Universitat de Girona
Escola Politècnica Superior

Incoming Exchange Student - Master Thesis

Erasmus □ **Techno** □ **Other (specify):** ERASMUS

Title of master course: Smart Cities

Title of master thesis: Strolling Dial-a-Ride Problem: Definition and Applicability

Document: Master Thesis

Student (Name & Surname)**:** Ivo Bernardo

EPS Advisor: Esteve del Acebo
Department: Informàtica i Matemàtica Aplicada

Delivered on (month/year): 09/2015

Strolling Dial-a-Ride Problem: Definition and applicability

Ivo Rosa Bernardo

Universitat de Girona

Escola Politècnica Superior

Institut d'Informàtica I Aplicacions

Girona, 2015

Strolling Dial-a-Ride Problem: Definition and applicability

Master Thesis Document

Ivo Rosa Bernardo

TFM Director - **Esteve Del Acebo**

External Specialist - **Albert Trias i Mansilla**

Universitat de Girona

Escola Politècnica Superior

Institut d’Informàtica I Aplicacions

Girona, 2015

Abstract

Small/medium size cities of today, need to present solutions to their citizens, taking into account the demand and their needs. In terms of public transportation, this demand can be satisfied by a new type of service, based on the presented problem of this document: the Strolling Dial-a-Ride Problem.

This problem is a variant of the Dial-a-Ride Problem, in which is expected that the passengers walk to get picked up and to reach their destination point, after they are delivered.

In this document, the Strolling Dial-a-Ride problem is explained and results from a computer simulation, of the problem, are presented and discussed.

Keywords: Strolling Dial-a-Ride Problem, Dial-a-Ride Problem.

Acknowledgements

First of all, I would like to thank the Universitat de Girona and the Escola Politècnica, specially Dra. Maria Luisa Marsal Llacuna and Dr. Josep Lluis de la Rosa for giving me the possibility of studying and learning in the Universitat de Girona.

Secondly, I would like to thank to Dr. Esteve Del Acebo for his guidance during this project and for the opportunity given for studying and testing this problem.

I would also like to thank all the personal of the Agents Research Laboratory, specially Dr. Albert Trias i Mansilla for his help, patience and clear answers, that without them the project would not be complete.

Last, but not least, I would like to thank my family, specially António Bernardo, my father, and Maria Aline Bernardo, my mother, for the strenght and motivation given to me, that were crucial for this stage of my life. I would also like to thank Daniela Rodrigues, for her motivation and advices during the construction of this project.

Index

Index of Figures

Index of Tables

1. Introduction

Cities of today have a main goal that is to improve their capacity to maintain their citizens happy, by becoming smarter.

One of the most important characteristics of a city is the capability of a citizen's mobilization. Although all the characteristics that makes a city, a "Smart City", are important, it is the mobility that dictates the pace of its citizens in the urban areas, by improving their use of time and resources.

Large cities of the world present complex networks of public transportation that respond to the demand made by their citizens, however small/medium size cities might have difficulty on using their public transportation, when trying to satisfy the majority of their citizens.

This document proposes a solution for a new variant of the Dial-a-Ride Problem (DARP), called Strolling Dial-a-Ride Problem (SDARP), where it is explored by simulating the interaction between a number of citizens that want to be transported by a number of vehicles, in a way that it is likely to understand the possibility of applying this solution and its success.

The SDARP is a problem based on the DARP, that is a known problem of optimization, where vehicles transport passengers, from their origin points to their destination points. Each passenger indicates two time intervals, the first one concerns the time of the day that he is available to be picked up by the vehicle and the second indicates when he wants to be at the delivered at its destination point.

The difference between the DARP and the SDARP is the existence of pre-defined pickup and delivery points, where the citizens have to walk to be picked-up and, to reach their destination points, they also have to walk from their delivery points.

2. Definition of the problem

Through the “World Urbanization Prospects” report, produced by the Department of Economic and Social Affairs of the United Nations, 54% of the world population today lives in urban areas and it is expected that this percentage will rise to 66% in 40 years, meaning that 2/3 of the world population in 2050 will live in cities.

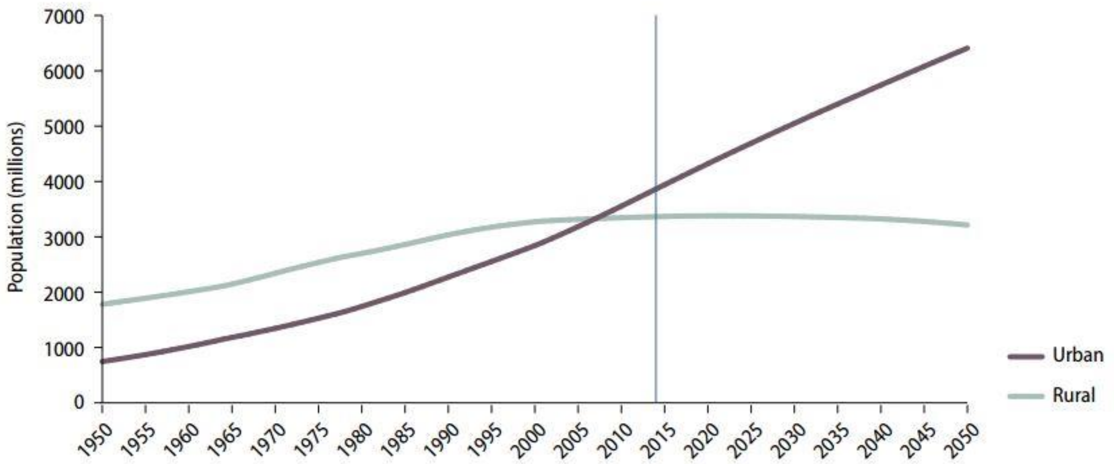

Figure 1 - Urban and rural population of the world, 1950 - 2050, World Urbanization Prospects – (United Nations,2014)

Although this increase is more noticeable in developing countries, the European continent will suffer an increase of population in urban areas. From 1950 to 2010, the european population that lives in cities went from 50% to 75% and the prediction is that in 2050, that percentage will rise to 85% (Caragliu et al. 2011).

Even if the predictions may not have the expected results, the actual percentage of population that lives in urban areas in Europe is very high. Cities need to adapt to this reality and maintain/improve the quality of life of their citizens.

This way, some ideas are being studied and some key-concepts started to rise, like the “Smart City” concept.

The “Smart City” concept may be portrayed as a big organic body, in which every crucial part is connected, in a way that makes a city sustainable and livable (Chourabi, 2012).

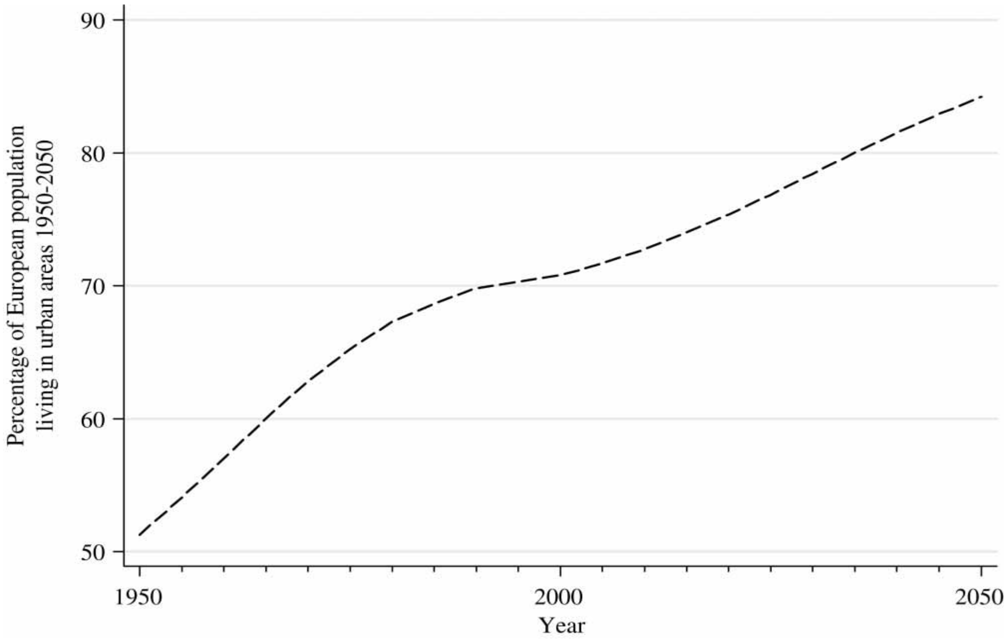

Figure 2 - Percentage of EU population living in urban areas (Caragliu et al. 2011)

The six main characteristics of a “Smart City” are presented in Table I - Characteristics of a "Smart City" and their combination is what makes a city “Smart”.

The “Smart City” concept is used in a very vague way and usually defines cities that are advanced in terms of technology, specially Information Technology. The concept is much more than that, because a city can only be called “Smart” when it uses its resources in a sustainable way and finds solutions that can offer the best possible life conditions for its citizens. The use of IT offers tools that can gather data about the citizens; that help on understanding the social behavior; and that can predict, or simulate, solutions. The best use of these tools helps a city to become more “smarter”.

In terms of physical mobility, a “Smart City” needs to know which solution fits better its citizens, in a general point of view. This solution can be the improvement of an existing infrastructure or the creation of a new one.

As it is described in Table I - Characteristics of a "Smart City", the main characteristics of a “Smart City” are connected and when, for example, if the mobility suffers a substantial improvement, other characteristics are affected in a positive way.

Table I - Characteristics of a "Smart City"

Characteristics	Constituents
Smart Economy	• Innovative spirit • Entrepreneurship • Productivity • International embeddedness
Smart People	• Qualification • Social balance • Creativity • Participation in public affairs
Smart Governance	• Defined political strategies • Transparent Governance • Ability to understand the needs • Power to act
Smart Mobility	• Accessibility • Sustainable public transportation systems • Availability of ICT infrastructure
Smart Environment	• Concern with pollution • Environmental protection • Waste control • Sustainable resource management
Smart Living	• Cultural and education facilities • Health-care • Housing Quality • Social-Cohesion

In terms of economy, a better mobility solution can be one that can offer the same infrastructure, but with a more reasonable price, can be one that offers a better service, with the same global cost or it can be one that improves the mobility, with a cheaper price. The mobility can also influence the social characteristic of a city, because it offers a better quality of life for its users, by being more efficient and more adaptable for their desires. From the environmental point-of-view, the mobility solution can offer conditions, where the citizen understands that it is better to use that solution, instead of others, like its private vehicle.

A study, made by Caragliu et al. (2011), indicates that there is a reciprocal relation between the efficiency of public transportation and high levels of wealth. The authors say that a good public transportation network can help to solve the associated problems with medium/high density urban areas and, at the same time, can reduce the direct, and indirect, costs of traffic jams.

With this point of view, each city needs to understand the movements of its citizens, in a way that is possible to study and apply mobility solutions. In the case of small/medium size cities, it does not make sense the application of colossal solutions for a demand that is lower than a large city. In these cities, the demand is always high and, therefore, the construction of infrastructures that can support that demand makes sense. In medium size cities, the solutions must be adaptable to their reality, in a way that the satisfaction of its citizens is prioritized.

Imagining a small/medium size city with a small/medium size population. Assuming that this small/medium size city is not a suburb of a large city and that the great majority of its citizens lives, and works, there, what are the types of movements inside this city?

Usually, these cities present a very basic public transportations network, where the buses are the main way to commute and their frequency is very low, meaning that the interval of time between the appearance of a bus, in a bus-station, is very high. In regular conditions, the citizens prefer to use different ways to commute, like the private vehicle, instead of waiting too much time for the next bus.

What are consequences of this situation, in a city? Citizens might abandon the usage of this public transportation system and that is a direct consequence, however there are some indirect consequences that can be predicted, like: the increase of the number of private vehicles in the streets of the city; increase of pollution; and the reduction of citizen’s sidewalk, for the construction of parking spaces.

Considering a type of city like the one described above, the following problem appears: is it possible to create/improve the conditions of public transportation, where it is possible to understand the demand during the day, in a dynamic way, on which it is probable to avoid the gradual increase of private vehicle’s traffic?

3. State of the Art

3.1. Dial-a-Ride Service

Cities offer vast options of public transportation, where citizens have various possibilities on choosing how to reach their destination. This offer of public transportation is supported by the complex and organized networks of trains, metros and buses that allow users to choose the options that best suits their needs, in order to arrive at their destinations.

This network of transports satisfies the majority of a city population, however elderly and disabled people show some difficulties in using the conventional transport systems, therefore it was developed an alternative public transportation system that can satisfy these minorities of citizens.

A known alternative is the paratransit service that is a demand responsive service. This service works in a way that the service responds to requests that specify the origin and destination to move to. A very well know paratransit system is the Dial-a-Ride, where passengers call the communications center asking for a trip, where they give information related to it and it is usually a shared public transportation service (Hall et al. 2012).

A service like Dial-a-Ride is one where the routes and schedules are not fixed. The purpose of it is to move a passenger from point A to point B, where the most common use of the service is to provide transportation to elderly and disable people from their homes to hospital or clinics.

In terms of its position in the classes of public transports, the Dial-a-Ride Service appears between the bus and the taxi. This happens because it does not have fixed routes and schedules as the buses usually do, but also for being less flexible than the taxi that normally offers a direct route service, without the need of sharing with other passengers.

Since the service does not have fixed routes and schedules, users have higher flexibility of leaving their origins and arriving to their destinations by the time they desire.

Usually when a passenger communicates with the Dial-a-Ride service, he/she specifies a time window, that is an interval of time, where the first value is the earliest pickup time and the second indicates the latest time to be picked up by a vehicle. This values are extremely important, because with this information the service can understand when a passenger can be picked up and when it is not feasible to pick up the passenger, because if the vehicle arrives after the latest time of the time window, the passenger is no longer available to travel. The same principle is applied for the delivery action.

This way, users have an active role on the service and the purpose of the last is to satisfy the transport requirements specified by passengers.

The Dial-a-Ride Service is supported by a fleet of vehicles that can be homogeneous or heterogeneous, where for example a fleet of heterogeneous vehicles can have different types of seats (Madsen et al. 1995) all mixed in different combinations, while a homogeneous fleet has the same specifications in every vehicle.

Passengers that indicate the service of their desire to travel, communicate with the central dispatch office that is responsible for creating those requests in the system. Within the conditions given by the passengers and the available fleet at the moment, the central dispatch office tries to find the best feasible solution for that user. This office is the core of a service like the Dial-a-Ride (Beaudry et al. 2010). When the Dial-a-Ride service was implemented, a team of people belonging to the central dispatch office, were responsible for the insertion of requests into the routes and schedules of all vehicles. Nowadays, with the capabilities of communication, calculation and geographic positioning technologies, it is possible that a computerized dispatching office works without human intervention, in the routing and scheduling phases, and is capable of performing a similar job with more efficiency (Fu, 2002).

3.2. Autonomous Dial-a-Ride Transit (ADART)

With the advances in technology, a new service of Dial-a-Ride can be created, like it was said in the previous chapter. In 1995, Robert B. Dial described a system similar to the classic Dial-a-Ride, but more sophisticated and with a more commercial approach. This system is the Autonomous Dial-a-Ride Transit (ADART) and, according to the author, the sophistication is associated with the decentralization of the routing and scheduling systems. The ADART has a fully automatic distribution system, in which the dispatching hardware and software are inside the vehicle computer. It is this decentralized system that gives the definition has being autonomous.

A system like ADART should present a service with various vehicles spread out through the entire network and each one of these has its own computer. These computers are responsible to work with the passenger information, in order to understand where they are, where they would like to be and when they can be picked up and delivered.

According to the same author (Dial, 2003) and as it is represented in Figure 3 - Schematic of Communication in ADART (Dial, 2003), the communication is done between three types of agents: the passenger, the on-board computer in each vehicle; and the central computer. The passenger interacts with the system, through the central computer, that ends up communicating with the on-board vehicle.

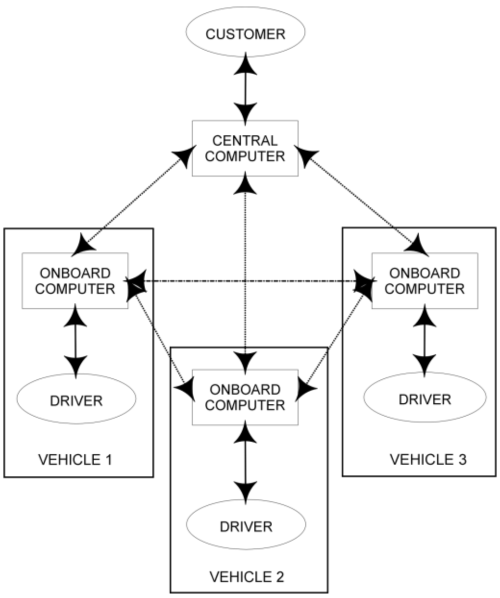

Figure 3 - Schematic of Communication in ADART (Dial, 2003)

After the central computer establishes a communication with the computer of a vehicle, the vehicle's computer calculates the cost of inserting this request (defined as insertion cost) into its schedule. After this, that computer communicates with the other computers to see if any of those have a lesser insertion cost. If so, the vehicle transfers that request to the vehicle with lower value of insertion, otherwise, the vehicle keeps that request to itself (Dial, 1995).

3.3. Dial-a-Ride Problem (DARP)

The Dial-a-Ride Problem (DARP) is a known transport problem about planning routes and schedules for a number of vehicles, existing in a fleet of a Dial-a-Ride Service (Cordeau and Laporte, 2007).

The standard DARP consists on creating feasible solutions for requests made by users, where they specify the locations for pickup and delivery actions (Beaudry et al. 2010).

In this problem there are restrictions associated and one of the most used are the time windows. These are given by the passengers, which indicate the time of the day that they are available for leaving their origin point and, also, the time they are willing to reach their destination. (Hall et al. 2012)

With the time windows feature, a more specific DARP is the Dial-a-Ride Problem with Time Windows (DARPTW), where the service must find a solution for those passengers according to their availability for travelling. This way, the departure and arrival times depend on the preferred pickup and delivery time windows specified by the passengers (Coslovich, 2006).

Another restriction associated to the DARP is the fleet of vehicles. This fleet can be located at a single depot or in multiple depots spread throughout the network and, the composition of the fleet can be homogeneous or heterogeneous (Coslovich et al. 2006).

In practice, the Dial-a-Ride Problem is a challenging problem, because it needs to satisfy two important perspectives: the service and the passengers. The first one wants to minimize the cost of the operation, by using the minimum vehicles to serve all the requests and, also, wants to minimize the total travel distance. From the passenger point of view, the service tries to minimize the travel and waiting time for every passenger, in order to have a better service (Fu and Teply, 1999).

The DARP is a problem of transportation, where passengers make requests with specifications of both location and time. The purpose of the problem is to maximize the success of serving all requests, within a group of restrictions associated with them, while trying to reduce the costs of operation that can be related to the number of used vehicles and distance travelled by them. Also, the DARP is different from a regular pickup and delivery problem, because it needs to evaluate the passenger inconvenience, in terms of travel and waiting time.

The DARP can work in a static way, where the requests are known in advance, before the time of service starts. The problem can also work in a dynamic form, where the requests are made during the time of service (Colorni and Righini, 1999).

Both static and dynamic DARP have exact and heuristic methods developed. Exact methods are methods that try to find the optimal solution for an optimization problem, as for the heuristic methods, its purpose is to find a reasonable solution. The goal of the heuristic is to find a solution in less time than the one required on an exact method, however the solution is not optimal.

In the next chapters, there will be given a synopsis of exact and heuristic methods, for both static and dynamic DARP.

3.4. Dial-a-Ride Problem – Static

One variation of the DARP is the static problem, where the service knows *a priori* the requests made by passengers. The planning and schedules are created before the service starts (Colorni and Righini, 1999).

3.4.1. Exact Methods

Psaraftis (1980) and Desrosiers et al. (1986) developed exact dynamic programming algorithms. The first author considers a constraint of maximal position shift has the quality of service and, the second incorporates the user inconvenience by considering the ride times.

Cordeau (2006) developed a branch and cut algorithm, where a 3-index mixed-integer problem formulation is used. The largest instance has 36 requests.

A similar strategy was used by Ropke et al. (2007), where two branch and cut algorithms are used, where 2-index problem formulations are used. The largest instance has 96 requests.

3.4.2. Heuristic Methods

In 1985, Sexton and Bodin (1985) developed an algorithm, for one vehicle, that solved the problem by iterating the recreation of routes and the attribution of tasks. The sum of two factors, related to the differences between the real and the expected travel time, and the differences between the real and expected time of arrival, belong to an inconvenience function that is minimized, in order to present a feasible solution of a service like Dial-a-Ride.

One of the big steps for developing multi-vehicle algorithms, appeared in 1986, with Jaw et al. (1986). In this model, the passengers have to specify the pick-up and delivery time windows and, also, the maximum travel time. All these factors are evaluated by the algorithm, through an inconvenience objective function related to the passenger and the algorithm chooses the option that adds less value to that function.

When concerning the developments of dynamic programming for a fleet of vehicles, Madsen et al. (1995) published a solution that had a real-life base: the transport of elder and/or disabled people in Copenhagen, that was requested by the Copenhagen Fire-Fighting Service. The authors built a model that was able to plan routes and schedules in a static way, where the objective to choose could be different. However, all the objectives depend on the number of available (heterogeneous) vehicles, the length of travel and some parameters connected with the passengers. The requests have a particular difference with the ones of other static DARP, because the passengers can only specify one time window. Before the algorithm processes the requests, these are ordered in a way that the first one to be inserted in the system, is the one with most difficulty of inserting. Madsen et al. (1995) algorithm can work in a dynamic environment and it was tested in real situations.

Toth and Vigo (1996) developed a heuristic for a real problem in the city of Bologne. The heuristic developed by the authors inserts the new requests of transport in all the current routes and, after the evaluation of the cost of inserting them in every route, chooses the one with less impact in the total operation cost.

Cordeau and Laporte (2003) created a heuristic that uses tabu search, in order to find a feasible solution. The same specifications of time windows are required in this model. The restrictions of this heuristic are related to the capacity of the vehicles, the maximum travel length and the maximum travel time, for each request. The search algorithm removes and shuffles the transportation request in the exiting routes, in order to choose the solution with better results. The objective of this solution is to minimize the total length of the service.

Another kind of objective could be the minimization of the number of used vehicles, as described in Rekiek et al. (2006). In the first phase of the algorithm, where passengers are aggregated to a determined common route, it is applied a genetic algorithm. Following that, it is implemented the insertion algorithm, that belongs to the second phase. The software developed by the authors provided quality solutions in a run time of 30 minutes. The results of the tests were made with random instances and with datasets from the city of Bruxels, however the authors only specify the run time of the software and do not give information related to the travel distance and time of each vehicle.

In the same year, Xiang et al. (2006) studied the problem, where the objective was to minimize the combination of the following aspects: fixed and variable costs of the vehicles; cost of the driver; waiting time; and active time. The search mechanism of this model uses a secondary objective function, focused on inactive time of the vehicles, in other words, the idle times.

Wolfler Calvo and Colorni (2006) studied the Dial-a-Ride Problem with a fixed fleet of vehicle. In the heuristic developed by the authors, an objective function is used, where the algorithm tries to serve the maximum number of users and tries to minimize theirs inconvenience.

This is evaluated by the waiting time and the excess of traveling time. The heuristic creates a number of main routes and, by solving an assignment problem, builds sub tours that could be added to those routes, in the routing phase.

Table II - List of restrictions in heuristic methods for Static DARP

Reference	Restrictions
Sexton and Bodin (1985)	Time Windows Vehicle Capacity
Jaw et al. (1986)	Time Windows Vehicle Capacity Relation between minimum and actual ride time
Madsen et al. (1995)	Time Windows Heterogeneous Fleet Vehicle capacity Maximum route duration Maximum deviation between minimum time and actual time
Toth and Vigo (1996)	Time Windows Vehicle Capacity Maximum ride time
Cordeau and Laporte (2003)	Time Windows Vehicle capacity Maximum route duration Maximum ride time
Rekiek et al. (2006)	Time Windows Vehicle capacity
Xiang et al. (2006)	Time Windows Different types of vehicles Vehicle Capacity Maximum route duration Driver breaks Driver working time
Wolfler Calvo and Colorni (2006)	Time Windows Vehicle Capacity

The tests of this model were conducted in instances where the number of customers was between 10 and 180 and when compared with a simple insertion algorithm, the heuristic created by the authors performs better in terms of unserved customers and level of service

3.5. Dial-a-Ride Problem – Dynamic

Another variation of the Dial-a-Ride Problem is the dynamic DARP, where requests are made during the time of service.

The problem replicates a central dispatch office that has to assign the new requests into a vehicle route and schedule, in real time and, contrary to the static Dial-a-Ride Problem, the dynamic system is more flexible, because it can react to unexpected inconveniences, like traffic jams or vehicle's breakdowns. This means that the problem tries to find solutions that re-optimize routes and schedules during the time of service (Colorni and Righini, 1999; Beaudry et al, 2010).

3.5.1. Exact Methods

When dynamic requests are taken into account, it is very difficult to consider a solution as "optimal", because the time that is necessary to find a solution like that is way too large, when compared to the time needed to make a decision. Therefore exact methods were not been specially developed.

Psaraftis (1980) designed a dynamic algorithm for one vehicle. This algorithm was responsible for planning a new solution, concerning the route and the schedule. It is an adaptation of the static case, developed by the same author.

3.5.2. Dynamic Methods

According to Teodorovic and Radivojevic (2000), another way to find a solution to the dynamic DARP is by using fuzzy logic. The authors created two approximate reasoning algorithms, where the first one is responsible for choosing which vehicle is the most appropriated for a request and the second is charged with re-routing and scheduling. The first algorithm makes the decision after evaluating the extra travelled distance of the vehicle and its waiting time, while the second algorithm, weights the extra travelled distance and travelled time for each passenger inside the vehicle. These evaluations are performed through rules that exist in the algorithm, where the combination of the values of **small, medium or large**, reports if the preference for that request is **very weak, weak, medium, strong or very strong**. The system of the authors was tested in artificial conditions with 900 requests.

Attanasio et al. (2004) developed a parallel algorithm that, at the beginning of planning, creates a static route, based on the known requests. When a new transport request occurs, the various parallel threads insert this request in its routes, randomly, and a tabu search is used to find a feasible solution. After this, the next phase is applied, where the requests are re-optimized. Tests were performed with various instances, with a maximum number of requests of 144.

In 2006, Coslovich et al. studied a new variant of the dynamic Dial-a-Ride Problem with Time Windows. This variant has the characteristic of inserting unexpected requests that occur during the time of service. These requests appear when the vehicle is picking-up or delivering a passenger and the unexpected passengers ask if it is possible to be transported to theirs destination. The authors created a two-phase algorithm, where the first one works off-line and between the two stops that vehicle is travelling. When the vehicle is between these stops, the algorithm creates a neighborhood with all the possible detours, respecting the temporal restrictions imposed by the passengers inside the vehicle. In the second phase, the algorithm tries to insert the unexpected passengers in the current routes of all the vehicles. The results present a bad performance, when the number of unexpected passengers increases, however in some cases the algorithm is capable of finding a static solution for the dynamic arrival of passengers. In some cases, the algorithm was capable of inserting 92% of unexpected passengers in the current routes.

Table III - List of restrictions in heuristic methods for Dynamic DARP

Reference	Restrictions
Teodorovic and Radivojevic (2000)	Time Windows Vehicle Capacity
Attanasio et al. (2004)	Time Windows Maximum ride time Vehicle Capacity
Coslovich et al. (2006)	Time Windows Vehicle capacity Maximum route duration
Xiang et al. (2008)	Time Windows Heterogeneous Fleet Maximum trip duration Driver breaks Maximum working time
Beaudry et al. (2010)	Time Windows Heterogeneous Fleet Vehicle Capacity Maximum ride time

A different solution found for the dynamic DARP was studied by Xiang et al. (2008). In the model, the authors considered stochastic events, such has the breakdown of vehicles, the nonappearance of customers, traffic, etc. The insertion of new requests is made by local search that performs on all the current routes, meaning that each request is inserted in all routes, until the best possible solution is found. The diversification strategy was implemented to the local search, where a secondary function is utilized, with the objective of diminishing the lost cost, meaning the cost associated with the non-moving state of the vehicle. The algorithm was tested in various instances, with a maximum number of 610 requests.

Another two-phase strategy was conducted by Beaudry et al. (2010) when studying the transportation of patients to hospitals. In this dynamic problem some restrictions are analyzed, such has the different types of emergency and available transportation. In the first phase, the requests are inserted in the current routes, taking into account the spatial and temporal proximity between them and the requests already processed by the algorithm. After that, the second phase uses tabu search to improve the solutions already found, moving the already processed requests from a route to another, with the objective of finding a better solution. This model was tested with real data from a hospital in Germany, in which was possible to reduce the waiting time of the passengers, with less vehicles being used.

4. Description of the Innovation

The demand for transportation increases, when the solution presented by a city interests a number of citizens. It is the responsibility of a city to understand which solutions fits better the satisfaction of its citizens.

The question asked at the end of the chapter 2 might have a solution the Strolling Dial-a-Ride Service.

This service is based on the Dial-a-Ride Service, described in 3.1 and also in the Autonomous Dial-a-Ride Transit explained in the 3.2, where it takes advantages of the type of service that the Dial-a-Ride Service gives, by going to pick and deliver the passenger in more preferable points and, can use the principles of the Autonomous Dial-a-Ride Transit, for making the insertion of requests automatically, without the need of a central dispatch center, and also for being more commercial.

The main goal of the Strolling Dial-a-Ride service is to transport passengers from the pre-defined points of the city, called Action Points (AP). These AP can be pickup and delivery points as the action performed by the vehicle is a pickup or a delivery, respectively. In this service, passengers are expected to walk from their origin point to their pickup-point and, also, have to walk from their delivery point to their destination point.

When a passenger makes a request to this service, it is mandatory that he gives two time-windows for both pickup and delivery. The passenger also needs to give information related to the maximum walking distance that he is willing to do.

In order to reproduce this service, a problem was created, called the Strolling Dial-a-Ride Problem, where it was conducted a computerized simulation. In this simulation, there a network of AP, where the distance between them is equal and the problem approaches a dynamic point-of-view, meaning that the requests made by passengers occur during the time of service. The simulation will have to find the best possible solution, through a heuristic, for both service and passenger.

A service like this presents a theoretical advantadge, when compared with the Dial-a-Ride Problem: the value of traveled distance is going to be shorter. Since the vehicles do not need to pickup, and deliver, the passengers in their origin and destination points, respectively, the traveled distances are expected to be shorter and, with that, the direct costs related to energy consuption are expected to be shorter, too.

5. Concept Proof

One of the most important tools for science is simulation. Simulation is the act of producing a framework, where the same conditions of the problem, or experiment, are applied in order to obtain results. With the use of simulations, one can test the same theory numerous times, in order to get enough data to understand its results or, if the objective is very clear, one can understand which are the conditions of the simulation in order to get the results that support that objective.

The simulator used in this study has the goal of representing the interaction between a set of homogeneous vehicles, that belong to a Strolling Dial-a-Ride fleet, and passengers that want to be transported. The scenario of this simulation has a set of Action Points (AP) that are spread through the simulation scenario. This scenario is a window with dimensions of N x M and every AP is equidistant from each other. This AP are interaction points between the passengers and the vehicles, meaning that it is here that the vehicles stop to pickup or/and deliver the passengers and the passengers are expected to walk to and from.

The passengers that make the requests at a given time of the day, indicate their origin point and their destination point. The time availability of being transported is represented by what is considered a time-window. For each request, there are two time-windows, one for the pickup action Pickup [MinTime;MaxTime] and another for the delivery action Delivery [MinTime; MaxTime].

The passengers also need to specify the maximum distance that they are willing to walk from their origin point to their pickup point and from their delivery point to their destination point.

The main goal of the simulator is to study the results of a heuristic algorithm that can be able to create a feasible solution. This algorithm simulates an auction between all the vehicles of the existing fleet, where every vehicle indicates the cost of adding this request to its task list (pickup and delivery tasks), if possible. After this, the one with the less costly insertion is the one chosen to pickup and deliver the passenger(s). The algorithm uses a cost function that determinates the cost of inserting a request, where the total traveled distance from the vehicles, the waiting time and walked distance of each passenger are taken into account. For each one of these variables, there is a factor associated, that

With this, the simulation can be simplified in the following way:

- the map of the simulation has the size of N x M;
- a set of AP = { ap0, ap1, …, apT}, with $T = \left(\frac{N}{ACdirect}\right) \times \left(\frac{M}{ACdirect}\right)$;
- APdirect is the distance between the Action Points;
- the fleet of the service is composed by a set of vehicles V={v0, v1, …, vN};
  N – latest number of the set V
- a set of passengers, P={p0, p1, …, pN};
  N – latest number of the set P
- a set of requests, R={r0, r1, …, rN}.
  N – latest number of the set R

Each vehicle (vk) has the following characteristics:

- Ck: capacity of the vehicle;
- Sk: speed of the vehicle (constant and the same for every vehicle).

Each passenger (pk) has the following characteristics:

- Sk: speed of the passenger (constant and different for the type of passenger).

Each request (rk) is made by a passenger, belonging to P. It has the following characteristics:

- (x0,y0) – coordenates of the origin point. (0,0) ≤ (x0,y0) ≤ (N,M);
- (xf,yf) – coordenates of the destination point. (0,0) ≤ (xf,yf) ≤ (N,M);
- Pickup[MinTime,MaxTime] – values of the pickup time-window. Pickup MaxTime > Pickup MinTime;
- Delivery[MinTime,MaxTime] – values of the delivery time-window. Delivery MaxTime > Delivery MinTime;
- The earliest delivery time needs to be bigger than the latest pickup time. (Delivery MinTime > Pickup MaxTime);

- N – number of passengers of rk;
- xmax – maximum distance that the passenger(s) is willing to walk.

5.1. Strolling Dial-a-Ride Simulator

The simulator for the Strooling Dial-a-Ride Problem was made using the language C#. This language is usually used for Object Oriented Programming and it was used for simulate the movement and interaction between vehicle, requests and passengers. The simulator has a graphical feature, that allows the user to watch the simulation.

5.1.1. Classes of the Simulator

The simulator has six classes that are used to create objects that are crucial for the simulation.

The MovingObject class is a very important class, because it has atributes and a method that are inherinted by the Vehicle class and Passenger class. The MovingObject class defines the atributes of identification (ID), the actual position (x and y), the desired position (x1 and y1), the speed of each object (speed) and the distance performed by each object (MovingDistance). This class also defined the method Move() that is responsible by changing the position of each Vehicle and Passenger, during the course of the simulation.

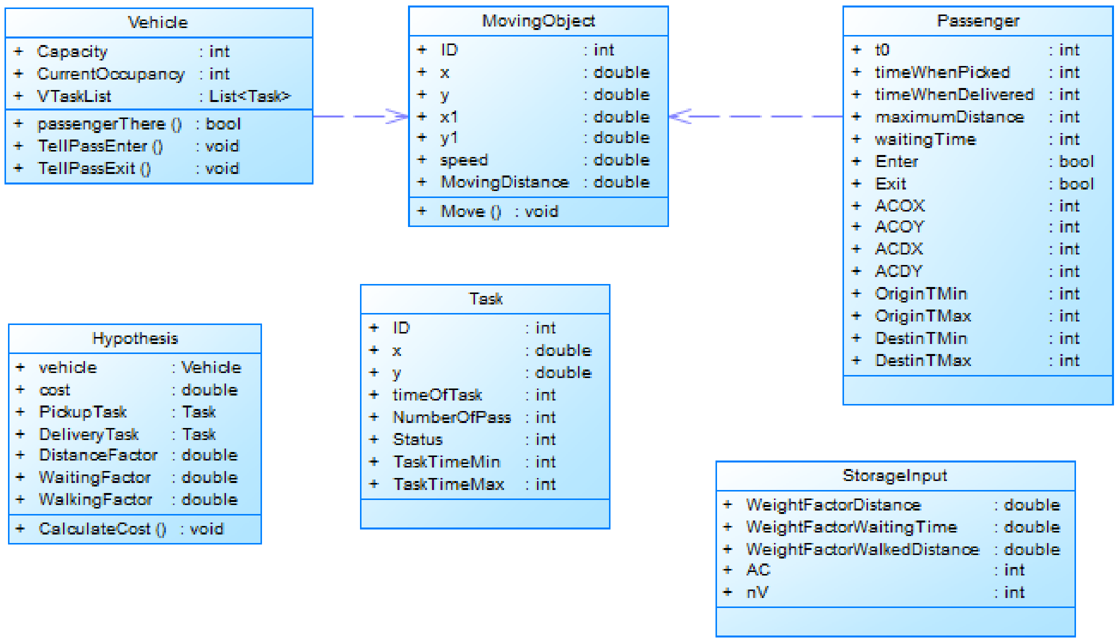

Figure 4 - Strolling Dial-a-Ride Simulator Class Diagram

The Vehicle class has three specific atributes: Capacity; CurrentOccupancy; and VTaskList. The first one is responsible for defining the amount of seats that a Vehicle has; the second atribute stores the value of the number of Passengers that are inside the Vehicle; and the last one is a list of Task.

The class Task is one that is used for creating the two objectives for each request: pickup and delivery. In this simulator, each Task is a pickup, or a delivery, and it is defined by the Status. Each Task needs the ID of the Passenger, the position (x,y) in order to know where this task is performed and the minimum and maximum times of the time-window, in order to know the availability of the Task.

The StorageInput class is used to save the values of the weight factors, the distances between each Action Point (AP) and the number of vehicles present in the simulation (nV).

The class Hypothesis is the class responsible for creating the possibility of inserting a request in a vehicle. It needs information related to vehicle, the pickup task and the delivery task. Uses the weight factors to calculate the cost of inserting this request into a vehicle.

5.1.2. Structure of the Simulator

The general structure of the simulator is showed in the Figure 5 - General Structure of the Strolling Dial-a-Ride Simulator, where in the **Initialize ACs()**, the simulator calculates the number of Action Points needed for the simulation, taking into account the distance between them, and determinates the position of each Action Point. In the **Initialize Vehicles()**, the simulator creates the vehicles with all its characteristics, such has its original position in the network, its ID and initializes its list of tasks.

The **Initiliaze Passengers()** creates the passengers based upon the input made by the user and fills all the needed characteristics, such has their origin and destination point, their maximum allowed distance to walk in the network and their pickup and delivery time-windows.

```
Initialize ACs();
Initialize Vehicles();
Initialize Passengers();
Initialize Simulation();
{
        While(t<= tmax)
        {

                SimulationStep();
                Draw Vehicles();
                Draw Passengers();
        }
}
```

Figure 5 - General Structure of the Strolling Dial-a-Ride Simulator

In the **Initialize Simulation()**, the whole process of simulation starts. While **t** (step of simulation) is less or equal than the tmax (maximum time of simulation) the following steps are made.

In the first one, **SimulationStep()**, the simulator begins to check if at the current **t**, there new requests to be processed. If there are, the algorithm of the simulator checks if in any vehicle it is possible to insert that request in the vehicle task list. This verification is based upon the capacity of the vehicle between the tasks, if the insertion of the a new pickup and delivery tasks influence badly the already processed requests, meaning that if these new tasks can delay the other tasks that have been already processed. This algorithm acts like an auction between the vehicles present in the network, which means that each vehicle communicates to every vehicle if it is possible to insert that request with a certain cost. This cost is associated with a cost function, that is a function that determines the cost of accepting a request and its described in the chapter 5.2. Cost Function.

With this simulated communication process, the vehicle with the least insertion cost is the one chosen to insert the request tasks into its task list, in order to be able to make them.

Also, the simulator gives back the information needed for the passenger to know where the pickup and delivery will be performed .

```
SimulationStep
{
        Check New Requests();
        Process New Requests()
        {
                if(New requests are feasible)
                {
                        Add pickup and delivery tasks
                        to the chosen vehicle;

                        Inform passenger of Pick and
                        Delivery Points;
                }
        }

        Calculate movement of the vehicles();
        Calculate movement of the passengers();

}
```

Figure 6 - Simulation Step structure of the Strolling Dial-a-Ride Simulator

After the process of new requests is complete, if there are new requests, the simulation continues by calculating the movement of the vehicles and the movement of the passengers.

When the simulation step is over at the current **t**, the simulator draws both passengers and vehicles, based upon the calculated movement of each one.

5.2. Cost Function

The purpose of the cost function utilized in this problem and simulation, is to choose the best insertion possibility, among all the possible insertions. This function calculates a cost, associated with two values for distance and a value of waiting time.

$$cost\ function = DF \times DetourDist + WT\ \times Wait\ T \times VSpeed + WalkDist \times WD$$

DF – Distance factor;

DetourDist – Vehicle Detour Distance;

WT – Waiting Time Factor;

Wait T – Waiting time of the passenger;

VSpeed – Vehicle speed;

WalkDist – Passenger walking distance;

WD – Walking distance Factor.

The first value of distance is based upon the detour that the vehicle needs to perform in order to make the pickup and delivery tasks. This detour is the addition that is given to the vehicle trip, meaning that at the current position of the new tasks in task list, the difference between the previous task and the next task, with the new task between them and without the new task, is the addition. This difference is the addition given to trip and it is this value that is atributed to the variable detour.

The second value of distance is associated with the total distance walked by the passenger between its original position and its pickup point and, also, between its delivery point and its destination point.

The value of waiting time is the time between the time of pickup and the first value of the pickup task time window, meaning that the waiting time is difference between the predictable time to be picked up and the minimum time to be picked up.

The three factors associated with the vehicle detour distance and the passenger waiting time, and walking distance, are values to manipulate between simulations. With these three factors is possible to see the effects of each one of the parcels. The values of the factors can be between 0 and 1, being the first the value where the parcel is complete dispized and the second value is where the parcel is fully prioritized.

This cost function was created in order to get the best insertion solution among all the possible ones, and not to give a real value of cost. If we took in consideration the real value of the cost, each distance and waiting time should be multiplied by its price. However, since this is a simulation of a new variant of the problem and not a simulation of a real life case, this values were not taken into account.

If the simulator just considered the three factors and the values of detour distance, waiting time and distance walked by each passenger, the value of waiting time would be irrelevant for the value of cost. This happens because, there is not a price for distance or for time, by the reasons explained before. If there were prices associated for each value, the cost function would be more balanced, however since it is not a real life case, in order for the waiting time to be relevant to the cost function, this waiting time is converted into the same unit of the distances. By multiplying a speed with a time, this one can be converted into a distance. The reason why it was chosen the vehicle speed it is to give the sensibility of inconvenience to the cost function of each waiting time.

5.3. Scale and relevant values of the Model

Since this simulator is able to create the interaction between the passengers of a small/medium with a fleet of vehicles, the size of the city is one of the most important features and the size was considered a small/medium size city.

The size of this city is considered 6km x 6km, representing in pixels, 700pixels x 700 pixels. This assumption is basis for all the future values of speed and distances.

In this model, the time of service was considered a usual time of service for a classical bus transportation system. This kind of system begins at 06:00 and ends at 21:00. Since the service acts during 15 hours and each step of the simulator represents one minute, the maximum time for simulation is 900 steps.

In terms of vehicle characteristics, the average speed considered is the same of a public urban transportation system, that is about 25 km/h. This speed takes into account the flutuation of traffic during the day and the time need to pickup and delivery passengers at the stops. In this model, this speed, when converted into the simulator units, it is about 50 pixels/step. The capacity of each vehicle is equal to 9 seats.

The average walking speed of the human being is 5 km/h and in this model the passenger has a converted speed of 10 pixels/step.

The Action Points were considered to have a distance between of 200 meters and in the present model the value is aproximately 25 pixels.

The capacity of each vehicle is equal to 9 seats, which is about the same amount of seats that a classic Dial-a-Ride system has.

5.4. Limitations for the requests

The generation of requests is done randomly in order get the very different types of trips, such has medium and large distance trips, at the scale of the city. Since the application of the Strolling Dial-a-Ride is for a small city, small distance trips were dispized. The small trips are considered trips that a passenger can walk in less than 10 minutes, therefore the need for paying the transportation for this amount of walking time is unecessary.

The generation of requests also took into account the possibility of not creating trips that are impossible to make, given the time-windows imposed by the passengers. For example, if a passenger gives a very narrow space of time between each time-window and the direct time between its origin and destination point is bigger, the request is rejected.

In order to get relevant results for simulation, these two limitations were considered for the generation of requests.

There are also two other limitations, but these are associated with the maximum walking distance for each passenger. This distance is equal to the sum of distances between the origin point and the pickup point, and between the delivery point and the destination point. The first limitation represents the minimum walking distance between the origin point and the closest Action Point (for pickup), and the distance between the destination point and the closest Action Point (for delivery). The second limitation is equal to what was considered the maximum confortable distance, that is equal to 15 minutes of walking. This distance is equal to 146 pixels.

6. Results

The results of the Strolling Dial-a-Ride simulation were obtained utilizing 30 passenger samples and a variable number of vehicles between 8 and 10. Each sample of passengers were created on Microsoft Excel taking into account the limitations explained in the previous chapter and adding an important characteristic of any transportation problem: rush hours. For these 30 samples, two rush hours were considered in the problem: one in the morning between 7:00 and 9:00; and another in the afternoon between 16:30 and 18:30. Between these hours, 65% of the passengers made requests for travelling and 35% of the requests occur during the course of the day. The vehicles are spread through the entire network and are all active during the simulation time. For all the simulations, breaks and driver’s changing were not considered.

The considered results of the Strolling Dial-a-Ride simulator were:

- **Total Distance – Vehicle [pixels]**: total distance traveled by all the considered vehicles;
- **Waiting Time – Passenger [simulation step]**: the average waiting time of all the inserted passengers;
- **Distance Walked – Passenger [pixels]**: the average walked distance traveled by all the inserted passengers.

6.1. Results of the simulations with all factors prioritized

The first situation took into account the prioritization of all the factors, meaning that the Travel Distance Factor, the Walking Distance Factor and the Waiting Time Factor were equal to 1. This means that the algorithm searched for a solution that were equally fair for both service and passenger.

Table IV - Results of the Simulations with all factors prioritized

	Distance Factor = 1	Waiting Time Factor = 1	Walking Distance Factor = 1
Number of Vehicles	Total distance - Vehicle	Waiting Time - Passenger	Distance Walked - Passenger
8	15765	3,45	69,83
9	14116	2,61	69,08
10	12885	2,18	68,62

When analysing the results, the total distance traveled by all the vehicles decreases when the number of vehicles, present in the simulation, increases.

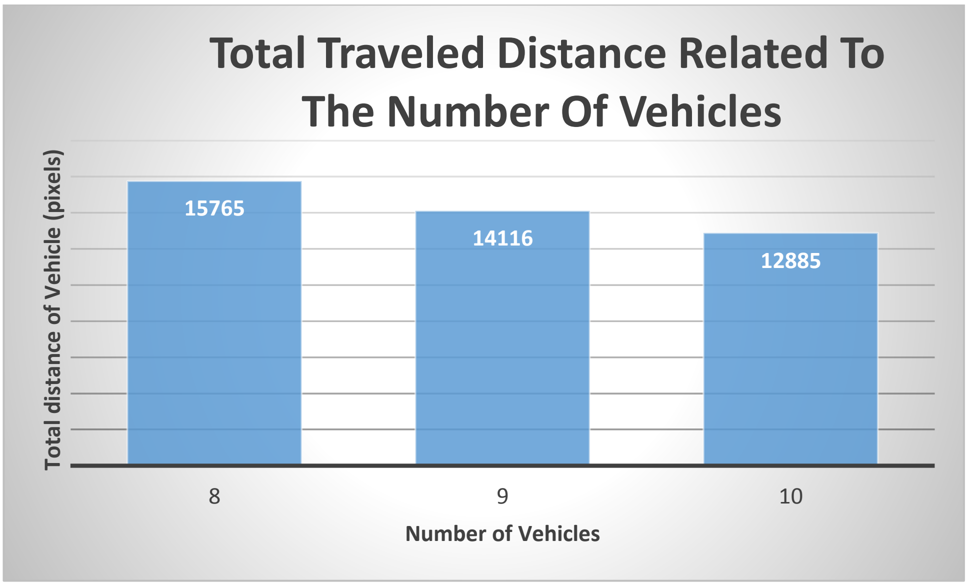

Figure 7 - Graphics of the Result of Total Traveled Distance Related to the Number of Vehicles

This is the consequence of having more options in the simulation area, that allows a vehicle not to travel too much, when there is one available that is closer to the pick up or delivery task.

In terms of waiting time, the values also decrease when the number of vehicles increases. The reason associated with this behaviour is that the increasing number of vehicles present in the network help the waiting time to be smaller.

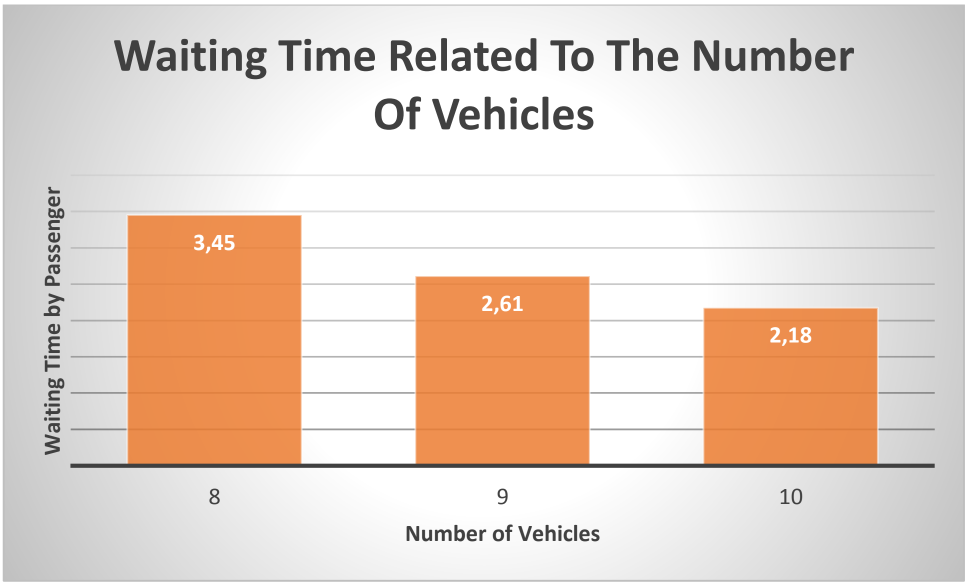

Figure 8 - Graphic Results of the Waiting Time Related to the Number of Vehicles

Since the simulator simulates a minute, for each step, the values of waiting time are favorable for a good quality of service.

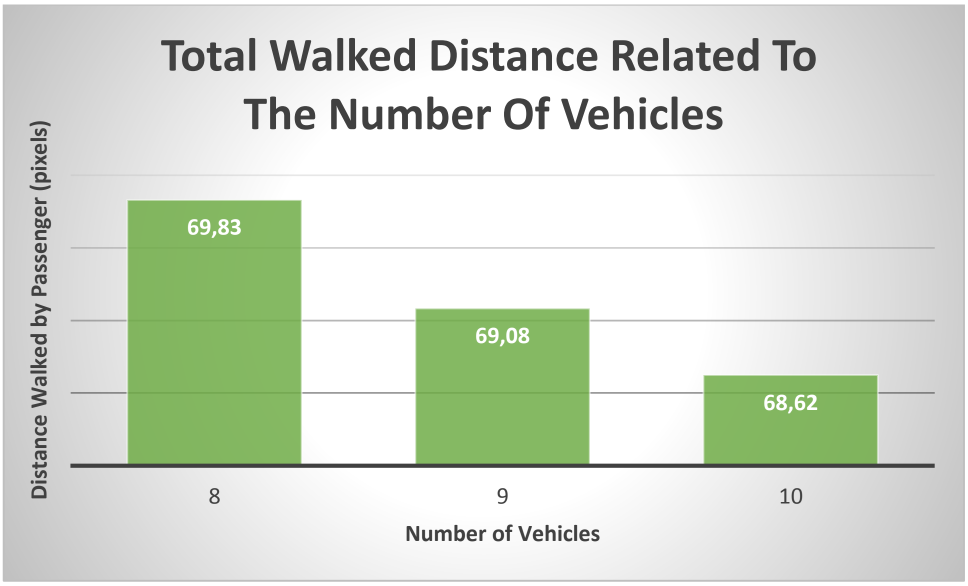

Figure 9 - Graphic Results of the Total Walked Distance Related to the Number Of Vehicles

The same data behaviour can be seen in the walking distance traveled by the passengers, that is about 69 pixels. This distance, when converted to metric measures, is the same as 590 meters, that is aproximately 7 minutes walking. This result gives a satisfactory quality of service, in terms of walking.

6.2. Results of the simulations with all vehicles total traveled distance prioritized

The second situation took into account the prioritization of the total traveled distance factor, meaning that the Travel Distance Factor is equal to 1 and the Walking Distance Factor and the Waiting Time Factor were equal to 0. This means that the algorithm searched for a solution that was better for the service and worse for the passengers.

Table V - Results of the Simulations with all vehicles Total Traveled Distance Prioritized

	Distance Factor = 1	Waiting Time Factor = 0	Walking Distance Factor = 0
Number of Vehicles	Total distance - Vehicle	Waiting Time - Passenger	Distance Walked - Passenger
8	14929	7,73	83,54
9	13137	6,78	83,73
10	11775	6,34	83,06

In this situation, the total traveled distance performed by the vehicles decreases, when the number of vehicles increases. The reason associated is the same of the previous situation.

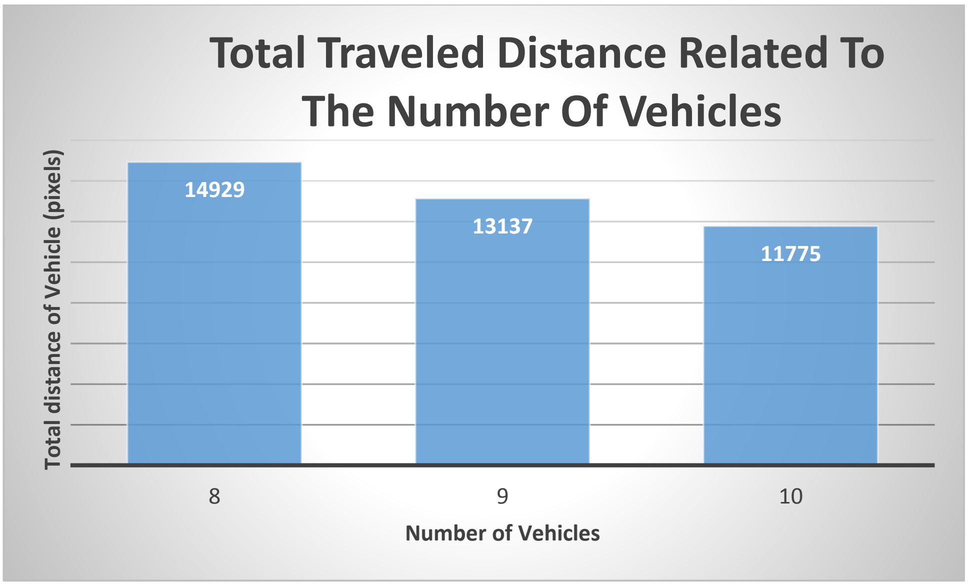

Figure 10 - Graphic Results of the Total Traveled Distance Related to the Number of Vehicles

The same behaviour appears for the waiting time, that it is between 7,73 and 6,34 minutes. With the increasing of vehicles in the network, the waiting time tends do diminish, which is what is expected, however the quality of service, concerning this characteristic is not good, but satisfactory.

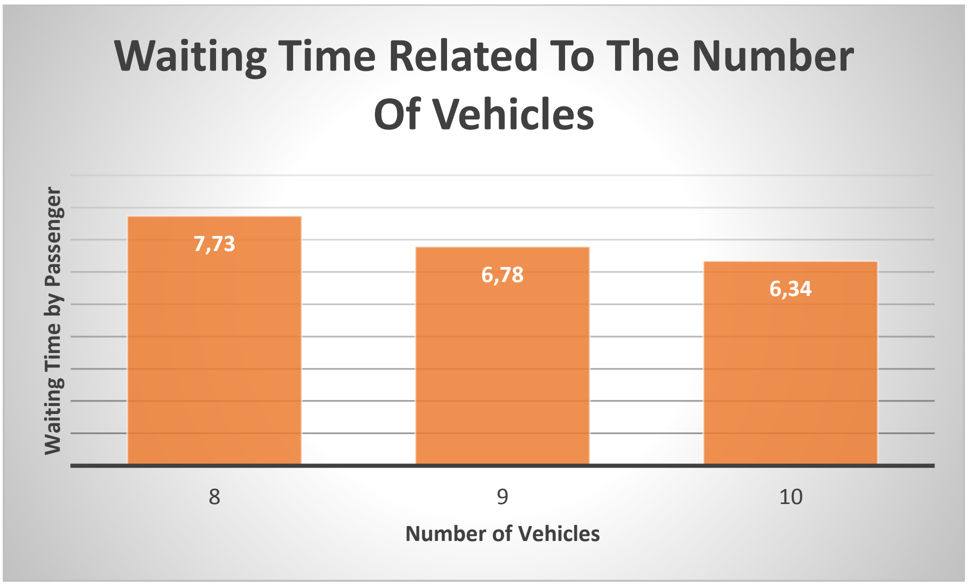

Figure 11 - Graphic Results of the Waiting Time Related to the Number of Vehicles

When the values of walked distance for the passengers is evaluated, the data behaviour is different. In this situation, the walked distance for 9 vehicles is higher than the with 8 vehicles.

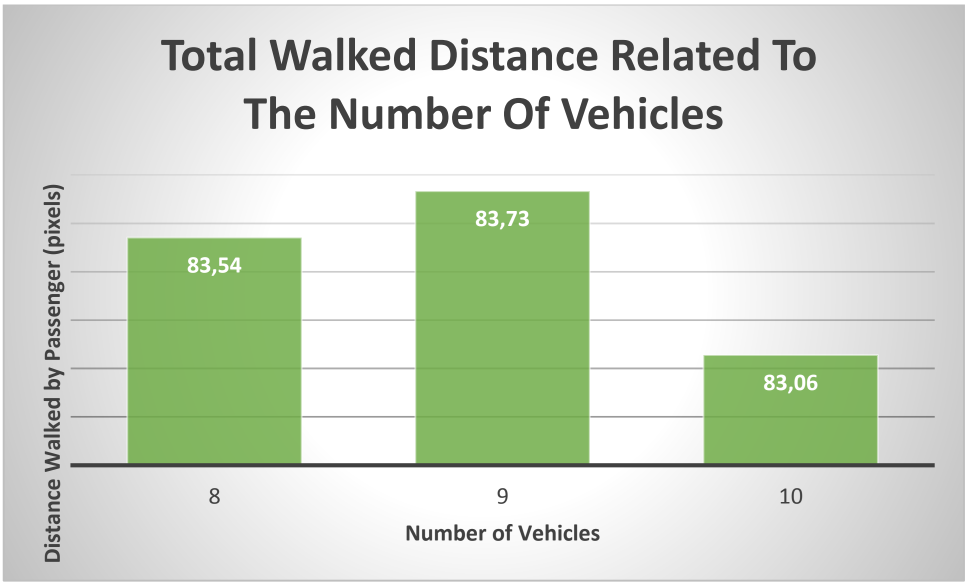

Figure 12 - Graphic Results of the Total Walked Distance Related to the Number of Vehicles

Although the values have a strange behaviour, when compared to the total traveled distance and the waiting time, the difference between the highest and shortest value is so small (aproximately 5 meters, in the metric system) that we can assume that the walking distance value as the same behaviour for every set of vehicles.

6.3. Results of the simulations with passenger waiting time and passenger walking distance prioritized

The third situation took into account the prioritization of the average waiting time and the average walking distance factors, meaning that the Travel Distance Factor is equal to 0 and the Walking Distance Factor and the Waiting Time Factor is equal to 1. This means that the algorithm searched for a solution that was better for the passengers and worse for the service.

Table VI - Results of the simulations with passenger waiting time and walkind distance prioritized

	Distance Factor = 0	Waiting Time Factor = 1	Walking Distance Factor = 1
Number of Vehicles	Total distance - Vehicle	Waiting Time - Passenger	Distance Walked - Passenger
8	19382	3,20	49,10
9	17918	2,62	46,85
10	16663	2,11	46,19

The behaviour of the total distance values is the same as in the previous situations.

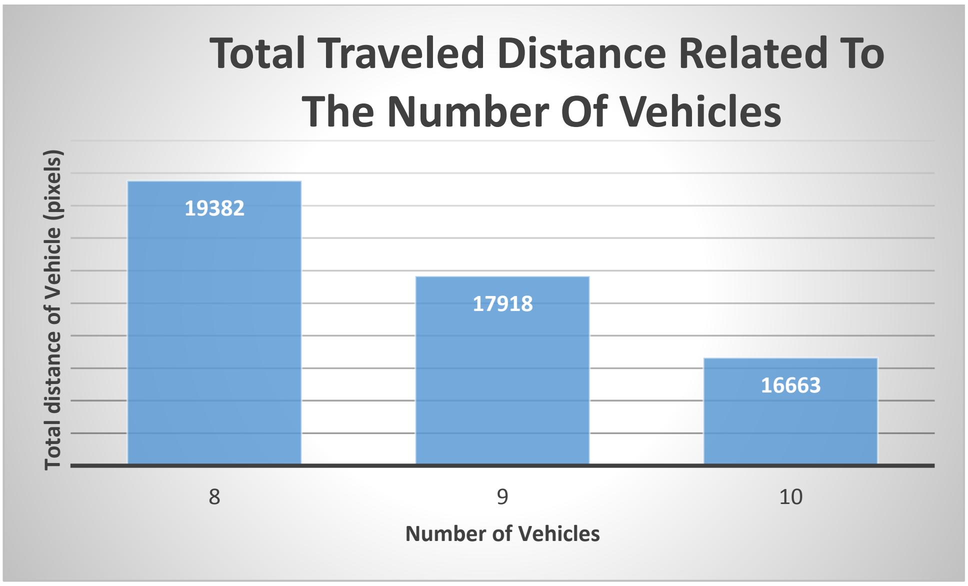

Figure 13 - Graphic Results of the Total Traveled Distance Related to the Number of Vehicles

In terms of waiting time, the values are also decreasing, when the number of vehicles increases. The values of waiting time are between 3,20 and 2,11 minutes. In terms of waiting time, this situation presents a good quality of service.

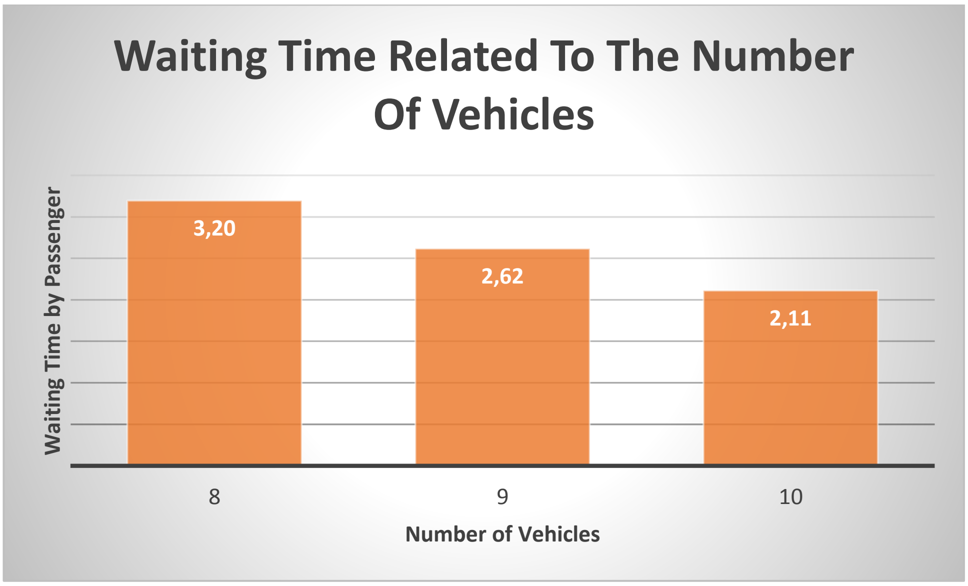

Figure 14 - Graphic Results of the Waiting Time Related to the Number of Vehicles

Regarding the walking distance of the passengers, there is a more noticeable difference between the highest value and the lowest one, that is equal to 3 pixels, that in metric terms is the same as 25 meters.

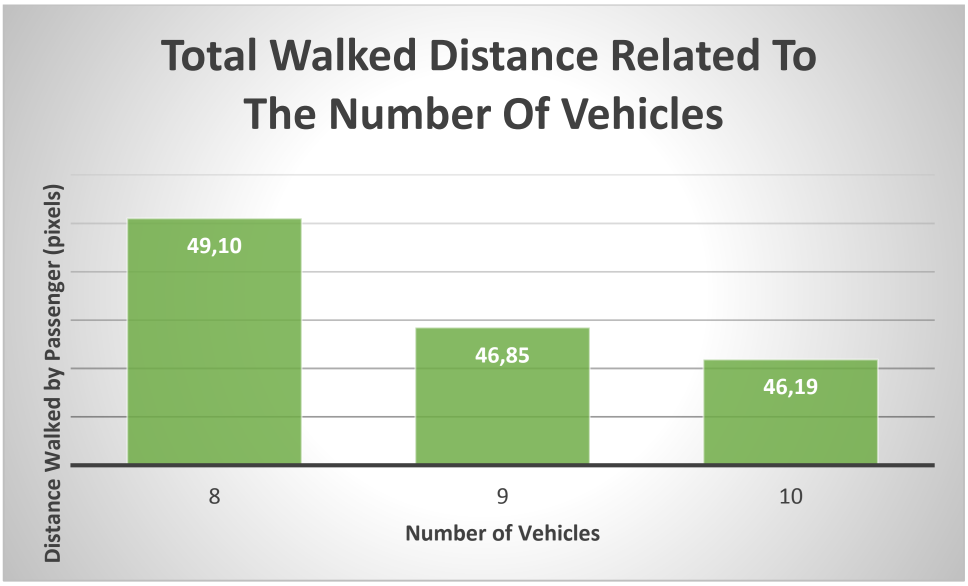

Figure 15 - Graphic Results of the Total Walked Distance Related to the Number of Vehicles

In terms of walking distances, the average passenger walks between 49,10 and 46,19 pixels, that in metric terms is about 420 and 395 meters, respectively. In terms of walking time, the passengers walk in the network for about 5 minutes. For the quality of service, regarding the walking distance, this situation provides a very reasonable service.

6.4. Discussion of the obtained results

After the individual analysis of each of the three situations, the following paragraphs are for discussing these results, by comparing the results of each simulated situation.

In terms of total distance traveled by all the vehicles, the behaviour of the data shows that when each point-of-view is prioritized, the results differ, as expected.

In the case of prioritizing the service, which means that only the minimization of the vehicles is applied, the total traveled distance is the shortest value of all the three situations. In the opposite situation, where the distance traveled by the vehicles is despised and the cost function only takes into account the factor related to the user, the total traveled distance rises to higher values. The average of the differences between these two opposite situations, is about 4707 pixels, that in metric units represents something aproximately of 40 kilometers, which is a big difference.

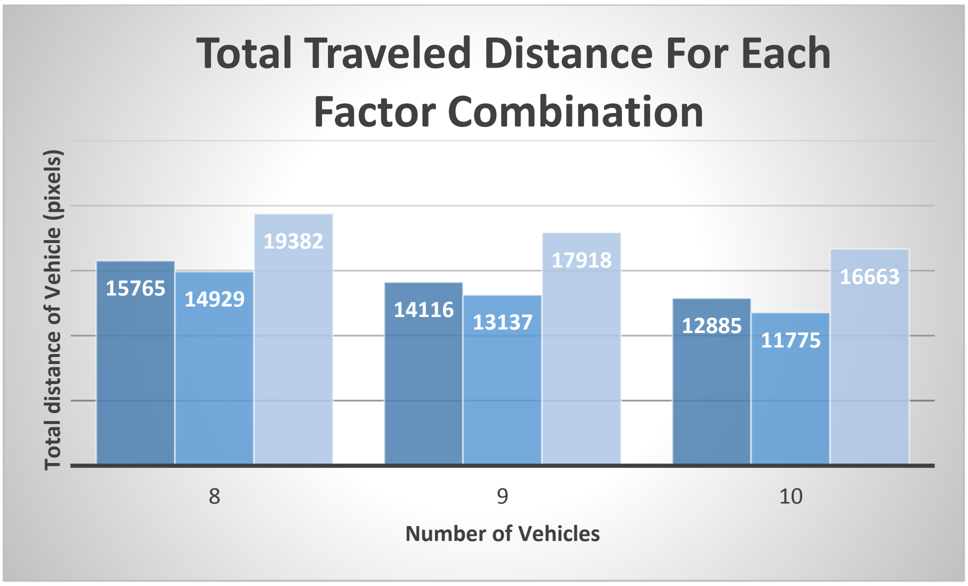

Figure 16 - Graphic Comparison Results of the Total Traveled Distance for each Factor Combination

The situation, where all the factors are prioritized, has a total traveled distance quite favorable, because the difference between this situation and the one that favours only the service is not has high, than the one described in the previous paragraph. This was expected, because has it was previous explained, the first prioritization of the cost function was the distance traveled by the vehicles, in order to make the system feasible for the service. This means that the difference between a service that only looks at itself and one that also takes into account the quality of service for the passengers, is not very big, in terms of total traveled distance performed by the vehicles.

Given all the studied combinations, the one with a more favourable situation, for the total traveled distance, is the one where only the service factor is applied and the total number of vehicles in the network is 10.

At the same time, the situation where all the factors are prioritized, presents very reasonable waiting time values. When the waiting time values of the situations of all factors are prioritized and the one that is only favourable for the passengers, are compared the difference between them is very small. However, the differences between these two situations and the one that favours only the service, are noticiable and has an average value of 4 minutes.

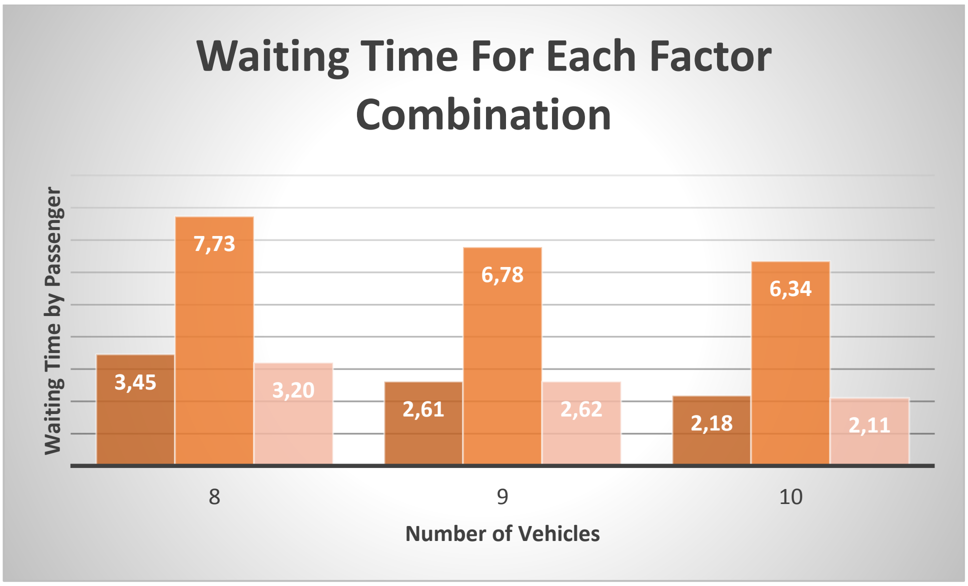

Figure 17 - Graphic Comparison Results of the Waiting Time for each Factor Combination

The best combination, regarding the waiting time is the one that has the situation of prioritizing only the factors of the passengers and with 10 vehicles, has expected.

In terms of the walked distance performed by the passengers between their origin point to their pick up point and from their delivery point to their destination point, the behaviour of the results between all the situations was the one expected, with the maximum walked distance related to less favourable passenger situation and the minimum with the most favourable.

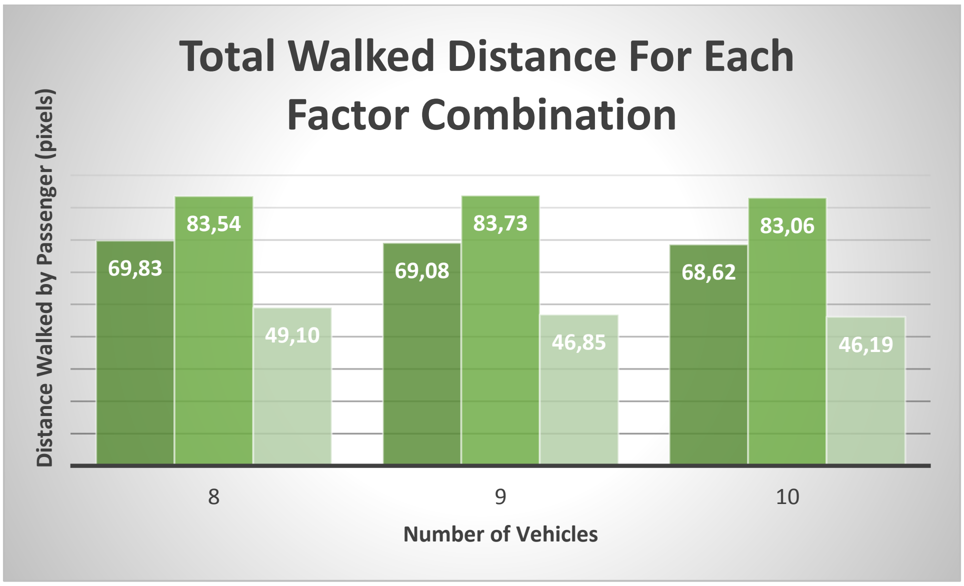

Figure 18 - Graphic Comparison Results of the Total Walked Distance for each Factor Combination

An aspect that was not expected was the very small difference between the walked distance, when the situation is the same and the number of vehicles increases. With these passengers samples and the number of vehicles, the differences between the distances walked by the passengers are so small, that it can be considered that this characteristic is independent of the number of used vehicles.

Analysing the data of the favourable situations for the passengers, it shows that although the waiting times are quite similar, the walked distance performed by the passengers presents differences for both favourable situations for the passengers. The average differences between the walking distances for the situations when all the factors are prioritized and only the passengers factors, is equal to 22 pixels, that in metric units is about 187 meters. This distance, in terms of walking time, is equal to 2 minutes.

Resuming, the simulator presented the expected results behaviour for the total traveled distance performed by the vehicles and for the passengers waiting time, where the best results are related to the prioritization of the factors that directly connected to these characteristics. The values of the characteristics studied are positive, specially the ones related to waiting time and total walked distance, performed by the passengers. Contrary to the expectations, the walked distance is independent of the number of vehicles, for the considered cost function and the number of passengers and vehicles.

7. Conclusions

In this document it is defined a new variant of the Dial-a-Ride Problem, called the Strolling Dial-a-Ride Problem. In this problem, the passengers of this type of transportation are expected to walk to be picked up and to get to their destination points, from their delivery points.

The possibility of applying this type of service, was studied with a computer simulation, where passengers and vehicles interacted in a network that represents a small/medium size city. This simulation took into account the possibility of communication between vehicles, in order to decide which one of them has the least insertion cost for a new requests. This decision is based on an auction, where, in this case, the one with the minimum value of insertion is the one responsible for picking up the passenger and deliver him next to his destination point. The insertion cost is based on cost function, that takes into account the detour distance of the vehicle to pickup/delivery the passenger; the waiting time of the new passenger; and the distance walked by the passenger. These three aspects can be prioritized by weight factors than can be changed between simulations, in order to see the influence of each aspect in the final results.The simulator is prepared for other types of cost functions and different numbers of vehicles and passengers.

The results obtained were positive, specially the ones that were oriented to the satisfaction of the passengers, presenting very low values of waiting time and walked distances. It was also noticeable that, in this cost function, the average value of walking distance made by the passengers, is irrelevant to the number of vehicles applied.

In terms of future work, the Strolling Dial-a-Ride Problem can be solved by other types of strategies and other cost functions. In terms of simulation, the problem can be tested with a real city network, with real data of passengers, in order to understand the effects of the Strolling Dial-a-Ride in a real-life case.

8. References

ATTANASIO, Andrea, et al. Parallel tabu search heuristics for the dynamic multi-vehicle dial-a-ride problem. *Parallel Computing*, 2004, 30.3: 377-387.

BEAUDRY, Alexandre, et al. Dynamic transportation of patients in hospitals.*OR spectrum*, 2010, 32.1: 77-107.

CALVO, Roberto Wolfler; COLORNI, Alberto. An effective and fast heuristic for the dial-a-ride problem. *4OR*, 2007, 5.1: 61-73.

CARAGLIU, Andrea; DEL BO, Chiara; NIJKAMP, Peter. Smart cities in Europe. *Journal of urban technology*, 2011, 18.2: 65-82.

CHOURABI, Hafedh, et al. Understanding smart cities: An integrative framework. In: *System Science (HICSS), 2012 45th Hawaii International Conference on*. IEEE, 2012. p. 2289-2297.

COLORNI, Alberto; RIGHINI, Giovanni. Modeling and Optimizing Dynamic Dial-a-Ride Problems. *International Transactions in Operational Research*, 2001, 8.2: 155-166.

CORDEAU, Jean-Francois. A branch-and-cut algorithm for the dial-a-ride problem. *Operations Research*, 2006, 54.3: 573-586.

CORDEAU, Jean-François; LAPORTE, Gilbert. The dial-a-ride problem (DARP): Variants, modeling issues and algorithms. *Quarterly Journal of the Belgian, French and Italian Operations Research Societies*, 2003, 1.2: 89-101.

CORDEAU, Jean-François; LAPORTE, Gilbert. The dial-a-ride problem: models and algorithms. *Annals of Operations Research*, 2007, 153.1: 29-46.

COSLOVICH, Luca; PESENTI, Raffaele; UKOVICH, Walter. A two-phase insertion technique of unexpected customers for a dynamic dial-a-ride problem.*European Journal of Operational Research*, 2006, 175.3: 1605-1615.

DESROSIERS, Jacques; DUMAS, Yvan; SOUMIS, Francois. A dynamic programming solution of the large-scale single-vehicle dial-a-ride problem with time windows. *American Journal of Mathematical and Management Sciences*, 1986, 6.3-4: 301-325.

DIAL, Robert B. Autonomous dial-a-ride transit introductory overview.*Transportation Research Part C: Emerging Technologies*, 1995, 3.5: 261-275.

DIAL, Robert B.; GHANI, Gisela. The ADART Driver Interface. 2003.

FU, Liping; TEPLY, Stan. On-Line and Off-Line Routing and Scheduling of Dial-a-Ride Paratransit Vehicles. *Computer-Aided Civil and Infrastructure Engineering*, 1999, 14.5: 309-319.

FU, Liping. A simulation model for evaluating advanced dial-a-ride paratransit systems. *Transportation Research Part A: Policy and Practice*, 2002, 36.4: 291-307.

HÄLL, Carl H.; HÖGBERG, Magdalena; LUNDGREN, Jan T. A modeling system for simulation of dial-a-ride services. *Public Transport*, 2012, 4.1: 17-37.

JAW, Jang-Jei, et al. A heuristic algorithm for the multi-vehicle advance request dial-a-ride problem with time windows. *Transportation Research Part B: Methodological*, 1986, 20.3: 243-257.

MADSEN, Oli BG; RAVN, Hans F.; RYGAARD, Jens Moberg. A heuristic algorithm for a dial-a-ride problem with time windows, multiple capacities, and multiple objectives. *Annals of operations Research*, 1995, 60.1: 193-208.

PARRAGH, Sophie N.; DOERNER, Karl F.; HARTL, Richard F. A survey on pickup and delivery problems. *Journal für Betriebswirtschaft*, 2008, 58.1: 21-51.

PSARAFTIS, Harilaos N. A dynamic programming solution to the single vehicle many-to-many immediate request dial-a-ride problem. *Transportation Science*, 1980, 14.2: 130-154.

REKIEK, Brahim; DELCHAMBRE, Alain; SALEH, Hussain Aziz. Handicapped person transportation: An application of the grouping genetic algorithm.*Engineering Applications of Artificial Intelligence*, 2006, 19.5: 511-520.

RØPKE, Stefan; CORDEAU, Jean-François; LAPORTE, Gilbert. Models and a branch-and-cut algorithm for pickup and delivery problems with time windows.*Networks*, 2007, 49.4: 258-272.

SEXTON, Thomas R.; BODIN, Lawrence D. Optimizing single vehicle many-to-many operations with desired delivery times: I. Scheduling. *Transportation Science*, 1985, 19.4: 378-410.

SEXTON, Thomas R.; BODIN, Lawrence D. Optimizing single vehicle many-to-many operations with desired delivery times: II. Routing. *Transportation Science*, 1985, 19.4: 411-435.

TEODOROVIC, Dusan; RADIVOJEVIC, Gordana. A fuzzy logic approach to dynamic dial-a-ride problem. *Fuzzy sets and systems*, 2000, 116.1: 23-33.

TOTH, Paolo; VIGO, Daniele. Fast local search algorithms for the handicapped persons transportation problem. In: *Meta-Heuristics*. Springer US, 1996. p. 677-690.

United Nations, World Urbanization Prospects: The 2014 Revision. Department of Economic and Social Affairs, United Nationa, New York 2014

XIANG, Zhihai; CHU, Chengbin; CHEN, Haoxun. A fast heuristic for solving a large-scale static dial-a-ride problem under complex constraints. *European Journal of Operational Research*, 2006, 174.2: 1117-1139.

XIANG, Zhihai; CHU, Chengbin; CHEN, Haoxun. The study of a dynamic dial-a-ride problem under time-dependent and stochastic environments. *European Journal of Operational Research*, 2008, 185.2: 534-551.