of the front-left-bottom corner of box b
lx_b, ly_b, lz_b -	Binary variable that indicates if the length of box b is along the X, Y or Z axis
wx_b, wy_b, wz_b -	Binary variable that indicates if the width of box b is along the X, Y or Z axis
hx_b, hy_b, hz_b -	Binary variable that indicates if the height of box b is along the X, Y or Z axis
$A_{b,d}, B_{b,d}, C_{b,d}, D_{b,d}, E_{b,d}, F_{b,d}$ -	Binary variable that indicates relative position between 2 boxes (above, below, to the right, left, in front and behind respectively)

4.2.2 Objective Functions

When developing mathematical models, one important aspect is the objective function. In the case of the routing of vehicles, some different objective functions are used in the literature. However, there are 3 main ones: namely the minimization of costs, of distances and of the amount of vehicles used. In this dissertation, the decision was made to investigate utilizing two of these.

Because the base of vehicle routing was initially the minimization of distances, objective function OF1 (4.1) was considered taking into account only the distance traveled by all the vehicles used.

$$OF1 = min \ \left(\sum_{i=0}^{nc} \sum_{j=0 \neq i}^{nc} \sum_{k=0}^{nv} dist_{i,j} \cdot \alpha_{i,j,k} \right) \tag{4.1}$$

Then, because comparability with the data and results of Wei et al. (2014) was to be achieved, the cost objective function OF2 (4.2) was also implemented.

$$OF2 = min \ \left(\sum_{k=0}^{nv} fc_k \cdot \gamma_{0,k} + \sum_{i=0}^{nc} \sum_{j=0}^{nc} \sum_{k=0}^{nv} dist_{i,j} \cdot \alpha_{i,j,k} \cdot vc_k \right) \tag{4.2}$$

4.2.3 Constraints

After having established the objective functions, the constraints to apply to the model had to be approached. For this, the constraints related to the VRP part (4.3 - 4.8) were based on the ones from Moura (2019).

$$\sum_{k=0}^{nv} \gamma_{i,k} = 1 \quad , \forall i \neq 0 \in CN \tag{4.3}$$

$$\sum_{i=0, i \neq j}^{nc} \alpha_{i,j,k} = \gamma_{j,k} \quad , \forall j \in N, \forall k \in V \tag{4.4}$$

$$\sum_{j=0, j \neq i}^{nc} \alpha_{i,j,k} = \gamma_{i,k} \quad , \forall i \in N, \forall k \in V \tag{4.5}$$

$$\sum_{b=1}^{nb} (\beta_{k,i,b} \cdot \psi_{i,b}) = d_i \cdot \gamma_{i,k} \quad , \forall i \neq 0 \in N, \forall k \in V \tag{4.6}$$

$$\sum_{i=q}^{nc} (wd_i \cdot \gamma_{i,k}) \leq Q_k \quad , \forall k \in V \tag{4.7}$$

$$\sum_{b=1}^{nb} \sum_{i=1}^{nc} \beta_{k,i,b} \cdot (l_b \cdot w_b \cdot h_b) \leq (L_k \cdot W_k \cdot H_k) \ , \forall k \in V \tag{4.8}$$

Constraint (4.3) ensures that every client is visited only once and by a single vehicle, while constraints (4.4) and (4.5) guarantee that that vehicle enters and exits every client it visits.

The following constraint (4.6) assigns every box b belonging to a client i, to the vehicle k that visits that client.

Constraints (4.7) and (4.8) are the capacity constraints that ensure that the total weight and volume of the boxes placed in a certain vehicle do not exceed its weight capacity and internal volume.

Even-though all of these constraints were derived from the model of Moura (2019), some modifications were necessary to make them accomplish the objectives of this project and some of the original constraints were left out due to non-applicability.

When analyzing the model from the original paper, it was noted that every box is indeed delivered to one client by one vehicle. However, we would like to make a clear correspondence between clients and boxes. To achieve this, a 0-1 matrix $\psi_{i,b}$ was created that assigns every box to it's final destination. This matrix was then introduced into equation (4.6).

Another modification made to equation (4.6) was changing the less or equal to an equal ensuring the demand of a client is met exactly.

x

$$kul_j - kul_i + (nc+1) \cdot \alpha_{i,j,k} \leq nc \quad ,\forall k \in V, \forall i,j \neq 0 \in N \tag{4.9}$$

Because Moura (2019) utilizes time windows as a way to eliminate sub-tours, and these are not implemented in this model, an additional equation (4.9) from the model portrayed in Kulkarni and Bhave (1985) was used for this purpose. This constraint attributes a value to every client relative to its position in the route of the vehicle it is served by, the first client having the highest value and the following ones decreasing along the route.

Having written the constraints of the VRP part of the problem, the 3D loading must then be integrated into the model. For this, equations based on the ones from Chen et al. (1995) were adapted to better suit the problem at hand.

$$x_b + l_b \cdot lx_b + w_b \cdot wx_b + h_b \cdot hx_b \leq x_d + (1 - A_{b,d}) \cdot M \ ,\forall b,d \in B, b < d \tag{4.10}$$

$$x_d + l_d \cdot lx_d + w_d \cdot wx_d + h_d \cdot hx_d \leq x_b + (1 - B_{b,d}) \cdot M \ ,\forall b,d \in B, b < d \tag{4.11}$$

$$y_b + l_b \cdot ly_b + w_b \cdot wy_b + h_b \cdot hy_b \leq y_d + (1 - C_{b,d}) \cdot M \ ,\forall b,d \in B, b < d \tag{4.12}$$

$$y_d + l_d \cdot ly_d + w_d \cdot wy_d + h_d \cdot hy_d \leq y_b + (1 - D_{b,d}) \cdot M \ ,\forall b,d \in B, b < d \tag{4.13}$$

$$z_b + l_b \cdot lz_b + w_b \cdot wz_b + h_b \cdot hz_b \leq z_d + (1 - E_{b,d}) \cdot M \ ,\forall b,d \in B, b < d \tag{4.14}$$

$$z_d + l_d \cdot lz_d + w_d \cdot wz_d + h_d \cdot hz_d \leq z_b + (1 - F_{b,d}) \cdot M \ ,\forall b,d \in B, b < d \tag{4.15}$$

$$\begin{aligned} A_{b,d} + B_{b,d} + C_{b,d} + D_{b,d} + E_{b,d} + F_{b,d} \geq s_{b,k} + s_{d,k} - 1 \ , \\ \forall b,d \in B, b < d, \forall k \in V \end{aligned} \tag{4.16}$$

$$\sum_{i=1}^{nc} \beta_{k,i,b} = s_{b,k}, \forall k \in V, \forall b \in B \tag{4.17}$$

$$x_b + l_b \cdot lx_b + w_b \cdot wx_b + h_b \cdot hx_b \leq H_k + (1 - s_{b,k}) \cdot M \ , \forall b \in B \tag{4.18}$$

$$y_b + l_b \cdot ly_b + w_b \cdot wy_b + h_b \cdot hy_b \leq W_k + (1 - s_{b,k}) \cdot M \ , \forall b \in B \tag{4.19}$$

$$z_b + l_b \cdot lz_b + w_b \cdot wz_b + h_b \cdot hz_b \leq L_k + (1 - s_{b,k}) \cdot M \ , \forall b \in B \tag{4.20}$$

$$lx_b + ly_b + lz_b = 1 \ , \forall b \in B \tag{4.21}$$

$$wx_b + wy_b + wz_b = 1 \ , \forall b \in B \tag{4.22}$$

$$hx_b + hy_b + hz_b = 1 \ , \forall b \in B \tag{4.23}$$

$$lx_b + wx_b + hx_b = 1 \ , \forall b \in B \tag{4.24}$$

$$ly_b + wy_b + hy_b = 1 \ , \forall b \in B \tag{4.25}$$

$$lz_b + wz_b + hz_b = 1 \ , \forall b \in B \tag{4.26}$$

Constraints (4.10) - (4.16) are responsible for the non-overlapping of the boxes inside a vehicle, meaning that no box should be placed partially or completely inside another box.

Constraints (4.17) integrates the problem by creating a relation between $\beta_{k,i,b}$ from the VRP part of the problem and $s_{b,k}$ from the 3D loading part.

Constraints (4.18) - (4.20) are responsible for assuring that the entirety of the boxes are placed completely inside the vehicle. The remaining constraints (4.21) - (4.26) ensure that every box is placed in only one direction with the faces parallel to those of the vehicle.

At this point, the model is able to arrange multiple boxes inside of a vehicle. However, because some additional requirements were necessary, new constraints had to be developed.

The first additional requirement was the implementation of Full Support. This guarantees that boxes placed above others, have their bases completely supported by the underlying surface. Constraints (4.27) - (4.34) ensure that this requirement is fulfilled by the model.

$$y_b \leq y_d + M \cdot (B_{b,d} + C_{b,d} + D_{b,d} + E_{b,d} + F_{b,d}) + M \cdot (1 - A_{b,d}) \ \ , \forall b,d \in B, b < d \tag{4.27}$$

$$y_d \leq y_b + M \cdot (A_{b,d} + C_{b,d} + D_{b,d} + E_{b,d} + F_{b,d}) + M \cdot (1 - B_{b,d}) \ \ , \forall b,d \in B, b < d \tag{4.28}$$

$$z_b \leq z_d + M \cdot (B_{b,d} + C_{b,d} + D_{b,d} + E_{b,d} + F_{b,d}) + M \cdot (1 - A_{b,d}) \ \ , \forall b,d \in B, b < d \tag{4.29}$$

$$z_d \leq z_b + M \cdot (A_{b,d} + C_{b,d} + D_{b,d} + E_{b,d} + F_{b,d}) + M \cdot (1 - B_{b,d}) \ \ , \forall b,d \in B, b < d \tag{4.30}$$

$$\begin{gathered} y_b + l_b \cdot ly_b + w_b \cdot wy_b + h_b \cdot hy_b \leq y_d + l_d \cdot ly_d + w_d \cdot wy_d + h_d \cdot hy_d + \\ M \cdot (A_{b,d} + C_{b,d} + D_{b,d} + E_{b,d} + F_{b,d}) + M \cdot (1 - B_{b,d}) \ \ , \forall b,d \in BN, b < d \end{gathered} \tag{4.31}$$

$$\begin{gathered} y_d + l_d \cdot ly_d + w_d \cdot wy_d + h_d \cdot hy_d \leq y_b + l_b \cdot ly_b + w_b \cdot wy_b + h_b \cdot hy_b + \\ M \cdot (B_{b,d} + C_{b,d} + D_{b,d} + E_{b,d} + F_{b,d}) + M \cdot (1 - A_{b,d}) \ \ , \forall b,d \in B, b < d \end{gathered} \tag{4.32}$$

$$\begin{gathered} z_b + l_b \cdot lz_b + w_b \cdot wz_b + h_b \cdot hz_b \leq z_d + l_d \cdot lz_d + w_d \cdot wz_d + h_d \cdot hz_d + \\ M \cdot (A_{b,d} + C_{b,d} + D_{b,d} + E_{b,d} + F_{b,d}) + M \cdot (1 - B_{b,d}) \ \ , \forall b,d \in B, b < d \end{gathered} \tag{4.33}$$

$$\begin{gathered} z_d + l_d \cdot lz_d + w_d \cdot wz_d + h_d \cdot hz_d \leq z_b + l_b \cdot lz_b + w_b \cdot wz_b + h_b \cdot hz_b + \\ M \cdot (B_{b,d} + C_{b,d} + D_{b,d} + E_{b,d} + F_{b,d}) + M \cdot (1 - A_{b,d}) \ \ , \forall b,d \in B, b < d \end{gathered} \tag{4.34}$$

A second requirement for this model was the implementation of fragility, meaning that if a box is marked as fragile, it cannot support any boxes on top of it. For this, constraints (4.35) and (4.36) were developed and ensure that a box marked as fragile is neither below a box nor has a box on top of it.

$$\sum_{b=1\ b<d}^{nb} A_{b,d} \leq M \cdot (1 - f_d)\ ,\forall d \in B \tag{4.35}$$

$$\sum_{d=1\ d>b}^{nb} B_{b,d} \leq f_b\ ,\forall b \in B \tag{4.36}$$

Finally, the last set of constraints (4.37) - (4.44) determines the type and the domain of the decision variables.

$$\alpha_{i,j,k} \in \{0,1\}, \qquad \forall i \in N, j \in N, k \in V \tag{4.37}$$

$$\beta_{k,i,b}, \in \{0,1\}, \qquad \forall k \in V, i \in N, b \in B, \tag{4.38}$$

$$\gamma_{i,k}, \in \{0,1\}, \qquad \forall i \in N, k \in V, \tag{4.39}$$

$$s_{b,k} \in \{0,1\}, \qquad \forall b \in B, k \in V, \tag{4.40}$$

$$kul_i \in \mathbb{N}, \qquad \forall i \in N, \tag{4.41}$$

$$x_b,\ y_b,\ z_b \in \mathbb{R}^+, \qquad \forall b \in B \tag{4.42}$$

$$lx_b,\ ly_b,\ lz_b,\ wx_b,\ wy_b,\ wz_b,\ hx_b,\ hy_b,\ hz_b \in \{0,1\}, \qquad \forall b \in B, \tag{4.43}$$

$$A_{b,d},\ B_{b,d},\ C_{b,d},\ D_{b,d},\ E_{b,d},\ F_{b,d} \in \{0,1\}, \qquad \forall b \in B, d \in B, b < d. \tag{4.44}$$

4.3 Relax-and-Fix Heuristic

In an effort to speed up the solving of the model, an implementation of a Relax-and-Fix heuristic was considered.

A Relax-and-Fix heuristic was first introduced by Belvaux and Wolsey (2000) for a lot-sizing problem. Since then, the heuristic has been used mostly in scenarios regarding lot sizing problems like in Stadtler (2003) and Federgruen et al. (2007) and also in other problems, such as the multicast routing in Ribeiro et al. (2019) or the vehicle-reservation assignment problem in the car rental in Oliveira et al. (2014), for example.

The Relax-and-Fix heuristic consists of an iterative process where some variables of a model are relaxed and the model is solved. The values of the non-relaxed variables are then fixed to those obtained in the previous resolution, while the other variables are set to their original domains and the model is solved again. If necessary, for the second step, instead of returning all relaxed variables to their domains, it is possible to add more steps of fixing and relaxing variables until all variables are fixed. From this resolution, in general, a feasible solution for the model is obtained.

Below, in figure 4.1, a flowchart of the Relax-and-Fix heuristic developed in this work is pictured. A detailed description of the algorithm is presented in the following subsections.

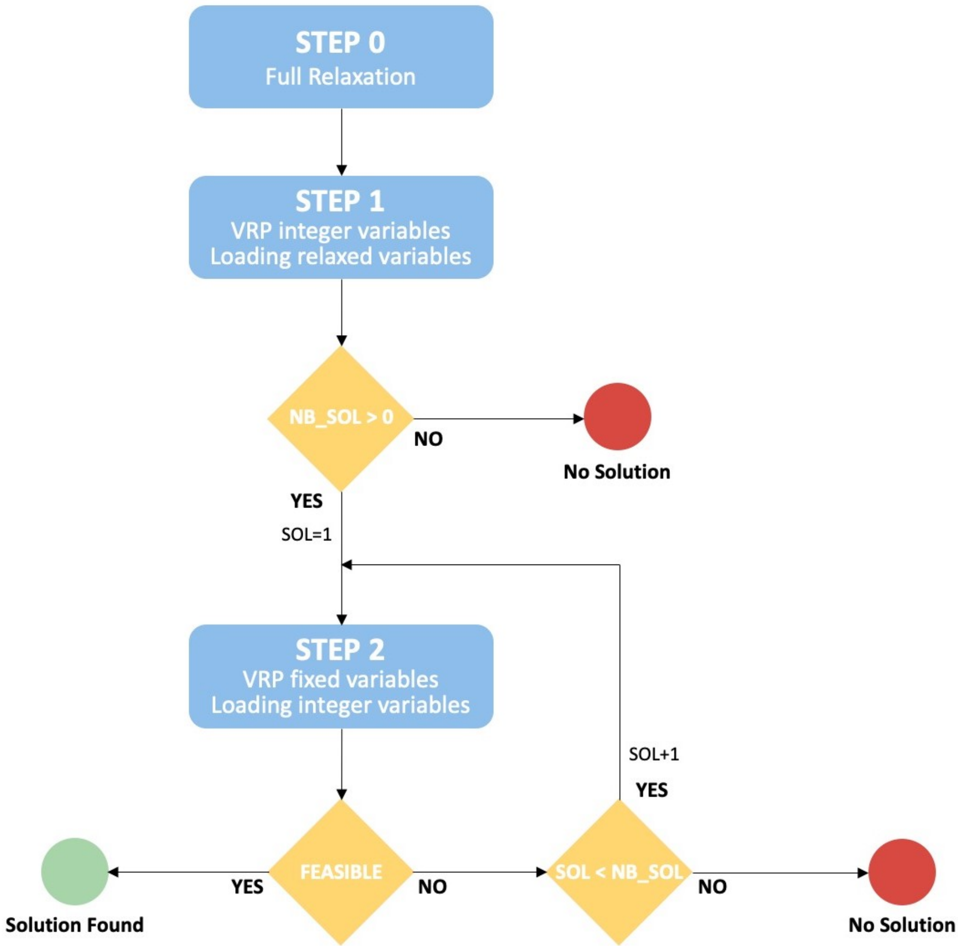

Figure 4.1: Flowchart of the process when the Relax-and-Fix heuristic is used

4.3.1 Full Relaxation - step 0

As the first step in this heuristic, a full relaxation of the model is made. In this relaxation, all decision variables are of type continuous. However, the respective intervals are maintained. For example, the binary variables become continuous in the interval [0,1]. And, after solving the model, the objective value obtained is stored for later comparison with the results of the following steps.

The domains of the decision variables to apply to this step are (4.45) - (4.52) that substitute (4.37) - (4.44).

$$\alpha_{i,j,k} \in [0,1] \,, \qquad \forall i \in N, j \in N, k \in V \quad (4.45)$$

$$\beta_{k,i,b}, \in [0,1] \,, \qquad \forall k \in V, i \in N, b \in B, \quad (4.46)$$

$$\gamma_{i,k}, \in [0,1] \,, \qquad \forall i \in N, k \in V, \quad (4.47)$$

$$s_{b,k} \in [0,1] \,, \qquad \forall b \in B, k \in V, \quad (4.48)$$

$$kul_i \in \mathbb{R}^+, \qquad \forall i \in N \quad (4.49)$$

$$x_b,\ y_b,\ z_b \in \mathbb{R}^+, \qquad \forall b \in B \quad (4.50)$$

$$lx_b,\ ly_b,\ lz_b,\ wx_b,\ wy_b,\ wz_b,\ hx_b,\ hy_b,\ hz_b \in [0,1] \,, \qquad \forall b \in B \quad (4.51)$$

$$A_{b,d},\ B_{b,d},\ C_{b,d},\ D_{b,d},\ E_{b,d},\ F_{b,d} \in [0,1] \qquad \forall b \in B, d \in B, b < d \quad (4.52)$$

4.3.2 VRP integer variables, Loading relaxed variables - Step 1

After the full relaxation of the model, a step of partial relaxation is required for the heuristic. In this case, it was decided to only relax the variables related to the 3D loading of the vehicles while the ones related to the routing are set back to their original type (binary, integer).

This results in a routing plan that serves all clients, but has no loading plan for the vehicles attributed to it. The feasibility of loading is to be tested in the next step.

Because there may not be a feasible loading plan for every solution obtained in this step, it was chosen to store a large number of feasible solutions from this step ordered in decreasing value of the objective function.

For this step, the type and domains of the decision variables are (4.37) - (4.41) and (4.50) - (4.52).

4.3.3 Fixed VRP variables , Loading integer variables- Step 2

After having the multiple solutions stored during the previous step (amount of solutions given by *nb_sol*), the decision variables associated with the routing will consecutively be fixed to the values obtained in the VRP solution (*sol*) being tested. Additionally, all the other decision variables related to the loading are also set to their original types.

A pseudo-algorithm for the process in this last step is pictured below in Algorithm 1.

The objective of this step is to find a feasible loading plan for the best solutions found in 4.3.2 and will not direct impact the value of the distance or the cost regarding the complete model.

After solving the model for the first solution of the previous step, if there is no feasible loading plan, the algorithm goes to the next solution on the list until it finds a solution. Because the solutions are ordered in increasing distance (or cost depending on the objective function), the first solution with a feasible loading plan, is the best value found and no further solutions must be tested.

Because this is an approximated method based on a MILP model, there is a possibility that a solution with feasible loading plan is not found within reasonable time.

Algorithm 1: Pseudo-Algorithm of step 2

```
end = 0
while (end = 0) and (sol ≤ nb_sol) do
    run model( step1_solutions [sol] );
    if infeasible or no solution found then
        if sol < nb_sol then
            not a feasible solution;
            sol ← sol + 1
        else
            no feasible solution from step 1;
            end ← 1
            sol ← sol + 1
    else
        feasible solution found;
        end ← 1
```

Chapter 5

Results

This chapter presents the numerical results obtained for the MILP model and the relax-and-fix heuristic. For the experiments, different variants of the tackled problem were considered and implemented to then allow for comparisons to be established.

5.1 Instances and Implementation

The first step towards the computational experiments of the MILP model is to implement it in an environment capable of solving it. For this, it was decided to use the Python 3.8.8 programming language and the Gurobi solver version 9.1.2.

For the MILP model, the time limit considered was 3600 seconds and for the Relax-and-Fix heuristic time limits of 600, 1800 and 600 seconds were set for steps 0, 1 and 2, respectively. These time limits were chosen in order to try to obtain results in less time than the MILP. For step 0 , the time limit was set in a way that finds solutions for every instance file. The limit for step 1 was set to half of the MILP to try to halve the solving time. For step 2 the time limit was selected in a way that would allow for the algorithm to test at least some solutions before surpassing the 3600 seconds mark in total. To run the experiments, a machine with an Intel i7-9750H CPU running at 2.6 to 4.5 GHz was used, which also sported 16GB of RAM.

5.1.1 Instances

After having implemented the model, problem instances were necessary. For this, the instances from Wei et al. (2014) were used. These instances are derived from the Solomon (1987) instances and contain information relative to a VRP. This information is separated into 3 main topics: vehicles, clients and orders of clients.

For the vehicles the instances present a list with the number of a vehicle, its weight capacity and internal dimensions given in height, width and length. Additionally, the instances also give a fix cost and a variable cost for the vehicles.

Regarding the clients, the instances always have the depot as client 0 and then all clients as the following numbers. For all the clients an x and y coordinate is given and also a weight demand.

Regarding the clients' orders, the boxes composing the orders of every client are listed, including the height, width, and length of every box and an indication of its fragility.

The instances can be separated into three main classes according to their number of clients. As can be seen in figure 5.1, the smallest instances (1-12) have 20 clients, the medium instances (13-24) have 50 clients, and the largest instances (25-36) have 75 or 100 clients.

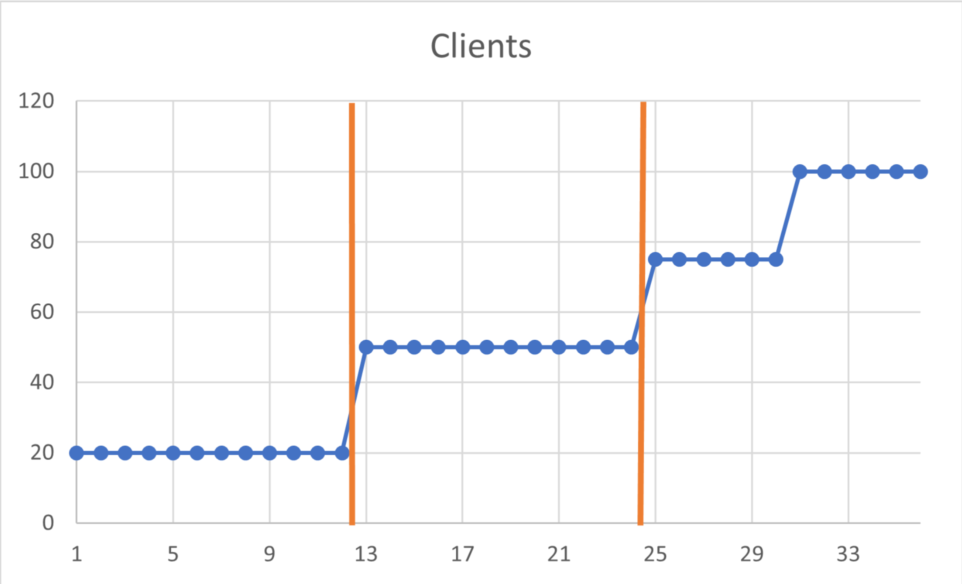

Figure 5.1: Number of clients in every instance

The original instances from Wei et al. (2014), however, were meant for a problem that considered multiple vehicles of the same type without limits for any type. In the model developed in this work, it is assumed that the full fleet of available vehicles is given and because of this some modifications to the instances were necessary. Since every one of the original instances has certain types of vehicles, the edit was made to take into account multiples of every type of vehicle. This edit allows for the solutions to choose multiples of the same vehicle, which was also necessary for all the boxes to get packed in the vehicles. The value by which the vehicles of each instance are multiplied is defined by the amount of overflow (calculated with the equations 5.1 and 5.2) of weight and volume capacity. The minimum amount of overflow permitted was 50% to ensure that the loading of the vehicles is not completely filled. In figure 5.2, the difference between the number of vehicles of the original and the modified instances can be seen.

$$weight\_overflow = \left(\frac{\sum_{k=0}^{VN} Q_k}{\sum_{i=1}^{CN} wd_i} - 1 \right) \cdot 100\% \tag{5.1}$$

$$volume\_overflow = \left(\frac{\sum_{k=0}^{VN} L_k \cdot W_k \cdot H_k}{\sum_{b=1}^{CN} l_b \cdot w_b \cdot h_b} - 1 \right) \cdot 100\% \tag{5.2}$$

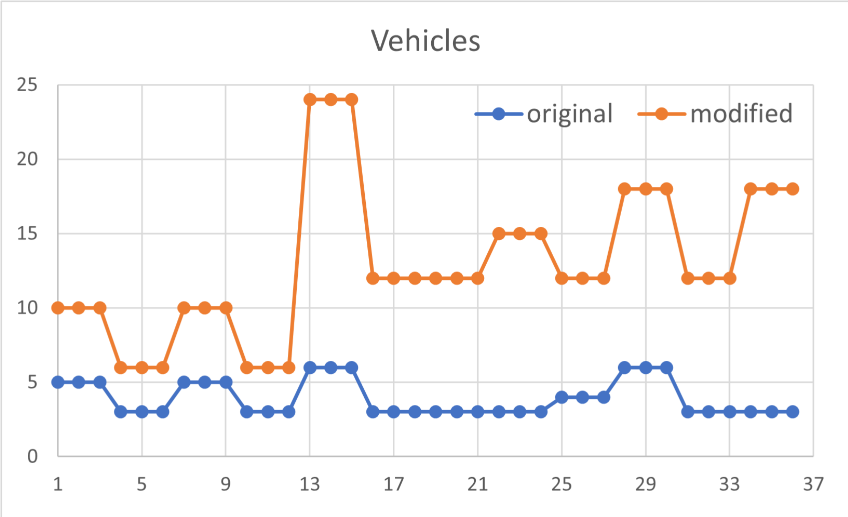

Figure 5.2: Number of vehicles in the original and modified instances

From Wei et al. (2014), it is also known that on average, there are around two boxes per client in every instance. The data relative to the original and modified instances can also be seen in table A.1.

5.1.2 Implementation

To be able to establish comparisons between methods and the influence of different constraints, some concepts had to be defined.

The first concept is the one of *gap*. The gap considered hereafter is given by equation (5.3) where Z_p is the best objective value found in the solution being analyzed (in either the MILP or the Relax-and-Fix) and $Z_{d\ MILP}$ being the objective lower bound achieved by the MILP.

$$gap = \frac{|Z_p - Z_{d\ MILP}|}{Z_p} \cdot 100\% \tag{5.3}$$

It is important to refer that in the numerical results presented hereafter, when the gap of the model is portrayed as 100 %, the model did not find any feasible solution within the time limit.

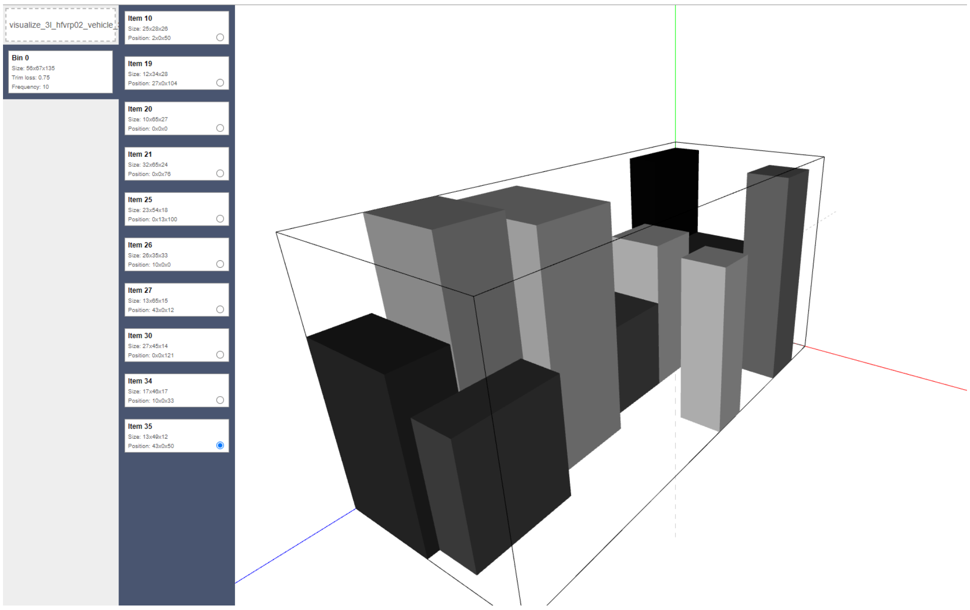

Figure 5.3: Example of the visualizer used to verify the loading plans

Lastly, a function was implemented into the program that writes *.json* files with the loading plan of every vehicle utilized in the solutions obtained. This was made to allow for visualization of the loading plan utilizing the online tool of Óscar Oliveira and Silva (2019). This tool has a visualizer which outputs a 3d environment where the user can see the boxes inside a certain space (vehicle) and check for issues like interference. An example of an output created for one of the files generated by the MILP is shown in figure 5.3.

5.2 Comparison between MILP and Relax-and-Fix

In this section, the numerical results from various experiments performed are presented. At first, the full model was tackled using the MILP model and the Relax-and-Fix heuristic described before. Then, the model was stripped of some of its constraints in order to establish their influence on the solver's performance.

5.2.1 Full Model

5.2.1.1 Performance of the objective functions

Firstly, the model was run using the complete mathematical model for both objective functions. The results from both of these runs displayed in figure 5.4, show that the model is more difficult to solve to optimality when using the cost objective function. On average, the gaps obtained for the small instances (1-12) with the cost objective function, is 42,4% higher than the ones obtained when minimizing distances.

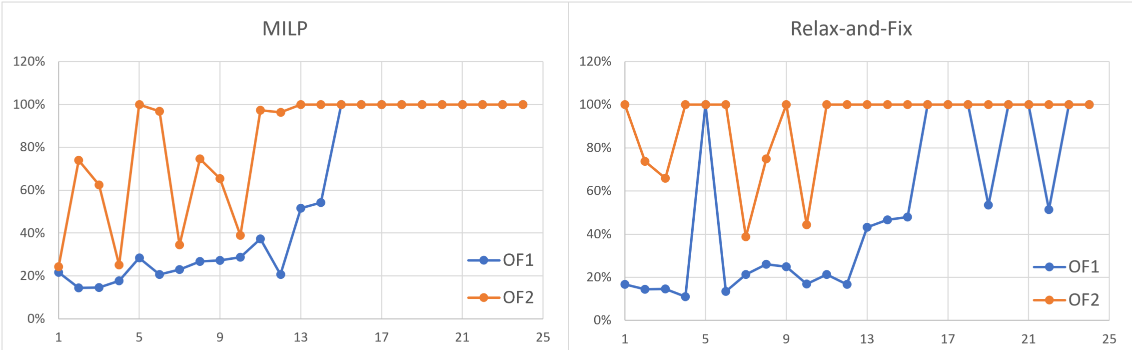

Figure 5.4: Gaps obtained for both objective functions in the MILP and Relax-and-Fix

When running the complete problem variant using the Relax-and-Fix, the same behavior is observable with the gaps for the first 12 instances being again considerably worse for the cost objective function (average difference of 58,4%) as can be seen in figure 5.4 .

These conclusions can be viewed in Tables A.2 and A.3 for OF1 and OF2 respectively.

5.2.1.2 Performance of the Relax-and-Fix Heuristic

In order to evaluate the performance of the heuristic the results of the previously mentioned experiments were then organized per objective function to allow for better analysis.

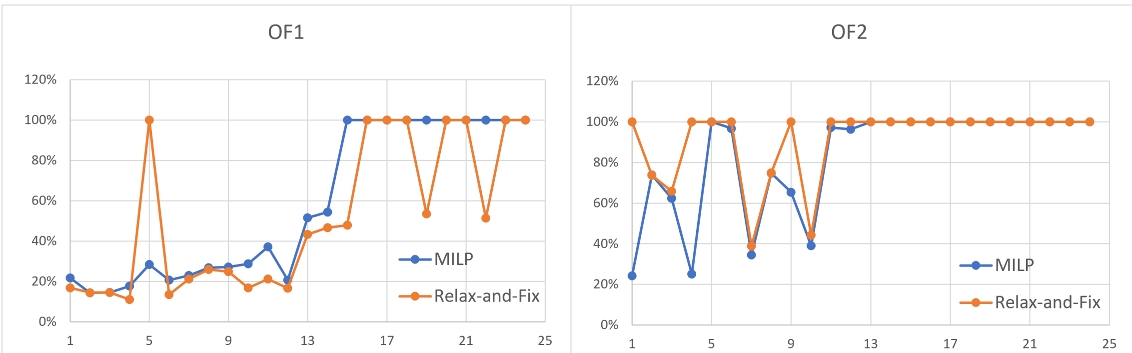

Figure 5.5: Gaps obtained for objective function 1 and 2, for the MILP and Relax-and-Fix

In figure 5.5, the gaps obtained for the distance objective function (OF1) are presented. It is observable that the Relax-and-Fix heuristic has better results for all instances except for instance 5 and the instances where no solution was found using any of the methods. The average difference between the gaps when discarding those instances is 13,7% and 6,1% when including them. The amount of instances where no solution was found during the running time was also reduced from 10 to 8. This means that the use of the Relax-and-Fix heuristic helped to solve the problem more easily.

Additionally, the average solving times for small and big instances are reduced from the 3600 seconds implemented as the time limit for the MILP model to 2872 seconds for the Relax-and-Fix.

When considering only the smallest instances, the average time is even reduced to 1989 seconds in the Relax-and-Fix.

For the cost objective function (OF2), the results point in the opposite direction. For this objective function, the heuristic is less well equipped to solve the problem, having consistently worse gaps than the mathematical model. The difference between the gaps of both methods is now of 17,3% for the first 12 instances, but, in this case, tending for the mathematical model. The number of instances without a feasible solution found within running time was also increased from 13 to 19 when using the heuristic. This means that the Relax-and-Fix heuristic has a deteriorating effect on solving the model with the cost objective function.

Having investigated the two objective functions, OF1 will be used for all experiments hereafter as it had considerably better results in both the MILP and Relax-and-Fix problems.

5.2.2 Without fragility (no fg)

To assert the influence that the fragility constraints have on the ease of solving the model, the instances were now run with a model without constraints (4.35) and (4.36).

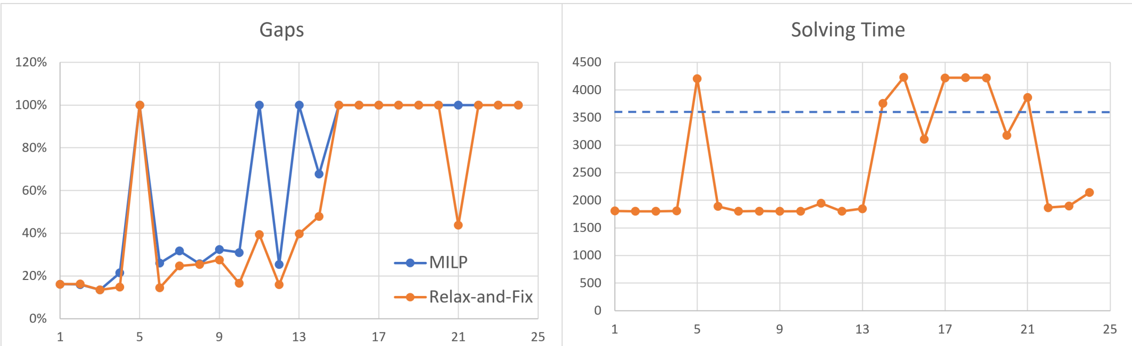

Figure 5.6: Gaps obtained when running the model without fragility and solution times of the Relax-and-Fix

When observing figure 5.6, it is possible to see that, when eliminating the fragility constraints, the results obtained for the MILP and Relax-and-Fix heuristic are similar for the first 12 instances with the average difference between gaps being 4,5% (in favor of the heuristic). For the following instances, the average difference is increased to 11,4%. We can also see that the solving times in this variant are also very close to the 1900 seconds for the small instances. For the instances above number 13 it centers around the 4000 seconds mark even though it only finds a feasible solution for the instance 21.

In this problem variant, the number of instances for which no feasible solution was found within the time limit was 13 for the MILP and 10 for the Relax-and-Fix. This improvement provided by the heuristic also comes with an improvement in solving time for almost all of the instances where a solution was found. The average time to the solution was 2100 seconds, and all of the solutions found have better gaps than the MILP.

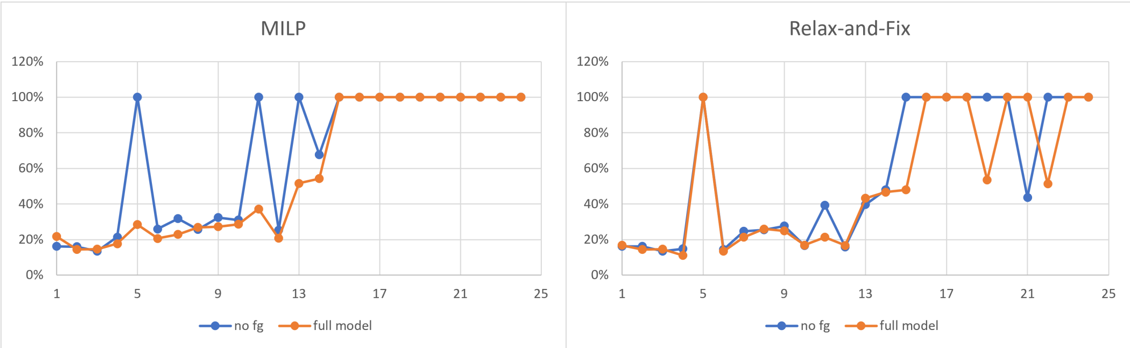

Figure 5.7: Comparative between full model and model without fragility on the MILP and Relax-and-Fix

As can be seen in figure 5.7, the results for the MILP of this problem variant are generally worse than the ones obtained for the full model. However, if the instances without feasible solutions found are left out, the results are indeed very similar to the ones obtained with the full model.

Regarding the application of the Relax-and-Fix, the problem variant being analyzed obtains results that are more similar to the full model than in the MILP but the number of instances without a feasible solution increases by 2.

The data regarding the computational experiments of the problem variant without fragility are presented in table A.4.

5.2.3 Without full support and without fragility (no fs + no fg)

Having checked the influence of the fragility constraints, the full support constraints (constraints (4.27 - 4.36)) were then also taken away from the model to assert their influence in the solving of the model.

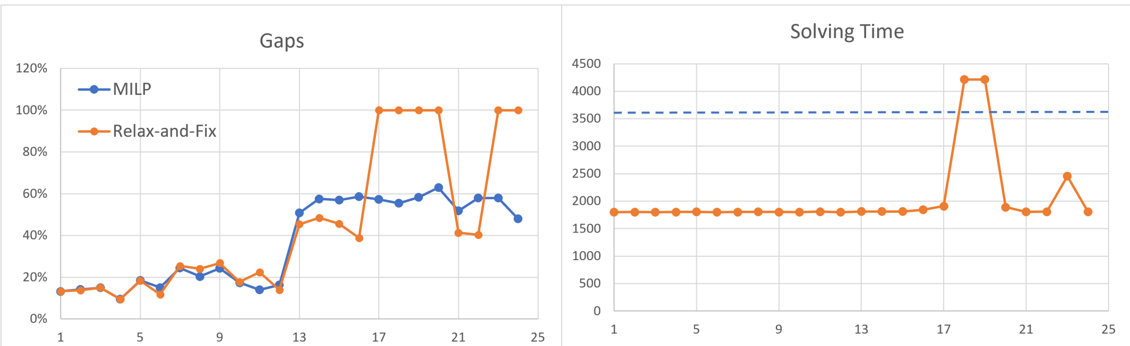

Figure 5.8: Gaps and solving times obtained for the model without full support and fragility

As can be seen in figure 5.8, the difference between the MILP and Relax-and-Fix heuristic when considering the problem variant without the full support constraints are quite a bit tighter with only an average difference of 0,8% between gaps for the smaller instances and 8,2% when considering the smallest and medium instances, both tending for the MILP model. Here, however, it is important to remember that the solutions found by the Relax-and-Fix heuristic have an average solving time of around 1900 seconds against the 3600 used by the MILP model.

The average time to solve this problem variant was 1807, seconds with solutions mostly similar for the small instances and slightly better for the medium instances.

However,this figure also shows that the Relax-and-Fix heuristic performs 10 to 20 percent better than the MILP model in the medium instances where it finds feasible solutions. However, there is a presence of 6 instances where the Relax-and-Fix heuristic was not able to find a feasible solution, whereas the MILP model always finds at least one.

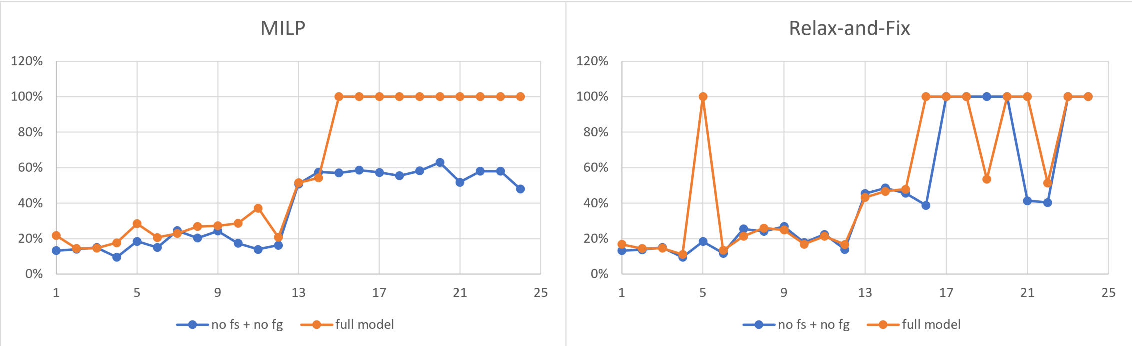

Figure 5.9: Comparative between full model and model without full support and fragility for MILP and Relax-and-Fix

When comparing the results for the problem variant without full support and without fragility with the ones obtained in the full model (figure 5.9), it is possible to identify that the application of the Relax-and-Fix heuristic is very advantageous in the mathematical model. With it, the amount of instances without a feasible solution found within the time limit is reduced from 10 to 0, and the average difference between the gaps of the full model and the reduced one is of 6,6% for the small instances and considerably higher (21%) for the medium ones.

Considering the implementation of this reduced model in the Relax-and-Fix heuristic, the results are less noticeable. The gaps are overall nearer to each other except for some instances where, for the reduced model, it is able to find feasible solutions, and for the full model, it was not. Overall, the gaps for the 24 instances are 7% lower for the reduced model.

The data regarding the computational experiments of the problem variant without full support and without fragility are presented in table A.5.

5.3 MILP problem analysis by problem variant

As can been observed in figure 5.10 and was to be expected, the model without the full support and without fragility constraints always has lower gaps than both other variants of the problem. However, it was also expected that when only the fragility constraints are stripped from the model, the gaps would be in between the full model and the more reduced model. This does not happen and, indeed, the problem variant without fragility constraints has predominantly worse gaps than the other two variants.

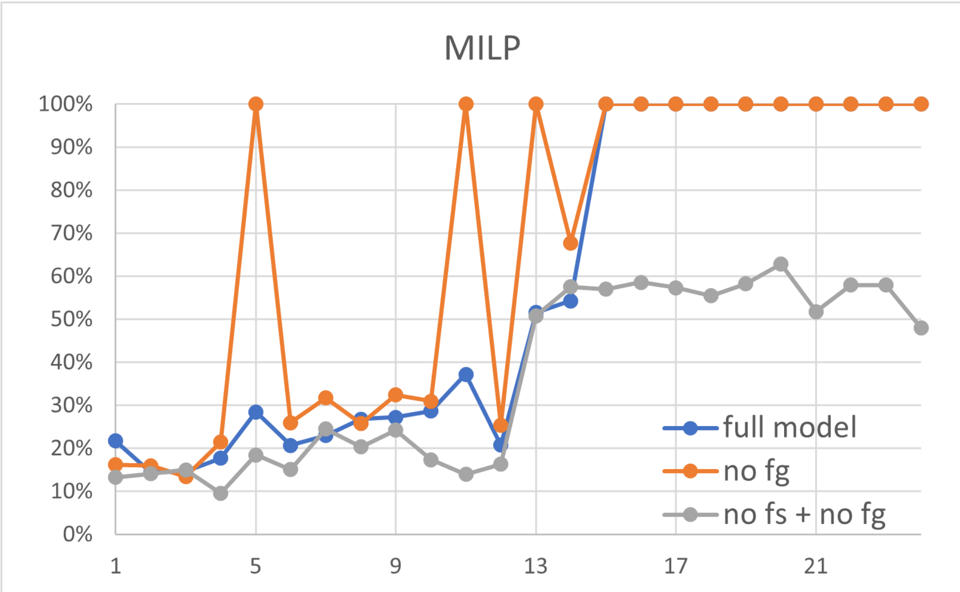

Figure 5.10: Comparative of the gaps obtained with all variants of the MILP

For the models that consider full support, it is also notable that beyond instance 14 no feasible solution is found in both models but, in the model without those constraints, all of the small and medium instances (1-24) have feasible solutions found within the time limit. Additionally, the problem variant without the fragility constraints also does not find feasible solutions for 3 more instances than the full model which was not foreseeable.

5.4 Relax-and-Fix problem analysis by problem variant

To evaluate the performance of the Relax-and-Fix heuristic at solving the 3 variants of the problem, on figure 5.11 can be seen a representation of the gaps and solving times obtained for them.

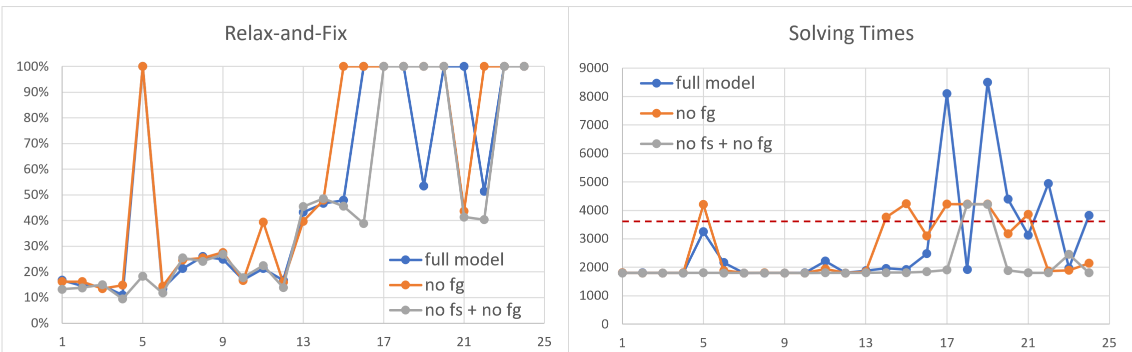

Figure 5.11: Comparative of gaps and solving times of all variants of the Relax-and-Fix problem

As can be seen in the aforementioned figure, the model without full support and without fragility consistently provides better gaps than the other 2 problem variants. The only instance where this problem variant does not do this is instance 19, where the full model is the only one to find a feasible solution.

As in the MILP model, this was to be expected because of the fewer constraints of this problem variant. However, also like in the MILP model, the problem variant without fragility, has again worse results than the full model. It is not able to find feasible solutions for 10 instances, whereas the full model is only not able to find solutions for 8.

In figure 5.11, a graph can be seen where the solving times of all 3 problem variants of the model are presented. As can be seen in that graph, for the small instances, the heuristic solving times for all problem variants are very similar to each other and grouped around the 1800 seconds mark. The only exception to this is instance 5 where both the full model and the model without fragility have higher solving times, the second even taking more time than the 3600 seconds imposed to the MILP model.

In the medium instances however, the solving times are less grouped around a certain value. For the problem variant without full support, the solving times are always under the 3600 seconds mark except for instances 18 and 19 where it, in fact, isn't able to find a feasible solution.

For the problem variant without fragility, there are 6 of the 12 medium instances where the solving time is above the 3600 seconds, and for the full model this value goes down to 5. However, from instance 15 upwards, it is important to note that the heuristic was not able to find any feasible solutions in these 2 problem variants.

Chapter 6

Conclusions and Future Work

Given the competitive nature of the current industrial and commercial landscape, companies are in constant search for new ways to reduce costs in order to improve profit margins. Because logistical costs are such a big part of the total operational costs of a company, it is one of the sectors where a big potential for cost reduction is foreseen. In this prospect, the work of this dissertation was developed to attempt to achieve a cost-saving and effective solution for a company. For this, a MILP model was developed at first and a Relax-and-Fix heuristic was then created to help obtain good solutions with shorter solving times. These were then implemented into an executable Python code and various computational experiments were performed with both the MILP and Relax-and-Fix models.

After hundreds of hours of experiments running the instances from Wei et al. (2014), the results obtained were then analyzed and it was possible to extract some conclusions.

The first conclusion to extract From the numerical results was that the difference in the quality of solutions obtained between the distance and cost objective function is quite different. The gap obtained for the same instances was consistently lower when utilizing the distance objective function. This is due to the lack of linearity of the cost objective function, which considerably influences the ease the solver has to achieve good solutions. The impact observed in the set of small instances with the MILP model was as big as 58,4% of difference in gaps with the distance objective function, greatly outperforming the minimization of costs.

Following the first experiments, it was also possible to conclude that the Relax-and-Fix heuristic developed is not very well equipped to solve the problem of cost minimization as it always returned worse results than its equivalent MILP.

After this, the performance of the Relax-and-Fix heuristic developed was evaluated with the help of more numerical results. For the full model, encompassing all constraints, the heuristic proved to achieve solutions of higher quality and more often than the equivalent MILP. This was also reinforced by the fact that it was able to find feasible solutions for more instances than the mathematical model, even on the medium sized instances.

After evaluating the performance of the heuristic in the full model, two further variants of the problem were tested. The first was a model that did not consider the concept of fragility. For this problem variant the conclusion was that, unexpectedly, this reduced model performed slightly worse than the full model. The gap difference observed was 9,2% in the MILP and 4,8% in the heuristic in favor of the full model.

Ensuing this evaluation, a further problem variant of the problem was introduced and tested, which additionally discarded the full support constraints. The result was as expected on these experiments, with the reduced variant considerably outperforming the full model in both the MILP and Relax-and-Fix. It must be mentioned that, for this problem variant, the MILP was able to find feasible solutions for all of the medium sized instances, which was not achieved in any other variant in either MILP of Relax-and-Fix form.

Overall, it was concluded that the Relax-and-fix heuristic provides the user with results that are, in the 3 problem variants, better than the MILP model in the case of the small instances (20 clients). This is also supported by the fact that the solving times of the heuristic in those instances are on average about half of the time of the equivalent MILP. For the medium instances, the Relax-and-Fix was also able to find feasible solutions for more instances than the MILP in all problem variants, except when leaving out full support. Additionally, in these instances, all solutions found were also with a lesser gap.

Having finished the work of the dissertation, some opportunities for future work were identified. One such opportunity would be to compare the results obtained with the results presented by Wei et al. (2014). For this, a further development of the model could also be done by adding a LIFO constraint.

Another possibly interesting development would be to analyze how minimizing distances affect the cost of the solutions found. Additionally, the further important analysis would be to establish connections between characteristics of instances and the results obtained for the MILP and Relax-and-Fix to see how they are influenced. This would allow for fine tuning of the instances and possibly the model, to obtain better results.

One constraint from Moura (2019) was also not utilized in this model and could possibly be introduced to improve the application in an industrial scenario. The constraint of Time Windows is important as it introduces a new dimension (time) which provides the model a greater versatility. This can also afterwards be used to implement other constraints relative to this dimension, like a maximum of work hours for a driver of a vehicle.

Load balance, or weight distribution, in a vehicle is another constraint that may be important for some applications where the objects to arrange in a vehicle are heavy and may affect the vehicles abilities or stability. However, for this constraint to be implemented, the instances would have to contain information about the weight of every individual box. The instances from Wei et al. (2014) only give the information on a per client basis and therefore cannot directly be used.

Because the push towards a green industry has been quite important, integrating a Pollution-sensitive VRP would also be interesting as companies are increasingly investing in technologies reduce their carbon footprint. The CO2 emissions of a vehicle are highly load-dependent and due to this, a variable would have to be introduced that represents the load a vehicle carries when it traverses any arc similar to the work proposed by Xiao et al. (2012). A cost could also be attributed to the emissions, and this could be added to OF2.

In terms of Relax-and-Fix, other strategies for implementing the heuristic could also be explored like clustering of clients or boxes, generating different steps for the heuristic that might be favorable for solving this problem.

Lastly, to further validate the efficiency of the MILP and Relax-and-Fix, other data sets from the literature could be used for computational experiments to compare against other methods present in the literature.

Bibliography

Belvaux, G. and Wolsey, L. A. (2000). bc—prod: A specialized branch-and-cut system for lot-sizing problems. *Management Science*, 46(5):724–738.

Bortfeldt, A. and Yi, J. (2020). The Split Delivery Vehicle Routing Problem with three-dimensional loading constraints. *European Journal of Operational Research*, 282(2):545–558.

Chen, C. S., Lee, S. M., and Shen, Q. S. (1995). An analytical model for the container loading problem. *European Journal of Operational Research*, 80(1):68–76.

Dantzig, G. B. and Ramser, J. H. (1959). The Truck Dispatching Problem. Technical Report 1.

Escobar-Falcón, L. M., Álvarez-Martínez, D., Wilmer-Escobar, J., and Granada-Echeverri, M. (2021). A specialized genetic algorithm for the fuel consumption heterogeneous fleet vehicle routing problem with bidimensional packing constraints. *International Journal of Industrial Engineering Computations*, 12:191–204.

Eurostat (2019). Freight transport statistics - modal split. (April 2019):1–11.

Federgruen, A., Meissner, J., and Tzur, M. (2007). Progressive interval heuristics for multi-item capacitated lot-sizing problems. *Operations Research*, 55(3):490–502.

Iori, M. and Martello, S. (2010). Routing problems with loading constraints. *Top*, 18(1):4–27.

Junqueira, L., Morabito, R., and Yamashita, D. S. (2012). Three-dimensional container loading models with cargo stability and load bearing constraints. *Computers & Operations Research*, 39(1):74–85.

Junqueira, L., Oliveira, J. F., Carravilla, M. A., and Morabito, R. (2013). An optimization model for the vehicle routing problem with practical three-dimensional loading constraints. *International Transactions in Operational Research*, 20(5):645–666.

Kulkarni, R. and Bhave, P. (1985). Integer programming formulations of vehicle routing problems. *European Journal of Operational Research*, 20(1):58–67.

Kurpel, D. V., Scarpin, C. T., Pécora Junior, J. E., Schenekemberg, C. M., and Coelho, L. C. (2020). The exact solutions of several types of container loading problems. *European Journal of Operational Research*, 284(1):87–107.

Leung, S. C., Zhang, Z., Zhang, D., Hua, X., and Lim, M. K. (2013). A meta-heuristic algorithm for heterogeneous fleet vehicle routing problems with two-dimensional loading constraints. *European Journal of Operational Research*, 225(2):199–210.

Moura, A. (2019). A model-based heuristic to the vehicle routing and loading problem. *International Transactions in Operational Research*, 26(3):888–907.

Oliveira, B. B., Carravilla, M. A., Oliveira, J. F., and Toledo, F. M. (2014). A relax-and-fix-based algorithm for the vehicle-reservation assignment problem in a car rental company. *European Journal of Operational Research*, 237(2):729–737.

Pace, S., Turky, A., Moser, I., and Aleti, A. (2015). Distributing fibre boards: A practical application of the heterogeneous fleet vehicle routing problem with time windows and three-dimensional loading constraints. *Procedia Computer Science*, 51(1):2257–2266.

Ribeiro, C. C., Santos, T. d. A., and de Souza, C. C. (2019). Multicast routing under quality of service constraints for vehicular ad hoc networks: mathematical formulation and a relax-and-fix heuristic. *International Transactions in Operational Research*, 26(4):1339–1364.

Silva, E. F., Toffolo, T. A. M., and Wauters, T. (2019). Exact methods for three-dimensional cutting and packing: A comparative study concerning single container problems. *Computers and Operations Research*, 109:12–27.

Society, A. (2018). Data and Decision Sciences in Action. pages 91–101.

Solomon, M. (1987). Algorithms for the vehicle routing and scheduling problems with time window constraints. *Oper. Res.*, 35:254–265.

Stadtler, H. (2003). Multilevel lot sizing with setup times and multiple constrained resources: Internally rolling schedules with lot-sizing windows. *Operations Research*, 51(3):487–502.

Toth, P. and Vigo, D. (2002). *The Vehicle Routing Problem*.

Tsai, J.-F., Wang, P.-C., and Lin, M.-H. (2015). A global optimization approach for solving three-dimensional open dimension rectangular packing problems. *Optimization*, 64(12):2601–2618.

Tseng, Y.-y., Yue, W. L., and Taylor, M. A. P. (2005). The role of transportation in logistics chain. Eastern Asia Society for Transportation Studies.

Wei, L., Zhang, Z., and Lim, A. (2014). An adaptive variable neighborhood search for a heterogeneous fleet vehicle routing problem with three-dimensional loading constraints. *IEEE Computational Intelligence Magazine*, 9(4):18–30.

Xiao, Y., Zhao, Q., Kaku, I., and Xu, Y. (2012). Development of a fuel consumption optimization model for the capacitated vehicle routing problem. *Computers and Operations Research*, 39(7):1419–1431.

Yi, J. and Bortfeldt, A. (2018). The Capacitated Vehicle Routing Problem with Three-Dimensional Loading Constraints and Split Delivery—A Case Study. pages 351–356.

Óscar Oliveira, D. G. and Silva, E. (2019). Resources for two-dimensional (and three-dimensional) cutting and packing solution methods research. *Proceedings of the 16th International Conference on Applied Computing*, 53(3):131–138.

Appendix A

Appendix 1 - Tables of Characteristics and Experiments

Table A.1: Characteristics and details of all instances

Instance	Clients	Boxes	original instances			modified instances		
			vehicles	weight overflow	volume overflow	vehicles	weight overflow	volume overflow
3l_hfvrp01	20	44	5	-21%	1%	10	58%	103%
3l_hfvrp02	20	41	5	-21%	-24%	10	58%	53%
3l_hfvrp03	20	37	5	-21%	-24%	10	58%	52%
3l_hfvrp04	20	38	3	-18%	-11%	6	64%	78%
3l_hfvrp05	20	48	3	-18%	6%	6	64%	112%
3l_hfvrp06	20	40	3	-18%	-3%	6	64%	94%
3l_hfvrp07	20	39	5	-21%	-13%	10	58%	73%
3l_hfvrp08	20	40	5	-21%	-22%	10	58%	56%
3l_hfvrp09	20	45	5	-21%	-13%	10	58%	74%
3l_hfvrp10	20	46	3	-18%	-20%	6	64%	59%
3l_hfvrp11	20	43	3	-18%	-2%	6	64%	96%
3l_hfvrp12	20	45	3	-18%	-21%	6	64%	58%
3l_hfvrp13	50	86	6	-51%	0%	24	97%	300%
3l_hfvrp14	50	104	6	-51%	21%	24	97%	383%
3l_hfvrp15	50	98	6	-51%	28%	24	97%	411%
3l_hfvrp16	50	103	3	-40%	20%	12	138%	379%
3l_hfvrp17	50	97	3	-40%	10%	12	138%	341%
3l_hfvrp18	50	98	3	-40%	19%	12	138%	376%
3l_hfvrp19	50	94	3	-60%	100%	12	60%	700%
3l_hfvrp20	50	98	3	-60%	114%	12	60%	756%
3l_hfvrp21	50	99	3	-60%	95%	12	60%	679%
3l_hfvrp22	50	103	3	-67%	138%	15	67%	1091%
3l_hfvrp23	50	109	3	-67%	161%	15	67%	1205%
3l_hfvrp24	50	103	3	-67%	133%	15	67%	1064%
3l_hfvrp25	75	144	4	-47%	3%	12	58%	210%
3l_hfvrp26	75	163	4	-47%	28%	12	58%	284%
3l_hfvrp27.	75	154	4	-47%	27%	12	58%	281%
3l_hfvrp28	75	142	6	-29%	-11%	18	113%	168%
3l_hfvrp29	75	148	6	-29%	-13%	18	113%	161%
3l_hfvrp30	75	141	6	-29%	-19%	18	113%	142%
3l_hfvrp31	100	212	3	-59%	94%	12	65%	676%
3l_hfvrp32	100	191	3	-59%	98%	12	65%	691%
3l_hfvrp33	100	200	3	-59%	145%	12	65%	882%
3l_hfvrp34	100	197	3	-73%	221%	18	65%	1825%
3l_hfvrp35	100	211	3	-73%	237%	18	65%	1924%
3l_hfvrp36	100	209	3	-73%	218%	18	65%	1809%

Values marked in red mean that the overflow of volume or weight would be negative. The 3 other colors represent the division of the instances in 3 classes according to the number of clients.

Table A.2: Comparative table of results for the complete method with OF1 in both MILP and Relax-and-Fix

inst	MILP		Relax-and-Fix				
	gap	**time**	**gap**	**t(step0)**	**t(step1)**	**t(step2)**	**t(total)**
3l_hfvrp01	22%	3600	17%	1	1801	6	1808
3l_hfvrp02	14%	3600	14%	1	1801	3	1805
3l_hfvrp03	15%	3600	15%	1	1801	2	1804
3l_hfvrp04	18%	3600	11%	1	1801	3	1805
3l_hfvrp05	28%	3600	100%	1	1801	1448	3250
3l_hfvrp06	21%	3600	13%	1	1801	365	2166
3l_hfvrp07	23%	3600	21%	1	1801	2	1803
3l_hfvrp08	27%	3600	26%	1	1801	2	1804
3l_hfvrp09	27%	3600	25%	2	1801	2	1805
3l_hfvrp10	29%	3600	17%	1	1801	6	1808
3l_hfvrp11	37%	3600	21%	1	1801	417	2219
3l_hfvrp12	21%	3600	17%	1	1801	2	1804
3l_hfvrp13	52%	3600	43%	8	1806	79	1893
3l_hfvrp14	54%	3600	47%	27	1807	154	1989
3l_hfvrp15	100%	3602	48%	9	1806	112	1928
3l_hfvrp16	100%	3600	100%	6	1804	677	2488
3l_hfvrp17	100%	3600	100%	50	1804	6298	8152
3l_hfvrp18	100%	3600	100%	6	1804	121	1930
3l_hfvrp19	100%	3603	53%	5	1804	6697	8506
3l_hfvrp20	100%	3600	100%	6	1807	2594	4407
3l_hfvrp21	100%	3600	100%	6	1804	1328	3138
3l_hfvrp22	100%	3600	51%	7	1806	3143	4956
3l_hfvrp23	100%	3600	100%	20	1805	171	1997
3l_hfvrp24	100%	3600	100%	8	1805	2020	3833

(times given in seconds)

Table A.3: Comparative table of results for the complete method with OF2 in both MILP and Relax-and-Fix

inst	MILP		Relax-and-Fix				
	gap	**time**	**gap**	**t(step0)**	**t(step1)**	**t(step2)**	**t(total)**
3l_hfvrp01	24%	3600	100%	4	1801	17	1822
3l_hfvrp02	74%	3600	74%	1	1801	4	1806
3l_hfvrp03	62%	3600	66%	1	1801	15	1817
3l_hfvrp04	25%	3600	100%	1	1801	7	1808
3l_hfvrp05	100%	3600	100%	2	1801	2408	4211
3l_hfvrp06	97%	3600	100%	2	1801	12	1815
3l_hfvrp07	35%	3600	39%	1	1801	18	1821
3l_hfvrp08	75%	3600	75%	1	1801	6	1808
3l_hfvrp09	65%	3600	100%	2	1801	18	1820
3l_hfvrp10	39%	3600	44%	1	1801	16	1818
3l_hfvrp11	97%	3600	100%	1	1801	185	1987
3l_hfvrp12	96%	3600	100%	3	1801	27	1831
3l_hfvrp13	100%	3600	100%	8	1806	60	1874
3l_hfvrp14	100%	3600	100%	27	1807	709	2543
3l_hfvrp15	100%	3600	100%	10	1807	1282	3099
3l_hfvrp16	100%	3600	100%	7	1805	104	1915
3l_hfvrp17	100%	3601	100%	6	1804	747	2557
3l_hfvrp18	100%	3600	100%	7	1804	11	1822
3l_hfvrp19	100%	3600	100%	6	1807	40	1853
3l_hfvrp20	100%	3600	100%	6	1804	94	1905
3l_hfvrp21	100%	3600	100%	6	1805	0	1811
3l_hfvrp22	100%	3600	100%	3	1810	0	1813
3l_hfvrp23	100%	3600	100%	5	1805	0	1810
3l_hfvrp24	100%	3600	100%	4	1806	0	1810

(times given in seconds)

Table A.4: Comparative table of results for the model without fragility in both MILP and Relax-and-Fix

inst	MILP		Relax-and-Fix				
	gap	**time**	**gap**	**t(step0)**	**t(step1)**	**t(step2)**	**t(total)**
3l_hfvrp01	16%	3600	16%	2	1801	7	1809
3l_hfvrp02	16%	3600	16%	1	1801	2	1804
3l_hfvrp03	13%	3600	13%	1	1801	3	1804
3l_hfvrp04	21%	3600	15%	1	1801	8	1809
3l_hfvrp05	100%	3601	100%	4	1801	2406	4211
3l_hfvrp06	26%	3601	14%	1	1801	88	1890
3l_hfvrp07	32%	3600	25%	1	1801	2	1803
3l_hfvrp08	26%	3600	25%	1	1801	3	1805
3l_hfvrp09	32%	3600	28%	2	1801	2	1804
3l_hfvrp10	31%	3600	17%	3	1801	1	1805
3l_hfvrp11	100%	3601	39%	1	1801	146	1948
3l_hfvrp12	25%	3600	16%	1	1801	2	1804
3l_hfvrp13	100%	3601	40%	8	1806	42	1856
3l_hfvrp14	68%	3603	48%	29	1807	1953	3789
3l_hfvrp15	100%	3602	100%	10	1807	2423	4240
3l_hfvrp16	100%	3600	100%	6	1805	1305	3117
3l_hfvrp17	100%	3600	100%	6	1805	2416	4226
3l_hfvrp18	100%	3600	100%	6	1805	2416	4227
3l_hfvrp19	100%	3601	100%	6	1804	2415	4226
3l_hfvrp20	100%	3600	100%	6	1805	1374	3185
3l_hfvrp21	100%	3600	44%	6	1804	2062	3873
3l_hfvrp22	100%	3600	100%	9	1806	61	1876
3l_hfvrp23	100%	3600	100%	8	1806	91	1905
3l_hfvrp24	100%	3600	100%	8	1806	337	2151

(times given in seconds)

Table A.5: Comparative table of results for the model without full support and without fragility in both MILP and Relax-and-Fix

inst	MILP		Relax-and-Fix				
	gap	**time**	**gap**	**t(step0)**	**t(step1)**	**t(step2)**	**t(total)**
3l_hfvrp01	13%	3600	13%	1	1801	1	1803
3l_hfvrp02	14%	3600	14%	1	1801	1	1803
3l_hfvrp03	15%	3600	15%	1	1800	1	1803
3l_hfvrp04	10%	3600	10%	1	1800	2	1803
3l_hfvrp05	18%	3600	18%	1	1801	5	1806
3l_hfvrp06	15%	3600	12%	1	1800	1	1802
3l_hfvrp07	24%	3600	25%	1	1801	2	1804
3l_hfvrp08	20%	3600	24%	1	1801	3	1804
3l_hfvrp09	24%	3600	27%	1	1801	1	1803
3l_hfvrp10	17%	3600	18%	1	1801	1	1803
3l_hfvrp11	14%	3600	22%	1	1800	8	1809
3l_hfvrp12	16%	3600	14%	1	1801	1	1803
3l_hfvrp13	51%	3600	45%	7	1805	6	1818
3l_hfvrp14	58%	3600	49%	8	1806	7	1821
3l_hfvrp15	57%	3600	46%	9	1806	6	1820
3l_hfvrp16	59%	3600	39%	5	1804	41	1849
3l_hfvrp17	57%	3600	100%	4	1803	106	1914
3l_hfvrp18	55%	3601	100%	5	1803	2411	4218
3l_hfvrp19	58%	3600	100%	5	1803	2411	4219
3l_hfvrp20	63%	3600	100%	4	1803	87	1895
3l_hfvrp21	52%	3600	41%	5	1804	4	1812
3l_hfvrp22	58%	3600	40%	7	1804	5	1816
3l_hfvrp23	58%	3600	100%	7	1806	651	2464
3l_hfvrp24	48%	3600	100%	5	1805	0	1810

(times given in seconds)