Traffic Analysis, Prediction and Optimization using Machine Learning Algorithms

Pedro Miguel Terras Lopes

Thesis to obtain the Master of Science Degree in

Telecommunications and Computer Science Engineering

Supervisor: Prof. Fernando Henrique Côrte-Real Mira da Silva

Examination Committee

Chairperson: Prof. Ricardo Jorge Fernandes Chaves
Supervisor: Prof. Fernando Henrique Côrte-Real Mira da Silva
Member of the Committee: Prof. Rui Miguel Carrasqueiro Henriques

October 2021

Declaration

I declare that this document is an original work of my own authorship and that it fulfills all the requirements of the Code of Conduct and Good Practices of the Universidade de Lisboa.

Acknowledgments

Em primeiro lugar, agradecer ao Professor Fernando Mira da Silva pela paciência, inputs valiosos e pela ajuda ao longo desta tese.

Em segundo, agradecer ao meu colega Rui Loureiro, por me enquadrar na sua tese, providenciar todos os dados necessários e se disponibilizar sempre que necessário para me tirar dúvidas.

Por fim, deixar os agradecimentos mais especiais de todos. Obrigado Pai, Mãe e também a ti Ricardo. Foram os vossos sacrifícios, educação e ensinamentos que me fizeram chegar até aqui. Profundamente agradecido por desde cedo me mostrarem o mundo e tal como ele é. Obrigado por nunca me terem cortado as asas.

Sem vocês nada disto era possível, este trabalho também é vosso!

Abstract

A balanced public transport system is essential for environmental, social and economic sustainability and is a prerequisite for ensuring the quality of life of a city's inhabitants. In such manner, it is crucial to compose a traffic network capable of match passengers' expectations. However different and conflicting interests have to be addressed. Passengers want to travel as fast as possible, waiting as little as necessary and making as few transfers as possible. On the contrary, operators seek to minimize costs in order to achieve higher profits.

Determining customised routes and their frequency adjusted to demand can highly affect the costs and efficiency of the urban transportation system. A change in transit frequency can influence not only route capacity but also waiting time. Variation in waiting time will affect the passengers' route choice strategy.

By saying that, this approach comprises a difficult dilemma dependent of several external factors, for instance governmental policies or available technology solutions. In addition, it is worth to mention as well, the particular complexity of public transportation planning process, that includes route design, setting frequencies or fleet allocation.

Traditionally, transit operators relied on manual surveys to gather transit activity details, although they revealed themselves as expensive and inefficient. Currently, as technology evolves, traffic companies make use of new solutions, namely Automatically Data Collection (ADC) systems, which provide a detailed information about transit and passengers' needs.

In this thesis, we extend previous work of other authors, namely [1] that were focused on origin, destination and interchange inference and transit network design. In this context, our work tackles simultaneously the two first stages of the overall operational planning process for public transportation networks, network design and frequency setting.

Moreover, it will be studied different metrics regarding public transportation activity, such as waiting and travel time, transfers, but also bus empty seats that constitute an unproductive cost to transit companies.

It will be considered transit data from the city of Lisbon to assess the efficacy of the computed methods.

Keywords: Transit Network Design, Frequency Setting, Genetic Algorithm

Resumo

Um sistema equilibrado de transportes públicos é essencial para a sustentabilidade ambiental, social e económica e é um pré-requisito para garantir a qualidade de vida dos habitantes de uma cidade. Desta forma, é crucial compor uma rede de tráfego capaz de corresponder às expectativas dos passageiros. No entanto, existem diversos interesses que entram em conflito. Os passageiros querem viajar o mais rápido possível, esperando o mínimo necessário e fazendo o menor número possível de transferências. Pelo contrário, os operadores procuram minimizar os custos a fim de obter lucros mais elevados.

O desenho de itinerários e respetiva frequência ajustados à procura pode afetar altamente os custos e a eficiência do sistema de transporte urbano. Por exemplo, uma mudança na frequência de trânsito pode influenciar não só a capacidade das rotas, mas também o tempo de espera. A variação do tempo de espera afeta diretamente a escolha da rota dos passageiros.

Ao dizer isto, esta abordagem compreende um dilema difícil dependente de vários fatores externos, por exemplo, políticas governamentais ou soluções tecnológicas disponíveis. Além disso, vale a pena mencionar também a complexidade particular do processo de planeamento do transporte público, que inclui a construção de rotas, a definição de frequências ou a atribuição de frotas.

Tradicionalmente, os operadores de trânsito baseavam-se em inquéritos manuais para recolher detalhes da atividade de trânsito, embora se revelassem dispendiosos e ineficientes. Atualmente, à medida que a tecnologia evolui, as empresas de trânsito utilizam novas soluções, nomeadamente sistemas de Recolha Automática de Dados (ADC), que fornecem uma informação detalhada sobre o trânsito e as necessidades dos passageiros.

Nesta tese, estendemos o trabalho anterior de outros autores, nomeadamente [1], que se centravam na origem, destino e inferência de intercâmbio e desenho de redes de trânsito. Neste contexto, o nosso trabalho aborda simultaneamente as duas primeiras fases do processo global de planeamento operacional de redes de transporte público, concepção de redes e definição de frequências.

Além disso, serão estudadas diferentes métricas relativas à actividade dos transportes públicos, especialmente tempo de espera e de viagem, número de transferências, mas também lugares vazios de autocarro que constituem um custo improdutivo para as empresas de trânsito.

Serão considerados dados de trânsito da cidade de Lisboa para avaliar a eficácia dos métodos concebidos.

Palavras-Chave: Algoritmo Genético, Configuração de Frequência, Desenho de Rotas de Trânsito

Contents

List of Tables

List of Figures

Glossary

ABC Artificial Bee Colony.

ACO Ant Colony Optimization.

ADC Automatic Data Collection.

ADCS Automatic Data Collection System.

AFC Automatic Fare Collection.

APC Automatic Passenger Count.

AVL Automatic Vehicle Location.

CP Comboios de Portugal.

FS Frequency Setting.

GA Genetic Algorithm.

GTFS General Transit Feed Specification.

IST Instituto Superior Técnico.

ML Machine Learning.

OD Origin Destination.

ODX Origin Destination Interchange.

OSM Open Street Map.

RFID Radio Frequency Identification.

TND Transit Network Design.

TNDFSP Transit Network Design and Frequency Setting Problem.

Chapter 1

Introduction

In the 21st century, urban mobility has become one of the key issues to metropolitan areas. The continuous rural exodus and the increased preoccupation about the environmental issues has conducted many different governmental policies, mainly focused on the sustainable growth of urban areas.

It is safe to say that public transportation is considered a crucial backbone of it, due to the fact it constitutes a reliable mean of transportation at an affordable fare in high density population zones, whilst emitting low carbon emissions.

Public transportation operators face a wide range of challenges related to its transit operation, such as unpredictability of events or extreme weather conditions. At the same time, these companies often struggle to strike the right balance between its level of service at a competitive cost and the user's expectations, who greedily demand reduced waiting times at stations, accuracy in bus or train arriving time, but also a modern fleet with sufficient capacity.

Reducing users' costs usually leads to increase in operator costs, and vice versa. However, determining the most efficient operation for both users and operators, for given levels of service and cost requirements, is essential for the success of such systems [10].

Even though operating and monitoring a complex bus network is an arduous task, the current use of electronic ticketing solutions enabled transit agencies to gather large data sets of system usage that can be a powerful tool in order to optimise network routes, its frequency and all the planning process.

Frequency setting is one of the the main activities in the tactical planning of public transport operations. Allocating frequencies of bus services in a city network is a multi-criteria problem that typically considers the operational costs, the passenger demand coverage and the service reliability [11].

This thesis will address Transit Network Design (TND) Problem and Frequency Setting (FS) Problem together. It will be applied Automatic Data Collection System (ADCS) data divided by different parts of the day during a week in order to assess the appropriateness of the given route set.

The objective function contains several metrics regarding passenger and operator costs. The latter includes empty seats. As far as passenger costs is concerned, waiting and in-vehicle time, transfers and unsatisfied demand will be measured.

The data sources treated in this thesis, were kindly provided by Carris and Metropolitano de Lisboa, both transit operators running in Lisbon area.

1.1 Objectives

This thesis proposes to address both Transit Network Design Problem (TNDP) and Frequency Setting Problem (FSP) simultaneously by developing a robust Genetic Algorithm that receives real transit data as input.

Our work is built upon the work of [1] that developed origin, destination and interchange (ODX) inference methodologies, essential to guarantee a good feasibility of input data. Furthermore, he has also computed a precise representation of the road network and a generation of routes method.

In this dissertation, we computed methods capable of calculating in-vehicle and waiting time but also trip transfers, that can be considered as costs to passengers when travelling on public transports. Moreover, we computed empty seats calculation, which is considered a cost to transit operators. Metrics impact on final result will be studied along.

This thesis represents a relevant contribution to public transportation planning because optimizes FSP and TNDP as a whole, and takes into account demand variation during the day. In relation to frequency optimization, we developed two ways of adjust it during Genetic Algorithm progression, Local Search and Random Search, that can enhance further approaches to Frequency Setting Problem.

The developed methodology used and is applied to the bus network of the city of Lisbon as a case study.

More concretely, this thesis intends to fulfill the following objectives:

- **Solve transactions without inferred destinations problem**

  [1] had an obstacle in his destination inference method, which led to a low number of transactions with deduced destinations. For this reason, we propose a solution to overcome data drawbacks.

- **Divide day by demand fluctuation**

  As we know by our experience as passenger, buses appearances as well as demand vary along the day. For instance, there is a significant variation between peak hours and night ones. So, in this work we divide demand (input OD matrix) in 4 parts given transactions timestamp, assuming that there is a change in resources allocation. A possible optimization solution will have a route set and 4 headways associated with each route.

- **Analyse different optimization search methods**

  In our work, there will be two different approaches regarding frequency adjustment: Local search and Random Search. The former contemplates small interval changes on values during algorithm development, the latter completely randomizes them.

- **Study metrics impact on service quality**

  Our objective function relates passenger costs, such as in-vehicle and waiting time, with operator costs, namely empty seats, that are given different weights. We will study metrics impact on final solution by varying those metrics, and check results.

- **Apply Route and Frequency Assignment Optimization**

  Improve previously developed GA-based optimisation method for the TNDP to include FSP as well. The computed methodology is applied to the entire bus network in Lisbon. This represents the main objective of the work.

1.2 Structure of this document

The methods utilized in this thesis can be grouped into three different categories: ODX inference processes, and optimization methods used in the Transit Network Design problem and Frequency Setting problem.

Chapter 2 provides the fundamental context for the problem as well as methods developed in this thesis, along with the required data sources. Likewise, Chapter 3 illustrates the most noteworthy literature regarding ODX inference, Transit Network Design alone, but also together with Frequency Setting.

Chapter 4 describes the methodology used by [1] for ODX inference. Chapter 5 depicts problem formulation, metrics calculated along with Genetic Algortihm method used to the Transit Network Design with Frequency Setting problem applied to Lisbon.

Chapter 6 describes the results of the computed optimization methods.

In the end of the document, Chapter 7 states some conclusions and presents future research developments to the subject.

Chapter 2

Background

This chapter starts by explaining some aspects regarding public traffic optimization, such as network design and frequency setting. After that, it is specified public transportation planning stages and some algorithms that tackle differently optimization process, mainly genetic algorithm. Finally data sources and its formats are pointed out.

2.1 Traffic Optimization

The aim of transit operators is to ensure efficient public transportation systems that are user and environment friendly, while keeping low operating costs.

Reducing the transportation cost, which is directly affected by the total travelled distances and consumed fuel costs, can be achieved by an optimum route planning by removing the wastes from the supply chain. Notwithstanding, a reasonable level of service and a commitment to public interest must be preserved. As a result of this, here are some scheduling resource and service-related constraints to take into consideration [12] [13]:

- **The total fleet size**: The total number of buses available for plying on the network is generally limited.
- **Minimum fleet size**: If a route is to be operated, at least a minimum number of buses should be assigned to that route.
- **Limited bus capacity** each bus has a finite capacity.
- **Minimum stopping time**: Buses must stop for at least a minimum stipulated time, at a bus stop.
- **Maximum stopping time**: Buses cannot stop for more than a maximum stipulated time at a bus stop as this causes discomfort to the on-board passengers.
- **Policy headway** There must exist a minimum service on any route. The time gap between buses on any route cannot be allowed to become more than a pre-defined policy headway for that route.
- **Maximum transfer time** no passenger should have to wait too long for a transfer.

Optimizing a public transport system is not trivial, as it depends on several aspects, such as governmental policies, available technology or budget constraints.

2.2 Transit Network Design

The aim of the transit route network design is to determine a route configuration that achieves a certain desired objective, subject to the constraints previously highlighted. As noted by [14], network design elements are part of the overall operational planning process for public transportation networks. The process includes four steps:

- design of routes
- setting frequencies
- developing timetables
- scheduling buses and drivers

The first stage is optimization of bus routes based on the demand matrix. The next stage involves proper determination of bus frequencies on each route with respect to the demand matrix. Scheduling optimal fleet to the routes based on the predetermined timetables in second stage, budget limits and location of the depots will be considered in this stage. In the final stage, fleet crew and their roster table will be assigned [15].

A good or efficient route set is one that satisfies the following: [13]

- The **entire transit demand is served**, that is, the percentage of **unsatisfied** demand is **zero**.
- A large percentage of transit demand is served through **direct connections**, that is, percentage of demand satisfied with zero transfers is high.
- The average **travel time** per transit user is as **low** as possible.
- **Operation cost** is as **low** as possible.

2.3 Frequency Setting

The Frequency Setting problem determines the number of trips for a given route to provide the best level of service as possible in a planning period. Including service level, cost and fleet size restrictions. Since frequency changes affect directly route waiting and travel time, so this parameter impacts the passengers' perception of the level of service which may lead to an increment or decrement of the system usage.

Bus transport planners have to consider the bus network demand, fleet capacity and route characteristics in determining bus service frequencies. At the same time, the bus operators have to provide the bus service operation and management based on the cost and benefit analysis of their investments.

The service frequency refers to the number of times a specific service is offered during a given time interval. Normally, frequency is expressed per hour. As with everything else public transportation planning, frequency of service (existing or planned) changes throughout the day. Based on this, bus arrivals during rush hours are fairly higher than on off-pick moments, such as night hours.

The **time between two vehicles** offering a service, which conveys exactly the same information as frequency, is known as the **headway**. The service headway is the **inverse** of the **frequency**.

$$Headway = \frac{1}{Frequency} \tag{2.1}$$

The passenger's arrival follows the **uniform distribution**, so the **average waiting time is assumed to be equal to half of the headway of a route**.

$$WaitingTime = \frac{Headway}{2} \tag{2.2}$$

We define a **trip** as a set of **intermediate journeys**, depending whether there are interchanges thorough trip progression. In the case of multiple routing options, the waiting time is estimated by the average of the sum of the half of the headways of all possible routes.

$$\text{Journey Waiting Time} = \frac{\frac{\text{Alternative 1 Headway}}{2} + ... + \frac{\text{Alternative n Headway}}{2}}{n} \tag{2.3}$$

For the trips that include transfers, the waiting time is calculated as the sum of all the intermediate journeys headways that concern the trip.

$$\text{Trip Waiting Time} = \sum \text{Journey Waiting Time} \tag{2.4}$$

Especially for high frequency services, passengers' perception of reliability are largely determined by the predictability of their trip times, rather than adherence to a published schedule. Passengers are unlikely to consult a schedule before starting their journeys when using high frequency services [2]. In opposition, for infrequent service, passengers are likely to rely on the schedule and arrive at the stop shortly before the bus is scheduled to depart [16].

Frequency on each route should lie within a range. For instance, high frequency on a route increases crowding or congestion in the route [17]. On the other side, low frequency leads to rise in waiting time and travel time [18].

2.4 Transit Route Network Design with Frequency Setting

The Transit Network Design and Frequency Setting Problem (TNDFSP) is a multi-constrained combinatorial optimization problem with a vast search space for which the evaluation of candidate solutions route sets can be both challenging and time consuming, with many potential solutions rejected due to infeasibility [19].

The TNDFS Problem addresses the conflicting interests of passengers and transit agency. The key objectives of the problem are to **minimize** the **passenger time** and the **operating cost**. For improving the service level, it is essential to reduce the in-vehicle travel time by providing a shortest path between each pair of stops, to reduce the waiting time by increasing the frequency of the vehicles and mitigate the number of transfers by considering more direct trips for passengers. For operators, it is economical to reduce the fleet cost and carbon tax by restraining the fleet size and imposing a penalty for emissions, respectively. There are two objective functions in this problem: the first objective function represents the passengers time, and the second objective function comprises of the operating cost. Passenger time in the first objective varies with respect to multiple parameters of waiting time and transfers [20].

Generally, with respect to service frequency, a TNDFSP is solved in two ways [21]:

- **Sequentially**. In the sequential procedure, initially, optimization of candidate route sets is carried out followed by the passenger assignment and frequency setting procedure. In this procedure, searching for the optimum set of routes is carried out without considering service frequencies. However, because the total travel time is unknown in the first process, the optimally depends on ratios of direct and indirect passenger travel demand.

- **Simultaneously**. In this method, frequencies are set simultaneously with the optimization of candidate route sets. Thus, the fitness of the solution would not only depend on ratios but also on the total travel time, fleet size, and service frequencies.

In the first phase, the transit network design (TNDP) part consists of finding a set of transit routes that together forms an efficient transit network, serving users' travel demand represented by an origin–destination matrix with minimum total travel time and transfers, which include penalties for transfers and trips which contain more than 2 transfers (unsatisfied demand). Transit routes take into consideration existing transportation network available and predefined stop points. During the second phase, assigning frequencies (FSP) to transit routes establishes waiting times and total capacity levels for routes, and makes possible the calculation of the total necessary fleet for network operation and also vehicles utilization.

Higher frequencies directly affect operator costs, once they directly influence the total fleet required; on the other hand, it also contributes to a better quality public transportation system, as passengers will wait less for their transport. These conflicting objectives of user and operator costs make the TNDFSP problem multi-objective in nature [10].

2.5 Public Transportation Planning

The planning problem of designing public transport networks is highly complex. The main goal of most transit agencies is to offer to the population a service of good quality that allows passengers to travel easily at a reasonable fare. However, they are subject to budgetary and management restrictions (number of buses or drivers, as well as bus capacity) that make difficult meet costumers' expectations.

By saying that, this organisation procedure treats a wide range of settlements, from longstanding ones to real-time choices. There are four distinct stages: strategic, tactical, operational and real-time.

At the **strategic** level, transit planning is concerned not only with the design of transit routes and networks, but also with types of vehicle and stop spacing. This involves long-term decisions focused on designing a network of routes to meet passenger demand.

Operational-level planning aims at constructing vehicle and crew schedules that minimize total costs. This includes to vehicle scheduling, driver rostering, maintenance planning, as well as parking and dispatching [22].

Tactical planning is typically performed on an annual or biannual basis. The aim is continually to address the tradeoff between service quality and system costs, e.g., by reflecting changes in demand or the available budget [23].

Even when a solution for the planning process is given, operation of transport systems is affected by uncertainty of travel times. Extreme weather, accidents or passenger demand at bus stops may cause such constraints. To address these situations, **real-time control** strategies are implemented to guarantee an efficient service during the operation of the system. For instance, that may be holding vehicles in a station or even skipping specific ones [24].

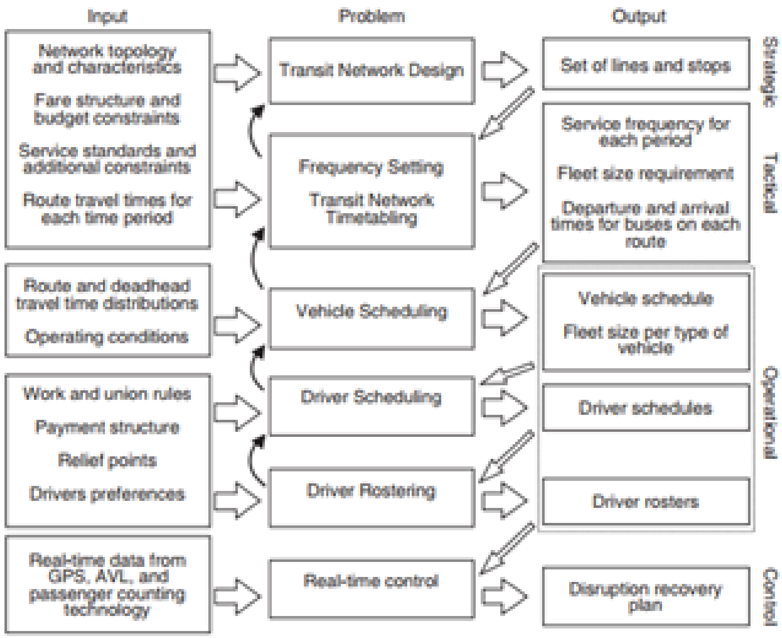

Figure 2.1: Interaction between stages of the planning process and real-time control strategies reproduced from [2]

The determination of frequencies and vehicle capacities is thus a crucial service design decision when planning public transport services. These decisions are considered both at the strategic and tactical levels. At the strategic level, frequency setting interacts with passengers' route choices (OD matrix) and the designated line capacity has consequences for the choice of public transport technology. At the tactical level, both service frequencies and vehicle capacity (e.g., number of train cars, ordinary or articulated bus) can be altered on a seasonal basis and vary by time of day and day of the week. Service unreliability can severely affect line capacity by reducing the effective frequency [25].

2.6 Optimization Algorithms

Since the main objective of this thesis is to optimize a traffic network by making use of machine learning (ML) algorithms, we will present a few techniques used in the literature to solve aforementioned problems with reference to optimal network.

Optimization technique is an approach to find the best solution to complex problems by minimizing or maximizing one or more objective functions stand on one or more decision variables that give a value of the objective function [3]. There are different ways to tackle optimization problems.

Optimization problems can be divided into two major categories:

- **exact** algorithms, which give, as name suggests, precise solutions
- **approximate** algorithms, which may give or not exact solutions

Approximate algorithm further divided into two major categories as **heuristic** and **meta-heuristics** algorithms.

Heuristic refers to experience-based techniques for problem solving, learning, and discovery [26]. The main objective of heuristics approach is to increase the efficiency of well-known methods and separate not-fit conditions from fit ones by decreased risk exposure, the error rate for prediction, and also increased competitive advantages [27].

A **meta-heuristic** algorithm is formally defined as an iterative generation process which guides a subordinate heuristic by combining intelligently different concepts for exploring and exploiting the search spaces using learning strategies to structure information in order to find efficiently near-optimal solutions [28]. Basically, a meta-heuristic is a top-level strategy that guides an underlying heuristic solving a given problem.

The most attractive feature of a meta- heuristic is that its application requires no special knowledge on the optimization problem to be solved, hence it can be used to define the concept of general problem-solving model for optimization problems or other related problems [29]. Genetic Algorithms and Ant Colony Algorithm belong to this category.

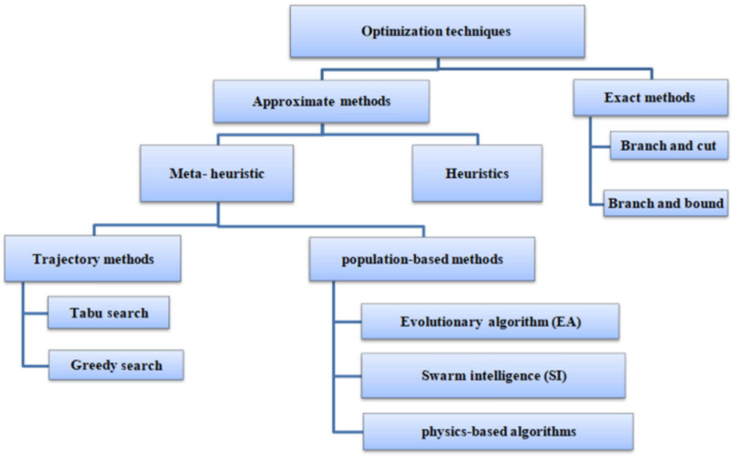

Figure 2.2: Classification of optimization techniques from [3]

2.6.1 Reinforcement Learning

Reinforcement learning is a method of machine learning where the learning is achieved based on the discovery of the ambient model and the optimal way to behave in it.

In this learning style, the algorithms try to predict the output for a problem based on a set of tuning parameters. Then, the calculated output becomes an input parameter and new output is calculated until the optimal output is found.

The term reinforcement indicates the process of forming and strengthening of these associations by the reinforcer, which encompasses both rewards (positive reinforcers) and punishments (negative reinforcers).

In order to produce intelligent programs (also called agents), reinforcement learning goes through the following steps [30]:

1. Input state is observed by the agent.

2. Decision making function is used to make the agent perform an action.

3. After the action is performed, the agent receives reward or reinforcement from the environment.

4. The state-action pair information about the reward is stored.

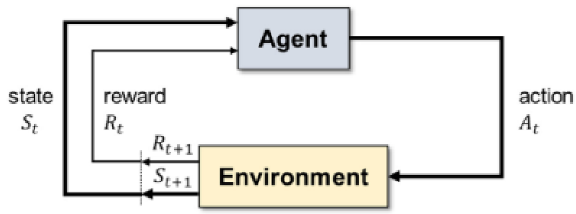

Figure 2.3: Basic setup for a reinforcement learning configuration from [4]

2.6.2 Ant Colony Algorithm

Ant colony optimization algorithm (ACO), which belongs to swarm intelligence methods, takes inspiration from the foraging behaviour of some ant species. These ants deposit pheromone on the ground in order to mark some favourable path that should be followed by other members of the colony. Ant colony optimization exploits a similar mechanism for solving optimization problems [31].

Ants living in colonies initially send the leader ants alone to find food. These leaders seek around to find food sources. If the leaders find food, they leave a special smell, pheromone, on their way back to the colony. Each ant leaves a constant amount of pheromone per unit of time, forming a pheromone trail. Thanks to this trace, other ants can reach the food source. Ants are capable of finding the shortest route from food sources to their nests without using their sight [32].

If the ant chooses to follow the pheromone trail it will leave its own pheromone, strengthening the trail. The stronger the trail, in other words, the path which has higher amounts of pheromone, the more appealing it is for ants so the higher the probability of other ants choosing that path. Eventually, the less optimal paths become less and less appealing due to pheromone evaporation.

ACO optimizes a problem by having an updated pheromone trail and moving these ants around in the search space according to simple mathematical formula over the transition probability and total pheromone in the region.

At each iteration, ACO generates global ants and calculates their fitness. Update pheromone and edge of weak regions. If fitness is improved, then move local ants to better regions, otherwise select new random search direction. Update ant pheromone and evaporate ant pheromone. The continuous ACO is based on both local and global search [3].

As illustrated by the next figure, when facing an obstacle, there is an equal probability for every ant to choose each of the tracks. As the left trail is shorter than the left one and so required less travel time, it will end up with higher level of pheromone. More the ants will take the right path, higher the pheromone trail is. This fact will be increased by the evaporation stage.

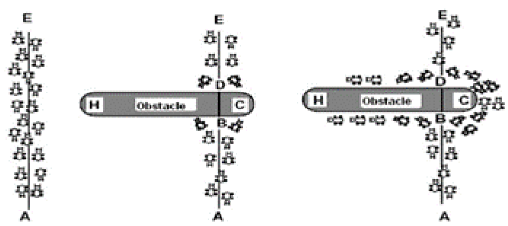

Figure 2.4: Ants finding the shortest path around an obstacle from [5]

2.6.3 Genetic Algorithm

Genetic algorithm (GA) is a search heuristic that mimics the process of natural selection and genetics. It belongs to a group of evolutionary algorithms. This procedure combines a Darwinian survival-of-the-fittest with a randomized, yet structured information exchange among a population of artificial chromosomes. The main attractions of the GA approach involve simplicity of procedures, global perspective and inherent parallelisation [33].

This algorithm was introduced by Professor John Holland in 1975 and since then has been applied in many research areas.

These algorithms encode a potential solution to a specific problem on a simple chromosome-like data structure, thereafter, apply recombination operators to these structures in such a way as to preserve critical information [34].

GA begins with an initial population of strings (chromosomes), created randomly. Each chromosome is formed by a set of variables (genes), encoded in binary values. In fact, each string represents all the problem variables to the solution context according user criteria. The creation of strings in the initial population of GA is as simple as tossing an unbiased coin. The successive coin flips (head=l, tail=0) can be used to decide genes (bits) in a string [35].

After initial population of possible solutions obtained, it is necessary to determine how fit a gene is. This method, named fitness function, is the most important feature of GA, because it provides comparable scores among chromosomes.

In general, a fitness function F(x) is first derived from the objective function and used in successive genetic operators. For maximization problems, the fitness function can be considered to be the same as the objective function i.e.,

$$F(x) = f(x) \tag{2.5}$$

However, for minimization problems, the goal is to find a solution having the minimum objective function value. Thus, the fitness can be calculated as the reciprocal of the objective function value so that solutions with smaller objective function value get larger fitness [33] [35].

$$F(x) = \frac{1}{[1 + f(x)]} \tag{2.6}$$

As stated above, GA simulates procedures that actually happen in nature. Improvements, from one generation to the next, are achieved using a set of three operators, **reproduction/selection**, **crossover** and **mutation**. Each one plays a crucial role and serves a well-defined objective.

Reproduction/selection operator is the first applied on a population. Basically, chooses the best chromosomes, in other words, the ones with higher score in fitness function, and copies them to the upcoming generation. This is how we guarantee to strings (individuals) with high scores to be selected more often than those with low scores, which may not be selected at all [36] .

The **crossover** operator is the most significant phase in GA, it combines two chromosomes to originate new offspring (children) to the next generation. Crossover is performed by randomly choose a bit which separates head and tail of the string. Eventually, the tail of the two parents are swapped to get new children.

In certain new offspring formed, some of their genes can be subjected to a **mutation** with a low random probability. This implies that some of the bits in the chromosome can be flipped. Mutation operator occurs to maintain diversity and avoid premature convergence.

In the end, after the completion of the previous procedures, there is an evaluation of the new chromosomes. If the termination criteria are not met, the population is again operated by above three operators and evaluated.

In practice, unlike simple neighbourhood search methods that terminate when a local optimum is reached GAs could run forever from a generation to the other. So it is essential to strike a termination criterion. Usually, this can be a predetermined threshold to the fitness function, a limit on the number of generations (iterations) or a given duration. When the population is converged, the GA is concluded [37].

From an optimization point of view, one of the main advantages of GAs is that they do not need gradient information of the optimization function. Evaluation of the objective function is the only requirement. Also GAs can handle nonlinear problems defined on a discrete, continuous, or mix search spaces, constrained or unconstrained [38].

Below, there is a genetic algorithm template reproduced from [37]. Crossover and mutation are performed under user-specified conditions.

Algorithm 1 Genetic Algorithm Template

```
Choose an initial population of chromosomes.
while termination condition is not satisfied do
    repeat
        if crossover condition satisfied then
            Select parent chromosomes
            Choose crossover parameters
            Perform crossover
        end if
        if mutation condition satisfied then
            Select chromosome(s) for mutation
            Choose mutation points
            Perform mutation
        end if
        Evaluate the fitness of offsprings
    until sufficient offspring created
    Select the new population
end while
```

2.7 Origin, Destination and Interchange Inference ODX

Due to the fact that Carris is an open AFC system, as passengers are just required to tap its ticket on boarding, it is fundamental to infer origin and destinations.

The aim of OD inference is to process data from ADC systems, and for each stage, obtain an origin (place and time) and a destination (place and time). It is assumed that, for the trips that contain transfers, the origin is the boarding stop of the first leg of the trip and the destination is the alighting stop of the last leg of the trip.

For a trip that begins at a faregate, as it happens in Metro Lisboa, the origin of the trip is known undoubtedly. For those beginning at fareboxes on-board vehicles, the timestamp of the farebox transaction is matched to the vehicle's location, as recorded by the AVL (Automatic Vehicle Location) system, to determine boarding stops [24]. Inference rates for origins are quite high (about 95%), as concluded by [22] in a study conducted in London. The main barrier to origin inference method are occasional failures and imprecisions of the AVL system [16] [39].

Identifying the alighting bus stations is more challenging, because of the absence of available alighting location information. Destinations are assumed to be the stop closest to his next bus boarding. Regarding that, a few assumptions are made [40] [22] [23] [24]:

- Passengers will not walk a long distance to board at a rail/bus station different from the one where they previously alighted.
- There is no private transportation mode trip segment between consecutive transit trip segments in a daily trip sequence.
- Last trip of a day ends at the origin of the first trip of the day (symmetry assumption)

However, there are some limitations to this method, that should be considered. The most relevant ones are [16]:

- When there is only one transaction on the card that day
- When a passenger is traveling in the opposite direction from the origin of the next trip
- When the current route does not pass within a reasonable distance of the next origin point.

In [22], destination inference procedure indicated 57.5% to 78.5% of destinations deduced using data from routes in the London bus network.

In addition, interchanges (transfers) are inferred between stages to reveal passengers' linked transit journeys, including those that span multiple modes. An interchange (or transfer) is defined as a transition between two consecutive journey stages that does not contain a trip-generating activity [24].

The interchange time can be calculated more accurately as the difference between the subsequent trip's boarding time and the previous trip's alighting time.

Reference [41] assumed that if the interchange time is longer than 30 minutes, then it is not considered a transfer.

2.8 Data Formats and Tools

This thesis will be developed using a wide range of data structures, such as GTFS (General Transit Feed Specification), Automatic Fare Collection (AFC), Automatic Passenger Count (APC), and Automatic Vehicle Location (AVL).

OpenStreetMap (OSM) will be used to design the optimized network, analyse traffic aspects, compare current and shortest paths between stations on the same route and compute travel durations.

2.8.1 GTFS

General Transit Feed Specification (GTFS) defines a common format used for transit operators to publish public transportation schedules and related geographic information. This format was created in 2005 by a Google Engineer, Chris Harrelson, with the aim of incorporating transit data into Google Maps.

GTFS comprises two different components:

- **Static**, that contains schedule, fare and geographic information
- **Real-time**, that presents predictions, service alerts and vehicle positions

A GTFS dataset is described by a collection of at least 6, and up to 13 CSV (comma separated values) files in .txt format. Files agency.txt, calendar.txt, routes.txt, stops.txt, stop_times.txt and trips.txt are considered mandatory to a valid GTFS data feed [42].

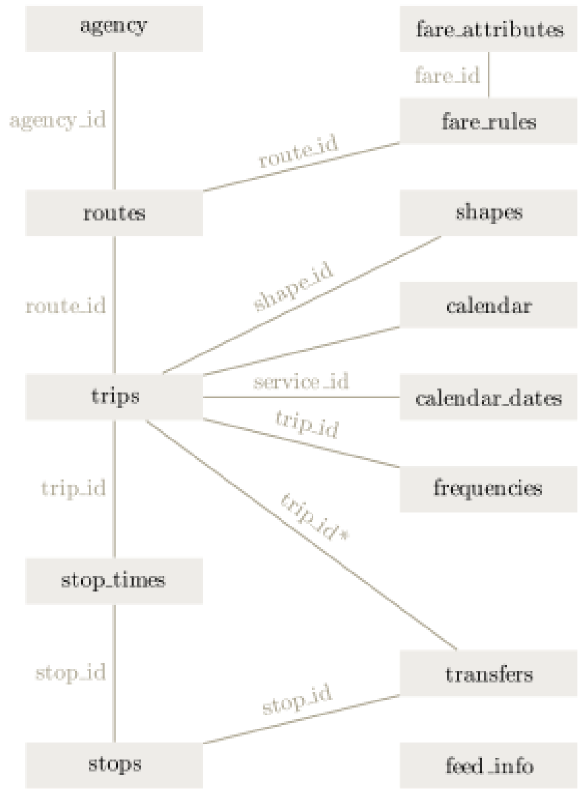

Figure 2.5: GTFS Class diagram from https://transportist.org/2019/03/12/gtfs-but-for/

The main advantage of using GTFS is to access the detailed schedule (stop_times.txt) of each trip ID [43].

Today, many operators publish publicly their routes and schedules in GTFS format.

2.8.2 Automatic Data Collection

Traditionally, transit operators resorted to manual surveys to gain an understanding of ridership profiles. Moreover, these surveys have also been used by travel demand modelers to develop trip tables. However, manual surveys can be considered costly, in small sample sizes and mostly inefficient [40].

In contrast, the last decades have seen the development of new technologies aimed at improving the information available for planning and operating public transport systems. Automatic Data Collection Systems are able to develop more passenger centric performance metrics.

The accuracy of ADC data can be determined, avoiding survey errors difficult to identify and bias that can appear in traditional methods [40]. Furthermore, agencies are able to collect more data at significantly lower marginal cost [44].

Operators have three sources of automatically collected data obtainable:

The **AFC (Automatic Fare Collection)** system includes several functionalities to monitor and control operations concerning issuing, sale and validation of transport tickets. In Lisbon, AFC system is denominated VIVA and integrates all the transit operators in Lisbon area.

This system records the fare related information whenever a passenger pays for a trip. It can be at a ticket vending machine, in a fare gate or, in case of Carris, in a validator on-vehicle. Its information comprehends card identifier and fare category (standard, student, senior), timestamp and location.

In the matter of AFC systems, they can be classified as open, hybrid or closed. While the Carris bus network is considered an open system, as the passenger only needs to tap in, in the case of Metro network, it is mandatory to tap at the entrance and exit of the stations. In respect of a hybrid example, in Lisbon area also, which has station from both categories, there is Comboios de Portugal, state-owned company which operates passenger trains.

Typically, open systems have been the main research interest in the development of a traveler's trip chain because the closed system provides both origin and destination (O-D) information of a trip [43].

The **APC (Automatic Passenger Count)** includes technology devices, such as sensors or CCTV cameras which accurately log boardings and alightings. Each record indicates timestamp, card ID and trip information.

AVL (Automatic Vehicle Location) system periodically indicates real-time vehicle localisation. This is used to inform passengers bus arrival time estimation. AVL data notifies about date, time, speed, trip information and of course, GPS coordintates.

2.8.3 OpenStreetMap

OpenStreetMap is an open collaborative project to create a free editable map of the world. Volunteers gather location data using GPS, local knowledge, and other free sources of information and upload it.

2.9 Data Sources

2.9.1 Carris

Carris is the main public transportation operator at the surface directed by Lisbon's City Hall. Its fleet is comprised by 706 buses from different sizes and fuel types, 48 trams, 3 funiculars and even an elevator (Santa Justa lift). As far as human resources are concerned, Carris employs nearly 2450 collaborators.

During the previous year (2019), a total of 139.5 million of passengers travelled with Carris [45].

Figure 2.6: Carris Network

2.9.2 Metro

Metropolitano de Lisboa is the public transit operator responsible for the metro system of Lisbon. The system includes 56 stations and 4 lines (Green, Yellow, Red and Blue) summing up 44.5km of extension. Regarding its capacity and dimension, Metropolitano de Lisboa assumes 1452 employees and 333 train carriages.

Figure 2.7: Lisbon Metro Network

Throughout 2019, 173 million of passengers used Lisbon’s underground system [46].

Chapter 3

Related Work

In the literature, there are several ways to manage transit route network design and frequency setting problem. Some works solved network design only, others approached the problem as a whole. From origin, destination and interchanges inference, network design process, but also involving frequency assignment , references tackled these issues differently.

3.1 Origin, destination and interchange inference

In the first place, based on demand pattern, it is required to analyse AFC logs to deduce trip beginnings and targets. Reference [47] proposes a methodology that estimates the OD matrix by using smart card and GPS data. It is described the characteristics of the smart card data and bus GPS data (AVL) which are matched to identify origins. Besides, alighting stations are inferred based on the characteristics already mentioned on the inference subsection, which were able to identify over 80% of the alighting stations on average, in a study conducted in Suzhou, China.

In [48], by including only those AFC transactions that occur within five minutes after an AVL event, inferred the origins of approximately 90% of all bus boardings. It is worth to mention as well, study [23] managed New York that indicated 90% of accuracy of inferred destinations. It was presumed that a rider's initial origin of the day is a close approximation of his final daily destination (daily symmetry rule) and that the most likely place for a passenger to alight a bus is the stop closest to his next bus boarding (closest stop rule).

According to literature, interchanges can be determined by a wide range of methods, such as application of a temporal threshold between the two stage's origin times, the evaluation of distances or inter-stage walk speeds.

Reference [49], with the aim of identifying transfers in the London network, stated some recommended fixed time thresholds:

- **Underground-to-bus transfers** from 15 to 25 minutes between Underground station exit and bus boarding
- **Bus-to-Underground transfers**, including bus in-vehicle time, from 30 to 50 minutes between bus boarding and Underground station entry
- **Bus-to-bus transfers**, including bus in-vehicle time and waiting time for the second bus stage, from 40 to 60 minutes from first bus boarding to second bus boarding

However, [50] set dynamic interchange thresholds. They assumed that transfer limit time has to be calculated considering the transfer access distance between the alighting and the boarding stop. The transfer access time is estimated by using an average walk speed of 1.2 m/s (or 4320 m/h) and the straight line distance between the stops.

$$\text{interchange time threshold} = \frac{\text{interchange distance}}{4,320} + 5 \tag{3.1}$$

It is added 5 minutes to the threshold due to the fact that some cardholders do not have perfect knowledge of the network and timetable, but also to prevent unreasonably short time thresholds when the two stops are extremely close together.

3.2 Transit Route Network Design using GA

Transit route network design problem (TRNDP) is an optimization problem where objectives are defined, its constraints are determined, and a methodology is selected and validated for obtaining an optimal solution [51]. Transit network design problem consists of a set of two subproblems, namely, the transit routing problem and the transit scheduling problem. Transit routing involves the development of bus routes on an existing road network with associated link travel times. Transit scheduling involves scheduling transit operations on each route. Both aspects should optimally satisfy some user-defined objectives [13].

To solve any "real-world" problem using GAs, it must first be mapped into a form on which the GA can operate. The most common coding method is to transform the variables to a binary string or vector.

On the one hand, in reference [6], for instance, a solution, or a route set, is represented as a string of nodes with demarcation lines indicating the end of a route and the start of another route.

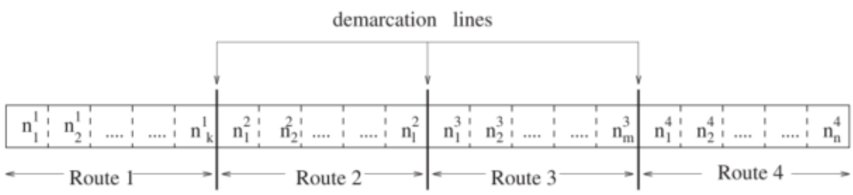

Figure 3.1: String representation of a route set from [6]

On the other hand, in [7], each gene (bit) on the chromosome represents which node is assigned to which route. This chromosome is defined as an array of length N, containing the assigned route number for every node *i*.

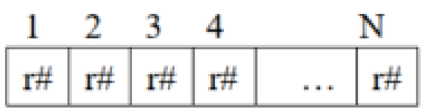

Figure 3.2: Chromosome representation from [7]

The route network design in [52] is done in two phases. Firstly, a set of potential routes competing for the optimum solution is generated by a candidate route set generation algorithm (CRGA). This algorithm consists in several steps:

1. Generate routes for every terminal node pair.
2. Generate a route by finding the shortest path between the origin and destination nodes for each pair identified in Step 1.

3. Check for minimum and maximum route length constraints. If the route satisfies the constraints, then the route is accepted as a candidate route.

4. Generate alternate routes by clamping every link of the shortest route generated in Step 2 successively and by finding the shortest path between the same origin and destination and then, release the link.

5. Check for each of the alternate routes whether it satisfies constraints such as the existence or duplication of the route, significant overlap with the shortest route (more than 75%) or maximum route length. If the route satisfies, then the alternate route is accepted as a candidate route.

6. Rank all the routes based on a performance index and store them as the candidate route set.

After that, the optimum set is selected using GA. The study suggested that the quality of the solution can be increased with a higher number of generations. This requires high computational effort and can be achieved by parallel computing.

By contrast, in study [6], the Initial Route Set Generation Procedure (IRSG) is heavily influenced by the activity level of the stations (nodes), given by the OD matrix, which corresponds to:

$$\text{activity level of station } i = A(i) + P(i) \tag{3.2}$$

A(i) is the number of transit trips culminating in node *i* and *P(i)* stands for the number of trips originating from node *i*. Then, a probability to be added to a route, directly proportional to the activity level, is assigned to each node.

In the case of [53], the GA chooses the bus lines among candidate bus lines that maximize the objective function, which is equal to the user benefits minus cost of the bus network. Therefore, the algorithm selects the routes which have the following three characteristics ([6] built a similar route evaluation procedure):

- Cover crowded regions of city.
- Minimize overlap between different bus lines.
- Minimize bus network length or bus network cost.

In this research, each gene represents one bus line and can have a value of either zero or one. Consequently, each chromosome represents one bus network. If the value of a gene *i* is one in chromosome *j*, it means that the bus line number *i* has been selected in the bus network *j*.

3.3 Transit Route Network Design with Frequency Setting

The main objective of the TNDFSP is finding a set of solutions that provides, for the users view, fast travel times for every origin destination pair with existing demand, the lowest possible average waiting time, few transfers as necessary in order to complete trips thus maximizing direct trips. From the operators' point of view, fewer buses to operate the network, minimum total bus-kilometres and the highest level of utilization as possible.

Firstly, in the literature, there was an example of determining bus service frequency with cost-benefit approach. [54] conducted a study in Malaysia, where the objective function was simply:

$$Benefits - Costs \geq 0 \tag{3.3}$$

Where the investment costs include all costs for vehicle, human resources, workshop, operational facilities, and supporting facilities, while the benefits are expected from ticket fare collected and fuel subsidy. It was implemented with real data provided by bus operators, include vehicle investment and maintenance costs, but also passenger numbers and ticket fare.

[8] proposed a solution method, in which a specific genetic algorithm (GA) is developed to solve the route design problem, while a frequency setting heuristic based on neighborhood search is integrated into the GA to solve the frequency setting problem.

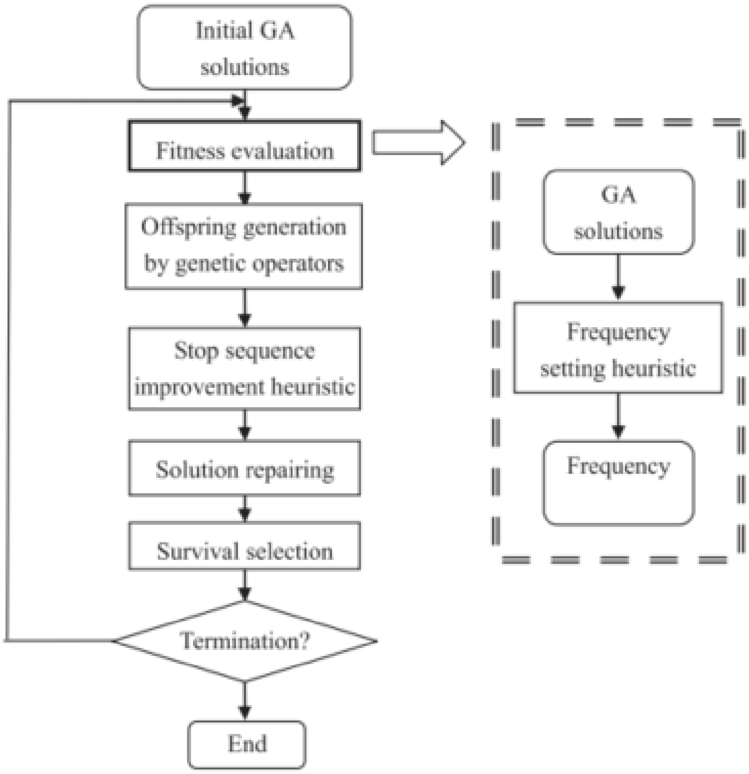

Figure 3.3: Detailed scheme of the hybrid GA implemented by [8]

[55] developed a mathematical model and employed the memetic algorithm to design a set of routes and associated frequency with the objective of minimizing the passenger cost to reduce the unsatisfied demand under the constraints of route length, route directness, maximum flows, fleet size, service capacity and route size. [56] computed a mixed integer linear programming model which handles the integrated network design, line planning and fleet investment. The objective is to maximize the total profit of the network so that a balance between the maximum trip coverage and the minimum total network cost is achieved.

[57] relied on the Artificial Bee Colony Algorithm (ABC) to design route structures and a proposed descent direction search method to determine an optimal frequency setting for a given route structure.

For each solution (i.e., route structure) obtained in the employed or the Artificial Bee Colony algorithm, the following procedure is used to determine the frequency setting:

1. Generate the initial frequencies.

2. Decide whether to obtain an optimal frequency for each route: If the lower bound of given solution is greater than the minimum upper-level objective value until iteration I or the route design is infeasible, then return the initial frequency of each route. Otherwise, proceed to the next step.

3. Solve the lower-level transit assignment problem.

4. If the termination criterion of the frequency determination algorithm is satisfied, then stop and return the optimal frequency setting. Otherwise, go to the next step.

5. Determine the descent direction of the lower- and upper-level objective values with respect to the frequency.

6. Find the step size of the frequency by solving a linear integer program and update the frequency with the obtained step size, then go to step 3

[58] proposed a bi-level formulation for assigning the frequency setting. The upper level concerned in two factors: the network cost which is the passengers' expected travel time that is based on uncertain bus passenger demand and operating costs and the network robustness, this is indicated by the variance in passengers' travel time. The lower level, he calculated the proportional flow eigenvalues using the optimal strategy transit assignment model.

[9] dealt with the problem of the bus frequency optimization with a bilevel approach as well. The bilevel formulation is presented as an interaction between decision takers passengers' route choices and a decision maker (setting frequency) in the following way.

First, it is designed an initial frequency for each route. Then, passengers are assigned to the network with the initial frequencies. This is the lower-level model. The objective of the upper-level model is to optimize the frequency of each route based on the number of passengers on each OD path from the lower-level model. Then, in the lower-level model, passengers are reassigned to the network with the designed frequencies from the upper-level model. The two models are repeatedly used to optimize frequencies of routes until a stable solution for the frequencies and the passenger assignment on the routes. Below there is a model of the previous process.

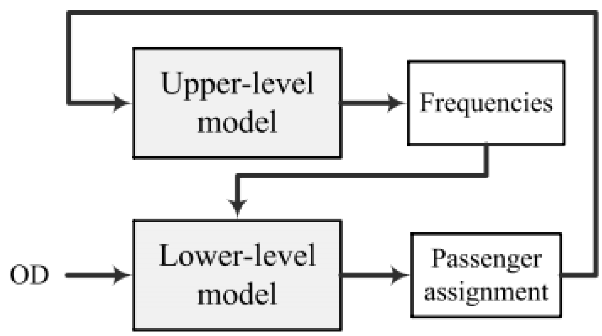

Figure 3.4: Process of the Bilevel Model from [9]

[59] called attention to a few important constraints when approaching this complex optimization problem:

- All transportation network nodes must be present in at least one route in order to be able to serve all transit demand.

- Each routes frequency must lie within a given predefined interval.

- Vehicle load factor must be less than a predefined maximum value to provide service quality for the passengers. Notwithstanding, a minimum capacity should be always taken into account.

- Route length (travel distance or travel time) must lie within a given range so that driver fatigue can be considered and the disturbances in the schedule can be avoided [60]
- Total fleet required to operate the designated routes is subject to a predefined upper bound, in order to cope with resources and budget constraints.
- Limit the maximal allowed number of routes due to the operator's resources limitation.

Chapter 4

ODX Inference

Although ADCS change the quantity, type and quality of data available to transportation planners, there is still a gap between what these systems can directly provide and what transit agencies need in practice. One example is the passenger trip OD matrices, one of the most important elements for transit planning, which cannot be obtained directly from the ADCS since most AFC systems record entries but not exits, which is the case of CARRIS.

OD matrices provide vital information for service planning, operations planning, and performance measurement of public transportation systems.

In this fourth chapter, we will feature available data and its format. Following this, it will be emphasized some steps of data cleaning and integration made by [1]. Additionally, a few aspects regarding origin, destination and interchange inference are described.

4.1 Input Data

Although CARRIS and METRO data are available , only the AFC data relative to the CARRIS bus network is applied. This decision was taken considering the aim of this work, which is focused on bus network optimization.

The AFC data relative to 5 days, from 7 (Monday) to 11 (Friday) of October, 2019 was used.

4.1.1 Lisbon AFC

LISBOA VIVA (since 2021, Lisboa Navegante) is an AFC card based on Radio Frequency Identification (RFID). It is the fare card applied to all the transit operators in the Lisbon Metropolitan Area, such as CARRIS bus network, Comboios de Portugal's (CP) rail network and the Metropolitano Lisboa underground network.

Figure 4.1: VIVA Lisboa card sample

4.1.2 CARRIS

A fare validation in Carris network allows the passenger to travel for 60 minutes, counting from the first entrance with unlimited number of bus journeys.

An AFC transaction record is generated each time a passenger swipes or taps a farecard at AFC equipment. The relevant information to our work is:

- Timestamp associated with the transaction, specifying the date and time, with second precision
- Card serial number, which is unique per passenger (card)
- Boarding stop identifier, where the passenger entered the bus
- Route being serviced by the boarded bus, which consists of three different fields:
  - Route ID, which identifies the route
  - Route direction, which gives the direction of travel and can take the value of ASC (ascending), DESC (descending) or CIRC (circular)
  - Route variant which differentiates between the several variations of the route

In theory, for a given route and trip, the fare collection timestamp from the farecard is exploited to search through the AVL data to determine the boarding stop. In the case of AFC system of CARRIS, boarding stop is already present in the AFC transaction. Despite having no further details, we will assume that a similar procedure was applied.

4.1.3 Metro Lisboa

As indicated in Background chapter, in contrast of CARRIS network, Lisbon's underground network is an AFC closed system. By saying that, origin and destination are known undoubtedly. In addition, metro AFC system does not record metro interchanges, on account of that passengers don't have to tap the card when transferring between metro lines. Passenger's destination is presumed to be the station where the fare ticket is tapped out of network.

Even though this underground data will not be employed, relevant fields are:

- Timestamp associated with the transaction, specifying the date and time, with second precision
- Card serial number, which is unique per passenger (card)
- Station identifier field, identifying the station where the passenger tapped the card
- Entry (IN) or Exit (OUT) field specifies if the tap corresponds to an entry or an exit, respectively

4.2 Origin Inference

Origins are inferred by matching fare transactions to locations through transaction time stamps and card-reader identifiers. When a card reader is installed at a fixed location, such as a fare gate in a station, the reader's location is simply assigned to the transaction. When the reader is installed in a vehicle, the transaction time stamp is compared with vehicle location data to determine boarding stops. [61]

4.3 Destination Inference

Following the assumptions presented on Chapter 2 concerning bus alighting times and locations, spatial information about the passenger's possible alighting location and subsequent origin location is needed. By saying that, the distance between the passenger's next boarding location, referred to as target-location, and each stop served by the current vehicle trip need to be calculated, in order to determine which stop is closest.

In the event of a transaction of a bus boarding, the stop and route field must be valid and cannot be on the last stop of the route . Otherwise, the origin is assumed unknown and the destination cannot be inferred.

If there are no other records in the card's daily history, the closest-stop rule cannot be applied. On the opposite, if the transaction is the last transaction of the day, the daily symmetry rule is applied, and the target location is defined as the first origin of the day. Otherwise, the target location is the next stage's origin.

If the distance between the alighting stop candidate and the next tap location is greater than a pre-established maximum interchange distance, the destination can´t be inferred, because is assumed that the passenger will not walk a significant distance.

Below there is a summary of the mentioned above, made by [1] and present in his work.

4.3.1 Bus Alighting Time Estimation

Besides deducing alighting locations, destination inference process also aims at inferring alighting times. However, we do not have AVL data available that could directly provide this valuable information.[39] [16] [22] made favourable use of data that is automatically collected that relates to the location of the bus, therefore a different approach had to be followed.

[1] exploits the GTFS file *stop times.txt*, which has, for every trip, the scheduled arrival and departure times for every serviced stop. In the case of the bus GTFS, the departure time and arrival time have the same value in every entry. By calculating the difference between the departure time for two consecutive stops, the result is the time the bus takes between those two stops. So, by adding up each consecutive stops plus a bus stop time per stop, it is possible to deduce route or trip duration.

It is worth to mention that, given the data circumstances, [1] subtracts bus stop time once, as the passenger does not have to wait on the stop he alights.

4.3.2 Destination Inference Results

Using the above mentioned techniques results for the destination inference process are:

Result	Count	Percentage
Destination Inferred	328660	47.70
Distance between candidate alighting location and next origin $\geq$ 750 meters	82819	12.02
Only one stage on card that day	277534	40.28
Total	689013	100.00

Table 4.1: Destination Inference Results

The percentage of destinations deduced is less than the half. The reasons rely on a significant percentage of cards (40.28%) that are tapped only one time that day, which does not enable us to consider closest stop rule. Furthermore, a substantial distance between the alighting stop candidate

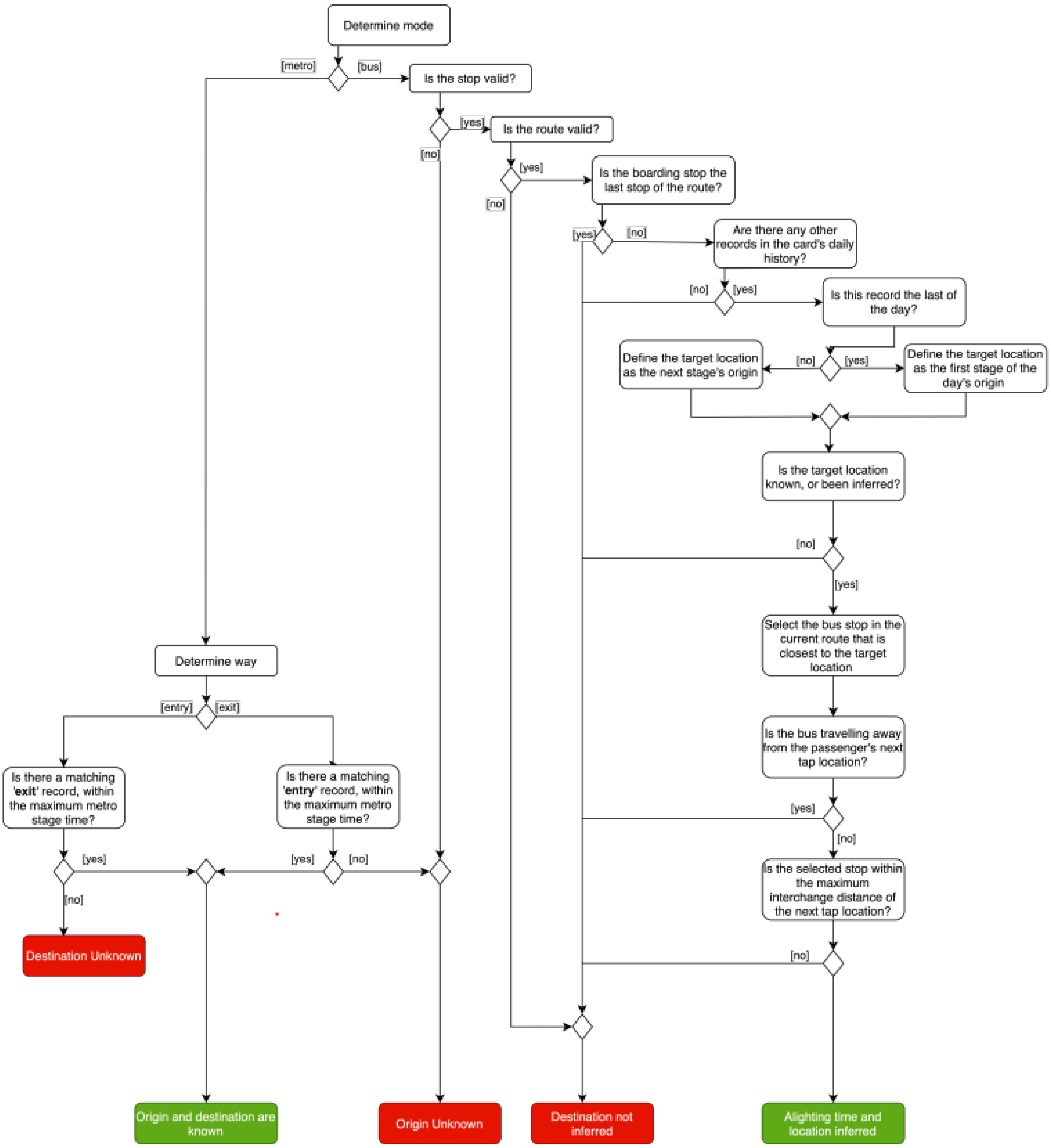

Figure 4.2: Destination Inference Fluxogram reproduced from [1]

and the next tap location (12.02%) does not allow us to infer destinations. It is assumed that passenger travelled by other means of transport.

As it would be expected, this result is quite similar to [1], where 45.24% of the destinations were inferred.

4.4 Interchange Inference

Interchanges are deduced by distinguishing them from trip-generating activities. Temporal and spatial data are used as proxies for passenger activity information, based in the assumption that trip-generating activities are longer in duration than convenience activities.

[1] applied binary, spatial and temporal tests sequentially to every stage alike [24], in order to infer if the two stages are linked and thus part of the same journey. A failing test means that the stage being tested is unlinked from the subsequent stage.

Our work is not focused on inferring transfers, however [1] in his thesis inferred just 2.51% of the cases. The reason indicated to this low value is due to the lack of AVL data.

4.5 Bus Alighting Stop Distribution

The destination inference process computed by [1] was capable of deducing 47.70% of bus alighting times and locations. This is quite inconclusive, as the majority of the journeys do not have a destination.

In this thesis, one metric of our work is empty seats in a bus during a route, which represents the difference between entrances and alightings in a route stop. In pursuance of the best optimization solution as possible, it is essential to have the idyllic overall estimate of the demand.

By stating that, in order to overcome this lack of destinations inferred, we do propose to assume an uniform distribution to the trips that do not have destinations deduced. This means that the number of passengers that enter in a specific stop, which do not have a destination, are distributed equally to the stops ahead of that route.

Below there is an example of stop 1607 (Sete Rios) from Route 758, Variant 0 and Direction ASC, before and after bus alighting stop distribution procedure. These validations are from 7 to 11 of October, 2019.

Destinations	Before	After
1616	82	125
10706	117	160
10704	79	122
10702	58	101
10830	41	84
10826	59	102
10814	29	72
10802	17	60
10906	45	88
10904	69	112
10902	78	121
11006	6	49
11007	54	97
Unknown	556	-

Table 4.2: Matrix of trips with origin on stop 1607, before and after distribution procedure

4.6 Summary

This chapter explained the ODX inference process implemented by [1] and applied in our thesis.

There was no origin inference methodology, as AFC system from Carris automatically infers the boarding stop of each transaction.

Deduced destinations percentage was $\approx 47.70\%$ of the transactions. This value is due to bus stages that are the only recorded transaction on that day and also candidate alighting location and next origin are too distant.

It must be noted our method to overcome the high rate of transactions without destination. We distributed uniformly, for each origin stop, every uninferred destination trip to the route stations ahead. Having this, all AFC transactions are valid to further steps.

Chapter 5

Transit Network Design with Frequency Setting

This thesis comprises two main stages: the generation of the set of routes and the determination of the frequency for those routes. The first stage comprises of designing an efficient set of routes on a given transit network to enhance the trip directness, service coverage area and reduce the average travel time for passengers. The second stage aims to assign frequency of buses on each route to lower the waiting time, and minimize the passenger time and the operating cost simultaneously [62].

As demand varies throughout the day, operator resources vary as well. In the same manner that during peak hours there is a heavy pressure on transit activity, during the night period this tends to become residual.

In our case, day will be divided in 4 parts, which means 4 OD matrices that acknowledge demand alterations. The time interval studied is from 7 (Monday) to 11 (Friday), October 2021. Each limit value corresponds to transaction timestamp, that is bus entry time:

- 6:30 a.m to 10:59 a.m, which will be named first matrix or morning matrix
- 11 a.m to 3:29 p.m, that will be titled as second matrix or lunch matrix
- 3:30 p.m to 7:59 p.m, denominated as third matrix or afternoon matrix
- 8 p.m to 0:30 a.m, designated as fourth matrix or night matrix

The choice of matrices hours aimed to divide week day on demand periods. Morning and afternoon matrices represent to first and second peak hours of the day, respectively. On these periods, most of the validations represent commuting trips, home-work or work-home. Lunch matrix aims to illustrate an intermediate demand level of passenger demand, the number of validation during this period are slightly lower than peak hours matrices. Last matrix describes a decline on passengers number during night hours.

In this chapter, we begin to present some preprocessing steps in the OD matrix as well as problem representation and its difficulties. Then, some procedures about frequency assignment and adjustment. In the end, it will be pointed up details regarding Genetic Algorithm.

5.1 Inputs

5.1.1 OD Matrix Preprocessing

We deduced origin, destinations and interchanges of AFC data under the conditions described on chapter 3. After that we computed 4 OD matrices given bus entries timestamps, as a representation of the passenger demand alterations during a day

It is worth to mention a relevant change as far as optimization process is concerned. Both us and [1] are only optimizing the bus network, so OD pairs from metro journeys are not considered, since no bus route can outperform the metro network.

5.1.2 Input Matrices

Each matrix divides the day, so it becomes easier to match resources according to demand.

Matrix	Number of Validations	Percentage
6:30 a.m - 10:59 a.m	158064	**32%**
11 a.m - 3.29 p.m	131580	**27%**
3:30 p.m - 7:59 p.m	165439	**33%**
8 p.m - 0:30 a.m	39079	**8%**
Total	**494162**	**100%**

Empirically, it is demonstrated that during peak hours there are way more passengers compared to night hours. Second matrix, as expected, represents a small decrease on passengers number.

5.1.3 Road Network data

OpenStreetMap data provides free detailed geographic information such as road links, turn restrictions, maximum speeds, etc, which are crucial for a good representation of the road network. It is a massive online collaboration, with hundreds of thousands of registered users worldwide.

The OSM data for the entire country of Portugal is available for download, but can be filtered to include only the necessary road links of Lisbon to describe the bus network.

5.2 Representation and Preprocessing

Considering that we are dealing with a real, complex network a representation of the road and route set has to be computed.

[1] built a road network representation that includes the stop locations and the road links that connect these stops. This representation is used to compute the paths between stops in the underlying road network. Furthermore, the route set representation describes the stops and bus routes that connect the stops. Each route set has a unique route set representation. This portrayal is used to compute the paths between stops in the bus network.

5.2.1 Road Network

Using the road network data from OpenStreetMap, [1] computed a graph that includes the bus stop locations and the road links that connect the stops.

Figure 5.1: Road network graph of Lisbon

There were some mismatches between stop locations on GTFS and OSM data, so it was applied a software (pfaedle)[1] which was utilized in our work, to map each GTFS stop to a location on the OSM road network.

Regarding bus network graph, a directed, weighted graph $G(N, E)$ was employed, where the set of nodes N represent the bus stops, and each edge $(i, j) \in E$ is a road link that connects stops i and j.

The weight of each edge, $w_{i,j}$, is the duration of the shortest path between stops i and j. An edge from stop s_1 to stop s_2 exists if s_2 is a neighbour of s_1, that is to say, if the path between s_1 and s_2. Neighbours of each stop are computed by relying only on the duration of shortest paths between stops.

5.2.2 Route Set Graph

Each possible solution to this work is a route set. A route set is characterised by a list of routes, where each route serves a sequence of stops at a specific frequency.

Given the nature of the problem, each route set is described by a weighted, directed graph $G = (N, E)$, where N is the set of nodes representing bus stops and E is the set of edges, representing the connections between each consecutive stops served by each route, titled route edges.

The weights of the route edges symbolize the duration of the path between the each consecutive stops. Thus, the weight of edge $e_{u,v}$, connecting stop u to stop v, is given by $d_{u,v} + b$ where $d_{u,v}$ is the duration of the shortest path between stops u and v, calculated using OSRM[2], and b is a constant used to account for the time a bus takes to stop at a stop and wait for the passengers to alight or board the vehicle, and is set to *30 seconds*.

Each stop must have one node per route, called route node, that serves that stop.

[1] 1https://github.com/ad-freiburg/pfaedle

[2] High-performance routing engine for shortest paths in road, bicycle and pedestrian networks

To allow transfers between routes of the same stop, an edge is added between every route node of each stop. We will refer to these edges as transfer edges. The edge weight of transfer edges is given by the parameter transfer weight and represents the time a passenger takes to transfer from one route to another. Considering we are optimizing frequencies, this weight will correspond to half of the headway of the intended route. This is justified by the fact that passenger's arrival follow the uniform distribution.

$$\text{Transfer Time} = \frac{\text{Route Headway}}{2} \tag{5.1}$$

In the case of [1], transfer_weight was set to 5 minutes in all routes. He assumed that every bus in the system had a fixed headway of 10 minutes.

Besides, it is worth to mention that [1] added two nodes per stop, origin and destination nodes, due to shortest path calculation purposes. Every origin or destination nodes are linked to routes nodes by an edge of weight 0.

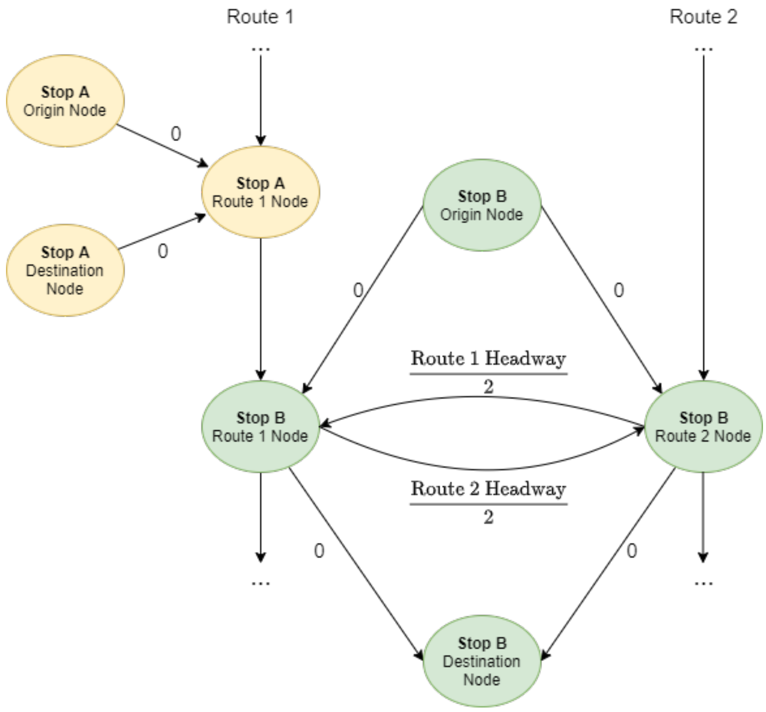

Figure 5.2: Example of route set graph topology

5.2.3 Preprocessing

[1] made a few adjustments with the intention of simplifying the problem and minimize computation time:

- Ignore stops that are not present in the AFC dataset, it is assumed they are inactive.
- Ignore tram-only stops that are only served by tram
- Clustering of stops that serve the same point of demand. This happens in areas wheres is a lot of demand concentration of people in the same physical stop. From a network design standpoint, these stops are one single stop.

5.3 Initial route set

In the first step of the genetic optimization process, it is necessary to produce an initial solution (route set).

Firstly, one single route set of R (input value) routes is obtained. Each of the R routes is a shortest path (based on travel time) between a selected pair of stops. Using a greedy algorithm, it is selected the R pairs (i, j) that have the highest number of passengers that travel along the shortest path between stops i, j. We will address this value as ds_{ij}.

$$ds_{ij} = \sum_{m \in S} \sum_{n \in S} d_{mn} \tag{5.2}$$

where S is the set of stops that are in the shortest path between i and j. d_{mn} is the number of entries for pair (m, n) in the OD matrix.

The greedy algorithm is summarised as:

1. Input the value of number of routes, R.
2. Initialise matrix DS, where $DS = \{ds_{ij} \mid i, j \epsilon [0, 1, 2, \ldots, |N - 1|]\}$
3. Find the pair (i, j) in DS with the highest ds_{ij} value.
4. i and j become the terminals of a new route.
5. Add every node in the shortest path between i and j to the route.
6. Remove every node pair (m, n) that are satisfied by the newly added route from DS
7. Check if the number of routes reaches N, stop. Otherwise go to Step 3

Secondly, an initial headway is assigned to each route created, equal across matrices. This value indicates the time between consecutive services in a route. In section 5.4, it will be specified this operation.

In the of the described procedure, there is an initial route set and its headways.

5.4 Genetic Algorithm

In this chapter, we will describe Genetic Algortithm [1] [63], but with some improvements to include frequency setting .

Each gene is a route and its respective headways. Each population is a possible solution to the problem, which is comprised by a route set and its headways that vary throughout the day. In our case, a **possible solution includes a route set and four headways, for each route**, since we have four OD matrices.

The objective function attempts to **minimize** costs of both parties which take part on public transportation activity, **passengers** and **transit operators**:

- Passenger Costs:
  - In-vehicle Time
  - Waiting Time
  - Transfers

- Operator Costs:
  - Empty seats

Passenger costs represents the amount of time a passenger has to experience when waiting and travelling on a bus, but also when transferring from a bus to another.

Operator cost in our thesis is represented by the number of empty seats on a bus during a route progression. Empty seats is seen as a unproductive cost to transit companies.

5.4.1 Objective Function

The objective function defines the aim of the optimization process. In our case, this equation describes the quality of a solution, which intends to minimize costs of all stakeholders associated with public transportation operation. This costs can be given by T:

$$T = passenger\_weight * (ATT + AWT + w * ATR) + operator\_weight * (AES)$$

Where each metric stands for:

- ATT: **Average In-Vehicle Time**. The average time a passenger spends in-vehicle when travelling
- AWT: **Average Waiting Time**. The average time a passenger has to wait at route stop
- ATR: **Average Number of Transfers**. The average number of transfers a passenger has to experience, w is a penalty equal to AWT
- AES: **Average Empty Seats**. The average number of empty seats across all network
- $passenger\_weight$: importance given to metrics that represent costs to passengers (AWT,ATT,ATR)
- $operator\_weight$: importance given to metric that represent a cost to transit company (AES)

Metrics are measured considering entity weight, which admits divergent interpretation of cost. While from passengers' point of view, cost means time taken travelling or waiting for the bus in the station, but also annoyance or discomfort on transferring between buses and routes. From operators perspective, empty seats are a sign of prejudice, as buses are not operating at a reasonable occupation, although fuel cost and vehicles depreciation remain stable.

As we are dealing with a route set which contains different headways across matrices, there is a fitness value for each matrix (part of the day), hence we define that the most suitable solution is the route set and its respective headways that have the lowest mean fitness value.

$$\text{Fitness Value of a Solution} = \frac{\sum_{i=1}^{4} \text{Matrix } i \text{ Fitness Value}}{\text{Number of Matrices}} \tag{5.3}$$

Where in our work *Number of Matrices = 4*.

Figure 5.3 illustrates the solution topology.

5.4.2 Operators

Genetic operators guide the algorithm towards a solution to a given problem. They mimic natural world processes, such as survival of the fittest, reproduction or mutation.

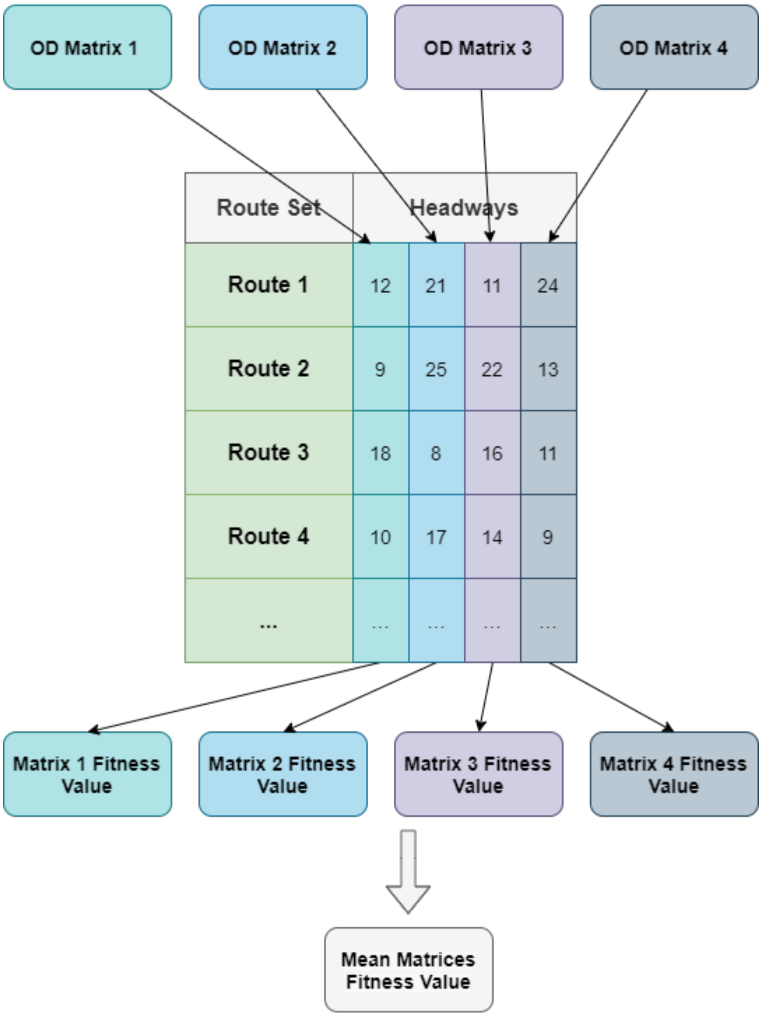

Figure 5.3: Solution Topology

5.4.2.1 Selection

In each generation, individuals are selected from the population to mate and generate offsprings for the next generation. Tournament Selection was used as the selection mechanism. The probability of selecting each individual i is given by

$$p_i = \frac{\Sigma_{j=1}^{N} f_j}{f_i},$$

where N is the population size and f_i is the value of the objective function of individual i.

5.4.2.2 Crossover

The crossover operator swaps routes at each route index, between the two selected parents with probability $p_{swap} = \frac{1}{R}$, where R is the number of routes.

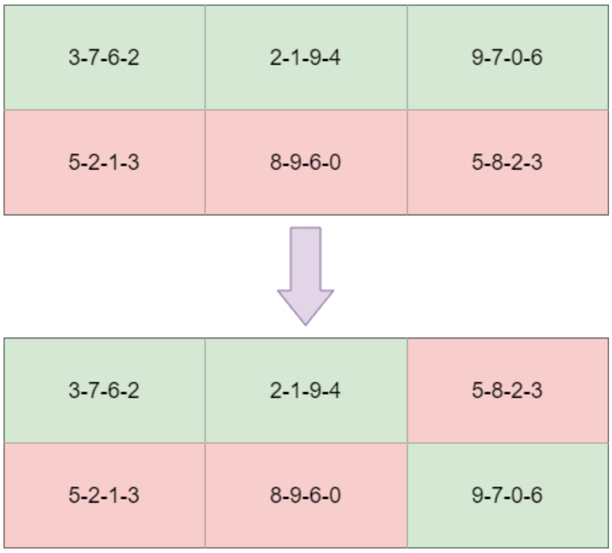

Figure 5.4: Routes at index 2 get swapped - Cross Operator

5.4.2.3 Mutation

The mutation operator is applied to the two offsprings that result from the crossover step. The mutation process is designed to select routes that have less demand more often, as the probability of select one route is: [1]

$$p_l = \frac{\frac{1}{ds_{ij}}}{\sum_{q \in L} \frac{1}{ds_{rs}}}$$

where i and j are the terminals of route l, L is the route set, r and s are the terminals of route q, and ds_{ij} is the total number of entries in the OD matrix that are satisfied, without any transfer, by using route l, that connects terminals i and j.

After route selection, it is applied a small modification with a high probability p_{ms} and a big modification with a low probability $1 - p_{ms}$.

- ***Small modification:*** selects one of the route terminals with the same (0.5) probability and applies one of two operators:
  - **deletes** the selected route terminal, with probability p_{delete}.

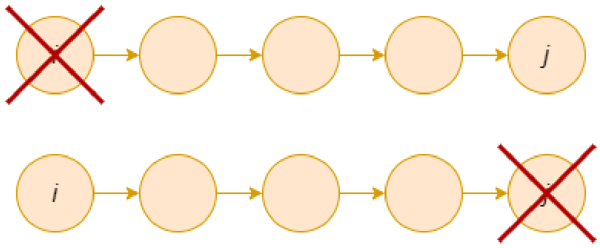

Figure 5.5: Delete - Mutation Operator

- **extends** the selected route, adding a new stop in the selected route terminal, with probability $1 - p_{delete}$

  The new stop is chosen at random from among the adjacent stops to the chosen terminal, in the road network graph. If the selected route terminal is the route's first stop, the new stop is prepended to the route. If the selected terminal is the route's final stop, the new stop is appended to the route.

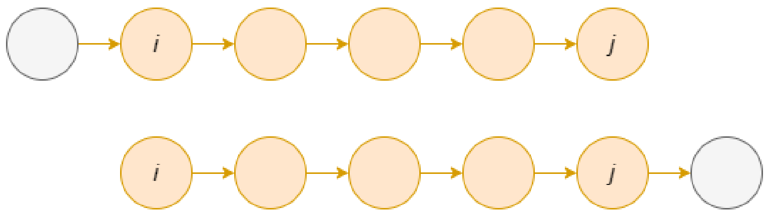

Figure 5.6: Extend - Mutation Operator

- ***Big modification:*** Selects one of the route terminals (say terminal i) with the same (0.5) probability, and then selects a new terminal k, with the following probability:

$$P_k = \frac{ds_{ik}}{\sum_{r \epsilon N} ds_{ir}}$$

The selected route will be replaced by a new route that is the shortest path between terminal i and terminal k.

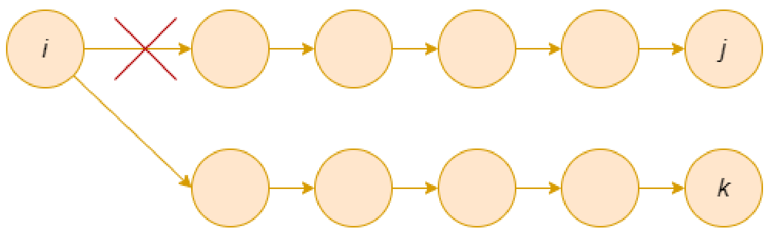

Figure 5.7: Extend - Mutation Operator

5.4.2.4 Elitism

One of the inputs of Genetic Algorithm parameters is $elite\_size$. This value ensures that the most fitting individuals from the current generation are copied to the next one.

5.5 Frequency Setting

In this section, we will describe frequency setting procedures during Genetic Algorithm progress. At start, headway interval is defined between variables ($min\_headway$) and ($max\_headway$).

After that, two separated frequency setting methodologies can be followed:

- **Random Search**: Randomizes all the headway values associated to the route set between all .
- **Local Search**: progressively vary all the headway values associated to the route set.

5.5.1 Initial and Update Frequencies Process

At the procedure start, as regard to random search, initial headways are completely randomized. As Genetic Algorithm evolves, for each new generation, it is maintained random change in headway values. On the opposite, local search starts off all headway with mean value between limit interval values, ($min\_headway$) and ($max\_headway$). After that, each new headway value is subject to a much smaller change operation. Consequently, each number may vary a difference of $[-5, 5]$.

5.5.2 Service Level across matrices

Between operator cost values associated with the same route set, our optimization algorithm does not allow disparities above a certain percentage throughout day planning.

In such option, it is guaranteed that there is a certain balance on cost of service (represented by empty seats) throughout the day. Empirically, waiting times, in other words route headways, are way higher at night than during peak hours, as a consequence of decline on demand. Likewise, keeping the same level of service (route headways) becomes financially unsustainable to transit company. However, we try to ensure that there is a consistent level of service, besides demand divergences.

So as a way of simulating resources management, we considered this operation in this thesis.

5.5.3 Small Frequency Adjustment

In the case of a disparity between two calculated operator costs (empty seats) above a pre-determined value, there is a method that attempts to converge these values.

On the one hand, by subtracting an integer $\in [0, 2]$ to the route headways (increase frequency) associated with the matrix that has higher fitness value and, on the other hand, by summing an integer $\in [0, 2]$ to the route headways (diminish frequency) of the lower fitness value matrix, we intend to normalize both values. The operatation is common to both frequency searches.

This means that through boosting route bus appearances (route frequency), matrix fitness value tends to decline.

5.6 Routes Overlapping

During genetic algorithm progress, when calculating fitness value of a matrix, there is a decisive step that aims to check whether exists equal segments among route set. By doing this, we intend to check if there are alternatives to an OD trip.

Below, there is a simple illustration of what is considered a route alternative. It is not considered an option even if it includes that exact OD pair, but when there is the same OD segment. This option is due to the fact that the route generation method (5.3) already contemplates shortest paths.

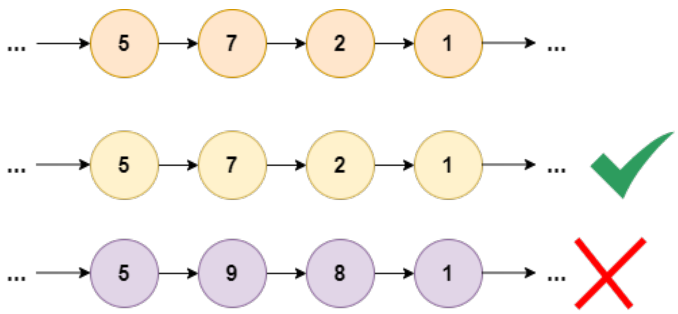

Figure 5.8: Example of route overlapping

The aforementioned constraint is quite suitable as it approximates our work with the reality, particularly when calculating some metrics, such as empty seats and waiting time (method explained on 2.3).

As far as empty seats is concerned, routes overlapping determination is absolutely vital, because can give us the number of bus appearances of routes that cover an intended OD segment.

Firstly, empty seats are calculated by iterating the bus capacity (in our case 85 seats, following Carris fleet capacity [64]), with the difference between boardings and alightings in each stop throughout a route progession. For each route, it will be computed the mean empty seats value of consecutive stops journeys.

In the end, the final average empty seats value to all the network corresponds to the mean value between every route empty seats value. Below a small illustration of Average Empty Seats metric calculation.

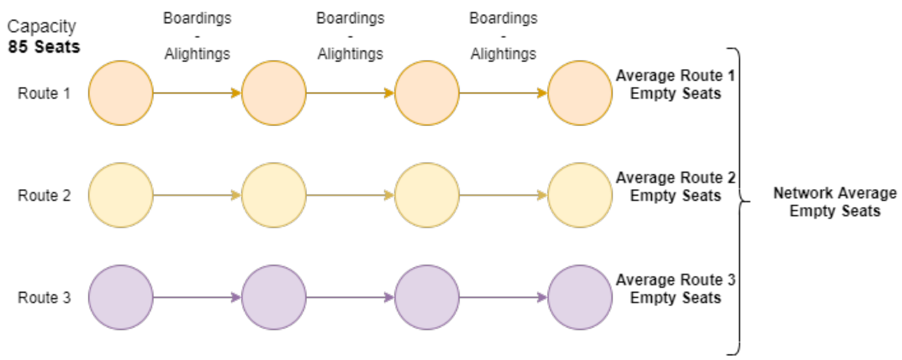

Figure 5.9: Average Empty Seats computation

5.7 Summary

In this fifth chapter, it is depicted problem representation and formulation, along with Genetic Algortihm method used to the Transit Network Design+ Frequency Setting problem applied to Lisbon.

It is highlighted the option of dividing the day in 4 equal parts of 4:30H duration. Such decision provides demand change notion across the day.

In the end, frequency adjustments methods, Random Search and Local Search, are described. Moreover, in case of that aim to avoid fitness values disparities.

Chapter 6

Results

This chapter presents the results of applying the optimization methodology earlier described to the bus network in Lisbon.

6.1 Evaluation and Metrics

In the first set of experiments, the routes were generated from scratch, using the initialization method described in Section 4. Several tests were conducted for the number of routes $R = 150$ in order to assess headway adjustment method (Local and Random) and metric weights variation (passenger and operator weight).

Additionally it is analysed in detail most optimized solution among experiments conducted.

Later, it is conducted another set of tests, where the existing (real) bus network was used as a starting point to the optimization process, serving as the initial solution. The analysis method is equal to the described on the above paragraph.

In the end of the chapter, it is presented an overall analysis of the results and computed methodology.

All tests do accept a maximum difference among matrices operator costs of 20%. We opted for this value because on small experiments, with less routes and iterations, the biggest disparity on matrices empty seats was frequently above this threshold.

Several combinations of values for the following parameters were used in both tests:

- **Operator Weight** ($operator\_weight$)
- **Passenger Weight** ($passenger\_weight$)
- Search Method:
  - **Local Search**
  - **Random Search**

To evaluate the results, the following metrics of objective function (described on Section 4) were considered:

- **Fitness Value**
- **Average In-Vehicle Time** (ATT)
- **Average Waiting Time** (AWT)

- **Average Number of Transfers** (ATR)
- **Average Empty Seats** (AES)
- **Passenger Cost** = ATT + AWT + AWT*ATR
- **Average Headway**

Genetic algorithm parameters were not the aim of this work, so we fixed them across all tests:

- pop_size=16
- $elite\_size$=4
- t=4
- p_{ms}=0.7
- p_{delete}=0.6

Across all tests, each route headway lies within an interval between $min\_headway$=7 and $max\_headway$=25.

6.2 Optimization from Scratch

In the initial tests with a smaller network of 150 routes, which are generated from scratch, we evaluated search frequency models and metric weights changes. We opted to assign values of 0.4 and 0.1 to operator metric, together with 0.6 and 0.9 to passengers', because we would like to test two situations. Firstly, where metrics have a balanced value (first case, 0.4 and 0.6), and the second, where there is a heavy importance given to passengers (0.1 and 0.9).

It is worth to mention that in our tests, passengers have always an higher importance assigned, since we have larger amount of information available associated with them and consequently more metrics to assess, than operators cost, which is limited regarding matrices values already. Another relevant reason is that the absolute value of empty seats is way greater than passenger costs putted together.

Search	Objective Function	ATT (min)	AWT (min)	Transfers (%)	Empty Seats	Average Headway (min)
Random	44.06	9.62	6.81	0.88	76.5	15.75
Local	44.04	9.63	6.62	0.88	77	15.8

Table 6.1: Initial Metric Values - $operator\_weight$=0.4, $passenger\_weight$=0.6

Search	Objective Function	ATT (min)	AWT (min)	Transfers (%)	Empty Seats	Average Headway (min)
Random	39.83 (-10.62%)	6.68	5.725	0.49	77	15.825
Local	41.9 (-5.11%)	8.31	6.12	0.69	76	15.07

Table 6.2: Final Metric Values - $operator\_weight$=0.4, $passenger\_weight$=0.6

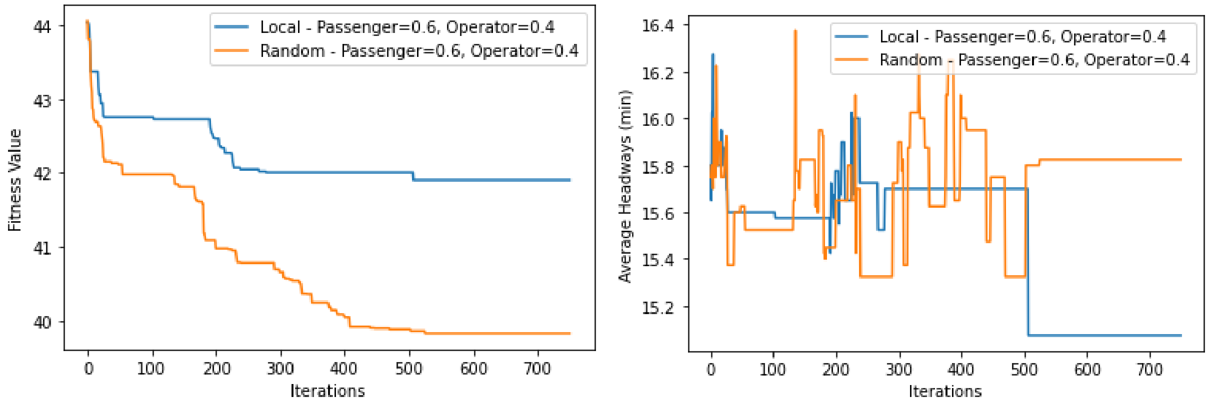

Figure 6.1: Fitness Value and Average Headways - $operator\_weight$=0.4, $passenger\_weight$=0.6

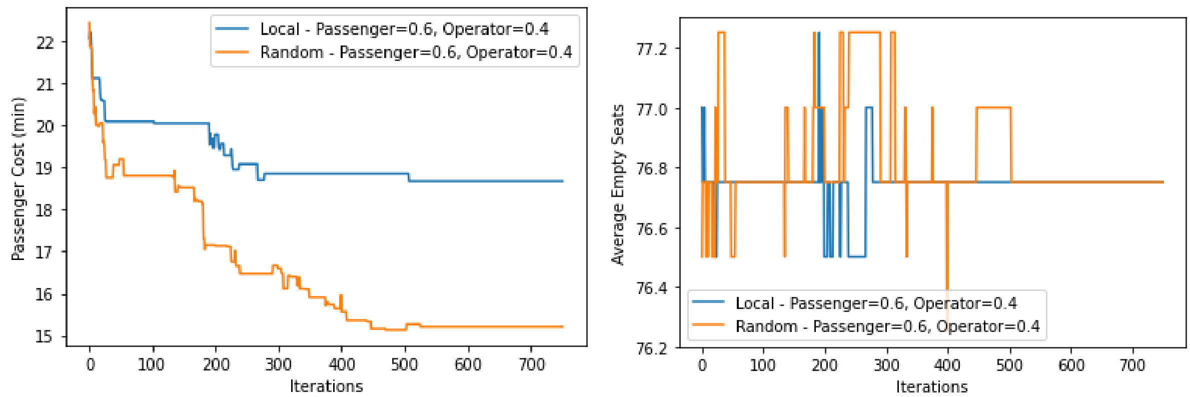

Figure 6.2: Passenger Cost and Average Empty Seats- $operator\_weight$=0.4, $passenger\_weight$=0.6

Search	Objective Function	ATT (min)	AWT (min)	Transfers (%)	Empty Seats	Average Headway (min)
Random	27.75	9.63	6.71	0.89	77	15.925
Local	28.21	9.63	7.02	0.88	76.75	16.075

Table 6.3: Initial Metric Values - $operator\_weight$=0.1, $passenger\_weight$=0.9

Search	Objective Function	ATT (min)	AWT (min)	Transfers (%)	Empty Seats	Average Headway (min)
Random	21.79 (−27.35%)	6.92	5.54	0.56	78	15.35
Local	23.17 (−21.75%)	7.45	6.01	0.61	77	15.75

Table 6.4: Final Metric Values - $operator\_weight$=0.1, $passenger\_weight$=0.9

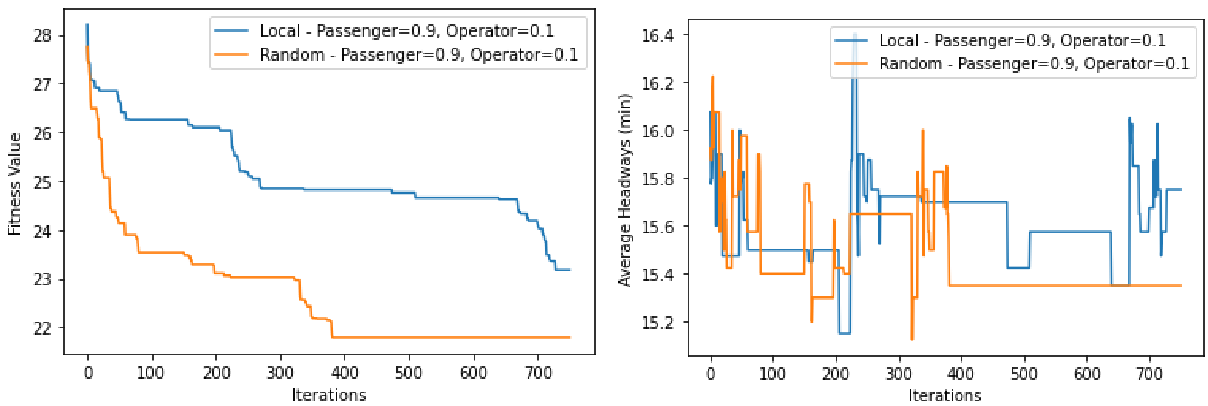

Figure 6.3: Fitness Value and Average Headways - $operator\_weight$=0.1, $passenger\_weight$=0.9

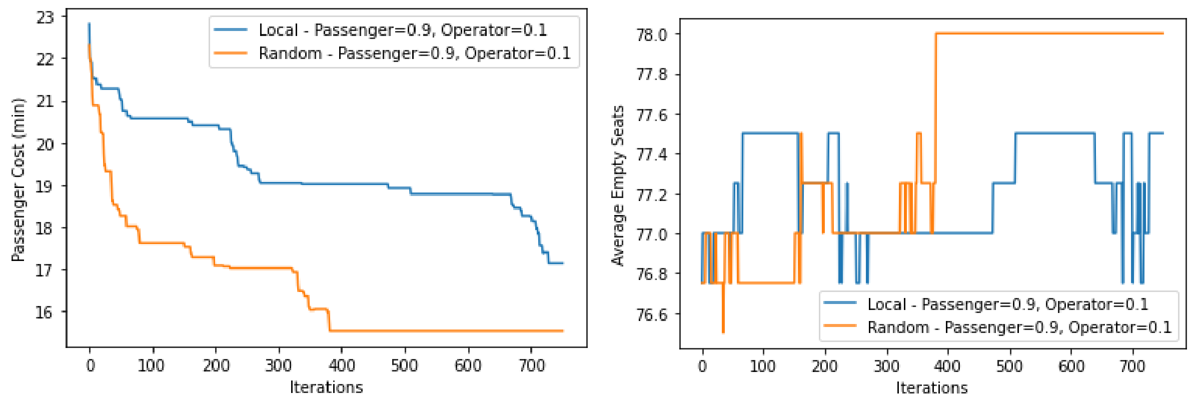

Figure 6.4: Passenger Cost and Average Empty Seats - $operator\_weight$=0.1, $passenger\_weight$=0.9

First experiments show that Random Search method has a better performance than Local Search.

Random Search computes optimal solution after around 400 iterations, as well as, it rapidly decreases fitness during the first 200 iterations. While Local Search on $0.4/0.6$ test soon reaches it optimal solution and barely changes during the rest of the process. On contrary, during $0.1/0.9$ test it gradually adjusts it optimal solution, however nearly in the end it has a quick fitness decrease.

Regarding cost weight changes, on the test where it is given a greater importance to the operator cost (0.4), there is a reduction on average empty seats, which is natural given the bigger preoccupation to the company cost.

6.2.1 Optimized solution in detail

Using Random Search and $operator\_weight$=0.1, $passenger\_weight$=0.9, which was the best parameters combination (-27.35% of improvement on fitness value), we present in detail metrics for each matrix.

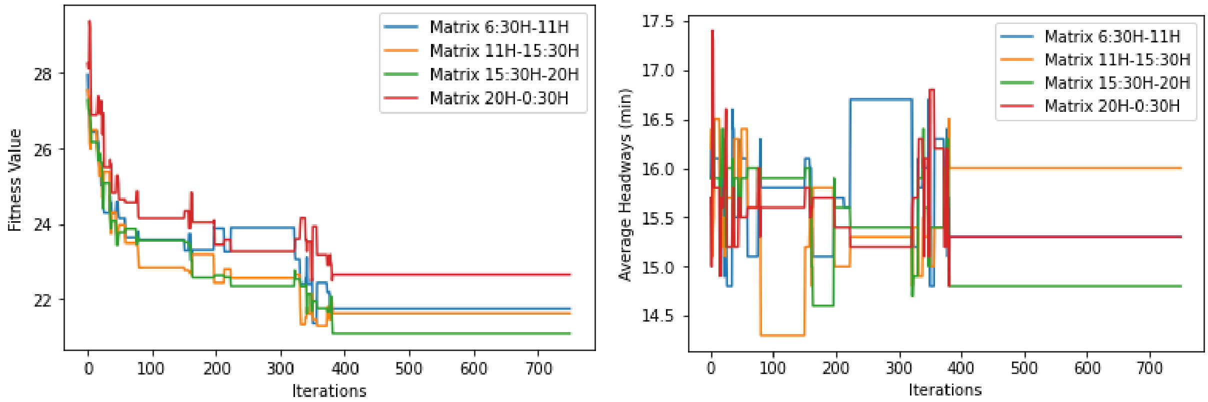

Figure 6.5: Fitness Value and Average Headways - $operator\_weight$=0.1, $passenger\_weight$=0.9

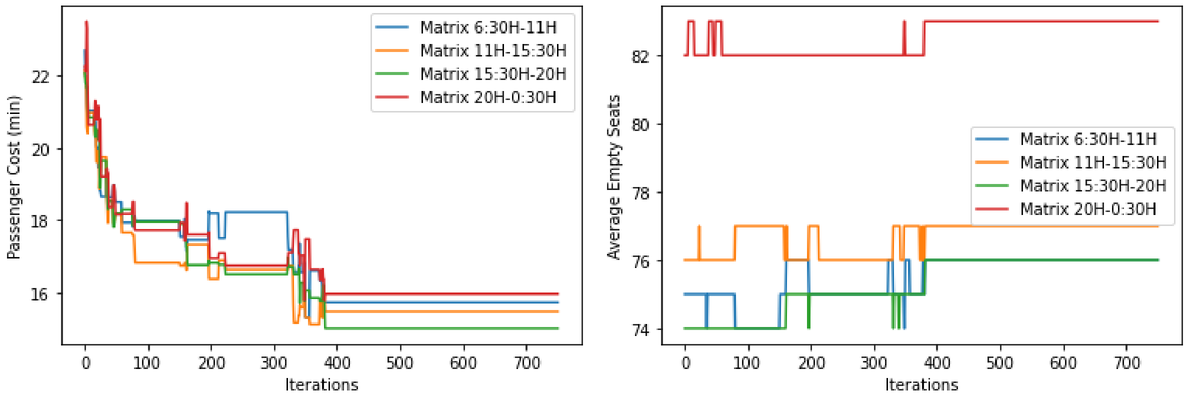

Figure 6.6: Passenger Cost and Average Empty Seats - $operator\_weight$=0.1, $passenger\_weight$=0.9

Night period clearly has a greater fitness value, mostly because it has a higher average empty seats value, which is somehow acceptable given demand level. However, passenger costs are rather balanced among all matrices.

The huge average empty seats difference between fourth matrix and the others is not mirrored on the increase of matrix fitness value, because in this experiment operator weight is reduced.

6.2.2 Fixed Maximum and Minimum Headways Test

Besides above tests, in order to get an overall result of the computed algorithm, we conducted two additional experiments. The first, fixing headways to the minimum value, $7min$, second one fixes it at its maximum value $25min$.

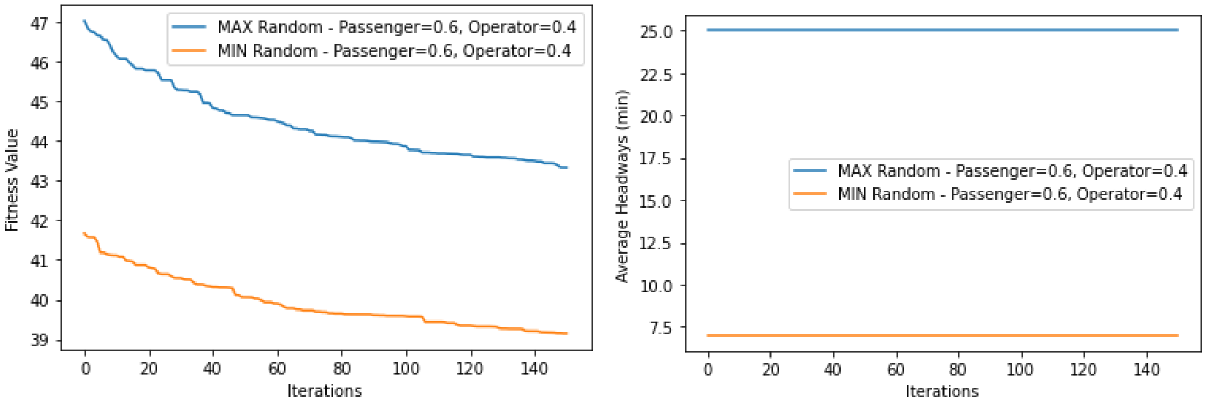

Figure 6.7: Fitness Value and Average Headways - $operator\_weight$=0.4, $passenger\_weight$=0.6

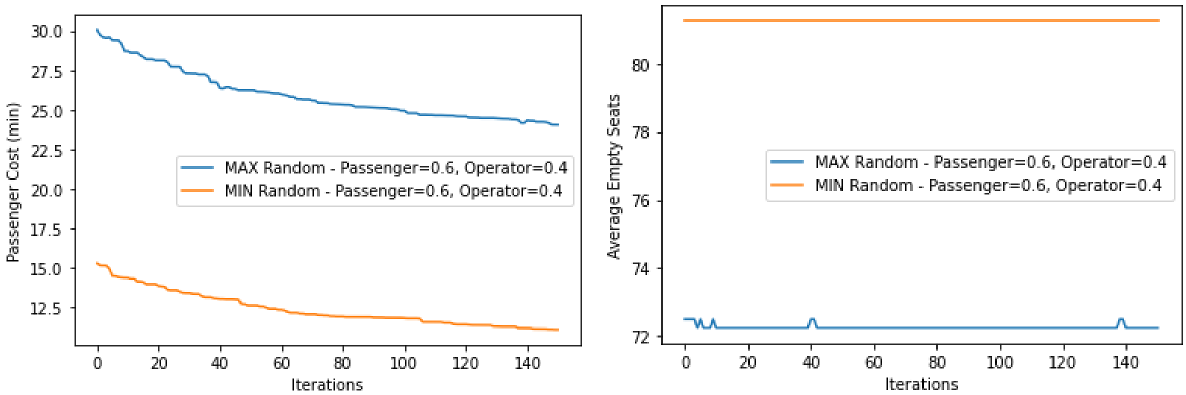

Figure 6.8: Passenger Cost and Average Empty Seats - $operator\_weight$=0.4, $passenger\_weight$=0.6

Immediately, there is a rapid conclusion that can be raised about this simple comparison. The higher headways (consequently, waiting times) are, the lower empty seats are. Naturally, if there are fewer bus appearances, buses tend to become more filled while cruising.

6.3 Optimization of the existing network

Similarly, we tested the methodology using the real route set of Carris as a starting point. Carris network comprises R = 308 routes.

The variations on passenger and operator weights were equal to the above experiments performed. Firstly, by assigning 0.6 to passengers' and 0.4 to operators metrics. On the following test, 0.9 to passengers costs and 0.1 to operator metric.

Search	Objective Function	ATT (min)	AWT (min)	Transfers (%)	Empty Seats	Average Headway (min)
Random	42.085	10.39	7.91	0.485	72	15.9
Local	42.05	10.4	7.94	0.47	72	16.13

Table 6.5: Initial Metric Values - $operator\_weight$=0.4, $passenger\_weight$=0.6

Search	Objective Function	ATT (min)	AWT (min)	Transfers (%)	Empty Seats	Average Headway (min)
Random	41.49 (−1.45%)	10.01	7.58	0.47	72	15.9
Local	41.55 (−1.20%)	10.16	7.6	0.46	72	15.7

Table 6.6: Final Metric Values - $operator\_weight$=0.4, $passenger\_weight$=0.6

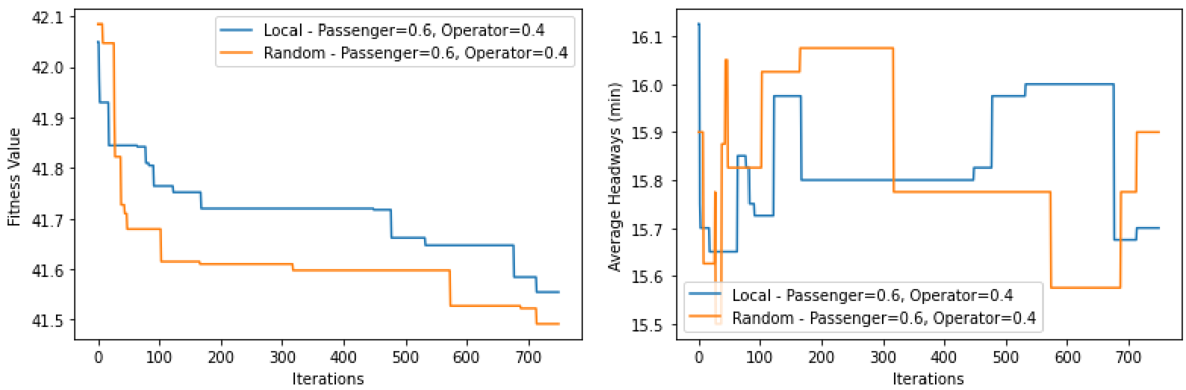

Figure 6.9: Fitness Value and Average Headways - $operator\_weight$=0.4, $passenger\_weight$=0.6

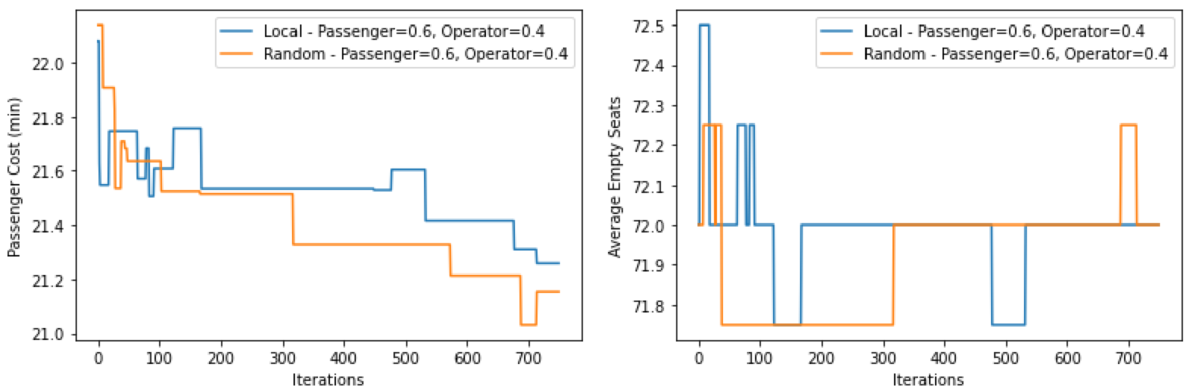

Figure 6.10: Passenger Cost and Average Empty Seats - $operator\_weight$=0.4, $passenger\_weight$=0.6

Search	Objective Function	ATT (min)	AWT (min)	Transfers (%)	Empty Seats	Average Headway (min)
Random	26.91	10.39	7.73	0.49	72.5	15.875
Local	26.78	10.4	7.70	0.47	72.25	15.475

Table 6.7: Initial Metric Values - $operator\_weight$=0.1, $passenger\_weight$=0.9

Search	Objective Function	ATT (min)	AWT (min)	Transfers (%)	Empty Seats	Average Headway (min)
Random	26.6 (−1.17%)	10.36	7.49	0.49	72.75	15.625
Local	26.64 (−0.53%)	10.4	7.57	0.47	72.5	15.475

Table 6.8: Final Metric Values - $operator\_weight$=0.1, $passenger\_weight$=0.9

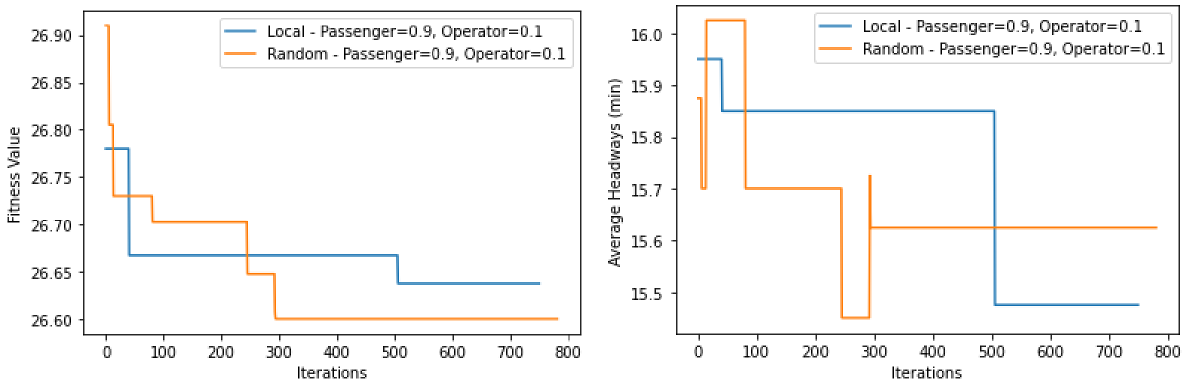

Figure 6.11: Fitness Value and Average Headways - $operator\_weight$=0.1, $passenger\_weight$=0.9

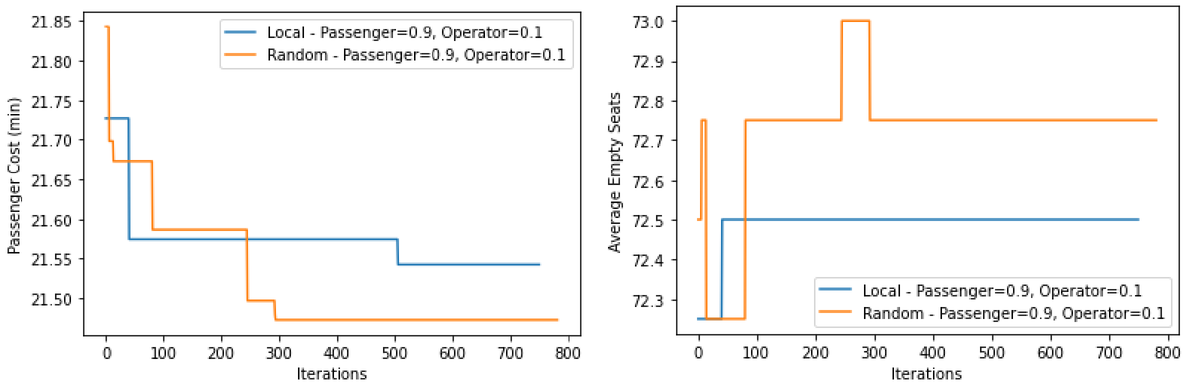

Figure 6.12: Passenger Cost and Average Empty Seats - $operator\_weight$=0.1, $passenger\_weight$=0.9

Similarly to optimization from scratch, Random Search had a better performance than Local Search.

Curiously, the fitness improvement, in equal conditions of weights assignment, was pretty low compared to experiments with R = 150 $(1.17 << 27.35)$. The reason may be the wider range of routes and headways number, which can lead to a more difficult optimization procedure.

6.3.1 Optimized solution in detail

Likewise, as it made on optimization from scratch analysis, we detail best solution matrices.

Using Random Search and $operator\_weight$=0.4, $passenger\_weight$=0.6, which had the best performance (-1.45%), we present in detail metrics for each matrix.

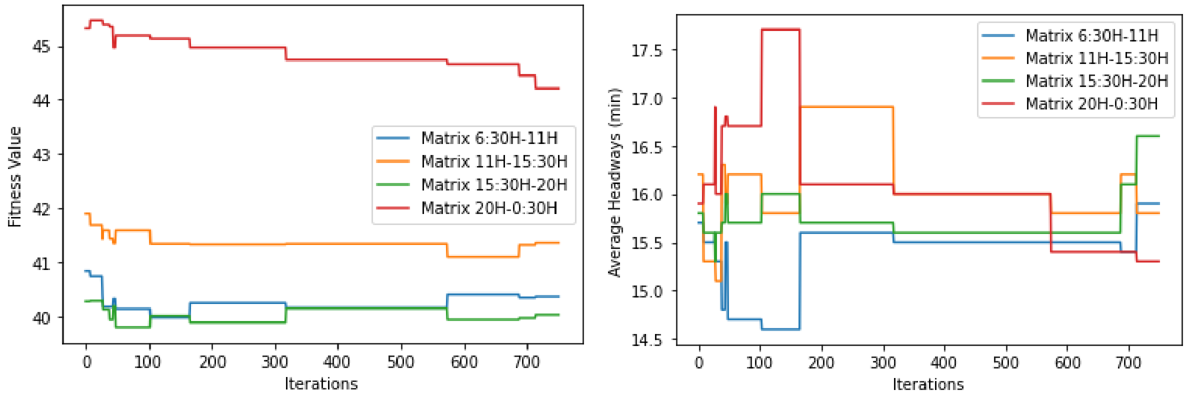

Figure 6.13: Fitness Value and Average Headways - $operator\_weight$=0.4, $passenger\_weight$=0.6

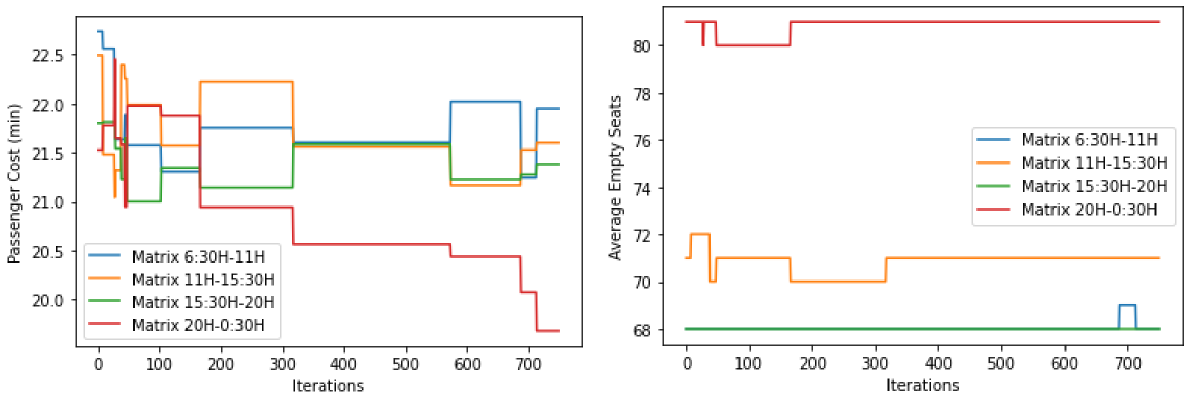

Figure 6.14: Passenger Cost and Average Empty Seats - $operator\_weight$=0.4, $passenger\_weight$=0.6

The two matrices that include peak hours (blue and green lines) had practically similiar results. Moreover, the third matrix (orange) produced a slightly higher fitness value, due to higher average empty seats metric.

Fourth matrix (red) performed the highest number of empty seats and the lowest on passenger costs and headways as well, providing that more bus appearances result in emptier buses.

6.4 Comparison to existing network

As far as real network comparison is concerned, *frequencies.txt* file present on GTFS data, describes trips headways of CARRIS real routes. Thus, we analyzed that file in order to assess our results, using same intervals.

Matrices	Optimized Average Headway (min)	Real Average Headway (min)	Difference (%)
6:30H - 11H	15.9	16.2	-1.8
11H - 15:30H	15.8	17.2	-8.8
15:30H - 20H	16.6	15.5	$+7$
20H - 0:30H	15.3	26	-70

Table 6.9: Comparison between real and computed headways on Random Search - $operator\_weight$=0.4, $passenger\_weight$=0.6

On the first three periods, the computed headway values are somewhat similar to real ones. In the first, practically the same result, and on the second and third, a relevant decrease and increase, respectively.

Notwithstanding, on fourth (night) matrix there is an enormous reduction of 70% on average headway values. On the following section, it is discussed some reason that may explain the above disparity.

6.5 Overall Analysis

We may conclude that Random Search is the best performing method as it minimizes best the ojective function across all tests performed. The possible reason is that by randomizing values, we can rapidly test a large scope of possible solutions, with diverse headway values.

Contrarily, Local Search slightly varies headway values through new generations. As a result of having a large set of routes and its associated headways, Local Search method leads to an homogeneity of headway values tested. Correspondingly, graphs plotted demonstrate the effective reduction on fitness values on Random Search experiments.

As far as metric weights is concerned, the heavier operator weight is, the higher fitness value is, which is somehow obvious, given absolute metric value differences. However, taking a closer look at the graphs, it is demonstrated on all experiments that when there is a certain balance between passenger costs (AWT + ATT + AWT*ATR) and average empty seats (AES). While there is a gradual reduction on passenger costs we can check also a compensation by increasing empty seats.

In perspective, it is consistent. A reduction on waiting time, caused by a decrease on headway (increase on bus appearances) and consequently leads to more empty seats. Plots of fixed max/min headway confirm that. Moreover, it is clearly observed that on matrices that include peak hours (first and third matrices, morning and afternoon, respectively), there are less average empty seats, as there is way more demand. On experiments, which was given a greater importance to operator, resulted in a reduction on empty seats.

Empirically, on our perspective as passengers, every public transportation operator adjusts service level, given demand. On the one hand, during peak periods, there are low route headways (high frequency) and reduced waiting time. On the other hand, during night hours there is an adjustment of the service level, mainly for management reasons. It is not financially feasible to keep high frequencies all day, as demand hugely vary as well.

By stating that, on both solution matrices plots, the previously mentioned does not occur. While there is a reduction on demand, mirrored by average empty seats metric matrices differences, the same adjustment on headways did not take place.

The reason to all this is that our methodology computes the most optimised solution as the most balanced among matrices fitness values (Figure 5.3), hence the difference in headways (also waiting times) is low, which is further balanced by the operator's cost control during the day. So the set of headways that harmonize passenger and operator costs are considered the most appropriate to the optimization process.

Regarding comparison to existing network, it is highly demonstrated that optimized solution is balanced on headway values across all periods. The most striking example of it is the forth matrix case, which resulted in 70% of average headways difference between computed solution and real headways.

Chapter 7

Conclusion

This thesis developed a methodology to tackle Transit Network Design and Frequency Setting Problem simultaneously. The computed method was applied to real transit data, provided by CARRIS, the bus operator of the city of Lisbon. The data sources utilized were GTFS and AFC data.

Firstly, it was inferred trip destinations, given that passengers only validate fare ticket at boarding on Carris Network. 47.7% of destinations were deduced. After that, a method to uniformly allocate uninferred destination trips throughtout routes was employed, with the intent of handle all available data to the forward steps. In the end of this process, we divided the demand in four different matrices that represent periods of a week day: morning, lunch, afternoon and night. These computed matrices were used as input data to approach the problem of the work.

With the resulting matrices, it was described network and road representations, as well as the Genetic Algorithm aspects, namely its operators and objective function to the optimization problem. Furthermore, it was explained a couple of actions about frequency adjustment, such as two different approaches regarding frequency optimization searching of solutions computed by GA: Local Search and Random Search. The results of the conducted experiments proved that Random Search has a better performance than Local method. One of the reasons is that small variations on headways, lead to an homogeneity on values and a small scope of probable values. On contrary, Random Search rapidly tests a wide range of values and finds optimal values.

The optimization process was evaluated taking into account costs from both parties involved in the public transportation activity. On the one hand, it dealt with the amount of time spent by passengers travelling and waiting for the bus. On the other hand, empty seats that represent an unproductive cost to the transit company. The aim is to minimize all associated metrics.

Several experiments were conducted, testing different parameters of metrics weight and search frequency method. Moreover, it was optimized bus network from scratch, as well as using the real bus network as a starting point.

As can be noted on plots, there is few difference among matrices average headways. Curiously, on peak times matrices, the headway is increased to compensate a drop on average empty seats. On the night matrix, the inverse happens. It is reduced average headways and consequently passenger costs, in order to balance the high number of empty seats.

The methodology managed to find the optimized solution with lowest average matrices values, which resulted on matrices metric values quite similar to each other. Besides that, it has not been expressed demand fluctuation on metric values. For instance, waiting time and headways are not increased on night matrix, to compensate demand decrease.

Under these circumstances, we do believe that computed methodology represents a great step on real public transportation planning solution. Not only by how metrics are calculated, but also by how methodology was developed. Providing that, there is more data, namely from transit companies activity, this methodology can preview a wider range of situations and give a closer picture of reality. Perhaps by assigning day periods different weights to optimization process, this work can be a useful tool when tackling Transit Network Desing and Frequency Setting problems.

7.1 Limitations

The main limitation identified was the way operator costs were computed. Empty seats alone are not a good parameter to represent operator costs by itself, it is a rather simple model. Without real information, for instance about fuel costs or fleet maintenance, it became quite difficult to develop a more robust representation. Given available data, we fancy a way to estimate empty seats to symbolize it. Even though, having only access to AFC data, sometimes imprecise and inconclusive, it could not depict a real picture of boardings and alightings still. In our case, average empty seats metric was calculated as the average of mean empty seats of each route.

A more feasible way of calculating it would be desirable, namely taking advantage of Automated Passenger Counters (APC) system, if available.

Another relevant absence of data observed was AVL. From the Transit Network design, it would represent higher destinations and interchanges inferred rates. Moreover, as far as frequency optimization is concerned, AVL data available would mean a more comprehensive notion regarding bus appearances and a more meticulous analysis on waiting time, due to the fact that provides real traffic impact on bus activity.

Ultimately, the lack of matching between route stops on opposite directions forms an additional barrier. Since all of opposite stops are on the other side of the street, providing that there are many one-way streets. Without further information, it becomes quite difficult to give a closer picture of a real network which is composed by symmetric routes.

7.2 Future work

The natural path to this work could be to continue optimization process to the next steps of the public transportation planning, particularly timetabling development and resources allocation, included on tactical and operational choices, respectively.

As mentioned in the subsection of Limitations, if we had at disposal a matching between opposite route stops, a procedure that generates symmetric routes would promote a real bus network representation. In this manner, it would be quite natural to add another metric, fleet allocation to network. By calculating round trips duration this could result on number of buses required to each route.

In addition, other relevant aspect to consider in the future could be assign economic costs to the objective function. Besides fuel costs that are directly impacted by network length, it would favorable to add some metrics, such as salaries or fleet maintenance costs. On revenues, ticket fares revenues would be applied. These data could be kindly provided by the operator to further researches.

Bibliography

[1] R. Loureiro, “Machine Learning Methods for the Optimisation of Urban Mobility,” Master’s thesis, Instituto Superior Técnico, 2020.

[2] O. J. Ibarra-Rojas, F. Delgado, R. Giesen, and J. C. Muñoz, “Planning, operation, and control of bus transport systems: A literature review,” jul 2015.

[3] Z. E. Ahmed, R. A. Saeed, A. Mukherjee, and S. N. Ghorpade, *Energy optimization in low-power wide area networks by using heuristic techniques*. INC, 2020.

[4] J. Shin, T. A. Badgwell, K. H. Liu, and J. H. Lee, “Reinforcement Learning – Overview of recent progress and implications for process control,” *Computers and Chemical Engineering*, vol. 127, pp. 282–294, 2019.

[5] M. D. Toksari, “Ant colony optimization for finding the global minimum,” *Applied Mathematics and Computation*, vol. 176, no. 1, pp. 308–316, 2006.

[6] P. Chakroborty and T. Dwivedi, “Optimal Route Network Design for Transit Systems using Genetic Algorithms,” Tech. Rep. 1, 2002.

[7] S. Saeidi, “A Genetic Algorithm for Route Optimization in Public Transportation Problem,” tech. rep., 2014.

[8] W. Y. Szeto and Y. Wu, “A simultaneous bus route design and frequency setting problem for Tin Shui Wai, Hong Kong,” *European Journal of Operational Research*, vol. 209, no. 2, pp. 141–155, 2011.

[9] B. Yu, Z. Yang, and J. Yao, “Genetic algorithm for bus frequency optimization,” *Journal of Transportation Engineering*, vol. 136, no. 6, pp. 576–583, 2010.

[10] R. O. Arbex and C. B. da Cunha, “Efficient transit network design and frequencies setting multi-objective optimization by alternating objective genetic algorithm,” *Transportation Research Part B: Methodological*, vol. 81, pp. 355–376, 2015.

[11] K. Gkiotsalitis, “Exact optimization of Bus Frequency Settings considering Demand and Trip time variations,” *Presented at 96th Annual Meeting of the Transportation Research Board*, no. January, 2017.

[12] P. Chakroborty, K. Deb, and R. K. Sharma, “Optimal fleet size distribution and scheduling of transit systems using genetic algorithms,” vol. 1060, no. December, 2000.

[13] P. Chakroborty, “Genetic Algorithms for Optimal Urban Transit Network Design,” tech. rep., 2003.

[14] A. Ceder and N. Wilson, “Bus Network Design,” *Bus Network Design*, vol. 1, no. 4, pp. 331–344, 1986.

[15] Sh. Afandizadeh, H. Khaksar, and N. Kalantari, “Bus fleet optimization using genetic algorithm a case study,” 2011.

[16] C. Vanderwaart, “Planning Transit Networks with Origin , Destination , and Interchange Inference by,” no. 2003, 2016.

[17] G. Fattouche, “How to improve high-frequency bus service reliability through scheduling,” *Proceedings of the ITRN*, no. 833, pp. 45–47, 2011.

[18] J. Walker, *Human transit: How clearer thinking about public transit can enrich our communities and our lives*. 01 2012.

[19] L. Fan, C. L. Mumford, and D. Evans, “A simple multi-objective optimization algorithm for the Urban transit routing problem,” *2009 IEEE Congress on Evolutionary Computation, CEC 2009*, pp. 1–7, 2009.

[20] M. Wardman, “A review of British evidence on time and service quality valuations,” *Transportation Research Part E: Logistics and Transportation Review*, vol. 37, no. 2-3, pp. 107–128, 2001.

[21] S. M. H. Mahdavi Moghaddam, K. R. Rao, G. Tiwari, and P. Biyani, “Simultaneous Bus Transit Route Network and Frequency Setting Search Algorithm,” *Journal of Transportation Engineering, Part A: Systems*, vol. 145, no. 4, p. 04019011, 2019.

[22] W. Wang, J. P. Attanucci, and N. H. Wilson, “Bus passenger origin-destination estimation and related analyses using Automated Data Collection Systems,” *Journal of Public Transportation*, vol. 14, no. 4, pp. 131–150, 2011.

[23] J. J. Barry, R. Newhouser, A. Rahbee, and S. Sayeda, “Origin and Destination Estimation in New York City with Automated Fare System Data,” no. 02, pp. 183–187, 2002.

[24] J. B. Gordon, H. N. Koutsopoulos, N. H. Wilson, and J. P. Attanucci, “Automated Inference of Linked Transit Journeys in London using Fare-Transaction and Vehicle Location Data,” *Transportation Research Record*, vol. 2343, pp. 17–24, 2013.

[25] O. Cats and S. Glück, “Frequency and Vehicle Capacity Determination using a Dynamic Transit Assignment Model,” *Transportation Research Record*, vol. 2673, no. 3, pp. 574–585, 2019.

[26] S. Desale, A. Rasool, S. Andhale, and P. Rane, “Heuristic and Meta-Heuristic Algorithms and Their Relevance to the Real World: A Survey,” *International Journal of Computer Engineering in Research Trends*, vol. 351, no. 5, pp. 2349–7084, 2015.

[27] H. Altinbas and G. C. Akkaya, “Improving the performance of statistical learning methods with a combined meta-heuristic for consumer credit risk assessment,” *Risk Management*, vol. 19, no. 4, pp. 255–280, 2017.

[28] I. H. Osman, K. Ct, and J. P. Kelly, “Meta-Heuristics: An Overview,” pp. 1–21, 1996.

[29] M. Gavrilas, “Heuristic and metaheuristic optimization techniques with application to power systems,” *International Conference on Mathematical Methods and Computational Techniques in Electrical Engineering - Proceedings*, no. June, pp. 95–103, 2010.

[30] M. Mohammed, M. B. Khan, and E. B. M. Bashier, *Machine learning : algorithms and applications*. Taylor Francis Group, LLC, 2017.

[31] Marco Dorigo, M. Birattari, and T. Stu¨tzle, "Ant Colony Optimisation," *IEEE Computational Intelligence Magazine*, vol. 780, no. November, pp. 33–42, 2006.

[32] T. Yigit and O. Unsal, "Using the ant colony algorithm for real-time automatic route of school buses," *International Arab Journal of Information Technology*, vol. 13, no. 5, pp. 559–565, 2016.

[33] K. Deb, "Introduction to Genetic Algorithms for Engineering Optimization," vol. 1975, pp. 13–51, 2004.

[34] D. Whitley, "A genetic algorithm tutorial," *Statistics and Computing*, vol. 4, no. 2, pp. 65–85, 1994.

[35] F. A. KIDWAI, B. R. MARWAH, K. DEB, and M. R. KARIM, "A Genetic Algorithm Based Bus Scheduling Model for Transit Network," tech. rep., 2005.

[36] A. Tošić and A. Horvat, "Optimization of traffic networks by using genetic algorithms," Tech. Rep. 4, 2012.

[37] R. COLIN and J. E.ROWE, *Genetic Algorithms: Principles and Perspectives*, vol. 7. 2002.

[38] J. Agrawal and T. V. Mathew, "Transit route network design using Parallel Genetic Algorithm," *Journal of Transportation Engineering*, vol. 129, no. 2, pp. 186–195, 2003.

[39] P. Kumar and A. Khani, "Transit Origin Destination Estimation using Automated Data," tech. rep., 2019.

[40] J. Zhao, A. Rahbee, and N. H. Wilson, "Estimating a rail passenger trip origin-destination matrix using automatic data collection systems," *Computer-Aided Civil and Infrastructure Engineering*, vol. 22, no. 5, pp. 376–387, 2007.

[41] M. A. Munizaga and C. Palma, "Estimation of a disaggregate multimodal public transport Origin-Destination matrix from passive smartcard data from Santiago, Chile," *Transportation Research Part C: Emerging Technologies*, vol. 24, pp. 9–18, 2012.

[42] "Gtfs google." `https://developers.google.com/transit/gtfs/reference`. Accessed: 2020-12-05.

[43] N. Nassir, A. Khani, S. G. Lee, H. Noh, and M. Hickman, "Transit stop-level origin-destination estimation through use of transit schedule and automated data collection system," *Transportation Research Record*, no. 2263, pp. 140–150, 2011.

[44] Y. Chen, "Understanding Bus Passenger Crowding Through Origin Destination Inference," no. 2004, pp. 1–172, 2014.

[45] "Carris." `https://www.carris.pt/en/carris/company/our-numbers/`. Accessed: 2020-12-05.

[46] "Metropolitano de lisboa." `https://www.metrolisboa.pt/institucional/conhecer/metro-em-numeros/`. Accessed: 2020-12-05.

[47] D. Huang, J. Yu, S. Shen, Z. Li, L. Zhao, and C. Gong, "A Method for Bus OD Matrix Estimation Using Multisource Data," *Journal of Advanced Transportation*, vol. 2020, 2020.

[48] A. Cui, "Bus Passenger Origin-Destination Matrix Estimation Using Automated Data Collection Systems," 2006.

[49] C. Seaborn, J. Attanucci, and N. H. Wilson, “Analyzing multimodal public transport journeys in London with smart card fare payment data,” *Transportation Research Record*, no. 2121, pp. 55–62, 2009.

[50] K. K. A. Chu and R. Chapleau, “Enriching archived smart card transaction data for transit demand modeling,” *Transportation Research Record*, no. 2063, pp. 63–72, 2008.

[51] K. Kepaptsoglou and M. Karlaftis, “Transit route network design problem: Review,” *Journal of Transportation Engineering*, vol. 135, no. 8, pp. 491–505, 2009.

[52] B. S. B. Pattnaik, S. Mohan, and V. M. Tom, “URBAN Bus TRANSIT ROUTE NETWORK DESIGN USING GENETIC ALGORITHM,” vol. 3, no. August, pp. 368–375, 1998.

[53] H. Sadrsadat, H. Poorzahedi, A. Haghani, and E. Sharifi, “Bus Network Design Using Genetic Algorithm,” tech. rep.

[54] Suwardo, “The Determination of Bus Service Frequency Using Cost-Benefit Approach,” *Jurnal Transportasi*, vol. 11, no. 1, pp. 29–38, 2011.

[55] H. Zhao, W. Xu, and R. Jiang, “The Memetic algorithm for the optimization of urban transit network,” *Expert Systems with Applications*, vol. 42, no. 7, pp. 3760–3773, 2015.

[56] D. Canca, A. De-Los-Santos, G. Laporte, and J. A. Mesa, “A general rapid network design, line planning and fleet investment integrated model,” *Annals of Operations Research*, vol. 246, no. 1-2, pp. 127–144, 2016.

[57] W. Y. Szeto and Y. Jiang, “Transit route and frequency design : Bi-level modeling and hybrid artificial bee colony algorithm approach,” *TRANSPORTATION RESEARCH PART B*, vol. 67, pp. 235–263, 2014.

[58] Z. Huang, G. Ren, and H. Liu, “Optimizing bus frequencies under uncertain demand: Case study of the transit network in a developing city,” *Mathematical Problems in Engineering*, vol. 2013, no. January, 2013.

[59] A. T. Buba and L. S. Lee, “A differential evolution for simultaneous transit network design and frequency setting problem,” *Expert Systems with Applications*, vol. 106, pp. 277–289, 2018.

[60] F. Zhao and A. Gan, “Optimization of transit network to minimize transfers,” tech. rep., 2003.

[61] G. E. Sánchez-Martínez, “Inference of public transportation trip destinations by using fare transaction and vehicle location data,” *Transportation Research Record*, vol. 2652, pp. 1–7, 2017.

[62] S. B. Jha, J. K. Jha, and M. K. Tiwari, “A multi-objective meta-heuristic approach for transit network design and frequency setting problem in a bus transit system,” *Computers and Industrial Engineering*, vol. 130, pp. 166–186, 2019.

[63] M. A. Nayeem, M. K. Rahman, and M. S. Rahman, “Transit network design by genetic algorithm with elitism,” *Transportation Research Part C: Emerging Technologies*, vol. 46, pp. 30–45, 2014.

[64] “Carris buses capacity.” `https://www.carris.pt/descubra/frota`. Accessed: 2021-10-14.