FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Exploring the Potential of DRT for Elderly Urban Mobility using Big Data

Marta Diogo Torgal Pinto

Mestrado Integrado em Engenharia Informática e Computação

Supervisor: Maria Teresa Galvão Dias

Co-Supervisor: Tânia Daniela Lopes da Rocha Fontes

February 24, 2020

Exploring the Potential of DRT for Elderly Urban Mobility using Big Data

Marta Diogo Torgal Pinto

Mestrado Integrado em Engenharia Informática e Computação

February 24, 2020

Abstract

According to European Commission statistics, there will be an increase in the percentage of the elderly in the next six decades, and it is expected that 66% of the population will be living in the cities by 2050. Elderly may have sensory, cognitive and motor impairments that can difficult the access to the Public Transport System. Thus, the main motivation of this dissertation is to explore the potential of a Demand Responsive Transport (DRT) system for the elderly, to complement the current Fixed Transport System. Therefore, the goals are to understand and characterise DRT services in urban areas as well as to study the usage of the PTS in space and in time of the population. In order to study the implementation of a DRT service, it is necessary to solve the Dial-A-Ride problem (DARP). Thus one objective is to implement a DARP algorithm. Other objectives were to perform sensitivity tests and a study case of a DRT system for the elderly in the Metropolitan Porto Area.

The static DARP was implemented considering the real-world problems that are complex, and have more than one variable to minimised. It has some constraints, such as the capacity of the vehicles, and a request must be picked up and dropped off only once, and within time window. Literature review states that is generated an initial solution and further optimisation of it. So, the DARP algorithm, a variant of the Vehicle Routing Problem, was divided into two algorithms, an Assigning Requests to Vehicles and a Multi-Objective Tabu Search Algorithm. The goal of the DARP algorithm is to minimise the total travelled kilometres, the deadheading kilometres and the number of vehicles. A set of computational test were performed to assess the performance and sensitivity of the DARP algorithm by using a combination of different parameters' values.

The computational results analyse the correlation between the variables, and it was found strong positive correlations between the number of requests and the number of vehicles (r_s =0.883, ρ<1%) and with the number of vehicles optimised (r_s =0.887, ρ<1%), and moderate negative correlations between the vehicle's capacity and with the deadheading km (r_s=-0.502, ρ<1%) and with the deadheading km optimised (r_s=-0.519, ρ<1%). The data was collected using an Entry-Only Automatic Fare Collection system in the PTS of the Metropolitan Porto Area. The PTS analysed is the bus provider *Sociedade de Transportes Coletivos do Porto*. Furthermore, it was implemented a case study focused only on the elderly population, using a sample of the validations. It was applied a case study with six different combinations of two parameters, the vehicle capacity (4,8,15) and the number of iterations (10,30). From the studied hypotheses, the best solution to be implemented in the PTS could be a vehicle with a capacity of eight. The final solutions generated were with only two vehicles, an average of 43 total travelled kilometres and 0.2 deadheading kilometres, rather than the other solutions found. So, this algorithm shows efficiency in solving this problem and could be applied to solve real-world DRT problems. This work can serve as basis for future work in simulating a DRT system complemented with the Fixed Transport System by using a software simulation and to study the impact that a DRT system could have in the Public Transport System.

Resumo

Os resultados estatísticos da Comissão Europeia demonstram que haverá um aumento da percentagem da população idosa nas próximas seis décadas e é esperado que 66% da população irá viver nas cidades em 2050. O envelhecimento pode causar limitações sensoriais, cognitivas e motoras, que podem dificultar o acesso ao sistema de Transporte Público. Desta forma, a principal motivação desta dissertação é explorar o potencial de um sistema de transporte a pedido para os idosos. Os objetivos desta dissertação são compreender e caracterizar os serviços Sistema de Transporte Público em áreas urbanas, bem como estudar a sua utilização no tempo e no espaço. A implementação de um algoritmo para resolver o "*Dial-A-Ride Problem*" (DARP), que é um sistema de transporte a pedido. Outros objetivos desta dissertação são realizar um estudo computacional, para avaliar a sensibilidade e a eficiência do algoritmo, e um caso de estudo para estudar a implementação de um sistema a pedido na área metropolitana do Porto para os idosos.

O DARP estático implementado teve em consideração que os problemas do mundo real são complexos e minimizam mais do que uma variável. As restrições implementadas no DARP são a capacidade do veículo, o cliente ser apanhado e largado apenas uma vez nas localizações indicadas por ele e não poder ser ultrapassada a janela de tempo predefinida. Desta forma, o algoritmo DARP foi dividido em dois algoritmos, o *Assigning Requests to Vehicles* e o algoritmo multiobjectivo *Tabu Search*. O objetivo do algoritmo DARP é minimizar o número total de quilómetros percorridos, o número total de quilómetros percorridos em vazio, e o número total de veículos da solução. De modo a avaliar o desempenho e a sensibilidade do algoritmo foram realizados testes computacionais, através da combinação de valores de diferentes parâmetros. Foi realizado também um caso de estudo, que explora a implementação de um sistema a pedido, através utilização de dados de validações reais dos idosos, de modo a avaliar um sistema de transporte público para os idosos.

Os resultados computacionais analisam a correlação entre variáveis e foram demonstradas correlações positivas fortes entre o número de pedidos e o número de veículos (r_s =0.883, ρ<1%) e o número de veículos otimizados (r_s =0.887, ρ<1%). E correlações negativas moderadas entre a capacidade dos veículos e o número de quilómetros em vazio (r_s=-0.502, ρ<1%) e os quilómetros em vazio otimizados (r_s=-0.519, ρ<1%). Para além disso, foi implementado um caso de estudo focado na população idosa que utilizou uma amostra das validações, de 2013, na Sociedade de Transportes Coletivos do Porto, a empresa que fornece o serviço de autocarro. O caso de estudo implementado utilizou uma combinação de diferentes parâmetros, a capacidade do veículo (4,8,15) e o número total de iterações (10,30). Através do estudo é possível concluir que a melhor capacidade do veículo é 8, pois gerou as melhores soluções com apenas dois veículos, percorrendo em média 43 quilómetros no total e 0.2 quilómetros em vazio. Este algoritmo demonstra eficiência ao resolver estes problemas e desta forma pode ser aplicado para resolver problemas no mundo real. Este trabalho poderá servir de base para no futuro ser realizada uma simulação de um sistema DRT complementado com o sistema de transporte fixo, para estudar o impacto que o sistema DRT pode ter no sistema de transporte público.

Acknowledgements

First of all, I would like to express my special thanks and gratitude to my supervisor Teresa Galvão and my co-supervisor Tânia Fontes, for their knowledge sharing, availability, contributions, ideas and advice, which were fundamental for the development of this work.

Then, I would like to thank all of my family and my boyfriend, for their words of encouragement, kindness, and for supporting me in this stage of my life. To my friends, I am also thankful for giving me the strength to continue working and for providing moments of leisure.

This work was financed by the European Regional Development Fund through the Operational Programme for Competitiveness and Internationalisation COMPETE 2020 Programme and national funds through Portuguese Science and Technology Foundation within opti-MOVES project (Ref: PTDC/ECI-TRA/32053/2017 - POCI-01-0145-FEDER-032053).

Marta Diogo Torgal Pinto

Contents

List of Figures

LIST OF FIGURES

List of Tables

LIST OF TABLES

Nomenclature and Abbreviations

A*	A star
ARV	Assign Requests to Vehicles
CVRPWPDTW	Capacitated Vehicle Routing Problem With Pick up and Deliveries with Time Window constraints
DARP	Dial-A-Ride Problem
DRT	Demand Responsive Transport
EU-28	European Union of 28 number states
FTS	Fixed Transportation System
GA	Genetic Algorithm
GRASP	Greedy Randomised Adaptive Search Procedure
MOP	Multi-objective Problem
MPA	Metropolitan Porto Area
OD	Origin-Destination
PTS	Public Transport System
TS	Tabu Search
VRP	Vehicle Routing Problem
MOTSA	Multi-Objective Tabu Search Algorithm

Chapter 1

Introduction

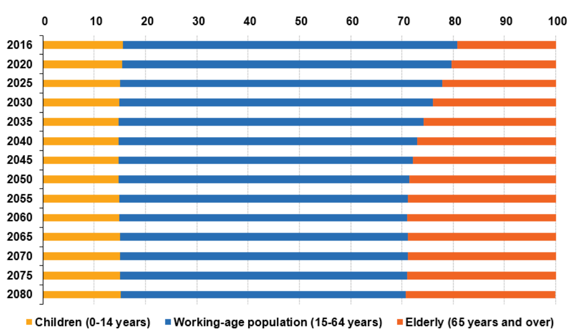
This chapter introduces the context, the objectives, the methodologies and the structure of this dissertation.

1.1 Context

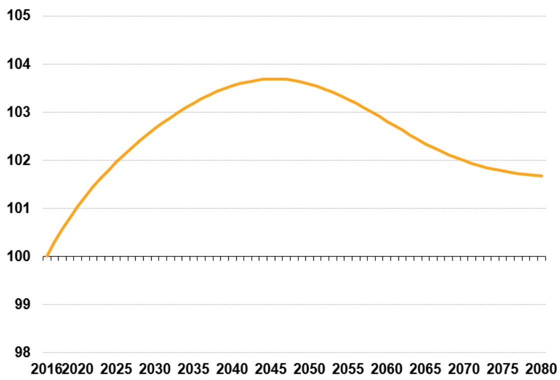
Statistical projections show that 66% of the population all around the globe will be living in cities by 2050 [Lee18]. In the EU-28's (European Union of 28 member states) the population will grow by 1.7% between 2016 and 2080, corresponding to an increase of 8.5 million people (519 million people by 2080) [Eur19b] (Figure 1.1b). With the peak around 2045, reaching 529 million of people, 3.7% more than the number of people observed on 1st of January of 2016.

It is expected a 9.9% increase in the percentage of the elderly population in the next six decades in EU-28 (53.3 million elderly people by 2080) (Figure 1.1a). [Eur19b]. The elderly may have cognitive, sensory and physical impairments [cited in RSMRR18]. Elderly do not usually drive cars relying on Public Transport System (PTS) to maintain an independent, active lifestyle [WSY+18], which promotes social integration and contributes to successful ageing [cited in YCC16]. Nevertheless, the mobility conditions make it difficult to access the PTS [WSY+18].

Nowadays, there is a growing social and environmental awareness that instigates measures to promote social inclusion and environmental sustainability. For example, by decreasing the number of private vehicles or single passenger vehicles in the roads, the public transport and soft modes of transport such as ride-sharing, walking and cycling [cited in BFFG11] are being promoted. Another way to promote PTS is by making public transport fares accessible to the entire population. So to develop or improve an inclusive PTS, it is necessary to have in consideration some aspects such as the accessibility, safety and comfort in transportation modes, and a walking environment adapted for people with mobility impairments.



(a) Population, by broad age group (% of total population)



(b) Projected population

Figure 1.1: Statistics from Eurostat, EU-28, 2016-2080.

1.2 Motivation

Traffic congestion and emission of pollutants are the results expected from the increase of the urban population, which justify the needs of a better PTS, able to improve user's quality of life, that positively impacts the cities economy and promotes environmental sustainability [MDSF98]. The Fixed Transportation Service (FTS) is one service of the PTS, that provides economic and ecological transport, by supporting collective transportation. However, some people can not benefit from it due to their mobility impairments, or even if the current PTS does not meet their needs. Therefore, the improvement of the PTS is currently focused on new ways to move within the city, such as the Demand Responsive Transport (DRT) system.

The main motivation is the PTS' improvement by using a DRT service focused on the elderly. This service is a flexible door-to-door, avoiding walking between home and the stop and this service can be adapted to wheelchairs with a less expensive fare than the taxi service.

Census performed in the year of 2011 [dE19b] allowed to obtain and analyse information about how people move in the Metropolitan Porto Area (MPA). Other information was obtained from statistics published by the *Instituto Nacional de Estatística*, in 2002 [dE19a] and 2013 [dE19c], and it was analysed how people move on MPA. From figure 1.2 can be verified that the majority of the population (60% approximately in 2011 and 2013, and a bit less in 2002) uses a personal vehicle to move in the city, as a driver and as a passenger.

1.3 Objectives

The objective of this work is to explore the potential of DRT for elderly urban mobility using big data. The main goals are:

- To study the mobility of the elderly population (analysing the most used zone/lines/area and the corresponding schedules);
- To understand and characterise DRT services in urban areas;

- To review the state of the art review on the Dial-A-Ride Problems (DARP) algorithms;
- To implement a DARP algorithm;
- To analyse the sensibility of the algorithms with computational tests;
- To apply the algorithm to a case study in the MPA using real-world data.

1.4 Dissertation Structure

This section explains the structure of the document.

Chapter 2 introduces the definition of PTS and DRT, and explains the DRT main concepts. Subsequently, a review of state of the art of the DRT systems already implemented is performed.

Chapter 3 contains a literature review of the DARP, the methodologies to solve it and an example for each methodology.

Chapter 4 introduces a detailed and practical explanation of the implemented solution for the DARP algorithm.

Chapter 5 details the computational results and the cases study, with the respective results' analysis.

Chapter 6 the conclusions and future work are discussed.

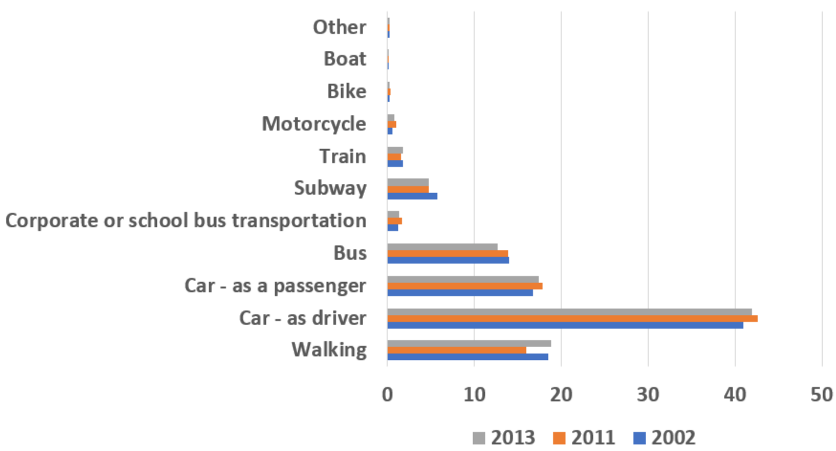


Figure 1.2: Most used means of transport in the Metropolitan Porto Area.

Introduction

Chapter 2

Demand Responsive Transport

PTS is a collective mode of transportation provided by several transportation modes, such as bus, metro, train or subway and others. With the use of the PTS, it is expected to reduce traffic congestion and environmental pollution. The traditional PTS has fixed routes and stops, and does not always provide equal mobility opportunities for all citizens.

The DRT system is one service of the PTS, created to overcome those problems and to get to the people that generally would not have access to the PTS. The DRT has flexibility in the choice of routes, modes of transport, schedules, service provider, vehicle allocation and payment systems. This type of flexible transport is usually focused on the service for people with disabilities [NWM+10].

In this chapter, the DRT service is presented in order to introduce the subject in matter. First, the definition and concepts of the DRT system are described and then examples of different real-world implementations are presented.

2.1 Demand Responsive Transport

DRT can also be called door-to-door Dial-a-Ride service [NWM+10], and this service was developed for people with reduced mobility as a complement to the FTS [Eur19a]. In the United States of America, a system called paratransit, also known as special transport services, was developed to transport people with low mobility in the urban areas[Fu02], [NWM+10].

A DRT system is an intermediate form of transport between PTS and taxi [GA03]. It is more flexible and expensive than the PTS providing similar flexibility as in the taxi service but with less cost [MN09]. In general, the main advantages of a DRT are:

- To adjusted to the needs of specific groups of the population, as reduced mobility;
- To overcome the limitations of the fixed transport system, in some periods, such as at night [IIdMedT19];
- To increase mobility convenience;

- To decrease travelling times of passengers.

There are two types of DRT, rural and urban, the former being more diversified and covering a more extensive range of systems. In order to characterise urban and rural DRT systems, the criterion used is the service-area. The rural DRT system is characterised mainly by the geographic size of the service-area used. For urban DRT systems service-area is defined mainly by population size and by the transit ridership and small urban DRT system may serve a population between 50,000 to 200,000; a large urban DRT system may serve a population from 200,000 to 1 million, and the largest urban DRT system a population higher than 1 million [EM09].

With the development and use of Information and Communication Technologies, the DRT system can book and schedule trips dynamically assigning passengers to trips and optimising the routes as are being requested [NWM+10]. Before the use of Information and Communication Technologies, on the initial DRT systems implemented, the operators had to book and schedule the user's trips manually, making it a service with a high cost of provision, lack of route flexibility and inefficiency on dealing with high demand [MN09], [MN03], while being mandatory for the customers to request in advance.

2.2 DRT Services Concepts

This section introduces the main concepts of the DRT service, namely routes, time, network and booking concepts, and vehicle allocation.

2.2.1 Routes and Time Concepts

In a DRT system, there is a different type of routes, unlike the fixed transit service which have predefined routes and schedules. Also, a DRT system can have four types of stops, as it is shown in figure 2.1.

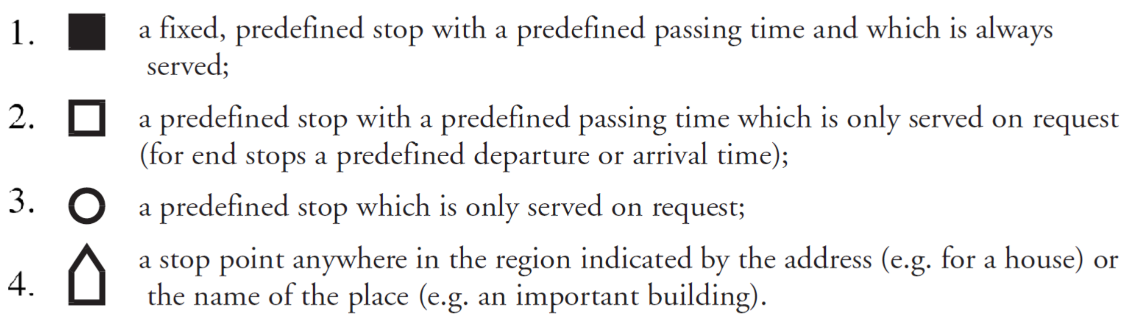


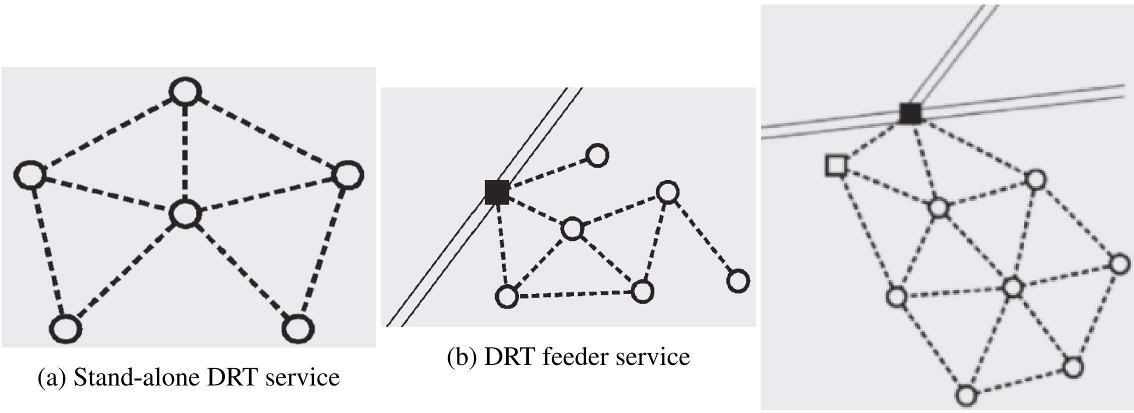
Figure 2.1: DRT types of stops
[GA03].

The stops of the DRT system can be predefined or not, and this will depend on the DRT service implemented. So different DRT services can be implemented, adapting to the circumstances in which they are inserted.

When the service for a DRT system is being planned, it is necessary to have into consideration the fixed predefined stops and the stops requested by the customers. In this way, from all four types of stops, several services schemes can be implemented, when combining the different routes serving the requests and at the same time optimising the service.

2.2.2 Network Concepts

The Network Concepts are (a) stand-alone DRT service, (b) DRT feeder service and (c) the DRT with multiple roles. The stand-alone DRT service (Figure 2.2a) is used generally in rural areas, offering transport to the local village, to the residents of a low-density area. The DRT feeder service (Figure 2.2b) was designed for the customers to have a connection to other PTS that will complement the trip, connecting the main stop of the DRT system [GA03]. The DRT with multiple service roles (Figure 2.2c) will provide access to the essential community services, such as travel facilities, shopping centre, services, school and work [GA03].



(a) Stand-alone DRT service

(b) DRT feeder service

(c) DRT service with multiple roles

Figure 2.2: DRT Network concepts.

2.2.3 Booking Concepts

The process of booking a trip usually includes three phases. First, the customer contacts the dispatcher centre with a route (pick up and delivery) and time (departure and arrival) information. The customer can perform it in several ways, such as phone call, SMS, internet, interactive voice response systems and an app. Then, if the requested service can not be executed, the operator sends several proposals offering a similar service to the customer. In the last phase, the booking confirmation is performed by the customer.

Scheduling a trip has four possible scenarios defining service [GA03]:

1. **Non-pre-booking trips**: the customer waits in a predefined stop with a predefined schedule for a vehicle of the service. The driver is the one responsible for deciding if the user can enter or not the vehicle, based on the number of empty seats in the vehicle at that moment. The destination of the user is booked on-board;

2. **Direct booking**; the customer should request the trip up to one or two hours before departure time, in order to the operator reorganise the service, and they will be picked up and dropped off on the location defined;

3. **Wide time window - trip notification**: the customer makes the booking request and receives a proposal from the operator with the departure and arrival times with a time window. The customer is notified moments before the trip of the accurate scheduled departure time, granting the operator more flexibility and optimisation in scheduling routes and timetables, and vehicle allocation;

4. **Collecting requests**: the operator will wait until the customers make all the requests. Based on the information of the departure and arrival time at the locations indicated by the customer, the operator will create a route which best optimises the service. After this, the customer is notified with all the details of the trip and can accept it or not.

2.2.4 Vehicle Allocation Concepts

A crucial decision when implementing the DRT service is the vehicle allocation (type and quantity of vehicles). The type of vehicle allocation is chosen by having in consideration the demand of the requested trips before each service. There are three distinct types of vehicle allocation:

1. **Fixed vehicle allocation**: there is only one type of vehicle, and it is determined by the service it operates on. Hence the flexibility is reduced. An example is a service that provides transport for people with low mobility with facilities for wheelchairs [GA03].

2. **Extendable vehicle allocation**: the service uses a type of vehicle that can be complemented with external transport to meet the demand, for example, a cooperation with a taxi company. This cooperation ensures that the system can fulfil all the requests throughout the day.

3. **Dynamic vehicle allocation**: is a more flexible service, that has various vehicles with several properties, such as capacity, accessibility and special facilities [GA03]. Some of the vehicles do not belong to a single company, thus can be operated by several companies.

2.3 DRT Implementations

A bibliographic review of the DRT services implemented in Portugal ([IIdMedT19],[HdFTP20], [dA20], [dTUdC19], [dVdC19], [dP19]) and in the United Kingdom ([NP12], [MN09]). Table 2.1 summarises the characteristics of the DRT services implemented with the information about the population density in each city ([dDPC19], [Wik19b], [Wik19a]).

In Portugal it was implemented DRT services in Funchal ([IIdMedT19], [HdFTP20]), Almada ([IIdMedT19], [dA20]), Coimbra ([dTUdC19]), Viano do Castelo ([dVdC19]), Portalegre ([dP19]), Beja ([IIdMedT19]) and Médio Tejo ([IIdMedT19]).

[1] possibility of a complementary route
[2] for the elderly the transport is door-to-door

Table 2.1: DRT Implementations.

Name	City (Country)	Service Area	Type of Vehicle	Predefined Routes	Stops type	Booking Time (before the trip)	Population Density (Hab/Km2)	References	Year of implementation
Collective taxis	Beja (Portugal)	Urban and Rural	Taxis	Yes[1]	4	Direct Booking	29,4	[IIdMedT19]	2000
Pantufinhas	Coimbra (Portugal)	Urban	Electric mini-buses	Yes	1	-	419,3	[dTUdC19]	2003
Linha Azul	Viana do Castelo (Portugal)	Urban	Electric buses	Yes	1	Non-pre-booking trips Booking	265,9	[dVdC19]	2004
Linha Azul	Portalegre (Portugal)	Urban	Mini-buses	Yes [2]	1,3	-	50,3	[dP19]	2004
Linha Eco	Funchal (Portugal)	Urban	Electric mini-buses	Yes	1,3	-	1.369,3	[IIdMedT19]	2006
FlexiBus	Almada (Portugal)	Urban	Electric mini-buses	Yes	1	Non-pre-booking trips	2.414,9	[IIdMedT19]	2010
Transporte a Pedido no Médio Tejo	Médio Tejo (Portugal)	Urban and Rural	Taxis	Yes	3	Collecting requests	69,9	[IIdMedT19]	2013
Ucall	Tyne and Wear (United Kingdom)	Suburban	-	Semi-fixed	1,2,3,4	-	2.091	[NP12]	2002
LinkUp	Tyne and Wear (United Kingdom)	Urban and Suburban	-	Yes	1,2,3,4	Direct Booking	2.091	[NP12]	2006

The common aspects of all DRT systems implemented in Portugal are:

- All the services have predefined routes and schedules;
- The services are operated by electric buses or mini-buses (except in collective taxis and *Transporte a Pedido no Médio Tejo* where taxis operate the service) [3];
- The service area is urban (except collective taxis and Transporte a Pedido no Médio Tejo where the service area are urban and rural);
- The target population of these services are mostly the senior and junior;
- To reduce car usage and traffic in the cities centres.

The main difference between the services is how the customers request the trips. The booking is performed by three types, Direct Booking, Non-pre-booking trips, and Collecting Requests. There are some services in which there is no information about it.

The collective taxis, in Beja, is a cooperation between the PTS service and the taxi company. This service only runs when the PTS is not available, such as at the weekends. The fare of each trip for each passenger is equal to the price in the PTS. People with reduced mobility can book a trip at least 30 minutes before the departure time and are transported to the requested location.

In Coimbra, the system was implemented in 2003, with three electric mini-buses and the residents have a free transportation service. Its service area is characterised by an ageing population, promoting, in this way, accessibility and reducing social exclusion. The restricted access to the

[3] there is no information on *Transporte a Pedido no Médio Tejo*

historical centre by private vehicles favours its conservation and increases the quality of life of the residents.

In Viana do Castelo the service implemented in 2004, operates with electric vehicles, with a total capacity of transportation of 22 passengers. A blue line establishes the routes in the pavement in the historical centre. This service also aids the mobility of the elderly in the city centre and it is a friendly, environmentally alternative.

In Portalegre, the project began in 2004 in the scope of the European Mobility Week. The vehicles of this service are mini-buses, and there is the main route that can integrate a complementary route. The goal is to maximise the use of parking spaces, thus reducing the number of cars in the historical centre.

In Funchal, the flexible public transport service operates with predefined routes and schedules, started in 2006. The customer needs to hail and ride. The route is performed by electric mini-buses adapted to transport people with mobility limitations and connects most of the city's parking lots.

In Almada, the service is composed by energy-efficient electric mini-buses, inaugurated in 2010. The passengers are picked up on predefined meeting points, the stops are made on request, and the service frequency is 20 minutes. This service runs between 7h and 19h on weekdays and Saturday mornings between 8h and 13h.

The *Transporte a Pedido no Médio Tejo* is a service implemented in 2013, in the Médio Tejo of Portugal. Like FTS, it has predefined routes, stops and schedules but it is operated by taxis with a capacity of 4 or 8 seats. However, a stop is only served if a customer requests it, and the requests should be made until the 15:00 of the previous day, by calling the travel dispatch centre, the trip is performed on the day, time and location predetermined in the booking.

Nexus is the regional public transport in Tyne and Wear, in the United Kingdom, operating since the late 1960s. In 2002 a pilot project called Ucall introduced a DRT service in Tyne and Wear with open-access. This service is operated as a semi-fixed route with intermediate stops. The customers can book a trip up to 8 days and at most 30 minutes before the trip. In the predefined stops, the customers could hail and ride; in this case, the decision to accept or not the customers is made by the driver. From 2006 UCall was extended, and a similar service was created, with a new brand LinkUp.

The LinkUp service is operated in a region with a high-density population and a service's area of 538 km^2, serving the entire population. Most of the services operate on fully flexible routes with no fixed times and no predefined timetables. However, some services operate daily on fixed routes and schedules, complementing the PTS.

Chapter 3

The Dial-a-ride Problem

This chapter describes the DARP, starting with a review of the literature for the problem (3.1). In section 3.2, a review of state of the art for DARP, specifically the exact methods, heuristics and metaheuristics, is presented. In section 3.3 the Multi-Objective Problems (MOP) are discussed.

3.1 The Dial-a-ride Problem

The DARP, a variant of Vehicle Routing Problem (VRP), consists of scheduling and routing vehicles to pick up and delivery a set of customers. The requests are usually made for the same day, a request from home to the desired location, and then a request from the desired location back home. The routes are computed to be performed at the minimum cost of operation with a feasible order in which the vehicles serve the requests [TV02]. The objective is to minimise the overall distance travelled, the total deadheading distance travelled and the total number of vehicles.

To solve this problem it is used mostly the Dijkstra algorithm and A-star algorithm, calculating the shortest distance between the origin node and the destination node. The A-star algorithm is a heuristic search algorithm, the searching efficiency and the results are affected by the evaluation function that determines the searching direction. The Dijkstra algorithm calculates the optimal path by iterating through all nodes, the search efficiency is much lower [CLY14].

Dial-a-ride services can be operated in one of two modes, static (off-line) or dynamic (on-line). The static mode only allows the customers to requests their trip in advance. Therefore all requests are known in advance, and the service is planned before the journey starts. The dynamic mode is a mix of static and dynamic mode where the customers can book a request in advance or immediately before the trip. Thus, the scheduling can be performed before the DRT service starts and altered or rescheduled, as new requests are made. As the service is designed for the elderly, the required time to enter and exit the vehicle can be substantially higher than the usual times, due to their reduced mobility. Furthermore, for this reason, an adequate boarding time should be considered for an accurate simulation [HHL12].

To implement a DRT service is necessary to have in consideration several variables, such as the requests made by the customers, the number of vehicles in the fleet, the capacity of each vehicle, the speed, the boarding time among others. Each request is composed of one departure location, arrival location and a time interval. In this work, the problem is that collective transport service has a homogeneous fleet of vehicles to serve a set of customers requests. It is required for each customer to be picked up and delivery by the same vehicle without violating the time window constraint.

The problem of this work is to solve a Capacitated Vehicle Routing Problem With Pick up and Deliveries with Time Window constraints (CVRPWPDTW) [HHL12], that is used to design x routes of minimum cost, one for each vehicle so that all customers are picked up exactly once and dropped off exactly once. The CVRPWPDTW is NP-Complete as it is a natural generalisation of the Travelling Salesman Problem[HPR13].

The VRP can be formulated as follows. Let G = (E,N) be a graph where N = {1,2,...,n} are the vertex set or node-set, and E are the edges or arc set, of directed edges. There is a fleet of similar vehicles of capacity Q, which collects customers q_i at each vertex (demand) $i \in N$. A non-negative cost c_{ij} is associated with each edge $(i, j) \in N, i \neq j$, this cost is the distance travelled. An extension of the VRP, the VRP with time windows, which has a time window $[a_i, b_i]$ constraint at each vertex. In the Vehicle Routing Problem with pick-up and delivery, each customer request corresponds to a pair of vertices [Kac09]. Usually, it is added a waiting time in case of the vehicle arrives before the lower bound a_i. The objective of a VRP is to minimise the distance travelled, for each vehicle, or the number of vehicles.

3.2 Approaches for DARP

The problem type is known to be an NP-Complete combinatorial problem, and therefore optimal solutions are hard to find in polynomial time. In this section, it is presented a review of methodologies for solving the VRP, more specifically in this case, the DARP. These approaches can be divided into exact methods, heuristics and metaheuristics.

Exact methods find optimal solutions, however at the expense of high computational time, especially when using large real-word data sets. There are studies focused on solving single-vehicle or multi-vehicle DARP ([Cor06], [RCL07], [LD04]) with exact formulations, by using branch-and-cut algorithms and others by using dynamic programming ([DDS86], [DDGS95], [KRKT87]).

Thus, in order to solve hard combinatorial optimisation problems can be used heuristics, which are divided into construction heuristics, improvement heuristics, and metaheuristics.

3.2.1 Construction Heuristics

Construction heuristics reviewed are insertion heuristics, savings heuristics, two-phase construction heuristics, route-first cluster-second and cluster-first route-second.

3.2.1.1 Insertion heuristics

The insertion heuristic can be used to construct an initial solution. The process of this heuristic is at each iteration it is selected a vertex from the not visited list of vertex, creating a feasible and with a least-cost solution. The vertex is inserted between two consecutive vertices already in the route. This process repeats itself until all vertices are visited. There are some variants of this heuristic, in which some constraints are verified before adding a new vertex to the route and if the constraints are not violated a new route construction begins [CW64].

3.2.1.2 Savings heuristics

In the savings heuristics, the construction of an initial solution is the starting point, by creating a route that connects the depot and each vertex, without violating a constraint that each vertex is visited exactly once. There is an ordered list of savings between each pair of vertices. The list guides the merging process, and it is sorted in non-increasing order. Two routes are selected at each iteration and merged without violating the constraints, and this process repeats itself until there are no more routes to merge [CW64].

3.2.1.3 Route-First Cluster-Second

Route-First Cluster-Second is an algorithm commonly used in VRP. The route construction is done by visiting all nodes while relaxing the constraint capacity of the vehicles and other constraints, and this way creates an unfeasible solution. Then, clusters are formed by converting the route into smaller and feasible routes that satisfy the constraints.

An exact polynomial-time algorithm is used to perform the partition in a VRP. As this algorithm works in an acyclic graph, it is possible to obtain a solution by solving the shortest path from node 1 to node x in polynomial time [cited in Pot09].

3.2.1.4 Cluster-First Route-Second

Cluster-First Route-Second is an algorithm also frequently used in VRP, first forming the clusters having in consideration the VRP constraints and that each cluster is associated with one vehicle. Then, for each vehicle, the routes are built, by visiting all nodes of that cluster[Pot09].

Another DARP was solved by implementing a mini-clustering algorithm which uses a parallel insertion method processing requests sequentially. The requests are grouped in time and space while assuring the quality of service for the users. There are some constraints followed in this heuristic, which must not violate the time window, capacity, pick up and delivery constraints. The requests in the same mini-cluster must be served by the same vehicle. The objective of the heuristics in [DDS+91] is to minimise the sum of all vehicle travelled kilometres.

3.2.2 Improvement Heuristics

This heuristic is based on the improvement of an initial solution, optimised with a local search heuristic. A set of neighbourhood solutions are generated by modifying the current solution. Then, they are compared with the current solution, and the best solution becomes the new current solution. By exchanging the position of one vertex (vertex move), another modification is exchanging two nodes between them, and the other modification is removing an edge from the current solution or adding another edge to the solution [Pot09]. This process repeats itself until there is no better solution in the neighbourhood solution. The modifications are based on the arcs or nodes of the solution and do not violate constraints. Thus the new solutions are feasible.

In the literature are presented several adaptations of those modifications. For example, the exchange of consecutive vertices, instead of a single vertex, can be applied to each route individually, modifying only the sequence of the vertices.

3.2.3 Metaheuristics

The metaheuristics reviewed in this subsection are Genetic Algorithms, Simulated Annealing, Greedy Randomised Adaptive Search Procedure, Ant Colony Optimisation and Tabu Search.

3.2.3.1 Genetic Algorithms

The Genetic Algorithm is a random search technique. The optimal global solution is found in the complex multi-dimensional search spaces, operates with a population of possible solutions. This algorithm uses operators that were inspired in the natural evolution process, which are also known as genetic operators, manipulating individuals in a population through the generations to improve their fitness gradually [H^{+}92].

There are three operators used in the genetic algorithms: (1) the selection, (2) the crossover and (3) the mutation. The selection operator (1) intends to reproduce more copies of individuals whose fitness values are higher. The crossover (2) selects a random point in the two individuals, swapping the parts after the point, this produces two new individuals (children). The mutation (3) is a procedure in which all bits of the individuals are checked where all bit values are randomly reversed.

The article [RDS06] proposed a method based on genetic algorithms to solve a VRP. The objective is to construct routes for transporting disabled people minimising the number of vehicles and at the same time satisfying the time window customers' requirements. Two formulations to solve the problem are considered, the first one with a fixed number of vehicles and the second without a fixed number of vehicles, in order to minimise the number of customers not served and to minimise the number of vehicles used, respectively.

3.2.3.2 Simulating Annealing

The Simulating Annealing algorithm is based on the analogy between the problem of solving the combinatorial optimisation problems and the annealing of solids [KGV83]. In the analogy, the feasible solutions of the optimisation problem are represented by the states of the solid. The values of the objective function correspond to the state of energy computed at those solutions where the optimal solution is the minimum energy state.

The cost function and mechanisms define how to generate neighbourhood solutions. The algorithm consists of a sequence of iterations. First, it chooses a random neighbourhood solution to the current solution in order to create a new solution. To decide whether the new solution is accepted, it must have a negative difference in the cost function, between the neighbourhood solution and the current solution. Otherwise, it is randomly chosen if it can be accepted to become or not the current solution [PK00].

A variant solution of the VRP with simultaneous pick up-deliveries and time window of goods was implemented, having in consideration the time window and capacity constraint, and that all the customers must be served. The simulated annealing accepts new current solutions, which do not improve with a probability, allowing to escape from local minima, becoming in this way a global optimum algorithm [WZMS13].

3.2.3.3 Greedy Randomised Adaptive Search Procedure

The Greedy Randomised Adaptive Search Procedure is a metaheuristic based on a multi-start strategy with two phases, the construction and the local search. It uses several initial solutions generated through repeated applications of a semi-greedy process. This process uses a list of candidates that can be incorporated into the building solution. After application of a greedy evaluation, it is created a restricted candidate list formed by the best elements. The next element to be incorporated is chosen randomly from the restricted candidate list, which may lead to an unfeasible solution.

In the local search phase, better solutions are searched for in the neighbourhood of the best solution, and whenever a better solution is found, it replaces the best solution. The initial solutions, the neighbourhood search technique, the strategy to evaluate the cost function value, can be affected by the speed and the effectiveness of the procedure. There are two ways to select the neighbourhood solution to implement, the best-improving and the first-improving. In the best-improving, all neighbourhood solutions are evaluated, and the best is selected. In the first-improving method, the neighbourhood solutions are evaluated iteratively, and the first solution found better than the current solution is implemented[RR10].

A Vehicle Routing Problem With Time Windows was studied in [Cha03] using GRASP in order to minimise the fleet size and the travelled distance. The solution of the VRP with the GRASP is a list of ordered customers visits, and each list corresponds to a vehicle. Besides that, at all times, the capacity of the vehicle must not be exceeded. By applying GRASP to VRP, it was constructed an initial solution which was further improved by applying the local search phase.

3.2.3.4 Ant Colony Optimisation

The Ant Colony Optimisation metaheuristic is based on the real-life behaviour of an ant colony, in which the ants use pheromones as a mean of communication. A colony of simple agents (artificial ants) that communicates indirectly through (artificial) pheromone trails used this as an analogy. The construction of the heuristic is implemented using a probabilistic decision based on the artificial pheromone trails function and a heuristics based on data input on the problem to solve. The use of the pheromone trails is a significant difference, which is taking into consideration by the accumulated search experience.

As opposed to what happens to other metaheuristics, the ant colony optimisation algorithm begins with an empty solution rather than with an initial solution. The solution is built incrementally by adding components of the solution without backtracking until a complete solution is obtained. A greedy algorithm is used to determine the heuristic of each partial solution. It has better quality than to use randomly generated solutions, despite being a greedy algorithm and often causes the heuristic to be tided in an optimal local [DS10].

A hybrid ant colony optimisation algorithm was used to solve the VRP with capacity and distance constraints with one central depot and a homogeneous fleet of vehicles. It also uses the 2-opt-heuristic, which is an exchange procedure that generates a so-called 2-optimal tour. If there is no possibility to reduce the tour by exchanging two arcs, it is called a 2-optimal tour. Also, the algorithm was evaluated for fourteen benchmark problems and then compared those results to other metaheuristics approaches such as Tabu Search (TS) and Simulated Annealing [BHS99].

3.2.3.5 Tabu Search

TS is a metaheuristic approach to solve combinatorial optimisation problems, which is an extension of the Local Search Algorithm and is characterised by the use of flexible memory. The search space of this algorithm is all the possible neighbourhood solutions. Unfeasible solutions can be accepted in order to obtain even better solutions, becoming the new current solution where relaxing some constraints, increasing the search space. The neighbourhood structure is what defines how the current solution can change to the neighbourhood solution.

After having all neighbourhood solutions from the search space, it is chosen the best solution for the next iteration, by selecting the best available move, and the process repeats itself until the termination criteria is reached. Only feasible solutions can replace other solutions as the best solution. TS has a short-term memory list of tabu moves, and this list prevents the algorithm of cycling away from local optima through non-improvement moves or cycling back to a solution previously analysed, by storing the previous moves in a list. This contributes to the algorithm to intensify and diversify the space search. The tabu list may prevent an attractive move, which is a disadvantage.

For this reason, the aspiration criteria were introduced. The aspiration criteria are used to allow a move even if it is in the tabu list. The move is allowed if the solution resulted from the move is better than any other solution obtained so far.

The optimisation algorithm could search for better solutions forever unless the optimal goal solution is known beforehand. Thus, a termination criterion needs to be implemented. The algorithm ends by reaching a fixed number of iterations or by reaching a fixed number of iterations without improvement or when the objective reaches a predefined threshold value.

The article [CL03] describes a TS heuristic for the static multi-vehicle DARP. The solution of the VRP aims to minimise the vehicle routes' costs and at the same time to serve all customers at their desired departure and destination locations, within the time window required on their departure and destination. The vehicle capacity, the maximum ride time for all customers and the route duration are other constraints considered in this article.

This algorithm is commonly used to solve VRP and was chosen in the DARP optimisation phase for being a great algorithm to solve combinatorial problems and simple to implement.

Figure 3.1 it presents a generic methodology used in all DARP solutions found in the literature review, that corresponds to the generation of an initial solution and further optimisation of it. This methodology was used in this dissertation.

3.3 Multi-Objective Approaches

Most of the studies on VRP focus on the single-objective optimisation, though it is growing the concern on the multi-objective optimisation problems as the real-world problems involve the comparison between more than one objective [JPKC08]. This work focuses on the multi-objective problem since it explores three objectives while optimising the solution, total travelled kilometres, total deadheading kilometres and the number of vehicles. The main concepts related to MOP are presented in the next paragraphs.

Commonly, VRP focus on the optimisation of the total travelled kilometres, however the focus of this work is a triple optimisation criteria, representing a MOP. The general formulation of a MOP (3.1) is [JST08]:

$$(MOP) = \begin{cases} min\ F(x) \ = \ (f_1(x), f_2(x), ..., f_n(x)) : \ n \ \in \ \mathbb{N}_{\geq 2} \\ \text{s.t. } g_i(x) \ \geq \ 0, \ i \ = \ 1, ..., m \end{cases} \tag{3.1}$$

Where a solution (decision vector) is x, $x = (x_1, x_2, ..., x_r) \in X$, and X defines the decision space. An objective vector is F(x), $F(x) = (f_1(x), f_2(x), ..., f_n(x)) \in Y$, where Y defines the objective space.

$$The\ feasible\ set = \Big\{ X_F = \ x \ \in \ X : \ g_i(x) \ \geq \ 0, \ i \ \in \ \mathbb{N}_{\geq 2} \tag{3.2}$$

A solution x is said to dominate 3.3 solution y, for $x, y \in X_F$ if the following conditions are true:

$$Pareto\ dominance = \begin{cases} F(x) \preceq F(y) : \ \forall j \in \mathbb{N} \\ f_j(x) \prec f_j(y) : \ \exists j \in \mathbb{N} \end{cases} \tag{3.3}$$

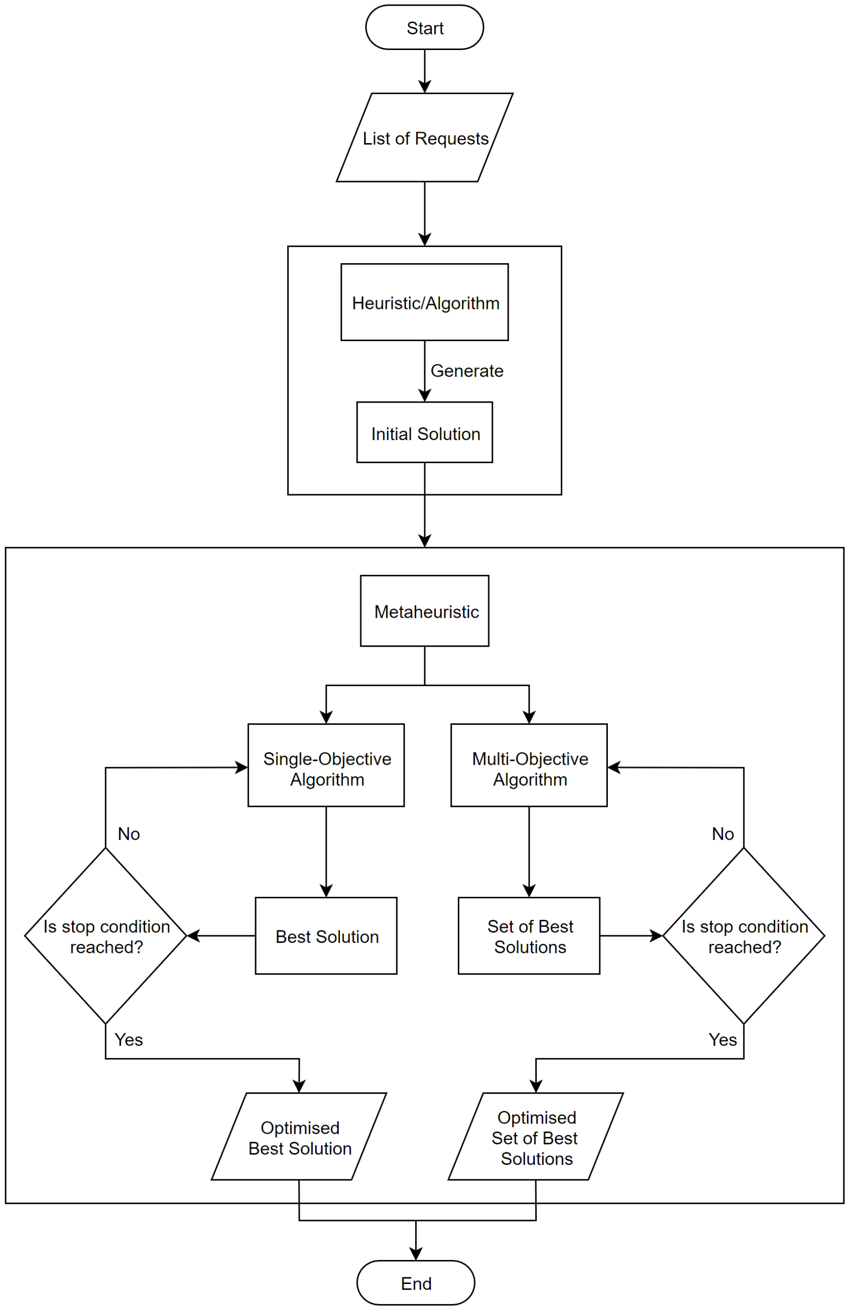


Figure 3.1: Generic Methodology Scheme

And a solution x is said to be non-dominated if the following conditions are true:

$$Non-dominated\ solution = \begin{cases} \nexists a \in A: F(a) \prec F(x) \\ x \in X_F \\ A \subseteq X_F \end{cases} \tag{3.4}$$

These non-dominated solutions, also known as Pareto Set, dominate any other solution from the feasible solution space. A solution $x \in X_F$ is said to be Pareto optimal if it is non-dominated regarding X_F. As the MOP is being solved, it should converge toward the Pareto Optimal Set, which is the set of all Pareto optimal solutions. The set of the corresponding objective vectors is called the Pareto optimal front or the trade-off surface.

The chapter [JST08] approaches two problems, the VRP with route balancing and the bi-objective covering tour problem. In order to solve the problems, a two-phase approach can be used based on the combination of single-objective techniques and a multi-objective evolutionary algorithm, provides diversification and intensification for the search space, respectively. This chapter aims to present an overview of the VRP multi-objective optimisation and what it can bring to VRPs. As more than one objective is minimised, it is possible to minimise the total travelled kilometres, and at the same time, the total travelled kilometres for the shortest route. It is also possible to evaluate customer satisfaction and to minimise the variables of the operators perspective, minimising the number of vehicles and optimising the effectiveness of the vehicles. So, with a multi-objective approach, it is possible to solve a real-world problem, which requires to minimise more than one variable.

Chapter 4

A Multi-Objective approach for DARP

This chapter details the solution of the static DARP. In subsection 4.1 it is presented the general methodology used. In the next subsections, it is presented the methodology used to develop the DARP algorithm, which is split into the Assigning Requests to Vehicles (ARV) algorithm, in subsection 4.2.2, and the Multi-Objective Tabu Search Algorithm (MOTSA), in subsection 4.2.3.

4.1 General Overview of the Solution

The solution implemented was based on the literature review of DARP (chapter 3.2). First, an initial solution with a good outcome should be feasible and do not violate constraints. Then a heuristic or metaheuristic algorithm should be used to optimise the solution. With a good initial solution, the solution optimisation could be reached using fewer iterations.

It is required a map to perform the routes between the desired locations. It was extracted from the Open Street Map and converted into three files using a parser, the roads, the edges and the nodes. The files have information about the coordinates of each node, the real-world nodes connection and about the street names and whether the edges are bidirectional. The distance between the nodes is calculated using the Haversine formula, later used in the A* algorithm. The Haversine formula determines the distance between two points on the surface of a sphere given their longitudes and latitudes, this was used due to the information in the node file being expressed in polar coordinates.

An adaptation of the Nearest Neighbour was used. However, instead of selecting the next request by the distance, it was selected chronologically. With all requests assigned to vehicles, the initial solution is complete.

To optimise the initial solution, a MOTSA was implemented. For each request in the list of not served requests, it is added a vehicle to serve it, and the vehicles are then added to the initial solution. It explores the neighbourhood solution by modifying the order in which the requests are served, named candidate solutions. By comparing amongst one another, the dominated candidates are eliminated, and the non-dominated candidates became the solutions to explore in the next iteration. The final solution is the set of feasible non-dominated solutions. The optimisation ends when there are no more solutions to explore or when it is reached the maximum number of iterations or the maximum number of iterations without improvement.

4.1.1 Storage of all Routes

As the route calculation was performed several times per iteration, the execution time of the DARP was high. It was improved with the calculation of all routes and storage in a database for further consultation. In this way, predetermined values are used, instead of calculating the routes as they are needed, and a compound index of two columns was created using the fields '*node_origin*' and '*node_destination*'.

4.2 DARP Algorithm

A DARP algorithm to solve a CVRPWPDTW was implemented. As it was not necessary to explore the dynamic mode of the DRT services, a static DRT service was implemented, which means that one of the parameters is a list of all the requests.

The list of requests is the set of all customers' requests. Each request consists of the departure and arrival time and location, the time-window and the number of customers. The request can be of individual passengers or groups of passengers. The list of requests includes the departure and arrival times called, origin node time and destination node time, respectively. The departure and arrival locations, named the origin node and the destination node, respectively.

The parameters necessary to implement the DARP, are represented in figure 4.1, are the list of the requests, the number of vehicles, the vehicles' specifications, the total number of iterations, the total number of iterations without improvement, a threshold number, a map and if it is to use the previously calculated paths or not.

Regarding the parameter of the number of vehicles, there are two possibilities, choose a positive integer number or have "all" vehicles necessary to answer the needs of the service. If there is a finite number of vehicles and if a request cannot be assigned to any available vehicle, the request is added to the list of not served requests. Once all customers must be transported, and the cost associated with each request not served is the same as adding another vehicle to the solution, the parameter "all" exists in order to the list of not served requests always remaining empty.

The vehicles' specifications, the speed and its capacity, the total number of iterations, the total number of iterations without improvement and the threshold number must be integers, the last three are only used in the optimisation algorithm. The threshold number limits the number of candidates chosen to become the solutions of the next iteration.

The DARP methodology is presented in the following subsections.

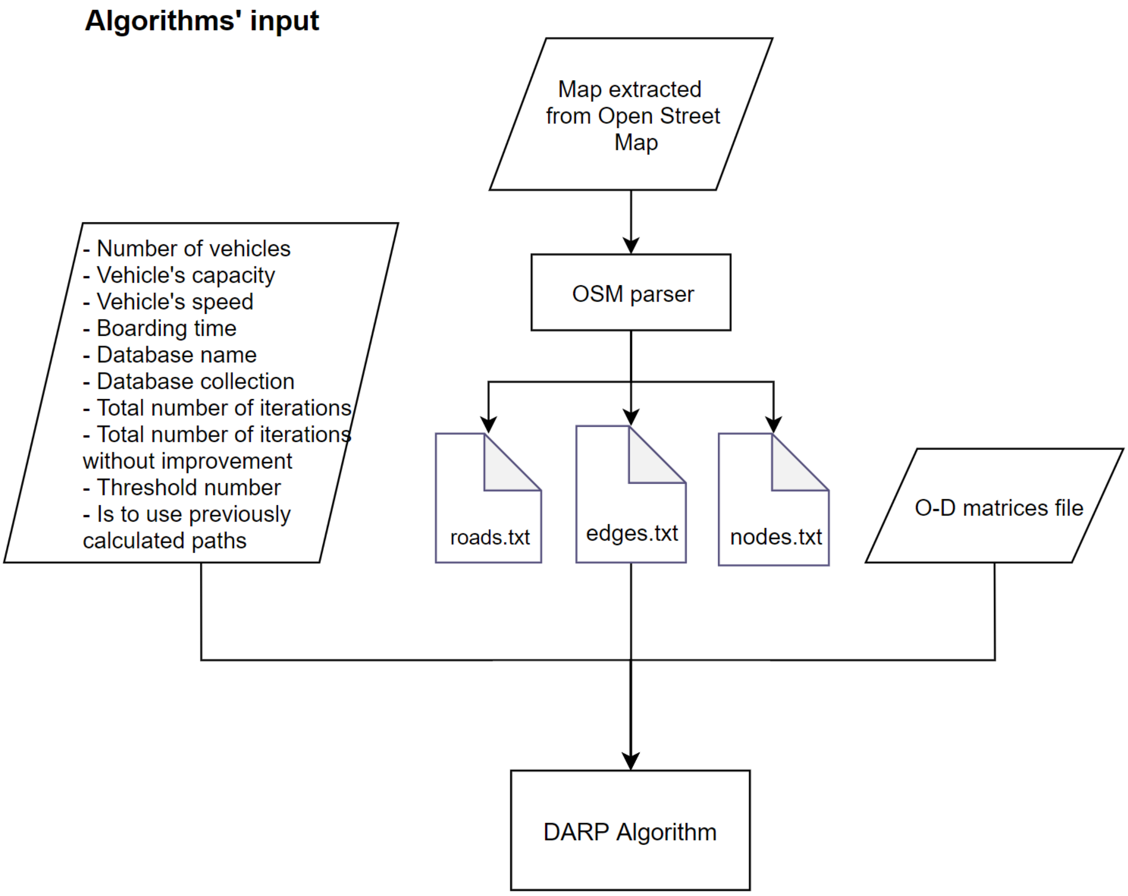


Figure 4.1: Algorithms' Parameters.

4.2.1 Requests

For the computation results presented, in section 5.1, requests were generated randomly, and for the case study, it was used a sample of real-world validations in the MPA. The explanation of the request generation is presented following.

The process starts by choosing randomly two nodes from the map, generating the requests randomly. Two constraints were defined: (i) all the requests must be feasible and (ii) all the nodes must be connected, for the heuristic always finding a route between two nodes. For each request, the origin time was created by adding 9 hours to the total travelled time, and each destination time was created by adding 10 hours, the total travel time and the total boarding time. The possible time windows are 10, 20, 30, 40, 50 and 60 minutes. For example, with a 20 minutes time window and origin time at 08:00, the time interval to pick up the customer is from 07:50 until 08:10. The requests are also composed by the number of costumes, that are randomly chosen between 1, 2 and 3.

4.2.2 Assigning Requests to Vehicles Algorithm

In order to generate an initial solution, it is used the ARV Algorithm to assign each request to a vehicle. In figure 4.4, it is presented the flowchart of it, which presents the methodology of scheduling the requests and assigning them to vehicles, in figure 4.5.

In this phase, the A* algorithm is used to calculate routes between locations or nodes or stops, computing the route with the shortest path, and thus obtaining the travelled time between nodes.

Every request must be feasible, this means, that it must be possible for the vehicle to travel between the departure and arrival locations without violating the customers' desired times for departure and arrival within the time window. If the constraints are violated, then the request is considered not feasible, and it is added to the list of not feasible requests. There is no cost added to the solution if the request is not feasible.

There are three possible states for the request: waiting (waiting to be picked up), boarded (on board of a vehicle) and alighted (dropped in the destination). At each iteration, the next request is chosen from the list of requests chronologically. The requests' state is updated as the requests are being assigned to vehicles if they are being picked up (the state changes from waiting to boarded) if they are being dropped off (the state changes from boarded to alighted). When the request is updated to alight, it is removed from the list of requests. At each iteration, a request is assigned to a vehicle, and the list of requests is updated. The route between the location of a depot and the origin node of the first request to be served is not relevant in this work. For this reason, each vehicle begins its journey in the first request. The algorithm ends when there are no more requests in the list to be assigned, resulting in the initial solution.

Constraints applied when assigning a request to a vehicle:

- Each request is always served by the same vehicle (picked up and dropped off);
- A request can only be dropped off after being picked up;
- A request must be picked up once and dropped off once.
- A request is only assigned to a vehicle if all customers already assigned to it, can leave the vehicle within time at the desired destination;
- A request can only be assigned to a vehicle if there are enough empty seats to accommodate all customers of the request;
- A request can only be assigned to a vehicle if the time window constraints are not violated;

These constraints are always verified when assigning a request to a vehicle and must not be violated.

4.2.2.1 Feature Enhancements Implemented

In this subsection, a couple of feature enhancements that were implemented are presented, which are used in both ARV Algorithm and MOTSA.

When a request is being assigned to a vehicle, the route between the current vehicle's position and the request's location is calculated. If the vehicle reaches the node to be served before the time window, it can wait for the request, or it can serve other requests while waiting.

While a request is being assigned, the path between the vehicle and the request to be served is analysed. If there are other requests to be served in the path, they will be served.

The maximum time to drop off is the destination time plus the time window, and there is no other time constraint to drop off a customer. For example, the time window is 10 minutes, and the destination time is 10:10:00, so the maximum time to drop off the customer is 10:19:59. This time window constraint relaxation when dropping was implemented to reduce the time spent inside a vehicle by the customer and provides more available seats earlier, maximising the number of customers transported by a vehicle.

Another feature enhancement implemented is delivering some customers when a vehicle has no more empty seats. They are delivered until at least the number of available seats is equal to the average number of customers per request. This can decrease the total number of vehicles in the initial solution by assigning more requests to that vehicle earlier than it would.

4.2.2.2 Example of how it works

In this subsection it will be presented a practical example of ARV Algorithm (figures 4.4 and 4.5) using the requests presented in listing A.1.

At the beginning of the ARV algorithm, the list of requests is ordered by the origin time to pick up the requests in ascending order. The first request to be served is the 'client_E'. After being assigned a vehicle to pick up the request 'client_E', the state of the request is updated to boarded. The list of requests is ordered once more, however this time the request 'client_E' is ordered by its destination time. At this moment the order of the list of requests is 'client_C', 'client_B', 'client_E', 'client_A', 'client_D'. So, the next request to be served should be 'client_C', however, the path from the vehicle's current node to 'client_C' has an intermediate stop, in this case, the origin node of the request 'client_B'.

It is verified if the constraints are violated and if not the request 'client_B' is picked up before the vehicle picking up the request 'client_C'. The current order of the vehicle's requests is 'client_E', 'client_B', 'client_C'. The list of requests is once more reordered and the next request to be served is the 'client_E', although the request was already picked up now, it is necessary to drop it off.

Even though the first request in the list is the 'client_A', other requests are served first, because the vehicle would reach the request 'client_A' before the origin time, violating the constraints. For this reason, other requests are served, without affecting the original request. In this way, instead of waiting for the customers, the operation time is optimised, by dropping the 'client_C' and 'client_B' and then picking up the 'client_A'. Then 'client_D' is picked up. The remaining requests are dropped off, and the initial solution is created.

This solution uses only one vehicle, with a total of 21.75 km and 3.69 km travelled without any client inside the vehicle.

4.2.3 Multi-Objective Tabu Search Algorithm

The algorithm has the following phases, (1) the initial solution generation that tries to ARV, presented in figures 4.4 and 4.5; (2) the creation of a candidate list of solutions based on the neighbourhood structure as presented in figure 4.6; (3) Application of a TS Algorithm for the MOP, as presented in figure 4.3. The result of the DARP Algorithm is a list of feasible non-dominated solutions. The cost of a solution is represented by the total travelled kilometres and the total deadheading kilometres and the number of vehicles. The algorithm's main goal is to minimise the cost of the solution while ensuring that all customers are served without violating constraints.

Figure 4.3 has a flowchart of the DARP Algorithm and the figure 4.6 has a flowchart that generates all the solution candidates for the next iteration of MOTSA from a solution.

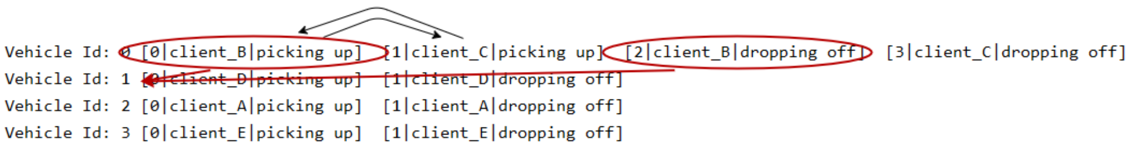


Figure 4.2: Explanation of the requests exchanges.

A solution is characterised by neighbourhood structures, the ordered requests, represented between '[' and'] ' (figure 4.2). In that figure is presented an example of a solution. The request-id composes each neighbourhood structure and whether the request is being picked up or dropped off. Every neighbourhood structure of the solution being analysed is associated with a unique position number. From the initial solution, it is created a list with all possible exchanges.

The type 1 of possible exchanges is exchanging the order of the neighbourhood structures, taking into consideration that the constraints cannot be violated. So, it is not possible for a request to be picked up by a vehicle and dropped off by another. As it is demonstrated in figure 4.2, the neighbourhood structure '[2|client_B|dropping off]', from vehicle id equals to 0, cannot be exchanged with the '[0|client_D| picking up]', from vehicle id equals to 1. For these cases, a new type of possible exchanges (type 2) is implemented, the exchange is moving a request from one vehicle to another. Examples of the exchanges are '(1,2)' and '(client_B,v1)', in the first case two neighbourhood structures are exchanged and in the second case the neighbourhood structures of request 'client_B' are moved to the vehicle with id equals to 4. A visual explanation of the two possible exchanges is demonstrated in figure 4.2, the exchange type 1 corresponds to the black arrows, and the exchange type 2 corresponds to the red arrows.

From all the possible exchanges it is verified if the exchange complies with the constraints mentioned in section 4.2.2 and whether it is inside the tabu list. By executing an exchange in a solution, it is generated a candidate solution, and this exchange is added to the list of tabu list, preventing the exchange from being reversed for the next three iterations. The new candidate solutions are added to the list of candidates, and its cost is calculated.

The cost of the candidate solution is calculated by computing the new paths, using the A* algorithm, from one neighbourhood structure to the other, in exchanges type 1. In exchanges type

2, it is applied the ARV Algorithm to the vehicle's requests, in order to obtain the cost of the solution.

Afterwards, the list of candidates is analysed and by comparing the costs between candidates a list of non-dominates candidates is built.

The threshold number is used to select a number of candidates from the list of candidates. In the one hand, if the threshold number is lower than the total number of candidates, they are all selected to become the new solutions. On the other hand, if the size of the list is bigger than the threshold number, a subset of the list is selected by choosing 1/3 of the threshold number of the candidates for each cost's variable with the lowest value. For example, if the size of the list is 20 and the threshold number is 12, there will be 12 candidates for the next iteration, four candidates with the lowest value of the total kilometres travelled, then four different candidates with the lowest total kilometres travelled without passengers and then four different candidates with the lowest number of vehicles.

Now, the candidate's solutions are added to the solutions list. Also, the solution list of non-dominated solutions is updated, comparing all the solutions amongst one another and removing the dominated solutions.

The stopping condition are:

- The total number of iterations is reached;
- The number of iterations without improvement is reached;
- All the solutions were explored for candidates;
- The non-dominated candidate list is empty.

The final result is a list of feasible and valid solutions, that corresponds to a set of the non-dominated solutions of the problem, the Pareto optimal set.

4.2.3.1 Example of how it works

In this subsection it will be presented a practical example of MOTSA (figures 4.3 and 4.6) using the requests presented in listing A.1.

The algorithm starts with the solution generated from the VRP algorithm, that was presented in section 4.2.2.2. At each iteration, it is verified from a list of all possible exchanges the ones that are feasible for the problem, in table 4.1. As the solution has only one vehicle, the requests being exchange between vehicles are not represented in the table. When a new solution is generated it is added to the solutions list and compared with the solutions in the best solution list, as the solutions are being dominated, they are removed from the list, and only the non-dominated solutions belong to the best solutions list. As it is possible to observe from the table 4.2, the initial solution the new solutions generated become better, the number of vehicles remain the same. However, the total kilometres travelled, and deadheading kilometres decreased. The final best solutions are the solution with id equals to 13, 14 and 15, 6.51 km, 6.23 km and 6.48 km total travelled kilometres, respectively. The deadheading kilometres for each solution, respectively, is 0 km, 1.37 km and 1.21 km. All the solutions use one vehicle to perform the service.

Table 4.1: List of all possible exchanges.

	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
0	-	(0,1)	(0,2)	(0,3)	(0,4)	(0,5)	(0,6)	(0,7)	(0,8)	(0,9)
1	-	-	(1,2)	(1,3)	(1,4)	(1,5)	(1,6)	(1,7)	(1,8)	(1,9)
2	-	-	-	(2,3)	(2,4)	(2,5)	(2,6)	(2,7)	(2,8)	(2,9)
3	-	-	-	-	(3,4)	(3,5)	(3,6)	(3,7)	(3,8)	(3,9)
4	-	-	-	-	-	(4,5)	(4,6)	(4,7)	(4,8)	(4,9)
5	-	-	-	-	-	-	(5,6)	(5,7)	(5,8)	(5,9)
6	-	-	-	-	-	-	-	(6,7)	(6,8)	(6,9)
7	-	-	-	-	-	-	-	-	(7,8)	(7,9)
8	-	-	-	-	-	-	-	-	-	(8,9)
9	-	-	-	-	-	-	-	-	-	-

Table 4.2: Table with the solutions generated in the Tabu Search Algorithm.

Iterations	**Solution id generated**	**Exchange**	**Total km**	**Deadheading km**	**Number of vehicles**	**From solution**	**Best solutions at the end of the iteration**
0	0	None	21.75	3.69	1	-	0
1	1	(0,2)	14.16	1.78	1	0	1,2
	2	(3,5)	16.56	0	1	0	
2	3	(1,2)	13.82	1.78	1	1	3,4
	4	(4,6)	15.57	0	1	2	
3	5	(8,9)	15.31	0	1	3	5,6
	6	(0,1)	12.26	1.21	1	4	
4	7	(5,7)	15.03	0	1	5	7,8
	8	(3,7)	12.04	0.41	1	6	
5	9	(3,7)	12.16	0	1	7	9,10
	10	(2,3)	8.93	0.41	1	8	
6	11	(6,9)	9.88	0	1	10	11,12
	12	(8,9)	6.91	0.41	1	10	
7	13	(8,9)	6.51	0	1	11	13
8	14	(7,8)	6.23	1.37	1	13	13,14
9	15	(5,6)	6.48	1.21	1	14	13,14,15

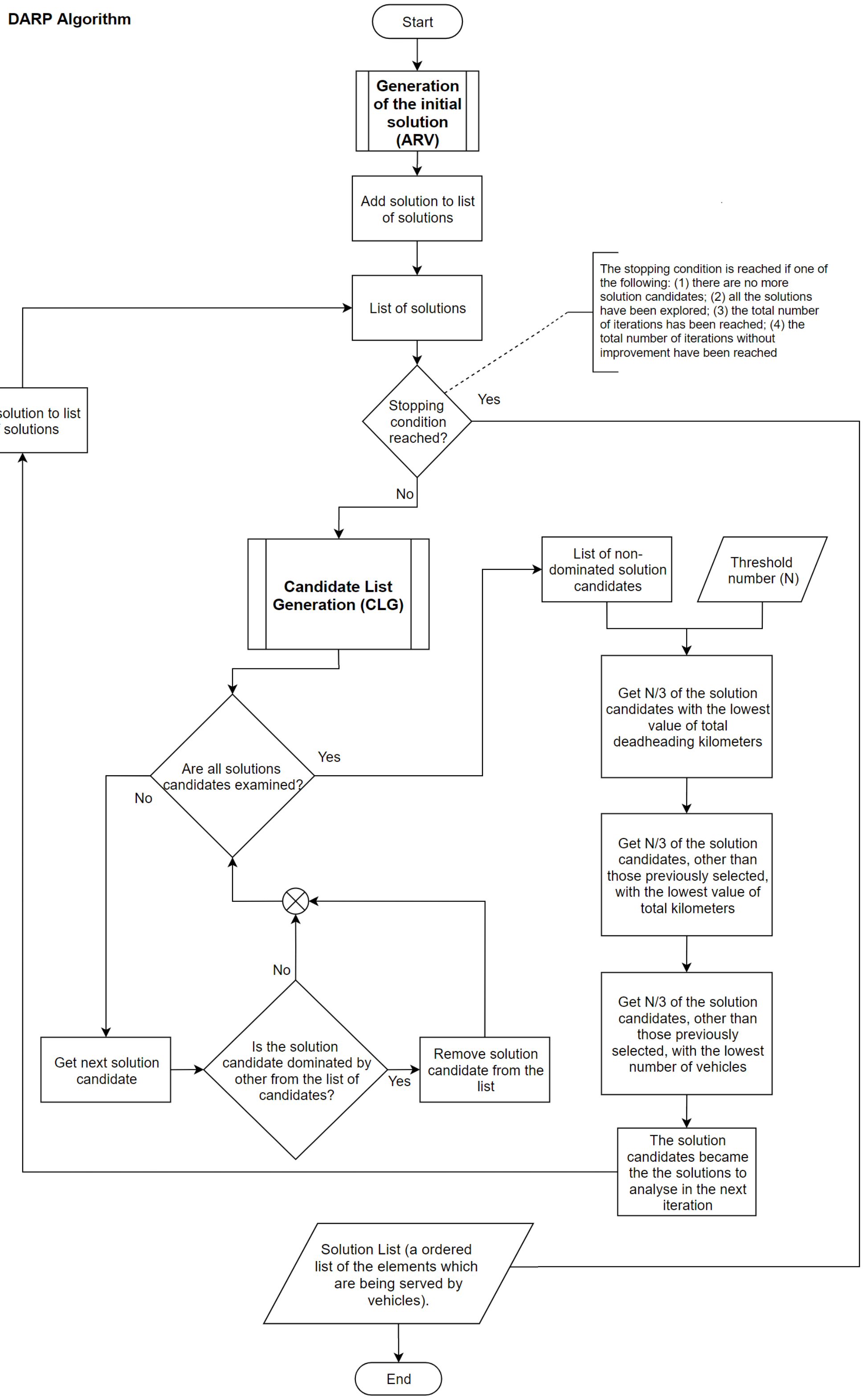


Figure 4.3: Flowchart of the DARP Algorithm.

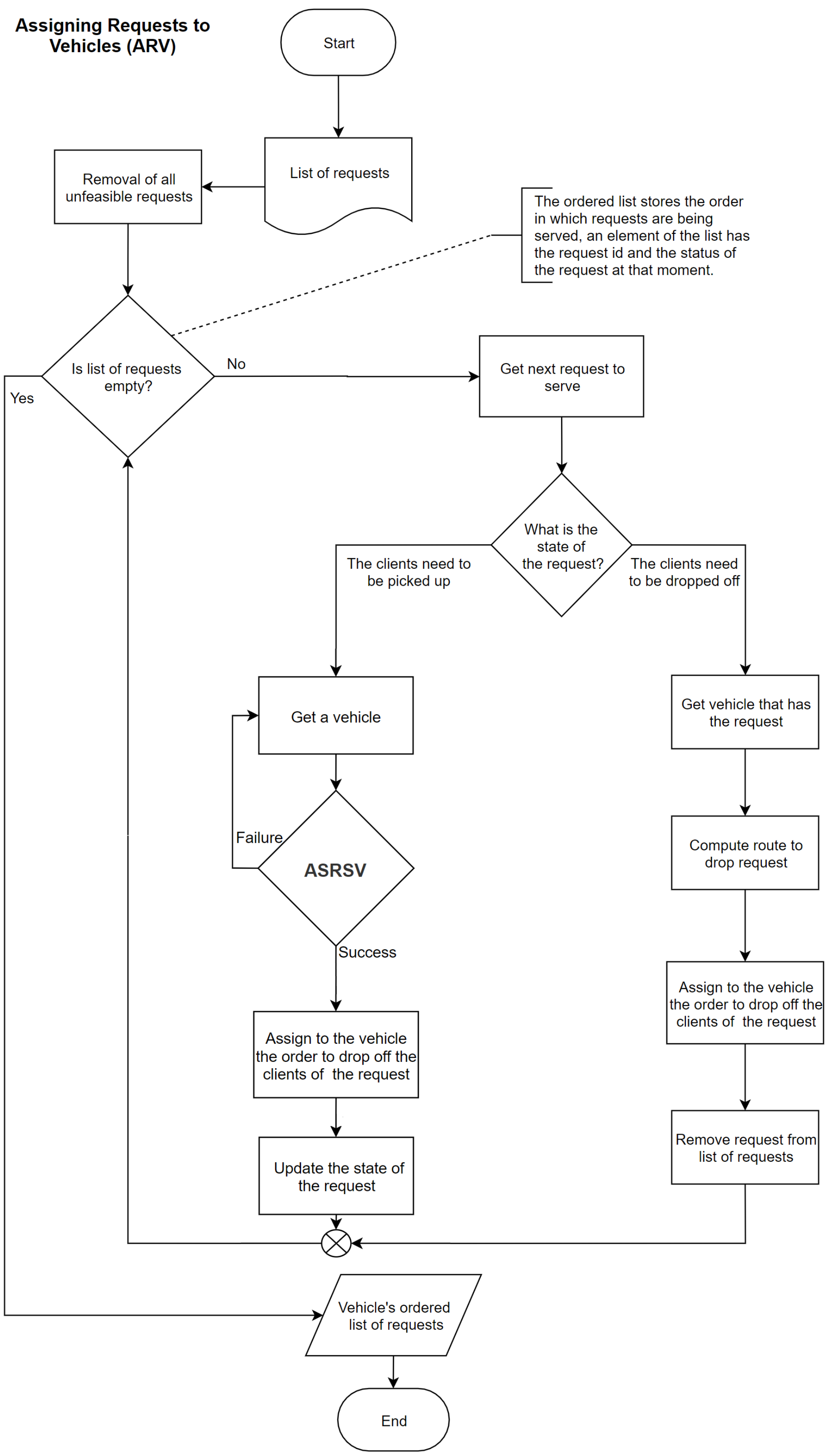


Figure 4.4: Flowchart of assigning all requests to vehicles.

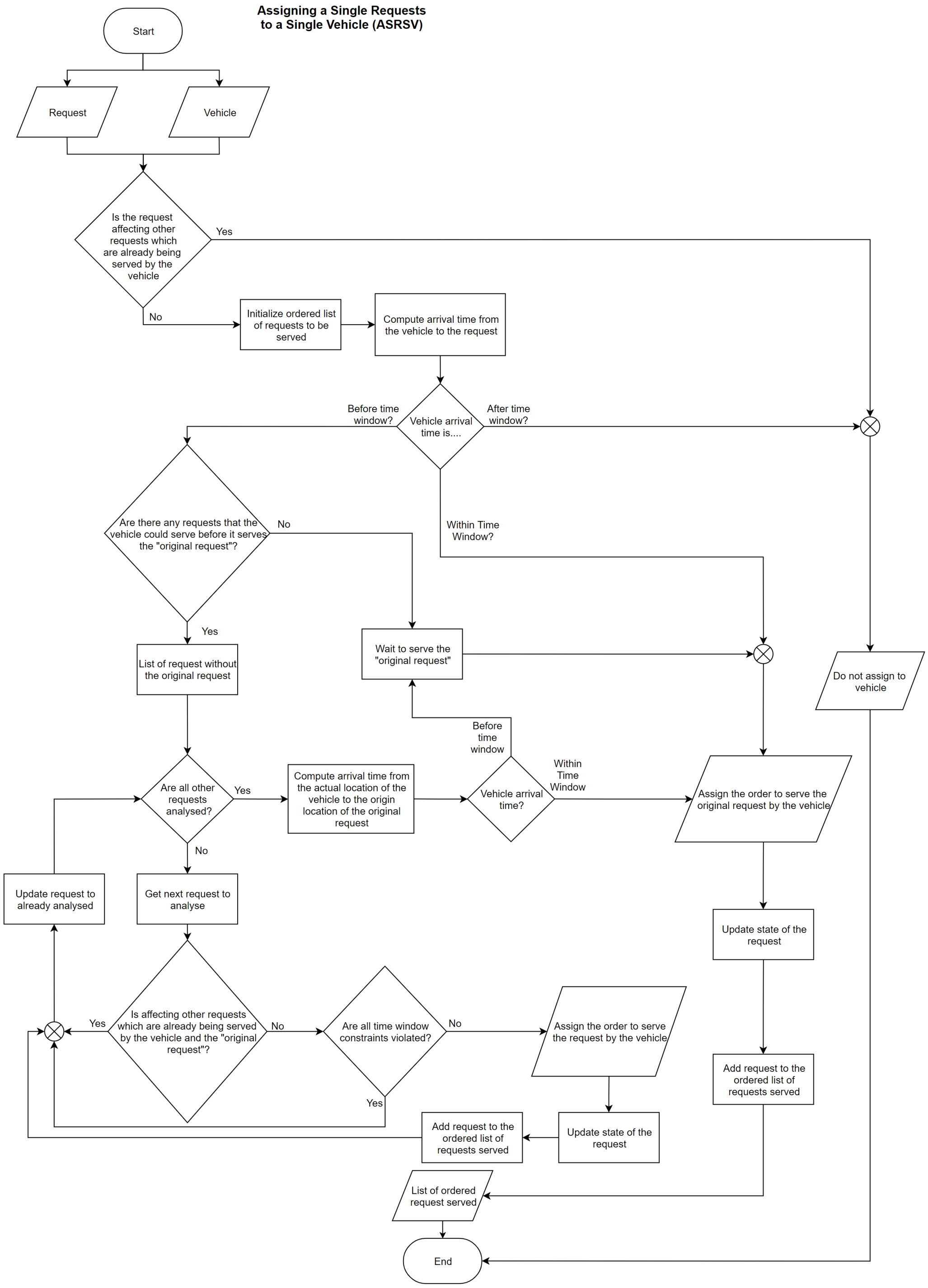


Figure 4.5: Flowchart of assigning a single request to a single vehicle.

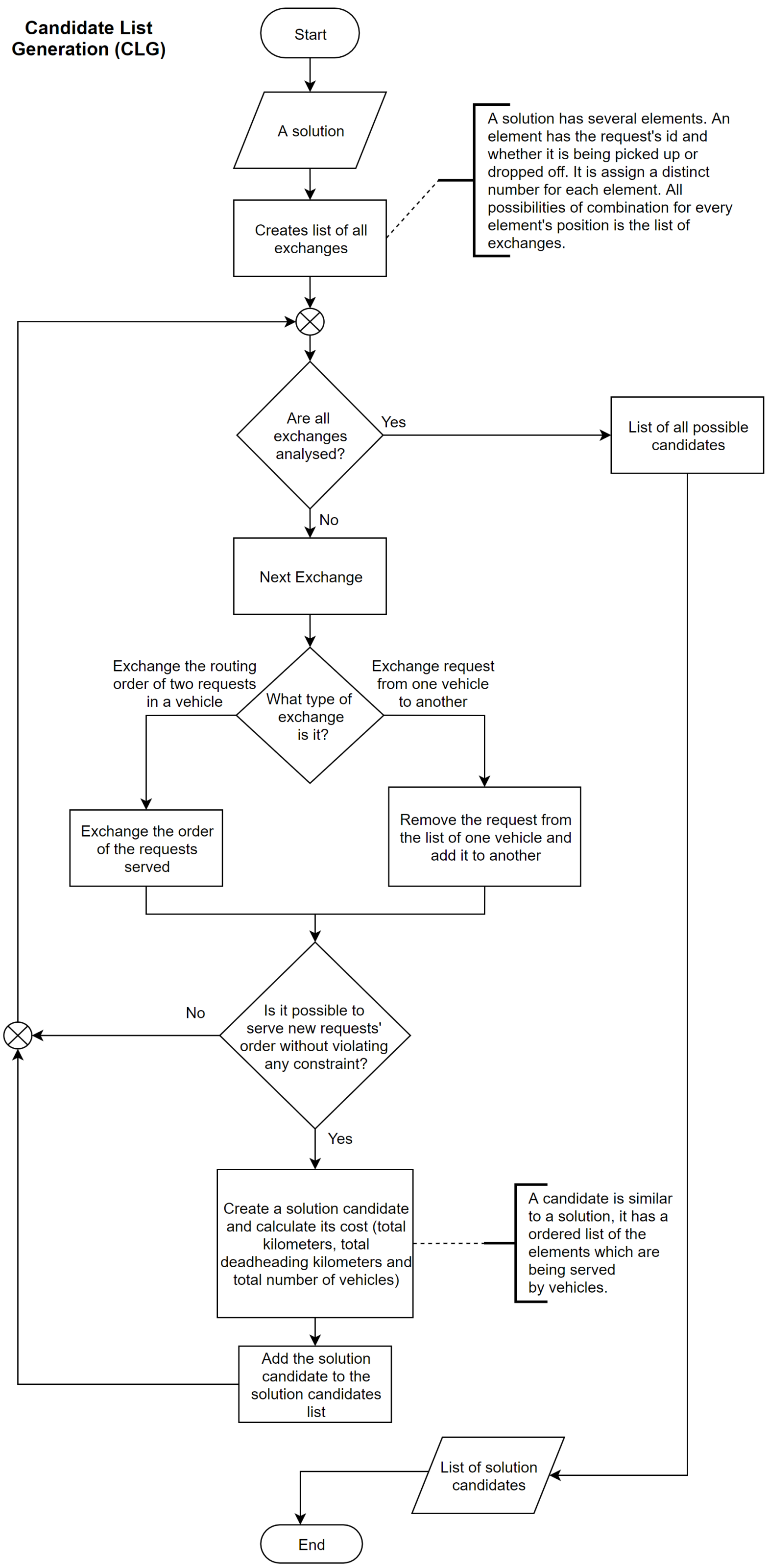


Figure 4.6: Flowchart of the generation of all feasible candidates from a solution.

Chapter 5

Results

In the first section 5.1, the results of an experimental computation performance of the DARP algorithm and an analysis of the model sensibility are performed. The case study is in section 5.2, in subsection 5.2.2, it is presented a small review of literature about the estimation of the Origin-Destination (OD) Matrices, to use in the case study. An analysis of the PTS was performed for the case study, described in subsection 5.2.1.

5.1 Computational Analysis

A set of tests were performed in order to assess the performance, sensitivity and effectiveness of the DARP algorithm. For this purpose, a combination of different values of the following parameters:

- Number of requests: [5, 10, 15, 20, 30, 50];
- Time boarding for each customer (minutes): [1, 2, 3, 4, 5];
- Number of seats in each vehicle: [4, 8, 15];
- Vehicle's speed (km/h): [20, 25, 30, 35, 40];
- Total number of iterations: [10, 30, 50];
- Candidate threshold number: [5,9,12,16].

The combination of the parameters generated 3730 different examples for the DARP algorithm to solve, evaluating the cost for each solution (triple-objective). The algorithm was implemented in Java and processed in a computer with 2.2GHz Intel Core and 16GB of RAM.

The *Spearman* correlation (r_s) was used to analyse the data from the final solutions (the variables in table 5.1). As the Kolmogorov–Smirnov test shows no evidence that the variables follow a normal distribution. Likewise, from figure 5.1 it is possible to notice that the variables do not follow a normal distribution, and a visual representation of the relationships between the variables, as they are detailed following.

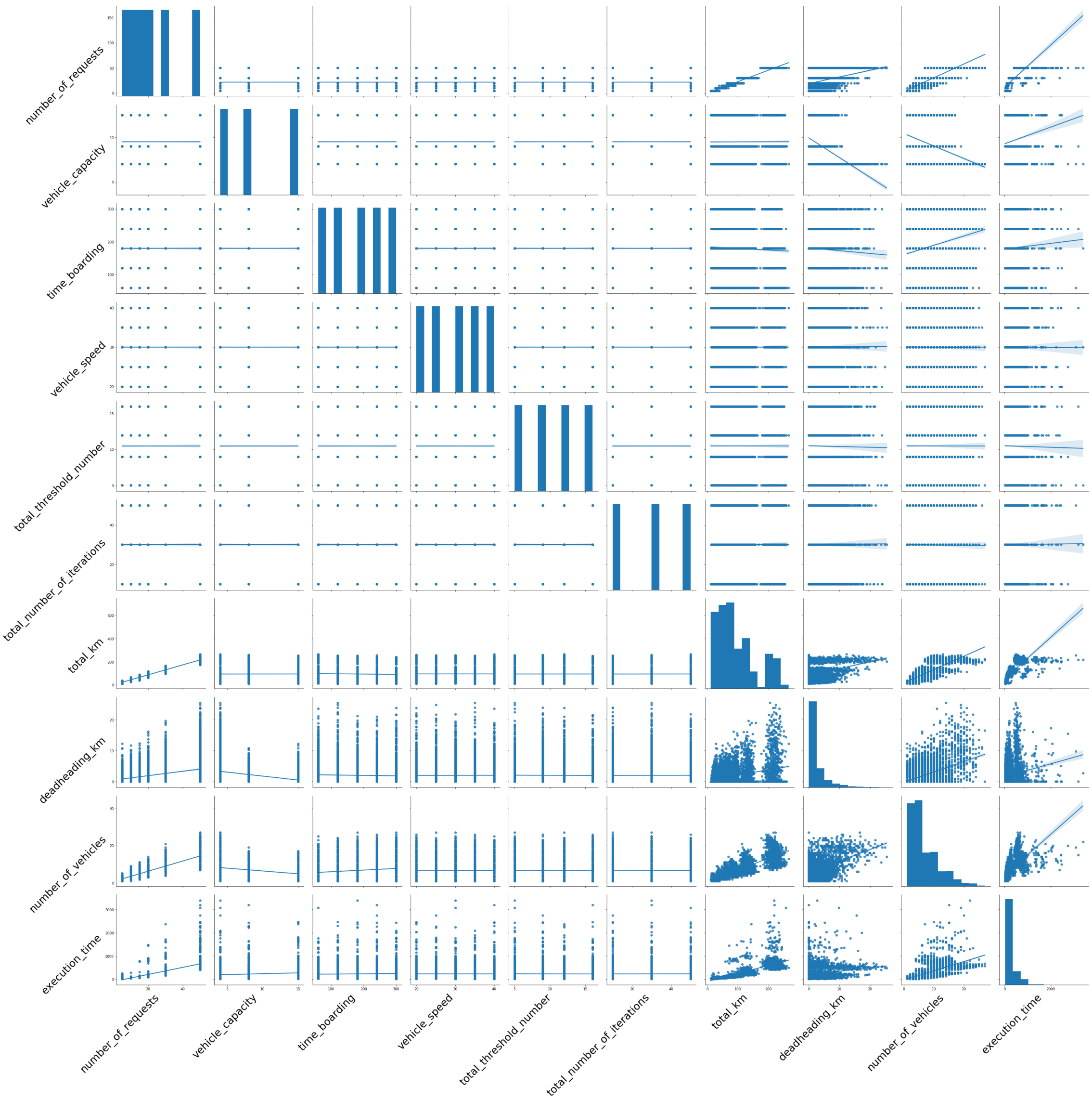


Figure 5.1: Correlation between all variables of the initial solutions.

A file was created with the output of all initial solutions and all final solutions, with the algorithm's parameters and with the initial solutions' costs and with the final solutions' costs. The costs of the initial solution are represented by 'total_km', 'deadheading_km' and 'number_of_vehicles'. The cost of the final solutions are represented by 'total_km_optimised', 'deadheading_km_optimised' and 'number_of_vehicles_optimised'.

The significance of the correlation between the variables is presented in table 5.1, the significance that are not described in the figure is higher than 0.05. A positive correlation means that both variables move tandem.

After the algorithms stopped, it was created a file with all of the algorithms' parameters, the cost of the initial solutions and the costs of the final solutions. There are strong positive relationships correlation observed between the number of requests and:

- the total travelled km (r_s=0.977, ρ<1%);
- the total travelled km optimised (r_s=0.976, ρ<1%);
- the execution time (r_s=0.954, ρ<1%);
- the execution time optimised (r_s=0.909, ρ<1%);
- the number of vehicles (r_s=0.883, ρ<1%);
- the number of vehicles optimised (r_s=0.887, ρ<1%).

Another strong positive relationship between the total travelled km and:

- the total travelled km optimised (r_s=0.989, ρ<1%);
- the execution time (r_s=0.929, ρ<1%);
- the execution time optimised (r_s=0.884, ρ<1%);
- the number of vehicles (r_s=0.843, ρ<1%);
- the number of vehicles optimised (r_s=0.850, ρ<1%).

So if the number of requests raises it will raise as well the number of the vehicle necessary to serve all requests, in the initial solution and in the optimised solutions, having similar correlations between them and the number of requests.

There is a strong positive correlation between the number of vehicles and:

- the total travelled km optimised (r_s=0.866, ρ<1%);
- the execution time (r_s=0.808, ρ<1%);
- the number of vehicles optimised (r_s=0.970, ρ<1%);
- execution time optimised (r_s=0.867, ρ<1%).

As the correlation between the number of vehicles and the number of vehicles optimised is (r_s=0.970, ρ<1%), being the correlation near one means that the initial solutions generated have almost the same number of vehicles as in the optimised solutions. Furthermore, a moderate correlation between the number of vehicles and the deadheading km (r_s=0.552, ρ<1%).

More strong correlations are observed between the execution time and:

- the total travelled km optimised (r_s=0.922, ρ<1%)
- the number of vehicles optimised (r_s=0.815, ρ<1%)
- the execution time optimised (r_s=0.850, ρ<1%)

Moreover, a strong correlation between the total travelled km optimised and the number of vehicles optimised (r_s=0.853, ρ<1%) and with the execution time optimised (r_s=0.887, ρ<1%). And a strong positive correlation between the number of vehicles optimised and the execution time optimised (r_s=0.871, ρ<1%).

There are some moderate positive correlations between deadheading km and:

- the number of vehicles (r_s=0.552, ρ<1%)
- the deadheading km optimised (r_s=0.629, ρ<1%)
- the number of vehicles optimised (r_s=0.551, ρ<1%)
- the execution time optimised (r_s=0.594, ρ<1%)

Also, there are some moderate negative correlations between the vehicle capacity and the deadheading km (r_s=-0.502, ρ<1%) and with the deadheading km optimised (r_s=-0.519, ρ<1%).

In figure 5.2 is presented the cost of the solutions at each iteration of the DARP algorithm for solving one of the tests. A colour and a shape represent each iteration. As it is represented at each iteration the each solutions' cost improve or remain the same, the latter due to not being dominated by other solutions. For a better understanding of the graph an auxiliary table A.1, in appendix, describes with detail the same optimisation, the total number iterations of iterations without improvement was reached. The stop condition was reached because there the same set of solutions remain in the non-dominated solutions list for five iterations. It was selected only the first four iterations for visual demonstration purpose. It is possible to infer that the solutions at each iteration are improving since at each iteration the solutions are closer to the origin of the axes than the initial solution, it minimises/remains the solutions' cost (in the 2d views).

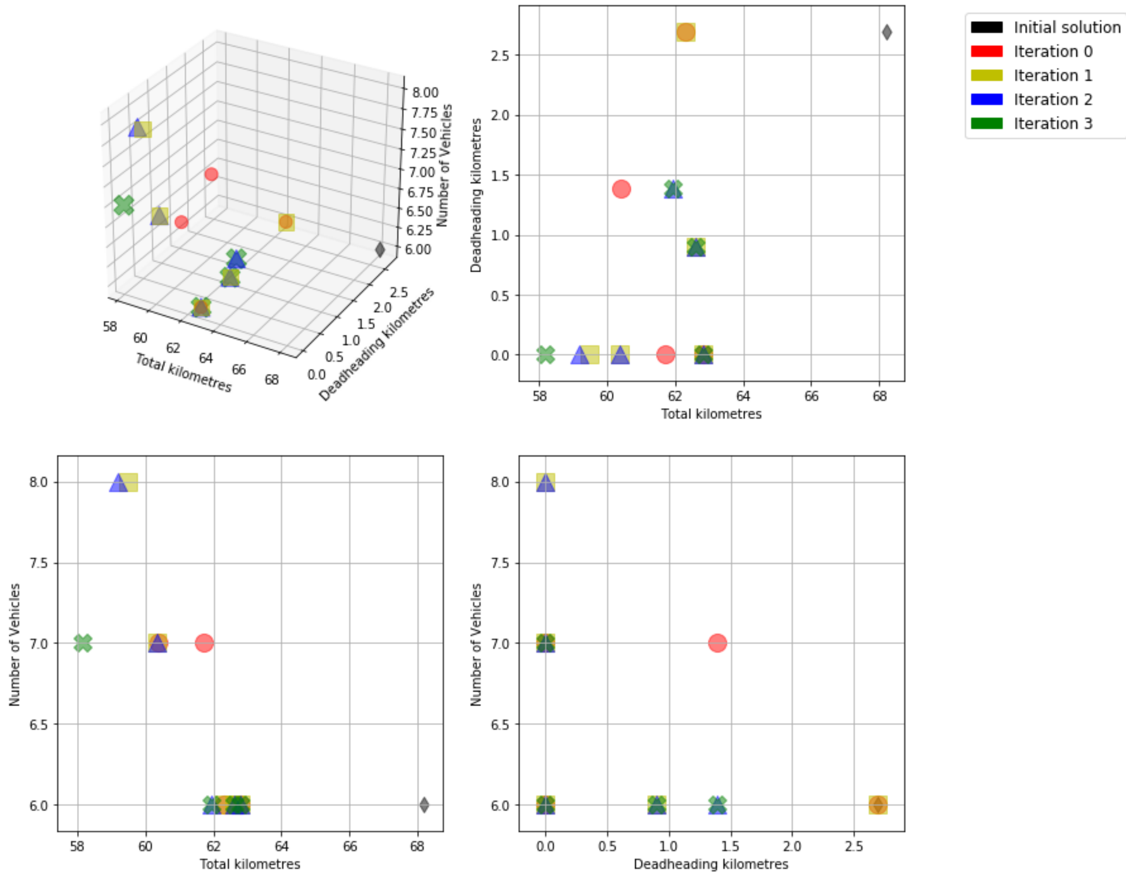


Figure 5.2: Iterations of the improvement of a DARP.

Table 5.1: Correlation of the final solutions.

	number_of_requests	vehicle_capacity	time_boarding	vehicle_speed	total_threshold_number	total_number_of_iterations	total_km	deadheading_km	number_of_vehicles	execution_time	total_km_optimised	deadheading_km_optimised	number_of_vehicles_optimised	execution_time_optimised
number_of_requests	1.000**	-0.084**	-0.034**	0.009	0.014*	0.007	0.977**	0.459**	0.883**	0.954**	0.976**	0.336**	0.887**	0.909**
vehicle_capacity	-0.084**	1.000**	0.102**	-0.005	-0.017**	-0.009	-0.073**	-0.502**	-0.417**	0.058**	-0.122**	-0.519**	-0.371**	-0.256**
time_boarding	-0.034**	0.102**	1.000**	-0.016**	-0.017**	-0.005	-0.079**	-0.079**	0.070**	-0.024**	-0.067**	-0.177**	0.069**	-0.114**
vehicle_speed	0.009	-0.005	-0.016**	1.000**	-0.004	0.000	0.013*	0.014*	0.003	0.002	0.013*	0.025**	0.002	0.006
total_threshold_number	0.014*	-0.017**	-0.017**	-0.004	1.000**	-0.017**	0.017**	0.005	0.018**	0.015*	0.017**	0.018**	0.013*	0.067**
total_number_of_iterations	0.007	-0.009	-0.005	0.000	-0.017**	1.000**	0.007	0.004	0.000	0.003	0.002	0.011	0.000	0.089**
total_km	0.977**	-0.073**	-0.079**	0.013*	0.017**	0.007	1.000**	0.462**	0.843**	0.929**	0.989**	0.332**	0.850**	0.884**
deadheading_km	0.459**	-0.502**	-0.079**	0.014*	0.005	0.004	0.462**	1.000**	0.552**	0.359**	0.481**	0.629**	0.551**	0.594**
number_of_vehicles	0.883**	-0.417**	0.070**	0.003	0.018**	0.000	0.843**	0.552**	1.000**	0.808**	0.866**	0.456**	0.970**	0.867**
execution_time	0.954**	0.058**	-0.024**	0.002	0.015*	0.003	0.929**	0.359**	0.808**	1.000**	0.922**	0.251**	0.815**	0.850**
total_km_optimised	0.976**	-0.122**	-0.067**	0.013*	0.017**	0.002	0.989**	0.481**	0.866**	0.922**	1.000**	0.365**	0.853**	0.887**
deadheading_km_optimised	0.336**	-0.519**	-0.177**	0.025**	0.018**	0.011	0.332**	0.629**	0.456**	0.251**	0.365**	1.000**	0.388**	0.476**
number_of_vehicles_optimised	0.887**	-0.371**	0.069**	0.002	0.013*	0.000	0.850**	0.551**	0.970**	0.815**	0.853**	0.388**	1.000**	0.871**
execution_time_optimised	0.909**	-0.256**	-0.114**	0.006	0.067**	0.089**	0.884**	0.594**	0.867**	0.850**	0.887**	0.476**	0.871**	1.000**

Notes: Bold values point out significant values of correlation at: * 5% level; ** 1% level.

Correlation:
- Strong ($|rs| \geq 75$ %)
- Moderate (75 % > $|r_s| \geq 50\%$)
- Low ($|r_s| < 50\%$).

5.2 Case study: transport elderly passengers in an urban context

As stated in Chapter 1, it is expected a growth of the elderly populations in the next three decades and that the elderly can have mobility limitations and thus difficulty accessing the PTS. So, in order for the elderly to have an independent and active lifestyle, it is necessary to improve the PTS. As in many other countries, DRT systems were implemented for the elderly and people with mobility impairments. Accessible public transport contributes to well-being because it is beneficial providing social interactions, reduces loneliness and the sense of belonging to a community [GJR14]. However, such systems have been mostly implemented in urban areas.

For these reasons, it was raised the necessity to perform a study of a DRT service for the elderly in an urban metropolitan area. So a case study was performed aiming to study how a DRT would perform in the PTS of the MPA for the elderly population, and what would be its operational cost. So, from the elderly passengers' validations, it was generated an OD matrix.

5.2.1 PTS usage analysis

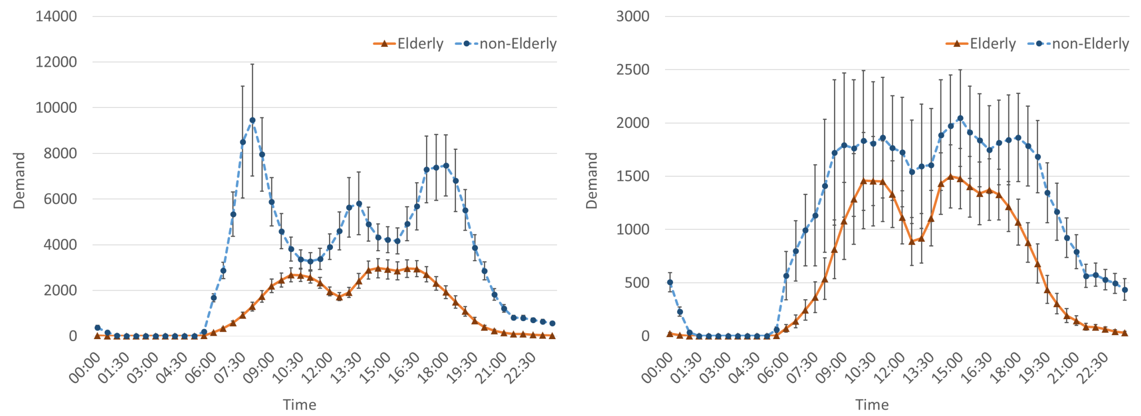
This case study will take into consideration the real MPA, where there is a PTS called *Andante*, a flexible ticket system, managed by the *Transportes Intermodais do Porto*. It was required to performed a PTS usage analysis, for that it was used the data from the validations of the PTS in MPA.

When analysing the data, two groups were identified, the elderly and the non-elderly. The non-elderly corresponds to all the population besides the elderly population. First, data analysis is performed, in order to study PTS usage on the focused population, this includes the study in zone/lines/area and schedules that the elderly population most use, only the weekdays were considered in this analysis, Only the weekdays were considered in this case study, due to the demand on the weekends decreases to half.

About 135.700 billions of trips were made along the year of 2013 by 3.017.357 travellers using the PTS, 57% of the trips were made by bus and 43% by metro. From the passengers, 75% has a monthly pass subscription, and from these, 41% are elderly, and 59% were non-elderly. The percentage of the elderly using the PTS daily is significant.

For the non-elderly travellers, three peak periods are identified, during the weekdays: a morning peak from 07:00 to 09:00, a lunch peak from 12:30 to 14:00, and an evening peak from 16:00 to 19:00. On the other hand, for the elderly, two peak periods were identified during weekdays, during the morning 09:30 - 11:30 and mid-afternoon 14:00 - 17:30. This case study will be focused on the elderly, as it is demonstrated in figure 5.4a and in figure 5.4b the usage of this provider of transport, metro, is significantly low by the elderly, corresponding to a usage of approximately 10%, figure 5.5b, figure 5.5a, and approximately 25% in the *Sociedade de Transportes Coletivos do Porto* (STCP) provider. Thus, only the STCP provider of transport will be considered in the case study.

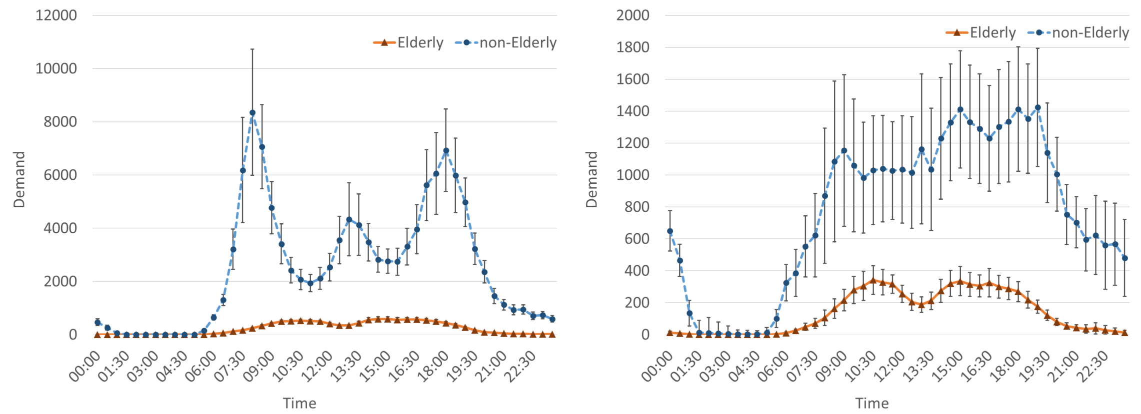
Table A.2, in appendix, shows the ten most used lines by the elderly and the usage rank of the same lines by the non-elderly. Also, it is presented the percentage of the elderly usage in the



(a) during weekdays

(b) during weekends

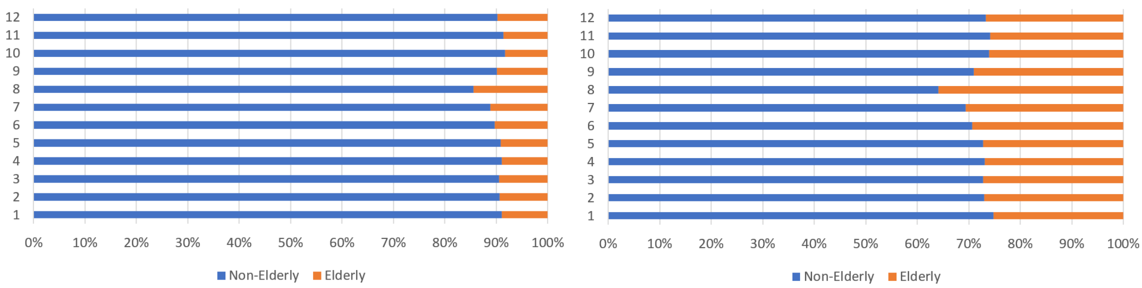
Figure 5.3: Average and standard deviation daily variation of the elderly and non-elderly traveller demand for STCP.



(a) during weekdays

(b) during weekends

Figure 5.4: Average and standard deviation daily variation of the elderly and non-elderly traveller demand for METRO do Porto.



(a) Metro provider

(b) STCP provider

Figure 5.5: Percentage of the elderly and non-elderly travellers for each provider.

entire year, per each bus line. Moreover, the percentage of the elderly and the non-elderly usage by month. The average percentage of the elderly population validations is 28.1%. The most used line by the elderly is the '600' and the most used line by the non-elderly is the '205'.

5.2.2 Estimation of the Origin-Destination Matrices

The validations of the PTS of the MPA were used in this case study to estimate the OD Matrices. As the system is in an open space, the passenger only validates at the beginning of each journey, being an Entry-only Automatic Fare Collection [NGF16]. The Automatic Fare Collection systems, in the beginning, were mainly used for revenue management. However, the data gathered contain information that can be used to improve transportation planning[BFS09].

There is no information about the alighting location, so it is necessary to estimate the OD Matrices. There are two assumptions when determining the alighting locations, described in [BFS09]:

1. most of the passengers start their next trip near or at the destination of their previous trip;
2. most of the passengers end their last trip of the day near the start of their first trip.

The methodology used was the first, because the validations are entry-only, there is only information about where the passengers start their trip, the destinations are the goal of the OD matrix estimations.

The data has information about the validations, the customer identification, the time of validation, the stop, the zone and the monthly signature title. Data collected from the 1 of January of 2013 to the 31 of December of 2013 was used. A database was used to analyse those data, and the MongoDB is the database used, with it the data is analysed separately, by months, in collections.

As the focus of this work is to study the implementation of a DRT in the MPA for the elderly, only a small percentage of the validations were used, by the following criteria:

- Elderly validations;
- Monthly subscription validations - only the passengers' validations that have a monthly subscription;
- Weekdays validations - the validations on the weekends dropped to less than half;
- Validations between 09h30 and 11h30 - due to the identified peak hour for the elderly;
- The month with more percentage of usage (August);
- Passengers' validations twice in a day, with an interval of more than 1 hour - This will allow identifying the origin and destination of a trip;.
- Passengers' with more than four validations at one stop - This would remove the customers that only used less than four times each stop with the characteristics explained above.

A matrix of the locations of the origin and the location of the destination was obtained and it is represented in the listing A.2. Nonetheless, it was necessary to retrieve the schedules of the customer' trips. From all validations it was retrieved the average time at which each customer used to validate, becoming the origin time of the request. It was assumed that every customer was willing to wait 15 minutes and that the destination time was 1 hour later than the origin time.

The previous characteristics were chosen accordingly to the focus of the work. The focus of this work is not to determine the OD matrices of the elderly, but to study how the DRT system would operate. Thus, a sample of selected validations was selected, some were discarded to reduce the map's area, figure 5.6, in order to decrease the computational time.

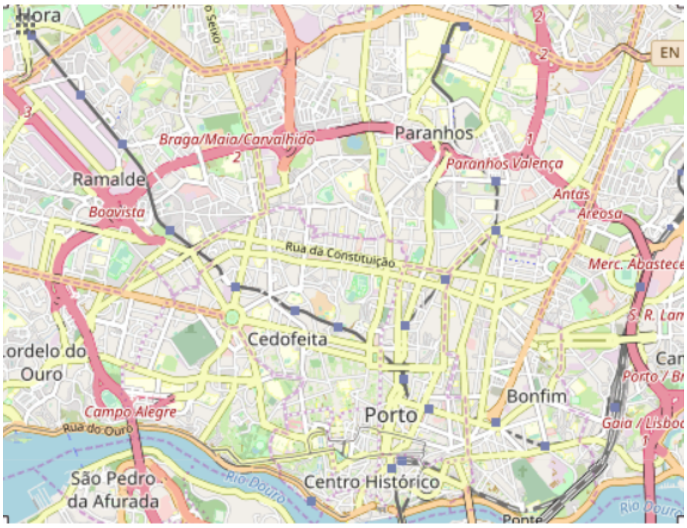


Figure 5.6: Map of the Case Study.

As the origins and destinations' matrix were complete, it was required to obtain the customers travelled map's area. For that, it was calculated the coordinates of the map having in consideration the following values southern-most latitude, western-most longitude, northern-most latitude, eastern-most longitude. With the help of a parser, the map was parsed into three files: the nodes files, the edges files and the roads files.

5.2.3 Application on the DARP algorithm to the case study

A set of tests were performed with a combination of different values for the parameters for the case of study, as mentioned as follows:

- Number of requests: 17;
- Boarding time for each customer (minutes): 2;
- Number of seats in each vehicle: [4, 8, 15];
- Vehicle's speed (km/h): 35;
- Total number of iterations: [10, 30];

– Threshold number: 12.

Table 5.2 summarises the results of the case study by using a vehicle with a capacity of 4 seats and a maximum number of 10 iterations. It generated 2675 candidate solutions, of these only 66 feasible candidate solutions were accepted, and 2609 candidate solutions were rejected. However, only 46 different solutions belonged to the non-dominated solution list, meaning that the solution list was improving at each iteration. The initial solution generated from the ARV belongs to the final non-dominated list, meaning that the ARV generated an excellent initial solution. There is an inverse relationship between the total travelled kilometres and the deadheading kilometres for the same number of vehicles.

Table 5.3 summarises the results of a similar to the previous case study, with the difference of a maximum number of iterations 30. This lead to more accepted solutions (23), 227 feasible solutions, and even more dominated candidate solutions (13035). One hundred eleven different solutions were in the non-dominated list of solutions considering this. In this case, as the number of iterations increases, the total travelled kilometres decreases for the same number of vehicles, concluding that the algorithm could find better solutions with more iterations. The same relationships observed in the previous table were also observed.

Table 5.4 summarises the results of the case study using a vehicle with a capacity of 8 seats and a maximum number of 10 iterations. Candidate solutions were generated (2483), of which only 54 feasible candidate solutions were accepted, and 2428 candidate solutions were rejected. However, only 50 different solutions belonged to the non-dominated solution list, which means the solution list was improving at each iteration. With a vehicle with more capacity allows the creation of more optimised routes for the same number of vehicles (2). In comparison with table 5.2 for the same number of iterations, but a different number of seats, 4 and 8, it was possible to obtain better dominated optimised solutions for the latter.

Table 5.5 summarises the results of the case study by using a vehicle with a capacity of eight seats and a maximum number of 30 iterations, having performed only 23 iterations. In this case, the metaheuristic stopped at 23 iterations because the total number of iterations without improvement was reached. The same has happened in table 5.3.

It generated 5662 candidate solutions of these only 108 feasible candidate solutions were accepted, and 5554 candidate solutions were rejected. However, only 68 different solutions belonged to the non-dominated solution list, meaning that the solution list was improving at each iteration. With a higher vehicle capacity, it is possible to create more optimised routes for the same solution's number of vehicles (2), it is differentiated from the previous examples. The restrictions applied to the ARV can limit the number of optimised candidate solutions generated.

In comparison with the previous table, for the same total travelled kilometres and deadheading kilometres it is possible to obtain routes with a smaller number of vehicles. Furthermore, by increasing the number of iterations, it was possible to find more and better solutions.

There are many solutions with equal triple-objective. However, the order in which the requests are served is different, due to the different neighbourhood structure found by the intensification and diversification of the metaheuristic and the elimination of candidates with the same routes that are already in the candidate list or in the solution list.

From tables 5.6 and 5.7 it is possible to observe that from a specific number of vehicles the initial and optimised solutions remain the same. So, above the capacity of eight, the cost of the solutions remains the same.

Summarising all it was observed from the tables of the case study that as the capacity of the vehicles increase there are more solutions with a lower number of vehicles in the optimised solution, as it was observed in the computational results.

Giving all the restrictions and features implemented in the algorithm ARV, there is a good initial solution, and the solution is feasible. Due to the restrictions implemented, the feasible search space is reduced, and the majority of the candidates are dominated thus removed from the list of candidates.

Table 5.2: Results of the case study with number of iterations of 10 and the vehicle capacity of 4.

Vehicle capacity		**Total number of iterations**		**Execution time**
4		10		2m46s
Initial solution	**Solution id**	**Total travelled km**	**Deadheading km**	**Number of vehicles**
	0	81.396	0	2
Optimised solutions	0	81.396	0.00	2
	62	49.344	0.00	3
	63	39.437	0.00	4
	64	43.097	4.225	3
	65	42.867	4.917	3
	66	43.961	3.275	3
	Number of iterations	**Number of generated solutions**	**Number of dominated candidates**	**Number of solutions that once were in non-dominated solutions list**
	10	66	2609	46

Table 5.3: Results of the case study with number of iterations of 30 and the vehicle capacity of 4.

Vehicle capacity	**Total number of iterations**			**Execution time**
4	30			11m15s
Initial solution	**Solution id**	**Total km**	**Deadheading km**	**Number of vehicles**
	0	81.396	0	2
Optimised solutions	0	81.396	0	2
	101	37.136	0	4
	110	37.136	0	4
	111	37.136	0	4
	112	37.136	0	4
	118	37.136	0	4
	119	37.136	0	4
	120	37.136	0	4
	127	37.136	0	4
	129	37.781	5.056	3
	134	37.781	5.056	3
	135	37.781	5.056	3
	137	42.02	0	3
	141	38.075	3.967	3
	143	37.781	5.056	3
	144	42.02	0	3
	145	42.02	0	3
	146	42.02	0	3
	150	38.075	3.967	3
	151	38.075	3.967	3
	154	42.02	0	3
	155	42.02	0	3
	156	42.02	0	3
	158	38.075	3.967	3
	159	39.052	3.275	3
	162	42.02	0	3
	166	39.052	3.275	3
	167	39.052	3.275	3
	178	39.052	3.275	3
	Number of iterations	**Number of generated solutions**	**Number of dominated candidates**	**Number of solutions that once were in non-dominated solution list**
	28	226	13035	111

Table 5.4: Results of the case study with number of iterations of 10 and the vehicle capacity of 8.

Vehicle capacity		**Total number of iterations**		**Execution time**
8		10		2m33s
Initial solution	**Solution id**	**Total km**	**Deadheading km**	**Number of vehicles**
	0	74.266	3.287	2
Optimised solutions	53	44.585	0.179	2
	54	45.443	0	2
	Number of iteration	**Number of generated solutions**	**Number of dominated candidates**	**Number of solutions that once were in non-dominated solutions list**
	10	54	2428	50

Table 5.5: Results of the case study with number of iterations of 30 and the vehicle capacity of 8.

Vehicle capacity		**Execution time**		**Execution time**
8		30		5m32s
Initial solution	**Solution id**	**Total km**	**Deadheading km**	**Number of vehicles**
	0	74.266	3.287	2
Optimised solutions	70	42.791	0.179	2
	72	42.791	0.179	2
	73	42.791	0.179	2
	74	42.791	0.179	2
	75	42.952	0	2
	76	42.791	0.179	2
	77	42.791	0.179	2
	78	42.952	0	2
	79	42.791	0.179	2
	80	42.952	0	2
	81	42.952	0	2
	82	42.791	0.179	2
	83	42.952	0	2
	84	42.952	0	2
	85	42.952	0	2
	89	42.952	0	2
	Number of iterations	**Number of generated solutions**	**Number of dominated candidates**	**Number of solutions that once were in non-dominated solutions list**
	23	107	5554	68

Table 5.6: Results of the case study with number of iterations of 10 and the vehicle capacity of 15.

Vehicle capacity		**Total number of iterations**		**Execution time**
15		10		2m40s
Initial solution	**Solution Id**	**Total travelled km**	**Deadheading km**	**Number of vehicles**
	0	74.266	3.287	2
Optimised solutions	53	44.585	0.179	2
	54	45.443	0.00	2
	Number of iterations	**Number of generated solutions**	**Number of dominated candidates**	**Number of solutions that once were in non-dominated solutions list**
	10	54	2428	50

Table 5.7: Results of the case study with number of iterations of 30 and the vehicle capacity of 15.

Vehicle capacity		**Total number of iterations**		**Execution time**
15		30		5m32s
Initial solution	**Solution id**	**Total km**	**Deadheading km**	**Number of vehicles**
	0	74.266	3.287	2
Optimised solutions	70	42.791	0.179	2
	72	42.791	0.179	2
	73	42.791	0.179	2
	74	42.791	0.179	2
	75	42.952	0	2
	76	42.791	0.179	2
	77	42.791	0.179	2
	78	42.952	0	2
	79	42.791	0.179	2
	80	42.952	0	2
	81	42.952	0	2
	82	42.791	0.179	2
	83	42.952	0	2
	84	42.952	0	2
	85	42.952	0	2
	89	42.952	0	2
	Number of iterations	**Number of generated solutions**	**Number of dominated candidates**	**Number of solutions that once were in non-dominated solutions list**
	23	107	5554	68

Chapter 6

Conclusions and Future Work

The elderly population will grow by 1.7 % in the next six decades, and half of the population will live in urban areas. This population growth is very significant since the elderly may have cognitive, sensory and physical impairments. The mobility conditions make it difficult to access the PTS. This study addressed the door-to-door service, complementing the FTS. This service does not have predefined stops or schedules, it is operated by bus, and the target is the population with limited mobility. Therefore, was studied a Capacitated Vehicle Routing Problem with Picked up and Delivery, and time window constraints, in a medium-sized metropolitan area, MPA. Each request is picked up and dropped off once and without violating the time window and capacity constraints. This algorithm was divided into two sub-algorithms, the ARV and the MOTSA, that generate an initial solution and further optimisation (resulting in a list of non-dominated solutions), respectively.

Usually, it is implemented a single objective DARP algorithm. However, in this dissertation, it was implemented a MOP because it is more similar to the real-world objective. A MOTSA was used with a triple objective criteria, using the concept of Pareto dominance, providing a set of different optimal solution.

From the computational results, it was obtained that the vehicle capacity and the deadheading km are inversely proportional, and the number of requests is directly proportional to the number of vehicles, the execution time, the total km, the number of vehicles optimised, the execution time optimised, and the total km optimised. Due to the complexity of the algorithm the execution time raises exponentially as the number of requests also raise.

A case study was performed using real-world validations from the STCP company of the MPA. With the results obtained from the case study, it is possible to conclude that the best solution for this case study is a vehicle with a capacity of eight passengers. The vehicle with the capacity to transport 15 passengers was excluded due to having the same optimised solutions as the vehicle with capacity of eight. It is possible to infer that it was generated an excellent initial solution for the vehicle with capacity of 4, due to appearing in the optimised solutions. The average optimised solutions, for the vehicle of capacity of 4, have about 80 total travelled kilometres for 2 vehicles, and about 44 total travelled kilometres for three or 4 vehicles, are dominated by an optimised solution from the solutions generated for the vehicle with a capacity of 8. So, it is possible to infer it was implemented a feasible and achievable solution by using the DARP algorithm.

For future work, it would be interesting to test this DARP algorithm for all the elderly population, as, in this dissertation, only a small part of the population was targeted. So, it could be extended this approach in future work, by performing a simulation using a software simulator and comparing, in this way, the combining action of the two public transportation system, the DRT system created from the case study and the FTS without the elderly population. Also, an aspect that can be used in the TS is the acceptance of unfeasible in order to diversify and intensify the search space.

An abstract of the work conducted in this dissertation was submitted to the 23rd EURO Working Group on Transportation EWGT 2020 that will be held in Paphos, Cyprus, on September 16 – 18, 2020. This is a conference focused on:

- Transport modelling and control;
- Transport economics and policy;
- Planning and operation;
- Innovative solutions;
- Connected and automated vehicles.

The accepted papers will be published in Transportation Research Procedia, from Elsevier. Also, the local organising committee will establish agreements with some top journals for special issues related to the works presented at the EWGT 2020 Meeting.

References

[BFFG11] Sebastian Bamberg, Satoshi Fujii, Margareta Friman, and Tommy Gärling. Behaviour theory and soft transport policy measures. *Transport Policy*, 18(1):228 – 235, 2011.

[BFS09] James J. Barry, Robert Freimer, and Howard Slavin. Use of entry-only automatic fare collection data to estimate linked transit trips in new york city. *Transportation Research Record*, 2112(1):53–61, 2009.

[BHS99] Bernd Bullnheimer, Richard F. Hartl, and Christine Strauss. Applying the ant system to the vehicle routing problem. In Stefan Voß, Silvano Martello, Ibrahim H. Osman, and Catherine Roucairol, editors, *Meta-Heuristics: Advances and Trends in Local Search Paradigms for Optimization*, pages 285–296. Springer US, Boston, MA, 1999.

[Cha03] Kim D. & Pardalos P.M. Chaovalitwongse, W. Grasp with a new local search scheme for vehicle routing problems with time windows. *Journal of Combinatorial Optimization*, 7:179–207, 2003.

[CL03] Jean-François Cordeau and Gilbert Laporte. A tabu search heuristic for the static multi-vehicle dial-a-ride problem. *Transportation Research Part B: Methodological*, 37(6):579 – 594, 2003.

[CLY14] L. Cheng, C. Liu, and B. Yan. Improved hierarchical a-star algorithm for optimal parking path planning of the large parking lot. In *2014 IEEE International Conference on Information and Automation (ICIA)*, pages 695–698, July 2014.

[Cor06] Jean-François Cordeau. A branch-and-cut algorithm for the dial-a-ride problem. *Operations Research*, 54(3):573–586, 2006.

[CW64] G. Clarke and J. W. Wright. Scheduling of vehicles from a central depot to a number of delivery points. *Operations Research*, 12(4):568–581, 1964.

[dA20] Câmara Municipal de Almada. Mini-autocarros flexibus. http://www.m-almada.pt/xportal/xmain?xpid=cmav2&xpgid=noticias_detalhe&noticia_detalhe_qry=BOUI=37509545&noticia_titulo_qry=BOUI=37509545, 2020. Accessed: 2020-01-07.

[DDGS95] Yvan Dumas, Jacques Desrosiers, Eric Gelinas, and Marius M. Solomon. An optimal algorithm for the traveling salesman problem with time windows. *Operations Research*, 43(2):367–371, 1995.

[dDPC19] PORDATA Base de Dados Portugal Contemporâneo. Densidade populacional. https://www.pordata.pt/Municipios/Densidade+populacional-452, 2019. Accessed: 2019-07-08 at 16h00.

[DDS86] Jacques Desrosiers, Yvan Dumas, and François Soumis. A dynamic programming solution of the large-scale single-vehicle dial-a-ride problem with time windows. *American Journal of Mathematical and Management Sciences*, 6(3-4):301–325, 1986.

[DDS+91] Jacques Desrosiers, Yvan Dumas, François Soumis, Serge Taillefer, and Daniel Villeneuve. An algorithm for mini-clustering in handicapped transport. *Cahiers du GERAD*, 1991.

[dE19a] Instituto Nacional de Estatistica. Local de residência (cidade, nuts - 2002) (1) meio de transporte mais utilizado nos movimentos pendulares (n.º) por local de residência (cidade, nuts - 2002) e principal meio de transporte; decenal. https://www.ine.pt/xportal/xmain?xpid=INE&xpgid=ine_indicadores&indOcorrCod=0007904&contexto=bd&selTab=tab, 2019. Accessed: 2019-06-26 21:55.

[dE19b] Instituto Nacional de Estatistica. Local de residência (cidade, nuts - 2011) (1) meio de transporte mais utilizado nos movimentos pendulares (n.º) por local de residência (cidade, nuts - 2011) e principal meio de transporte; decenal. https://www.ine.pt/xportal/xmain?xpid=INE&xpgid=ine_indicadores&indOcorrCod=0007093&contexto=bd&selTab=tab, 2019. Accessed: 2019-06-26 21:55.

[dE19c] Instituto Nacional de Estatistica. Local de residência (cidade, nuts - 2013) (1) meio de transporte mais utilizado nos movimentos pendulares (n.º) por local de residência (cidade, nuts - 2013) e principal meio de transporte; decenal. https://www.ine.pt/xportal/xmain?xpid=INE&xpgid=ine_indicadores&indOcorrCod=0008420&contexto=bd&selTab=tab2, 2019. Accessed: 2019-06-26 21:55.

[dP19] Câmara Municipal de Portalegre. Linha azul. http://www.cm-portalegre.pt/en/component/content/article/144-municipio/servicos-municipalizados/smat-transportes/diversos/726-linha-azul, 2019. Accessed: 2019-06-29 at 12h12.

[DS10] Marco Dorigo and Thomas Stützle. Ant colony optimization: Overview and recent advances. In Michel Gendreau and Jean-Yves Potvin, editors, *(eds) Handbook of Metaheuristics. International Series in Operations Research & Management Science, vol 146.*, pages 227–263. Springer US, Boston, MA, 2010.

[dTUdC19] Serviços Municipalizados de Transportes Urbanos de Coimbra. NotÍcias: 7º aniversario do "pantufinahs". http://www.smtuc.pt/revista/revista_agosto_2010.pdf, 2019. Accessed: 2019-06-29 at 14h00.

[dVdC19] Câmara Municipal de Viana do Castelo. Autocarros elétricos retomaram circuito, depois de terem transportado mais de 280 mil pessoas em nove anos. http://www.cm-viana-castelo.pt/pt/noticias/

autocarros-eletricos-retomaram-circuito-depois-de-terem-transportado-
2019. Accessed: 2019-06-29 at 12h30.

[EM09] Elizabeth H Ellis and Brian E McCollom. *Guidebook for rural demand-response transportation: measuring, assessing, and improving performance*, volume 136. Transportation Research Board, First edition, 2009.

[Eur19a] Interreg Europe. Demand-responsive transport. https://www.interregeurope.eu/fileadmin/user_upload/plp_uploads/policy_briefs/2018-06-27_Policy_Brief_Demand_Responsive_Transport.pdf, 2019. Accessed: 2019-06-07 15:00.

[Eur19b] Eurostat. People in the eu - population projections. https://ec.europa.eu/eurostat/statistics-explained/index.php/People_in_the_EU_-_population_projections, 2019. Accessed: 2019-06-08.

[Fu02] Liping Fu. A simulation model for evaluating advanced dial-a-ride paratransit systems. *Transportation Research Part A: Policy and Practice*, 36(4):291 – 307, 2002.

[GA03] M. Romanazzo G. Ambrosino, J.D. Nelson. *Demand Responsive Transport Services: Towards the Flexible Mobility Agency*. ENEA, Lungotevere Thaon di revel 76 00196 Rome (ITALY), 2003.

[GJR14] JUDITH GREEN, ALASDAIR JONES, and HELEN ROBERTS. More than a to b: the role of free bus travel for the mobility and wellbeing of older citizens in london. *Ageing and Society*, 34(3):472–494, 2014.

[H+92] John Henry Holland et al. *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. MIT press, 1992.

[HdFTP20] S.A. Horários do Funchal Transportes Públicos. Linha eco cidade. http://www.horariosdofunchal.pt/index.php?option=com_content&task=view&id=2201&Itemid=457, 2020. Accessed: 2020-01-07.

[HHL12] Carl Häll, Magdalena Högberg, and Jan Lundgren. A modeling system for simulation of dial-a-ride services. *Public Transport*, 4:17–37, 07 2012.

[HPR13] Karla L. Hoffman, Manfred Padberg, and Giovanni Rinaldi. Traveling salesman problem. In Saul I. Gass and Michael C. Fu, editors, *Encyclopedia of Operations Research and Management Science*, pages 1573–1578, Boston, MA, 2013. Springer US.

[IIdMedT19] I.P. IMT Instituto da Mobilidade e dos Transportes. Casos de boas práticas em sistemas de transportes flexíveis. http://www.imt-ip.pt/sites/IMTT/Portugues/Planeamento/Documents/AMP/boas_praticas_transportes_flexiveis.pdf, 2019. Accessed: 2019-06-25.

[JPKC08] D.M. Jaeggi, G.T. Parks, T. Kipouros, and P.J. Clarkson. The development of a multi-objective tabu search algorithm for continuous optimisation problems. *European Journal of Operational Research*, 185(3):1192 – 1212, 2008.

[JST08] Nicolas Jozefowiez, Frédéric Semet, and El-Ghazali Talbi. From single-objective to multi-objective vehicle routing problems: Motivations, case studies, and methods. In Bruce Golden and Edward Raghavan, S.and Wasil, editors, *The Vehicle Routing Problem: Latest Advances and New Challenges*, pages 445–471. Springer US, Boston, MA, 2008.

[Kac09] Janusz Kacprzyk. Studies in computational intelligence, volume 161. 2009.

[KGV83] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *Science*, 220(4598):671–680, 1983.

[KRKT87] A. W. J. Kolen, A. H. G. Rinnooy Kan, and H. W. J. M. Trienekens. Vehicle routing with time windows. *Operations Research*, 35(2):266–273, 1987.

[LD04] Quan Lu and Maged Dessouky. An exact algorithm for the multiple vehicle pickup and delivery problem. *Transportation Science*, 38(4):503–514, 2004.

[Lee18] George W. Leeson. The growth, ageing and urbanisation of our world. *Journal of Population Ageing*, 11(2):107–115, Jun 2018.

[MDSF98] Alan T. Murray, Rex Davis, Robert J. Stimson, and Luis Ferreira. Public transportation access. *Transportation Research Part D: Transport and Environment*, 3(5):319 – 328, 1998.

[MN03] Jenny Mageean and John D. Nelson. The evaluation of demand responsive transport services in Europe. *Journal of Transport Geography*, 11(4):255–270, 2003.

[MN09] Corinne Mulley and John D. Nelson. Flexible transport services: A new market opportunity for public transport. *Research in Transportation Economics*, 25(1):39 – 45, 2009. Symposium on Transport and Particular Populations.

[NGF16] A. A. Nunes, T. Galvão Dias, and J. Falcão e Cunha. Passenger journey destination estimation from automated fare collection system data using spatial validation. *IEEE Transactions on Intelligent Transportation Systems*, 17(1):133–142, Jan 2016.

[NP12] John D. Nelson and Thanawat Phonphitakchai. An evaluation of the user characteristics of an open access drt service. *Research in Transportation Economics*, 34(1):54 – 65, 2012. Gender and transport: Transaction costs, competing claims and transport policy gaps.

[NWM$^+$10] John D. Nelson, Steve Wright, Brian Masson, Giorgio Ambrosino, and Aristotelis Naniopoulos. Recent developments in flexible transport services. *Research in Transportation Economics*, 29(1):243 – 248, 2010. Reforming Public Transport throughout the World.

[PK00] D.T. Pham and D. Karaboga. *Intelligent Optimisation Techniques*. Springer, 2000.

[Pot09] Jean-Yves Potvin. A review of bio-inspired algorithms for vehicle routing. In Francisco Babtista Pereira and Jorge Tavares, editors, *Bio-inspired Algorithms for the Vehicle Routing Problem*, pages 1–34. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009.

[RCL07] Stefan Ropke, Jean-François Cordeau, and Gilbert Laporte. Models and branch-and-cut algorithms for pickup and delivery problems with time windows. *Networks*, 49(4):258–272, 7 2007.

[RDS06] Brahim Rekiek, Alain Delchambre, and Hussain Aziz Saleh. Handicapped person transportation: An application of the grouping genetic algorithm. *Engineering Applications of Artificial Intelligence*, 19(5):511 – 520, 2006.

[RR10] Mauricio G.C. Resende and Celso C. Ribeiro. Greedy randomized adaptive search procedures: Advances, hybridizations, and applications. In Michel Gendreau and Jean-Yves Potvin, editors, *Handbook of Metaheuristics*, pages 283–319. Springer US, Boston, MA, 2010.

[RSMRR18] V. Ramos-Sesma, M. Górgolas-Hernández Mora, and J.M. Ramos-Rincón. The elderly traveler. *Revista Clínica Española (English Edition)*, 218(8):426 – 434, 2018.

[TV02] Paolo Toth and Daniele Vigo. *The vehicle routing problem*. SIAM, 2002.

[Wik19a] Wikipedia. List of states and territories of the united states by population density. https://en.wikipedia.org/wiki/List_of_states_and_territories_of_the_United_States_by_population_density, 2019. Accessed: 2019-07-08 at 16h15.

[Wik19b] Wikipedia. Tyne and wear. https://en.wikipedia.org/wiki/Tyne_and_Wear, 2019. Accessed: 2019-07-08 at 16h30.

[WSY+18] R.C.P. Wong, W.Y. Szeto, Linchuan Yang, Y.C. Li, and S.C. Wong. Public transport policy measures for improving elderly mobility. *Transport Policy*, 63:73 – 79, 2018.

[WZMS13] Chao Wang, Fu Zhao, Dong Mu, and John W. Sutherland. Simulated annealing for a vehicle routing problem with simultaneous pickup-delivery and time windows. In Vittal Prabhu, Marco Taisch, and Dimitris Kiritsis, editors, *Advances in Production Management Systems. Sustainable Production and Service Supply Chains*, pages 170–177, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[YCC16] Esther H.K. Yung, Sheila Conejos, and Edwin H.W. Chan. Social needs of the elderly and active aging in public open spaces in urban renewal. *Cities*, 52:114 – 122, 2016.

REFERENCES

Appendix A

Appendix

```
client_A;3573222266;4517268372;10:32:56;11:09:56;3;00:20:00
client_B;6249303866;4378186267;09:49:30;10:51:30;1;01:00:00
client_C;4378171737;4378186279;09:43:06;10:49:06;3;01:00:00
client_D;4378171712;3573222228;10:40:00;10:54:47;1;00:30:00
client_E;6249303825;3573222266;09:33:57;10:30:57;1;00:10:00
```

Listing A.1: File with the list of requests for the ARV example.

```
client_A;128668293;25632412;08:33:34;9:33:34;1;00:15:00
client_B;122452424;25632398;09:32:23;10:32:23;1;00:15:00
client_C;122432550;26016441;08:34:29;9:34:29;1;00:15:00
client_D;25632412;126646458;10:17:28;11:17:28;1;00:15:00
client_E;111449499;90378707;11:24:29;12:24:29;1;00:15:00
client_F;111641067;25503962;08:48:31;9:48:31;1;00:15:00
client_G;25632227;90378707;08:32:26;9:32:26;1;00:15:00
client_H;138246838;126646458;10:24:19;11:24:19;1;00:15:00
client_I;25632467;26057813;09:36:25;10:36:25;1;00:15:00
client_J;26015899;122452427;08:26:26;9:26:26;1;00:15:00
client_K;111481288;129559333;11:28:23;12:28:23;1;00:15:00
client_L;25632398;122452414;08:31:36;9:31:36;1;00:15:00
client_M;25504120;25620516;09:41:24;10:41:24;1;00:15:00
client_N;128668293;26016441;08:36:29;9:36:29;1;00:15:00
client_O;122452427;112613604;09:17:31;10:17:31;1;00:15:00
client_P;111641067;122432550;10:24:23;11:24:23;1;00:15:00
client_Q;25620692;25620737;09:49:34;10:49:34;1;00:15:00
```

Listing A.2: File with the list of requests for the case study (OD matrix).

Table A.1: Iterations of the optimisation example as in figure 5.2

Iterations	Solution id	Total km	Deadheading km	Number of vehicles
Initial solution	0	68.204	2.694	6
0	1	60.395	1.389	7
	2	62.3	2.694	6
	3	61.702	0	7
	4	62.813	0	6
1	2	62.3	2.694	6
	4	62.813	0	6
	5	59.479	0	8
	6	62.6	0.899	6
	7	60.359	0	7
2	4	62.813	0	6
	6	62.6	0.899	6
	7	60.359	0	7
	9	61.924	1.389	6
	10	59.174	0	8
3	4	62.813	0	6
	6	62.6	0.899	6
	9	61.924	1.389	6
	14	58.164	0	7

Table A.2: Ranking of buses lines used by elderly (E) and non-elderly (nE) travellers.

Line (bus)			**’600’**	**’701’**	**’305’**	**’205’**	**’200’**	**’204’**	**’800’**	**’903’**	**’704’**	**’207’**
Elderly %			30.81	27.47	31.13	20.06	26.7	26.03	25.29	22.13	25.84	28.39
Elderly Ranking and Percentage usage	**Jan**	25.2%	1	2	3	4	5	6	7	8	9	10
	Feb	27.0%	1	2	3	4	5	6	11	7	8	10
	Mar	27.3%	1	3	2	4	5	6	11	7	8	9
	Apr	26.9%	1	2	3	4	5	6	8	10	9	11
	May	27.2%	1	2	3	4	5	6	7	10	9	11
	Jun	29.3%	1	2	3	5	4	6	10	9	8	11
	Jul	30.6%	1	3	6	2	4	7	11	10	8	9
	Aug	35.9%	1	7	5	2	4	6	10	9	8	11
	Sep	29.0%	1	2	4	3	5	6	11	9	8	10
	Oct	26.0%	1	2	3	5	6	4	11	8	7	9
	Nov	25.8%	1	2	3	5	6	4	11	8	7	9
	Dec	26.7%	1	2	3	4	8	5	10	7	6	9
Non-elderly Ranking and Percentage usage	**Jan**	74.8%	3	2	9	1	6	7	5	4	8	11
	Feb	73.0%	2	3	9	1	6	5	7	4	8	11
	Mar	72.7%	3	4	9	1	5	6	8	2	7	10
	Apr	73.1%	3	4	9	1	5	6	7	2	8	10
	May	72.8%	3	2	9	1	6	5	7	4	8	10
	Jun	70.7%	3	4	9	1	5	6	8	2	7	11
	Jul	69.4%	3	4	8	1	5	6	8	2	7	11
	Aug	64.1%	3	6	9	1	4	7	10	2	5	17
	Sep	71.0%	3	4	9	1	6	5	8	2	7	10
	Oct	74.0%	4	5	8	1	6	3	9	2	7	10
	Nov	74.2%	4	3	8	1	6	5	9	2	7	10
	Dec	73.3%	4	3	8	1	7	5	9	2	6	10