UNIVERSITY OF THESSALY
SCHOOL OF ENGINEERING
DEPARTMENT OF MECHANICAL ENGINEERING

DATA DRIVEN DESIGN OF A DEMAND RESPONSIVE TRANSPORTATION SYSTEM

by
SOFIA MARAGKOU
Applied Mathematician, National Technical University of Athens, 2017

Submitted in partial fulfillment of the requirements for the degree of Master of Science in Logistics & Supply Chain Management at the University of Thessaly

Volos, 2020

UNIVERSITY OF THESSALY
SCHOOL OF ENGINEERING
DEPARTMENT OF MECHANICAL ENGINEERING

DATA DRIVEN DESIGN OF A DEMAND RESPONSIVE TRANSPORTATION SYSTEM

by
SOFIA MARAGKOU

Submitted in partial fulfillment of the requirements for the degree of Master of Science in Logistics & Supply Chain Management at the University of Thessaly

Volos, 2020

Approved by the Committee on Final Examination:

Advisor	Dr. Athanasios Ziliaskopoulos, Professor, Department of Mechanical Engineering, University of Thessaly
Member	Dr. George Liberopoulos, Professor, Department of Mechanical Engineering, University of Thessaly
Member	Dr. Athanasios Lois, Network Operation Center, University of Thessaly

Date Approved: [February, 2020]

Acknowledgements

I wish to thank various people for their contribution to this thesis. First of all, I would like to express my deep gratitude to Professor A. Ziliaskopoulos, my research supervisor, for his guidance and useful critiques of my research work in general. I would also like to express my very great appreciation to Dr. A. Lois for his valuable and constructive suggestions during the planning and development of this research work. His willingness to give his time so generously has been very much appreciated. My grateful thanks are also extended to Professor G. Liberopoulos, member of the Committee, for inspiring during his classes to deal with stochasticity. Special thanks should be given to the chairman of VOLOSTAXI company for his willingness to share with me an extensive data set. Finally, I wish to thank my family and my friends, without whose support and encouragement throughout the years, I would not be here. This thesis is dedicated to the memory of my mother, who was always in my thoughts on this journey – you are missed.

DATA DRIVEN DESIGN OF A DEMAND RESPONSIVE TRANSPORTATION SYSTEM

SOFIA MARAGKOU
Department of Mechanical Engineering, University of Thessaly

Supervisor: Dr. Athanasios Ziliaskopoulos
Professor of Optimization of Production/Transportation Systems

Abstract

In this thesis, we attempt to define the steps needed in order to design a Demand Responsive Transportation system from scratch. The crucial factor that makes our proposed methodology differ from other approaches, is the fact that is fully based on historical data and not driven by extremely time consuming and expensive surveys. For this purpose, we collected and analyzed a huge historical data set concerning the requests for transportation in the city of Volos, Greece. Moreover, we focused on the Dial-a-Ride problem in its static version but also taking into account probabilistic information which became available through an extensive statistical data analysis. Probability was added to the dial-a-ride problem in three different ways, by considering probabilistic pickup and delivery points and probabilistic times in which each request occurs. Lat but not least, we constructed different test cases in accordance with our market's characteristics. Our work has led us to capture all the critical elements that need to be taken under consideration in the designing phase and define some crucial steps in the phase of deploying an on-demand system in an urban area, by making use of the advantages of technology in a world of data development.

Key words: demand responsive transportation system, on-demand transportation system, dial-a-ride problem, probabilistic information, data-driven design

Contents

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

<u>General Terms</u>

VRP = Vehicle Routing Problem

TW = Time Window(s)

PDPTW = Pickup and Delivery problem with Time Windows

VRPPDPTW = Vehicle Routing Problem with Pickup and Delivery with Time Windows

DVRP = Dynamic Vehicle Routing Problem

SVRP = Stochastic Vehicle Routing Problem

DSVRP = Dynamic and Stochastic Vehicle Routing Problem

VRPSD = Vehicle Routing Problem with Stochastic Demand

DVRPSD = Dynamic Vehicle Routing Problem with Stochastic Demand

DVRPSC = Dynamic Vehicle Routing Problem with Stochastic Customers

DOD = Degree Of Dynamism

SP = Stochastic Programming

LP = Linear Programming

ADP = Approximate Dynamic Programming

NP-hard = Non-deterministic Polynomial-time - hard

DSVNS = Dynamic Stochastic Variable Neighborhood Search

MSA = Multiple Scenario Approach

DARP = Dial-a-Ride problem

DARPTW = Dial-a-Ride problem with Time Windows

TSP = Travelling Salesman Problem

DRT = Demand Responsive Transportation System

<u>New terms specific to this thesis</u>

InsertionH = Static insertion algorithm for the Dial-a-Ride problem

RegretH = Static Regret algorithm for the Dial-a-Ride problem

Chapter 1. INTRODUCTION

The purpose of this thesis is the data driven design of a Demand Responsive Transportation system from scratch, which is based on probabilistic information that became available through an extensive statistical analysis of a huge historical data set.

This thesis is organized as follows. A brief description about Demand Responsive Transportation systems is given in this section. A detailed description and literature review about the Vehicle Routing Problem and the Dial-a-Ride Problem is given in section 2. The algorithms that were implemented in order to find a solution for our problem are described in section 3. In section 4 we present our statistical analysis concerning a large sized data set obtained on the city of Volos, Greece. Furthermore, the construction of our experiments and their computational results are contained in section 5. Finally, some concluding remarks and directions for the data driven design of an On-Demand Transportation system are presented in section 6.

1.1 Demand Responsive Transportation (DRT) systems

The provision of qualitative public transportation can be extremely expensive in low, variable and unpredictable demand scenarios, as it is the case of dispersed rural areas or some periods of the day in urban areas. Demand Responsive Transportation (DRT) systems try to handle this problem by providing a hybrid approach between a taxi and a bus, with routes and frequencies that may vary according to the observed demand. In this way, operators offer a more efficient service because of this additional flexibility, with routes planned shortly before their start, better occupancy rates and vehicles' characteristics better suited to users' mobility requirements. However, in terms of financial sustainability and quality of the service, the design of this type of services may be difficult.

The problems of designing and operating DRT services are closely related to the Vehicle Routing Problem, and in particular to the Dial-A-Ride Problem. DRTs extend the “classical” Vehicle Routing Problems (VRP) in a number of ways. It is clear that in the DRT context, vehicles have a limited capacity, demands should be served within a certain time window, each stop along the route can be both a pickup and a delivery point and there is the uncertainty and variability associated with the number of stops along the route. In the Dial-A-Ride Problem (DARP) the goal is not only minimizing the operating costs or the distance travelled by the vehicles but also maximizing the quality of the service, based on indicators such as the average passenger waiting time or the on-board (ride) passenger time.

DRT services can operate in a static or in a dynamic mode. In static mode, all requests are known before-hand (a priori or advanced requests), whereas in dynamic mode transportation requests are gradually revealed along the service operating time, with routes and schedules having to be adjusted to meet the demand. In practice, however, “pure” dynamic services are not common since some requests are usually known a priori.

Given the complexity of these problems, optimal solutions can take an enormous amount of time to be found, ruling out their usefulness in the context at hand. Besides, in a multiple criteria decision analysis the “optimal” solution is in general meaningless because it is impossible to satisfy all (usually contradictory) objectives simultaneously. So, we are interested in finding a set of efficient solutions.

In this thesis, we attempt to design a DRT system for an urban area from scratch. The crucial factor that makes our proposed methodology differ from other methodologies, is the fact that is fully based on historical data regarding the city of our research. A deliberate study on this data set revealed significant elements for the design of the on-demand transportation system. Moreover, in contrast with other approaches regarding the design of a DRT system, our methodology is not driven by surveys which are extremely time consuming and expensive. Thus, instead of questioning people, we made a request for a historical data set to the only taxi company in the city of our research. A huge historical data set has been provided to us, which included 959780 requests for transport in our region of research for the last 4 years (2016 - 2019). In this way, we have been able to know the demand for transportation in this city including its time and spatial characteristics, without trying to discover the requirements for such a transportation system through costly questionnaires. In conclusion, we achieved to capture all the critical elements that need to be taken under consideration in the designing phase and define some crucial steps in the phase of deploying an on-demand system in an urban area, by making use of the advantages of technology in a world of data development.

Chapter 2. LITERATURE REVIEW

A detailed description and literature review about the Vehicle Routing Problem and the Dial-a-Ride Problem is given in this section.

2.1 Vehicle Routing Problem (VRP)

Stochastic and dynamic point of view

Over the years, the VRP is studied widely but it is still a challenging research theme. One should categorize the VRPs based on the information availability and uncertainty as it is introduced in Schorpp (2010) and utilized in Pillac et al. (2013). This leads to *static*, *dynamic*, *stochastic*, and *dynamic and stochastic VRPs*. For the *static* VRP, the reader is referred to an overview of different VRP formulations in Toth andVigo (2001) and other reviews in Cordeau et al. (2007), Laporte (2007, 2009).

In stochastic optimization, the aim is to find, given a VRP where some data are stochastic, an a priori routing plan that minimizes the expected objective function. These approaches often have a recourse function to correct the plan when constraints are violated. In dynamic optimization, customer requests are not known beforehand and become available over time. Usually, optimization is carried out on the known parameters until an event (e.g. a customer request) happens. The plan is then adapted to service the new request if it is possible. Generally, stochastic information is also available for the dynamic version of VRP, either through historical data (as suggested in Gendreau et al. 1998, 1999 and Kilby et al. 1999) or through some available probabilistic models (as mentioned in Gendreau et al. 1996b and Yang et al. 2000). However, the exploitation of the stochastic information in order to service as many customer requests as possible is a crucial open research problem.

1. Dynamic vehicle routing problems

Sometimes, not all information concerning the problem is known beforehand. In the dynamic VRP (DVRP), also mentioned as *real-time* or *online* VRP, some data are available only during the execution. The arrival of new customer demands, service times and travel times are the most frequent dynamic events in VRPs. The DVRP is extensively studied in the literature, starting with the work of Psaraftis (1988) which shows the differences between static and dynamic VRPs. Larsen and Madsen (2000) present a classification of DVRPs according to the dynamism of the system, while Pillac et al. (2013) classify DVRPs concerning the type of dynamic events. A variety of algorithms coping with DVRPs are suggested in the literature and superior reviews are given in Psaraftis (1995), Larsen and Madsen (2000), Larsen, Madsen, and Solomon (2008), Jaillet and Wagner (2008), Schorpp (2010), Pillac et al. (2013) and Visentini et al. (2014). Including dynamic information leads to the increase of the complexity of the problem and new issues occur. The decision whether a new request is accepted or rejected is introduced as *service guarantee* in Van Hentenryck and Bent (2009) and discussed in Ichoua, Gendreau, and Potvin (2000) and Li, Mirchandani, and Borenstein (2009). Karsten Lund and Rygaard (1996) define the *degree of dynamism (DOD)* which shows how dynamic a system is concerning the number of dynamic request and the number of total requests (the ratio *dynamic customers/total customers* (Larsen et al. 2002)). Larsen and Madsen (2000) introduced further measurements which additionally consider the time aspect. Furthermore, a significant issue is to determine a suitable objective function, since attributes such as service level, number of serviced requests, minimization of response times or profit maximization are concerned in contrast to simply considering travel times or distances. Due to the fact that DVRPs require online decisions, a compromise between reactiveness and decision quality must be found. A highly accepted performance measure for online algorithms is the *competitive analysis* introduced by Sleator and Tarjan (1985) and further research on this topic is presented in Krumke (2002), Angelelli, Speranza, and Savelsbergh (2007), Jaillet and Wagner (2008). The adjustment of the planned solution according to the plan in execution is another crucial element and can be performed with different updating strategies.

2. Stochastic vehicle routing problems

The majority of real-world problems suffer from an initial level of uncertainty. Commonly, information concerning upcoming events is available through historical data, which can be adapted to information models. The processes of data collection, analysis and provision are widely discussed in Ehmke, Steinert, and Mattfeld (2012). The uncertainty in problem descriptions can be captured in various ways, mainly distinguishing between different formalization variants, as discussed in Bianchi et al. (2009). One could generally classify stochastic combinatorial optimization problems (SCOP) according to the time where uncertain information is revealed. This results in static SCOPs, where decisions are taken before the random variables are realized and dynamic SCOPs, where decisions are made after some random events have occurred. The stochastic VRP (SVRP) is basically any VRP where one or more parameters are stochastic, meaning that some future events are random variables with a known probability distribution. A classification according to the stochastic parameters is proposed in Gendreau, Laporte, and Sguin (1996) and various optimization problems with uncertainty are summarized in Sahinidis (2004). In general, pure SVRPs have a probability distribution of the random variables available and the optimization process is performed before they are realized. The planned routes are not changed or updated after the realization, thus, it is often referred to as a-priori *optimization*. One of the most commonly applied approaches to SVRPs is *stochastic programming (SP)* where a general introduction is given in Birge and Louveaux (1997) and SP in the context of transportation and logistics are discussed in Powell and Topaloglu (2003), Powell and Topaloglu (2005).

3. Dynamic and stochastic vehicle routing problems

In the last years, researchers have focused on the dynamic and stochastic VRPs (DSVRP). The term stochastic means that the information concerning upcoming events can be described by a random variable with a known probability distribution. Moreover, the term dynamic means that the available information evolves over time – occurs over the problem horizon – with accuracy at its worst during initial planning and at its best during operations. Because of the rapid technological progress concerning information and communication, it is now allowed to cope with real-world applications more accurately. In conjunction with efficiently handling dynamic events, stochastic knowledge about the revealed data is considered. The literature provides several approaches to deal with this evolution of information: through the incorporation of stochastic information during planning (Dror et al. 1989), through update methods, where a static problem is solved repeatedly (Berbeglia et al. 2010, Pillac et al. 2012a), through online algorithms where a specific action is taken when new information arriving (Jaillet and Wagner 2006), or through stochastic, dynamic strategies that combine the first two approaches (Goodson et al. 2013). When incorporating stochastic information in the planning phase, the aim is to develop a robust plan – one that remains relevant despite changes in information. Both dynamic and online algorithms are similar in that new routes or actions are only undertaken when the uncertain information becomes certain. Even after the vehicles are en route, stochastic, dynamic strategies that accommodate job location information on unknown future jobs demonstrate improvements (see for example Thomas and White (2004), Hyyti¨a et al. (2012), Cortes et al. (2009)). Other authors use probabilistic or advance information to make decisions about which jobs to serve (Jaillet and Lu 2011, Kim et al. 2004). Considering the problem from the decision-making point of view, Approximate Dynamic Programming (ADP) provides a tool to decompose the problem into a series of decisions over time. The key to use ADP effectively is to have a clear relationship on the transition from one system state to the next and to have a mechanism by which to evaluate the different decision policies. Powell et al. (1988) utilized ADP to study a problem of truckload pickup and delivery when the job requests are uncertain. Goodson et al. (2013) used a roll-out algorithm along the lines of ADP in order to serve loads when the size of the loads was uncertain. Flatberg et al. (2005) and Pillac et al. (2013) provide an overview on DSVRPs but with a strong focus on pure DVRPs, whereas Ritzinger and Puchinger (2013) give a review of DSVRPs but with an exclusive focus on various hybrid methods applied to this field. In recent years, the field of *anticipatory optimization*, dealing with the future realization of relevant parameters, has also been connected to dynamic decision-making. A summary of successful methodologies for anticipatory optimization and dynamic decision-making, categorized regarding different degrees of anticipation, is given in Meisel (2011). As already mentioned, the broad classification is based on the nature of the uncertain information as defined in Gendreau, Laporte, and Sguin (1996). However, the focus is on a further characterization based on the point in time where substantial computational effort for determining decisions or decision policies actually occurs. This results in two groups:

• The first group, *preprocessed decisions*, is about approaches where policies or solutions are computed before the execution of the plan.

• The second group, *online decisions*, consists of approaches where solutions are computed as soon as a dynamic event occurs.

Both groups consider dynamic systems and are based on some stochastic information. In order to tackle such systems properly, state-dependent decisions must be made, which is often named *policy* in the literature. The problem settings for both groups are usually modelled as *Markov decision process (MDP)* or formulated as *multi-stage stochastic models*. In Figure 1, the difference between the groups is illustrated. Solution approaches which belong to the first group (*preprocessed decisions*) determine the values for decision-making, respectively, policies, before the execution of the solution plan. Therefore, possible states need to be constructed in advance and evaluated based on possible dynamic events and stochastic information along a considered time horizon. The first variant is to analyze general policies based on the arising states and decisions beforehand and apply them during the operational process, e.g. always dispatch the nearest available vehicle. The second variant is to value possible states and their decisions before plan execution and use these preprocessed values in the dynamic planning process. Thus, during plan execution, these methods only exploit the precomputed values to make accurate decisions for the current system state. The second group (*online decisions*) differs in so far as a major part of the computation is made when a dynamic event occurs. Here, a current solution plan is followed during execution and whenever a new event occurs, a decision is calculated online with respect to the current system state and the available stochastic information. This procedure is also referred to as *rolling horizon procedure* or as *look-ahead strategy*. The solution of such an approach is either a single decision for the current situation or a re-optimized plan. Another possibility is to provide a greedy single decision requiring less computational effort first and run a more intensive re-optimization later in the background. The choice of the most adequate method to be used for reacting on dynamic events is strongly connected to the DOD and the given *reaction time*.

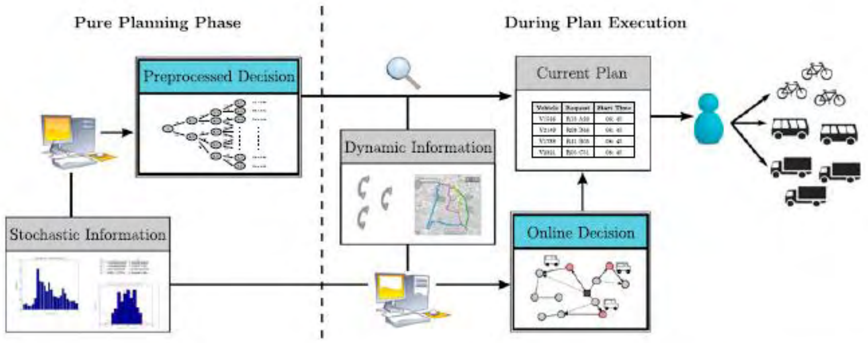

Figure 1. Illustration of the two groups: *preprocessed* and *online decisions*. The picture shows that *preprocessed decision* approaches spend most of the computational time processing stochastic information in the planning phase, before the actual execution of the plan. During the plan execution, the current plan is updated based on lookups to the preprocessed information. However, *online decision* approaches have to compute the results with respect to the dynamic and stochastic information during the operational process. Finally, a dispatching system receives an updated plan and is able to assign the vehicles.

DVRP with stochastic travel times

Travel times are essential data in VRPs because of their paramount importance for the considered network. Static deterministic travel times often differ from real-world, since variation in traffic density, mostly in urban areas, is not considered. In order to overcome this problem, there are the three ways to model travel times as *time-dependent*, *stochastic*, or *stochastic and time-dependent*. The latter variant is the combination of stochastic and time-dependent travel times and results in stochastic time-dependent networks, where link travel times are random variables with time-dependent distributions. Hall (1986) first studies the stochastic time dependent *shortest path problem*, also called the *least expected shortest path* or *least expected time path* where a shortest path, based on estimation of mean and variance travel times, is constructed. The solutions are called a-priori or *non-adaptive*, since no decisions are updated once the vehicle is en route. Work on this can be found in Hall (1986), Fu and Rilett (1998) and Miller-Hooks and Mahmassani (2000). VRPs applying stochastic and time-dependent travel time are presented, for example, in Fu (2002), Chen, Hsueh, and Chang (2006),Woensel et al. (2008), Nahum and Hadas (2009), Lecluyse, VanWoensel, and Peremans (2009) and Ta¸s et al. (2014). On the other hand, *dynamic* or *adaptive* solution approaches recognize the benefits of an adaptive decision-making process.

1. Preprocessed decision support

In order to handle uncertainty and real-time information about travel times efficiently, travel times are analyzed and modelled appropriately and incorporated into the algorithms. Usually, this is modelled as a MDP and it is popular to model the stochastic information about travel times via probabilities of congestion on links. The purpose is to calculate an optimal routing policy beforehand which can then be applied to the vehicle en route. Note that all the possible decisions are calculated and evaluated before the vehicle starts its route. One flaw is the dimension of the problems. Thus, for every possible state concerning time, location and traffic information, an optimal routing policy must be decided and stored. The computational effort can be demonstrated by the rather small test networks used in the literature. Interesting work which considers stochastic time-dependent travel times and which performs (optimal) routing decisions in advance is presented in Fu (2001), Miller-Hooks (2001), Kim, Lewis, and

White (2005), Gao and Chabini (2006), Gao and Huang (2012) and Güner, Murat, and Chinnam (2012). Also, Toriello, Haskell, and Poremba (2014) propose an approximated linear programming (ALP) approach for the dynamic TSP with stochastic arc costs. They also investigate a rollout policy, where the expected costs at any state biased by the optimal value of the LP relaxation of a shortest Hamiltonian path are considered.

2. Online decision making

Here, the focus is on DVRPs with stochastic and time-dependent travel times where the decisions for adapting the route are not preprocessed but calculated online. In Taniguchi and Shimamoto (2004) and Potvin, Xu, and Benyahia (2006), the routes are updated according to estimated travel times whenever a vehicle arrives a customer location. In the former work, the travel times are obtained by a dynamic traffic simulation based on a macroscopic approach and the results show the benefit of using the reactive algorithm compared to the static counterpart for test scenarios where a link is blocked for an hour at different times. In the latter case, travel times are obtained by a short-term forecast and dynamic perturbation model and a tolerance level, which is an allowed waiting time in the case of lateness, before a reassignment action, is initiated. Results show that a good strategy is to accept lateness in the case of small deviations in travel times, but react on events of a large magnitude. Yan, Lin, and Lai (2013) deal with the planning of courier routes and introduce a time–space network including several arcs associated with travel times and a probability. A stochastic planning model, incorporating unanticipated lateness penalty costs and a stochastic real-time adjustment model, which handles dynamic requests and aims for little route adjustment, are introduced. The results show that the model with route adjustment yields better results. Another fundamental work is presented by Schilde, Doerner, and Hartl (2014), using stochastic deviations from time-dependent travel speeds which are deduced from historical accident data. The positive effect of using such data instead of average time-dependent travel speeds is tested on different stochastic metaheuristic concepts: dynamic stochastic variable neighborhood search (DSVNS) which is based on Gutjahr, Katzensteiner, and Reiter (2007) and multiple scenario approach (MSA) introduced in Bent and Van Hentenryck (2004). The results are compared to the corresponding myopic approaches: a dynamic VNS (DVNS) and a multiple plan approach (MPA). In contrast to the work above, the true travel speed is revealed in each iteration of the algorithm. In general, the DSVNS performs best, but the solution quality highly depends on the DOD in so far as for highly and nondynamic instances, the DVNS works better. However, the MSA turns out to be unsuitable for this problem setting and, on average, obtains no improvements.

3. Conclusion

The algorithms cannot be fairly compared to each other because of the problem-specific constraints and the use of different test instances. However, each work shows that even though the reduction of operational costs is not wide, the reliability (i.e. level of customer service) increases considerably when uncertainty in travel times is considered in the solution approach.

DVRP with stochastic demand

Another extensively studied problem class is the VRP with stochastic demand (VRPSD), where there is uncertainty in the customer demands. The location of the customer is known beforehand, but the actual demand is revealed when arriving at the customer location. It occurs that the required demand cannot be met and the vehicle has to return to the depot for replenishment before serving the customer. Therefore, the objective of VRPSD is to minimize the total expected travel costs needed to serve all customer demands. Typical applications for the VRPSD are the supply of gas stations or garbage collection. Mainly, an a-priori solution which incorporates stochastic demand information is constructed and the customers are visited according to the plan without any route updates during the operation. Whenever the customer demand is not met, a so-called route failure occurs and a defined recourse actions (e.g. replenishment at the depot) must be applied. Typically, a two-stage approach is applied as proposed in Bertsimas (1992) and Gendreau, Laporte, and Sguin (1996). The most common concepts are chance constraint programming and stochastic programming with recourse. A great review on VRPSD is given in Campbell and Thomas (2008) and in Mendoza et al. (2010), Erera, Morales, and Savelsbergh (2010) and Juan et al. (2011, 2013). *Anticipatory insertion (AI)*, discussed in Thomas and White III (2004), is another concept for exploiting future information about customer demand. While a-priori solution approaches are widely studied, advanced planning technology has also facilitated another approach where routing or replenishment decisions are made dynamically, often called *re-optimization* approach. This different problem class is called *DVRP with stochastic demand (DVRPSD)*. In other words, VRPSDs are extended with the possibility of route adaption during the plan execution phase whenever new data are revealed.

1. Preprocessed decision support

In order to solve a DVRPSD, a widely known way is to consider all states (e.g. all possible demand realizations) beforehand and the value of each state according to its performance. Such an approach performs the evaluation of the states before the vehicle starts the tour and enables an accurate decision-making based on these values during the plan execution phase. From a computational point of view, it might be very expensive to determine all the predefined values, but it comes with the advantage that dynamic decisions can be provided quite fast. This is usually formulated as a *multi-stage stochastic dynamic programming* model and with the assumption that states and decisions are discrete and the value evaluation can be done by Bellman's equation. To overcome the curse of dimensionality, the exact value function can be replaced by an approximation. This formulation can either be implemented as an ALP as in Toriello, Haskell, and Poremba (2014) or as an *approximated dynamic programming (ADP)* discussed in Powell (2007). An ADP algorithm is discussed in Zhang et al. (2013) for the single-vehicle DVRPSD based on *value function approximation (VFA)*. An ADP algorithm based on VFA with lookup table representation is developed and then improved by a Q-learning algorithm with bounded lookup tables and efficient maintenance. The algorithms are tested on the instances of Secomandi (2001) and Solomon (1987) and results show that especially for larger instances (up to 60 customers), the computational time could be reduced with the improved algorithm with even slight improvements in solution quality. Another ADP-based approach is presented in Meisel, Suppa, and Mattfeld (2011) where the need of explicit anticipation of customer requests is discussed. The authors consider a single-vehicle approach, where the set of customers is divided into static and dynamic requests and for the latter, either a rejection or acceptance decision is made after its arising. The algorithm is tested on the instances in Solomon (1987) and the results are compared against the *distribute waiting time strategy* presented in Thomas (2007) and yield better results.

2. Partly online decisions

For the DVRPSD, this additional category is identified consisting of approaches where some of the computational effort is done beforehand to guide the online decision-making process. This grows out of the fact that all locations are known beforehand and this information can be exploited for precalculations. A common sequential approach is the *rollout algorithm (RA)*, which can be considered as a single iteration of policy iteration starting with a heuristically computed base policy. Based on its performance, an improved policy is obtained by an one-step look-ahead. Thus, decisions for the current state are determined by approximating the cost-to-go via the base policy looking one-step ahead. Bertsekas (2013) provides an extensive survey on RAs and Goodson, Thomas, and Ohlmann (2012) describes rollout policies for general stochastic dynamic programs. Secomandi (2000, 2001) provides the first computational results of a re-optimization policy for the DVRPSD by means of a RA. A one-step algorithm is developed where the cyclic-heuristic introduced in Bertsimas (1992) is used as base policy. Two versions (no-split and split delivery) are analyzed and compared to a static rollout approach. Secomandi and Margot (2009) develop an algorithm which determines re-optimization policies by computing the optimal policy for a restricted set of states (selected by a partitioning and sliding heuristic). Novoa and Storer (2009) extend the work in Secomandi (2001) by implementing different base policies. It is demonstrated that applying a two-step RA yields better results than the one-step RA. However, the best performing base policy is an adapted stochastic set-partitioning-based model (Novoa et al. 2006). Note that these works consider the single-vehicle DVRPSD, whereas Fan, Wang, and Ning (2006) solve the multi-vehicle DVRPSD by decomposing it to single-vehicle problems first and applying the RA of Secomandi (2001) to each of them.

3. Online decisions

Online decisions for the DVRPSD are determined either by applying online algorithms or, if computational time allows it, by recomputing the base sequence at predefined states (e.g. an event occurs). Erera and Daganzo (2003) divided the service region into two parts and after serving all customers of the first part a single real decision is made to assign the unserved customers to vehicles considering their remaining capacities. Pillac, Guéret, and Medaglia (2012) propose an event-driven, sampling-based MSA where a pool of scenarios is maintained.

The scenarios are realizations of customer demands and are optimized by an adaptive VNS. The selection of the next customer is performed by a consensus function and results are compared to some large instances in Novoa and Storer (2009). Cheung, Hang, and Shi (2005) present a dynamic stochastic drayage problem where the duration of a task (transportation from origin to destination location combined with some intermediary activities) is considered to be uncertain. An adaptive labelling approach within a rolling horizon procedure is developed where virtual routes and labels are used and adapted to approximate the expected future costs. In Thomas (2007), intermediate requests at known locations may arise while the vehicle is en route and waiting positions for serving these requests best possible are determined by a real-time heuristic, called center-of-gravity heuristic. A somehow surprising result is presented in Ghiani, Manni, and Thomas (2012) where an AI heuristic is compared to a sample-scenario planning approach (MSA) in the context of a dynamic stochastic TSP. The main contribution is to emphasize that AIs, thus a-priori solutions, generate comparable solutions while needing less computational effort. Even though the MSA shows poor performance, it has to be noted that the MSA is sensitive to the problem structure (e.g. no time windows are given) and to instance data. As for Thomas (2007), it can be deduced that an increase in the DOD has greater impact than an increase in the likelihood of requesting customers. In Goodson, Ohlmann, and Thomas (2013), besides different a-priori-based policies, a dynamic decomposition-based rollout policy is presented to effectively tackle the multi-vehicle case. The customers are re-partitioned at each decision point by executing the fixed-route heuristic from the current state. In a further work, Goodson, Thomas, and Ohlmann (2013) consider preemptive replenishment and present a RA in combination with a sampling-based approach, where a sample average approximation is applied to estimate the expected value of a restocking policy along a fixed route to a set of scenarios. Recently, Zhu et al. (2014) introduce the *paired cooperative re-optimization* strategy formulated as a bilevel MDP, which makes use of partial re-optimization (Secomandi and Margot 2009) and paired-vehicle cooperation (Ak and Erera 2007).

4. Conclusion

Research demonstrates the success of adaptive methods for the VRPSD (DVRPSD) compared to non-adaptive approaches. Research on *partly/preprocessed decision* procedures is a little bit more explored, but results on *online decision* methods show slightly better results. However, because different instances and recourse actions are applied, no fair comparison can be done among the algorithms. A variety of papers on DVRPSD is available in the literature with a strong focus on single-vehicle problems but the challenge is to investigate methods which are able to cope with more realistic problems, like larger sizes of instances as well as multi-vehicle VRPs.

DVRP with stochastic customers

Another widely studied problem class is the DVRP with stochastic information about customers (DVRPSC). This problem class mainly occurs when some customer requests are known at planning time, but others are revealed during the day of operation. In contrast to the previous section, combined stochastic information about customer locations and the time of request occurrence are considered. In many real-life routing problems, there is more uncertainty with respect to the required timing of the service than with respect to the service locations. The stochastic information is either available for each customer or it is aggregated geographically and/or temporally. For example, in the context of truckload, vehicle routing problems, this uncertainty can be classified along two dimensions – spatial and temporal. Spatial uncertainty is about situations in which the exact location of a job is not known (or known inaccurately) during planning, but becomes known at some point in operations. Temporal uncertainty, on the other hand, is about situations in which the exact service time or time window of a job is not known (or known inaccurately) until operations are underway. These two dimensions are so intertwined that focusing on one of them separately is typically difficult. Concerning truckload, pickup and delivery problems, researchers tend to focus on scenarios in which both the service time and the location of future jobs are unknown. For such problems, multiple solution techniques exist varying from reactive, online/dynamic, rolling horizon strategies to stochastic strategies (Berbeglia et al. 2010). However, the isolated study of temporal uncertainty is less present in the literature. Srour, Agatz, and Oppen (2016) tried to fill this gap by studying a pickup and delivery problem with time windows in which the pickup and drop-off locations of the service requests are fully known beforehand, but the time at which these jobs will require service is only fully revealed during operations. They developed a *sample-scenario routing strategy* to accommodate a variety of potential time realizations while designing and updating the routes. The results showed that advance time related information (by having customers preannounce their need for service), even if uncertain, can provide benefits in conjunction with having information on the service locations. The stochastic mechanism that they introduced, the MTS strategy with the sequence-based consensus function, was robust, even if they did not know the true probability distribution that characterized the uncertainty. The strength of MTS-seq was its extension of the original consensus function of Bent and Van Hentenryck (2004) by not only scoring the similarity in terms of the job to vehicle assignments but also in terms of the route sequence. This additional component was beneficial in handling the uncertainty in their problem setting.

1. Preprocessed decision support

Besides *myopic* and *look-ahead* policies (also known as *rolling horizon planning*) and *policy function approximation (PFA)*, a successful policy in operations research and the context of dynamic stochastic problems is VFA. In this context, ADP is a powerful framework for calculating the future impact of a decision and using an approximated value to improve it. This is demonstrated in an excellent tutorial on ADP in the field of transportation and logistics in Powell, Simao, and Bouzaiene-Ayari (2012) and illustrated on different problem classes in Powell (2007). Godfrey and Powell (2002a, 2002b) present an ADP algorithm for a stochastic dynamic resource allocation problem with randomly arising tasks, whereas Spivey and Powell (2004) applied the ADP concept to dynamic assignment problems and Simão et al. (2009) present an ADP for dealing with a large-scale fleet management. Maxwell et al. (2010) and Schmid (2012) demonstrate the successful use of ADP algorithms for the dynamic ambulance relocation and dispatching problem under uncertainty. Other approaches using policies based on VFA are studied in Mes, van der Heijden, and Schuur (2013), Mes, van der Heijden, and Schuur (2010) where a decentralized auction-based approach is considered and the exploit of incorporating historical job information and auction results into the planning process is investigated. Ulmer, Brinkmann, and Mattfeld (2015) investigate an anticipatory cost-benefit heuristic and an ADP approach for decision-making for courier, express and parcel services yielding better results than the myopic approach. Another research group studies the problem of designing motion strategies for mobile agents, where new customers arise randomly and remain active for a certain amount of time and the objective is to meet as many customers as possible within their active time window. The problem at hand is formulated as a DVRPSC and discussed in Pavone and Frazzoli (2010) and Pavone et al. (2009), Pavone, Frazzoli, and Bullo (2011).

2. Online decisions

As the main goal is to handle real-world problems efficiently, research efforts on approaches considering *online decisions* for the DVRPSC have increased in recent years. The aim is to appropriately react to new events by introducing rules that can be easily computed in real-time. These rules consider future events in the decision-making process by generating scenarios of potential outcomes. A common concept, called *sample scenario approach (SSA)*, is to generate multiple scenarios of future customer requests and include them into the planning process. After selecting the most appropriate solution for the future, the sampled customers are removed, but with the effect that the new solution is well prepared for possible future requests. Bent and Van Hentenryck (2004) introduce the MSA, an event-driven sampling-based algorithm, where new solutions are generated regarding to a set of scenarios and maintained in the solution pool. The key idea behind MSA is to continuously generate and solve scenarios which include both static and dynamic requests. The way of deciding which customer has to be visited next is fundamental for the MSA. Decisions use a distinguished plan selected by a consensus function. From among these scenarios, they select and enact the plan with the most similarity to all of the other scenarios in terms of vehicle to job assignments. Experimental results indicate that MSA produces dramatic improvements over approaches not using stochastic information. They also indicate a strong synergy between MSA and the consensus function, especially for problems with many stochastic customers. Therefore, Van Hentenryck, Bent, and Upfal (2010) present three algorithms sharing the same offline optimization algorithm and sampling procedure but differing in the way of selecting the next customer at each decision step: the *online expectation* algorithm, the *consensus* algorithm and the *regret* algorithm. Further work is discussed in Bent and Van Hentenryck (2005), where approaches without distribution are presented: a *machine learning approach* to learn the distribution about requests during execution of the algorithm and a *historical data approach*, exploiting information about past instances and in Bent and Van Hentenryck (2007) where waiting and relocation strategies are investigated. Schilde, Doerner, and Hartl (2011) investigate the potential of using stochastic information about future return trips for the dynamic stochastic DARP applying two SSAs: a DSVNS and a MSA algorithm. It is shown that the incorporation of very near future information in the planning process (short-term sampling) is most beneficial. Another finding is that the DOD has a strong impact on the DSVNS, but the MSA is not affected as strongly. Another short-term SSA is presented in Ghiani et al. (2009), where near future requests are anticipated for a dynamic stochastic *pickup-and-delivery problem (PDP)*. Instead of using a scenario pool, an individual number of samples is determined and applied to the best alternative solutions. Out of them, a distinguished solution is selected based on its expected penalty. Van Hemert and Poutré (2004) propose an *evolutionary algorithm* for the DVRPSC, where the samples consist of request occurring in fruitful regions, which are clusters of known customer locations that are likely to require service in near future. Flatberg et al. (2007) introduce an approach similar to the MSA but using a simple similarity score as consensus function. Additionally, the authors show how statistical knowledge of events can be learned automatically from the past, but no results are presented. Hvattum, Løkketangen, and Laporte (2006) formulate a multi-stage stochastic model and implement a *dynamic stochastic hedging heuristic* because of the problem size which is a time-driven SSA solving each sample scenario with a static algorithm and determining a plan based on common aspects of the generated solutions. Zhang, Smilowitz, and Erera (2011) present a SSA and a capacity reservation approach, both designed as two-stage models within a rolling horizon framework for a *multi-resource routing problem*. In order to reduce computational effort, a small set of decision epochs is defined and instead of using probabilistic models in the two-stage approach, the sample average over a set of scenarios is optimized. Besides sampling-based approaches, other concepts like stochastic modelling or stochastic strategies (e.g. waiting strategies) are investigated for the DVRPSC. Yang, Jaillet, and Mahmassani (2004) introduce a rolling horizon-based real-time multi-vehicle truckload PDP and present a MIP formulation for the offline problem which is applied at every decision epoch in the online strategy. They demonstrate that a rolling horizon strategy that also includes, via opportunity costs, probabilistic information on future job pickup and drop-off locations outperforms the simple reactive strategies. Another MIP-based RH approach is presented in Kim, Mahmassani, and Jaillet (2004) where dynamic decisions about the acceptance or rejection of requests are made based on an approximation of future vehicle utilization regarding spatial and temporal information about future requests. Others use probabilistic spatial information in order to choose where the vehicles should idle in anticipation of future requests (Larsen et al. 2004, Thomas 2007). Larsen, Madsen, and Solomon (2004) present a rolling horizon approach where geographical and temporal aspects about customer requests are exploited in different strategies to reposition vehicles during idle times. In Ichoua, Gendreau, and Potvin (2006), a PFA algorithm utilizes a threshold-based waiting strategy which decides how long a vehicle should wait at its last customer location according to the probability of future requests in this area. Sáez, Cortés, and Núñez (2008) present a *hybrid adaptive predictive control* approach based on a genetic algorithm where the demand pattern is obtained by a zoning method based on a fuzzy clustering model. Huth and Mattfeld (2011) present an algorithm which allows dynamic truck allocation for the *stochastic swap container problem* and investigate strategies for anticipating future demand realization. Results show that the use of a probability distribution is more beneficial than using expected values for anticipating future demands. Ferrucci, Bock, and Gendreau (2013) present a rolling horizon approach for the 0delivery of newspapers and apply a temporal and spatial clustering of future requests which guides vehicles into request-likely areas (cf. Van Hemert and Poutré 2004). It is claimed that according to the comprised stochastic information of the clustering, solving only one scenario is adequate. Vonolfen and Affenzeller (2014) present a waiting heuristic which utilizes historical request information based on an intensity measure and Albareda-Sambola, Fernndez, and Laporte (2014) introduce an adaptive service policy for a multi-period DVRPSC, where an auxiliary prize collecting VRP is solved at each time period. Additionally, a VNS-based adaptive policy is applied to solve larger instances.

3. Conclusion

Most research has been done on the DVRPSC based on VFA and ADP algorithms for various problem structures on different real-world instances, which makes an objective comparison of the results difficult. The research on *online decision* procedures distinguishes between non-sampling and sampling-based approaches. Both methodologies demonstrate the benefit with improvement of incorporating stochastic information about future customer requests into the solution. According to the advance of information and communication technologies, future research for both groups will be performed. However, it is assumed that sampling-based approaches obtain more attention because of the advantage of not depending on an accurate probability distribution.

DVRP with multiple stochastic aspects

The vast majority of research on DSVRPs considers not more than one stochastic aspect. Only few works investigate the impact of incorporating multiple stochastic aspects, e.g. consideration of stochastic travel times and customers. Nowadays, the availability of data is increasing rapidly and better data analyses and probability models are coming into view. As a result, knowledge about the impact and benefit of using more stochastic information in DSVRPs is essential. Furthermore, the impact of various stochastic aspects and the benefit of combinations of stochastic information have to be examined. Chang,Wan, and OOI (2009) propose a heuristic solution approach for a just-in-time trucking service with stochastic travel times and service times which belongs to the group of preprocessed decision support. Cortés, Núñez, and Sáez (2008) extend the approach in Sáez, Cortés, and Núñez (2008) which considers stochastic customers, by additionally incorporating expected traffic conditions. The strategy approach is analyzed on scenarios with predictable and unpredictable congestion and results show an improvement of 2.1% compared to the myopic approach. In Bent and Van Hentenryck (2003), stochastic information about customers' service times is considered and the MSA algorithm is applied. The aim is to investigate the behavior of the MSA algorithm on a less constrained but more stochastic problem. Results show that travel times are reduced while not degrading the service level and it is shown that the MSA is robust when stochastic information is not entirely accurate. Attanasio et al. (2007) present an approach with zoning technology for a same day courier service and introduce a forecast and an allocation module, where reliable near future predictions of travel times and demand are generated and handed to the allocation module which assigns customer requests and relocates idle couriers. Based on Hvattum, Løkketangen, and Laporte (2006), Hvattum, Løkketangen, and Laporte (2007) present a *branch-and-regret heuristic (BRH)* for the DVRPSC which finds better results regarding the number of vehicles used but not regarding the travelled distance. Furthermore, the performance is investigated when the demand of known customers is uncertain as well. The additional difficulty does not affect the performance of the myopic approach (MDH), while the BRH performs worse when demands are stochastic, but performs better than the MDH but with a smaller gap. The algorithms are also compared to the best results in Bent and Van Hentenryck (2004), and it is shown that the MSA performs better than the BRH. Note that the comparison is not completely fair because the MSA is a event-based algorithm, which means that it is responded immediately to new requests, which is not the case in the time-driven BRH. Schilde

(2012) did extensive research on the impact of using stochastic information during the planning process and demonstrates the effect of incorporating the combination of stochastic aspects (future transports and travel times) into the DSVNS and MSA algorithm. Results show that in contrast to the better performing DSVNS for approaches with a single stochastic aspect, the MSA outperforms the DSVNS when multiple stochastic aspects are considered. It is also shown that the combination leads to a more stable approach, indicating that increasing the amount of stochastic information has a positive effect on the robustness of the results.

Overall conclusion

To sum up, as it seems through literature, the appropriate handling of dynamic events in conjunction with stochastic information about possible future events usually produces better results than the myopic or pure a-priori approaches. However, these approaches are computationally expensive which limits, e.g. the number of scenarios that can be considered. On the other hand, nowadays through advanced computers, satisfactory results can be obtained within reasonable time and it is a fact that technological progress will increase the potential of advanced methods even more. As a result, more qualitative data will occur and provide more reliable and robust information to optimization systems. Not only information from stochastic models using historical data, but also real-time information about traffic, requests, weather, etc. can be considered. This will allow treating more complex fleet management problems. However, problem size can increase and on the other hand, more complex data can be gathered, thus allowing for the combination of multiple stochastic aspects. The difficult task is to determine the influence each individual aspect can have and how the interaction between these parameters will look like. Furthermore, nowadays, because of the development of parallel algorithms, we can handle huge data and time-consuming methods, such as parameter estimation, sampling methods or background optimization procedures. Also, the impact and the limits of using complex stochastic information about multiple types of future events will be investigated more widely and the already developed methods for more classical dynamic and stochastic vehicle routing variants will be extended. Finally, a progress should be done about the comparability of results of several approaches, as each of them deals with its own real-world application and its own data which is usually private, and as a result it is hard to draw clear conclusions about the best approach.

2.2 Dial-a-Ride Problem (DARP)

The Dial-A-Ride Problem

A dial-a-ride system is an application of demand-dependent, collective people transportation (Cordeau and Laporte 2007). Each customer requests a trip between an origin and a destination, to which a number of service level requirements are linked. The service provider attempts to develop efficient vehicle routes and time schedules, respecting these requirements and the technical constraints of a pickup and delivery problem (Parragh et al. 2008).

Dial-a-ride systems offer efficient door-to-door transportation, especially for people with mobility difficulties (elderly or disabled). Furthermore, there is a variety of advantages not only for drivers because of their stress and driving cost reduction, but also for non-drivers due to the improvement of their mobility. Also, dial-a-ride systems contribute to decrease traffic as well as to reduce energy use and pollution.

The dial-a-ride problem with time windows (DARPTW) features the satisfaction of a set of travel requests, with predetermined pick-up and drop-off locations and desired pick-up and delivery time by the customer, by a fleet of vehicles from one or more depots. Each vehicle has a given capacity and the time each customer spends in the vehicle is bounded by a threshold. Most studies on the DARP assume a fleet of homogeneous vehicles based at a single depot. However, there may be several depots, especially in wide geographical areas, and the fleet might be sometimes heterogeneous. For example, some vehicles are designed to carry wheelchairs only, while others may only cater to ambulatory passengers and some are capable of accommodating both types of passenger.

Planning of these transport systems is difficult due to all the constraints that need to be considered. As mentioned before, all persons that need to be transported have their own pickup and drop-off location, a time window at pickup and/or at arrival and a maximum ride time, which is usually about an hour. Also, in many cases the passenger may also have a wheelchair or may need a special driver or an attendant. If a person needs to be transported to a school, day center or sheltered workshop, the time window is defined by the opening hours of this destination. On the other hand, if a person is using a less regular type of transport, there is usually a time window at the pickup location of 15 minutes before and 15 minutes after the desired pickup time, to give some planning flexibility.

All these constraints in conjunction with the rising demand make more and more difficult to design vehicle routes and time schedules manually and more constraints lead to a more inefficient solution. One can imagine that if all customers can express all their wishes, making a good planning becomes really difficult. Thus, algorithms are needed to assure cost efficiency and service quality. Optimization of dial-a-ride systems involves solving a class of complex vehicle routing problems with pickup and delivery with time windows (VRPPDTW). What makes the DARP different from most routing problems is the human perspective. When transporting passengers, reducing user inconvenience must be balanced against minimizing operating costs. In addition, vehicle capacity is normally constraining in the DARP whereas it is often redundant in PDVRP applications, particularly those related to the collection and delivery of letters and small parcels (Molenbrunch et al, 2017). The most common cost elements are connected with regular fleet size and operation, occasional use of extra vehicles, and drivers' wages. Moreover, quality service criteria include route duration, route length, customer waiting time, customer ride time (i.e. total time spent in vehicles), and difference between actual and desired delivery times. Some of these criteria may be treated as constraints or as part of the objective function. Therefore, it is essential to find the right balance between user satisfaction and reducing costs by optimizing the planning.

Healy and Moll (1995) showed that the DARP is NP-hard. As a result, much effort has been spent, over the years, in finding efficient and effective solution methods for this problem class. The static deterministic variants of the DARP are extensively studied and a variety of sophisticated solution approaches have been presented by the research community.

Over the years, literature has also provided purely dynamic or stochastic approaches and a combination of them. The increasing interest in this area results in a fast-growing body of research and multiple applications and settings arising from real-world problems, where not all requests are known in advance and the travel times are somewhat uncertain. Vehicles breakdowns, traffic, weather conditions and roadwork are factors that can influence the travel time between two locations. As a result, stochastics is included to deal with these uncertainties. If the travel time is modelled as a random variable with a mean and a variance, the model makes sure that both the arrival within the time window as well as the maximum ride time, are satisfied with a minimum reliability. In this variance several uncertain factors can be included, making the arrival time at a location vary each time. Also, uncertainties like rush hour can be modelled with time dependent travel times, which vary for each time of the day. Moreover, a combination of both, where the stochastic travel times are also time-dependent, could be useful. Another uncertainty is the arrival of the requests. Part of the requests are static, which means that they are known before the schedule is made. For example, regular requests that can be the same every day or week, as passengers going to their work or school. The other part consists of dynamic requests that arrive during the time horizon and need to be integrated into the static schedule. Because these requests arrive during the time horizon, their exact details are not known while making the static schedule. Sometimes a dynamic request cannot be integrated in the current schedule anymore. In this case the operator adds a new vehicle to the schedule, otherwise the dynamic request has to be rejected. Of course, both of these options are not desirable.

In the last years, researchers have focused on dynamic and stochastic points of view. When solving sophisticated vehicle routing problems (VRP), the goal is to offer reliable, flexible and robust solutions. The term robust means that the solution is minimally disturbed by unforeseen events. It may be disturbed, but it is wanted this disturbance to be as small as possible.

The advance of information, communication technologies and the available data allows to gather related information for advanced vehicle routing. Nowadays, real-time information about vehicles and their positions, vehicle breakdowns, traffic information, and up-to-date information about customer requests and respective demand is also available due to innovative technologies. These technologies offer great opportunities to transportation companies operating in a dynamic environment. As a result, an extensive data collection and real-time decision support in vehicle routing are now offered and companies can modify their operations, whenever necessary. The data can be captured, stored and then be processed and analyzed using modern statistics in order to provide useful stochastic information to the optimization algorithms. Thus, routing algorithms provide real-time solutions by incorporating online information and considering possible future events. One could categorize such events as events that introduce new elements to the problem like a new order, a new vehicle, or a failing vehicle. Other events do not introduce new elements, but they change existing parameters. A congestion event changes the travel times between locations, loading or unloading delays change the service times required by orders. Receiving an almost continuous flow of information, companies always try to find ways to use it appropriately in order to improve their decisions.

All these advances offer the opportunity to improve solution quality in real-time systems, but they need to be leveraged by efficient optimization algorithms for solving dynamic and stochastic vehicle routing problems (DSVRP).

Static and Deterministic dial-a-ride problem

The dial-a-ride problem was first introduced by Wilson et al. (1971). Since then a variety of papers was written about this problem. The majority of them deal with the static case of the problem, where all requests are known in advance. An often-used heuristic for the static dial-a-ride problem is a tabu search. This algorithm searches for the best solution in the neighborhood at each iteration. To avoid cycling, solutions can be declared ’tabu’ for a number of iterations. In Cordeau and Laporte (2003a) a tabu search heuristic is used with the neighborhood defined as all solutions that can be obtained by removing a request from one route and transferring it to another route. A combination of a mixed-integer program and a tabu search heuristic is used by Melachrinoudis et al. (2007). Other algorithms that are used for the static problem are an insertion heuristic (Jaw et al. (1986), Diana and Dessouky (2004), Luo and Schonfeld (2007)), a sacrificing algorithm (Healy and Moll (1995)), a grouping genetic algorithm (Rekiek et al. (2006)) and an adaptive large neighborhood search (Pisinger and Ropke (2007)). Although most of these papers take into account time windows and maximum riding time constraints, the authors often ignore other realistic constraints. Most papers use homogeneous vehicles and do not consider customers with wheelchair. One of the papers that assumes heterogeneous vehicles and more realistic constraints is Xiang et al. (2006). Similar realistic constraints are formulated in Parragh et al. (2012). In this paper a column generation algorithm is combined with variable neighborhood search to solve the static heterogeneous dial-a-ride problem with driver-related constraints. An overview of additional research on the static dial-a-ride problem is given by Berbeglia et al. (2007).

Standard DARP

Definition

A classic definition of the Dial-a-Ride Problem (DARP) has been established by Cordeau and Laporte (2003). The problem is about the design of a set of minimum-cost vehicle routes and time schedules in a graph of nodes and arcs, under a set of constraints, ensured that all customer requests are satisfied. Nodes refer to the pickup and delivery locations of customers, included the vehicle depot. Each directed link between two nodes is an arc, characterized by a travel time and an associated cost which is included if the arc is part of the solution. Furthermore, each route begins and ends at the depot within specific time intervals and it does not violate a maximum route duration. Additionally, the service for the customers at each location starts within a time window. The maximum customer ride time cannot be violated and so does the vehicle capacity. In order to have a right route design, priority and pairing of a customer's origin and destination should be respected by visiting them in the correct order with the same vehicle. Also, the term service time refers to the time required for loading and unloading customers. As indicated in the classification scheme of Parragh et al. (2008), this definition distinguishes the DARP from other problems in vehicle routing. Most closely connected is the pickup and delivery problem with time windows (PDPTW), which is also characterized by demand-dependent transportation between paired pickup and delivery locations. Nevertheless, the definition of the PDPTW is based on the transportation of goods and as a result, fewer quality constraints need to be met. Especially, the maximum customer ride time constraint is the basic characteristic of the DARP and the reason that makes the scheduling problem more complex. Sometimes, it might be required to delay the start of service in pickup nodes in order to decrease the ride time of the customer involved.

Mathematical formulations

Cordeau (2006) proposes an arc-based mixed-binary linear program, shown by Eqs. 1–14. The three-index binary decision variable x_{ij}^{k} indicates whether vehicle k traverses the arc between nodes i and j. Each vehicle route starts at an origin depot and ends at a destination depot (Eqs. 4, 6). One and the same vehicle should visit and leave matched pickup and delivery locations i and $n+i$ (Eqs. 2, 3, 5), ensuring flow conservation and pairing. Decision variable L_i^k refers to the ride time of customer i (Eq. 9) and cannot exceed the maximum customer ride time L (Eq. 12). Decision variables B_i^k and Q_i^k express the service start in node i (Eq. 7) and the load upon leaving node i (Eq. 8), respectively. They should respect the time window of node i (Eq. 11) and the capacity Q_k of vehicle k (Eq. 13), respectively. The time span between the time a vehicle leaves the origin depot and the time it returns to the destination depot cannot exceed the maximum route duration T_k (Eq. 10). A minimum-cost set of arcs is made (Eq. 1), subject to all constraints and full demand satisfaction.

$$\text{Minimize} \quad \sum_{k \in K} \sum_{i \in N} \sum_{j \in N} c_{ij}^k x_{ij}^k \tag{1}$$

Subject to

$$\sum_{k \in K} \sum_{j \in N} x_{ij}^k = 1 \quad \forall i \in P \tag{2}$$

$$\sum_{j \in N} x_{ij}^k - \sum_{j \in N} x_{n+i,j}^k = 0 \quad \forall i \in P,\ \forall k \in K \tag{3}$$

$$\sum_{j \in N} x_{0j}^k = 1 \quad \forall k \in K \tag{4}$$

$$\sum_{j \in N} x_{ji}^k - \sum_{j \in N} x_{ij}^k = 0 \quad \forall i \in P \cup D,\ \forall k \in K \tag{5}$$

$$\sum_{i \in N} x_{i,2n+1}^k = 1 \quad \forall k \in K \tag{6}$$

$$B_j^k \geq (B_i^k + d_i + t_{ij}) x_{ij}^k \quad \forall i \in N,\ \forall j \in N,\ \forall k \in K \tag{7}$$

$$Q_j^k \geq (Q_i^k + q_j) x_{ij}^k \quad \forall i \in N,\ \forall j \in N,\ \forall k \in K \tag{8}$$

$$L_i^k = B_{i+n}^k - (B_i^k - d_i) \quad \forall i \in P,\ \forall k \in K \tag{9}$$

$$B_{2n+1}^k - B_0^k \leq T_k \quad \forall k \in K \tag{10}$$

$$e_i \leq B_i^k \leq l_i \quad \forall i \in N,\ \forall k \in K \tag{11}$$

$$t_{i,i+n} \leq L_i^k \leq L \quad \forall i \in P,\ \forall k \in K \tag{12}$$

$$max\{0, q_i\} \leq Q_i^k \leq min\{Q_k, Q_k + q_i\} \quad \forall i \in N,\ \forall k \in K \tag{13}$$

$$x_{ij}^k \in \{0, 1\} \quad \forall i \in N,\ \forall j \in N,\ \forall k \in K \tag{14}$$

P = set of pickup nodes, D = set of delivery nodes, N = set of all nodes (including the depot nodes), K = set of vehicles, c_{ij} = cost associated with arc (i,j), t_{ij} = direct travel time associated with arc (i,j), d_i = service duration in node i , q_i = net number of customers boarding in node i.

Stochastic and dynamic point of view

The standard DARP makes two assumptions concerning the nature of the information available to the service provider. Firstly, information is assumed to be static, which means that all relevant data (e.g. requests, travel times, …) is known before the route planning process and remains unchanged during the entire time horizon. Secondly, the data is assumed to be deterministic, meaning that it is not subject to variability or uncertainty. However, both assumptions rarely hold in real-life systems. Most service providers face dynamic changes in inputs and external conditions tend to induce stochasticity into the system. This section presents different causes that complicate the availability of information and explains how the subject has been addressed in the literature.

1. Travel times

Mainly in urban areas, logistic transportation operations often face problems due to travel speeds change, depending on the current traffic situation and as a result it leads to missed time windows and poorer service quality. Especially in the case of passenger transportation, it often leads to excessive passenger ride times as well, which is more important when the transported passengers are medical patients or elderly people. As a consequence, time-dependent and stochastic influences on travel speeds are relevant for finding feasible and reliable solutions. Thus, travel speeds should be treated as stochastic if we want to represent reality more precisely. The majority of published articles concerning vehicle routing problems assume travel speeds that are constant over time (e.g., Muelas, LaTorre, & Peρa, 2013;; Paquette, Cordeau, Laporte, & Pascoal, 2013; Parragh and Schmid, 2013). In real world, travel speeds are not constant but they are determined by factors such as traffic congestion caused by rush hours, accidents, temporary construction sites, large one-time events or bad weather conditions. Thus, assuming that travel speeds are non-stochastic or even time-independent often leads planned schedules to fail with respect to time windows or ride time limitations. Some authors treat travel speeds as time dependent, by dividing each day into discrete time intervals, each of which provides a characteristic travel speed for each road within a network (Ehmke, Steinert, & Mattfeld, 2012; Fleischmann, Gietz, & Gnutzmann, 2004; Ichoua, Gendreau, & Potvin, 2003; Kok, Hans, Schutten, & Zijm, 2010; Lorini, Potvin, & Zufferey, 2011; Potvin, Xu, & Benyahia, 2006; Schmid & Doerner, 2010; Xiang, Chu, & Chen, 2008). Even these approaches assume travel speeds to be deterministic though, with the statement that travel speed, in terms of average values for each interval, is known a priori and not influenced by any stochastic effects. Also, others (Eglese, Maden, & Slater, 2006; Fleischmann, Gnutzmann, & Sandvoí, 2004; Maden, Eglese, & Black, 2010) use a different approach and incorporate time-dependent travel speeds in the process of calculating shortest paths. In Kok et al. (2010) the vehicle routing problem with time windows is extended with time-dependent travel times as well as driving hours regulations. However, these algorithms do not consider stochastic information about future travel speeds in order to get better solutions but they treat travel times as deterministic, so they are limited to react to changes in travel speeds by recalculating the shortest paths. In the case of stochastic travel times the travel times follow a certain distribution with known mean and variance. In this way unexpected changes because of for example weather conditions or accidents can be modelled. This is done by Li et al. (2010), where the stochastic constraints resulting from the stochastic travel times are transformed into deterministic constraints by using an approximation function of the normal distribution. After this, a tabu search-based heuristic is used to solve the problem. These two could also be combined in a stochastic dial-a-ride or vehicle routing problem with time-dependent travel times. In this case the mean and variance of the travel times are time-dependent. Fu (2002a), in one of the first papers on the stochastic dial-a-ride problem, argues that in an urban environment, a system's reliability can be increased considering stochastic and time-dependent travel times. They allow to account for traffic congestion and avoid that delays are accumulated. A normal distribution is assumed for the travel times on each arc. The average travel time varies with the precise departure time, whereas the corresponding standard deviation is assumed constant. A route is considered as a sequence of schedule blocks (Jaw et al. 1986) with zero variance at the start of each block. The expected start of service in a node can be computed recursively and should respect the time constraints with a given probability. Routes being feasible in a deterministic context may be rejected if they exhibit a large variance, which increases fleet requirements. To estimate the travel times, Fu and Teply (1999) suggest three approaches, based on zones, distance and an artificial neural network. Xiang et al. (2008) include stochastic travel and service times in a dynamic problem context. Moreover, their model is able to deal with unexpected changes, such as vehicle breakdowns and cancelations of requests. A paper using time-dependent stochastic travel times on the dynamic side is Schilde et al. (2014). They consider the effect of exploiting statistical information available about historical real world accidents, using stochastic solution approaches for the dynamic dial-a-ride problem (dynamic DARP). They propose two pairs of metaheuristic solution approaches, each consisting of a deterministic method (average time-dependent travel speeds for planning) and its corresponding stochastic version (exploiting stochastic information while planning). The accidents are modeled as gradually expanding and shrinking circles, causing congestion on the arcs they cover. Travel time consists of an average and a stochastic influence of accidents, both being time-dependent. Vehicles always use the shortest-distance path. The results show that in certain conditions, exploiting stochastic information about travel speeds leads to significant improvements over deterministic approaches.

2. Requests

Most authors assume that all requests are known in advance and as a result, static routes and schedules can be constructed. In the dynamic point of view, additional information may occur during the planning or the execution phase. The most studied case includes part of requests being received in real time. These customers either follow the usual reservation principle or ask for immediate service, in which case a maximum position shift may be imposed to respect the order of booking (Psaraftis 1980). The service provider should be able to decide immediately whether an additional request can be inserted (Attanasio et al. 2004). For this purpose, Berbeglia et al. (2012) present a constraint satisfaction problem formulation which can be used to prove the infeasibility of a problem. Hyytiä et al. (2010) point out the risk of congestive collapse when the capacity of the control policy is exceeded, which suddenly causes an unacceptably high rejection rate. Specific problem contexts involving additional requests may be considered. For example, Hanne et al. (2009), Beaudry et al. (2010) study transportation systems in a hospital context, where emergency requests should be serviced within a very limited time frame. Coslovich et al. (2006) focus on unexpected users asking for service during the stop of a vehicle. Cremers et al. (2009) consider subcontracting requests to taxi services during peak moments. The taxis are cheaper when booked one day in advance, but some requests are only revealed at the beginning of the operation day. For the dynamic version it is also possible to use an insertion heuristic, which is done by Madsen et al. (1995) and by Coslovich et al. (2006). Another approach was chosen by Teodorovic and Radivojevic (2000), who used fuzzy logic to formulate and solve the problem. An overview of additional research on the dynamic version of the problem is given by Berbeglia et al. (2010) and Pillac et al. (2013). Horn (2002) describes a software system to bridge the gap between static and dynamic approaches. Also, in Cordeau and Laporte (2007) several models for the static as well as the dynamic dial-a-ride problem are presented. An overview of both the static and dynamic version is given by Cordeau and Laporte (2003b). Except for additional requests, other unexpected events related to customers or vehicles may be taken into account, such as user no-shows, cancelations of requests, changes of requests, vehicle breakdowns and traffic jams (Donoso et al. 2009, Häme 2011). Especially the latter two may have a considerable operational impact (Xiang et al. 2008).

The response to new information remains crucial, in spite of technological advances, such as vehicle localization systems or increased processing power. As a consequence, most authors focus on fixing existing solutions, rather than repeatedly applying static solution methods. Generally, such repair heuristics first look for a feasible solution once new information is available. Afterwards, continuous optimization is performed until the next event occurs. Parallel computation may be applied (Attanasio et al. 2004). Parallelization strategies differ in whether control is executed by a single processor or distributed, whether new best solutions are communicated to other processors or not, and whether search parameters and initial solutions differ or not. Nevertheless, a problem’s real-time nature may heavily affect the efficiency with which it can be solved. This is reflected in the competitive ratio, being the worst-case proportion between an algorithmic result and the corresponding static optimum.

Minimizing the time interval needed to complete all available requests, Feuerstein and Stougie (2001), Ascheuer et al. (2000) compute lower bounds on this competitive ratio. Lipmann et al. (2004) include incomplete ride information, meaning that destinations only become known when their corresponding origin is visited. Feuerstein and Stougie (2001) also find lower bounds on the competitive ratio for a minimization of the *average* completion time. Minimizing the maximum flow time, Krumke et al. (2006) show that a solution method for a single vehicle with unit capacity cannot be competitive. Yi and Tian (2005) maximize the number of requests for which service starts within a fixed time period after their release. They provide lower bounds for the single-vehicle case with either unit capacity or infinite capacity. Yi et al. (2006) add restricted information and a finite capacity to the work of Yi and Tian (2005).

Apart from the dynamic DARP, problem variants with a limited availability of information may involve known requests with a stochastic or probabilistic nature. Schilde et al. (2011) observe that some users, such as patients in a hospital, may be unable to specify their return time in advance. Rather than considering such inbound trips as dynamic requests, a statistical distribution can be used to anticipate possible inbound trips at various times. Ho and

Haugland (2011) consider requests that are served with a given probability. In real-life cases, such a problem arises when fixed routes are executed on a regular basis, but users are absent with a known probability. For example, elderly people may not feel fit enough to go to the daycare center on a particular day. In this case, the order of the remaining nodes in the route remains unchanged. The authors construct routes such that expected costs are minimized, using a recourse function that takes into account the skipping probabilities.

3. Customer behavior

Stochastic customer behavior may also have an impact to reliability. Heilporn et al. (2011) consider customers showing up late at their pickup location. In this case, the on-demand vehicle leaves at the scheduled time and a taxi is called in, causing a cost which exceeds the savings of skipping the corresponding delivery node. An arc-based mixed-binary linear program for this problem variant includes an expected delay cost. The probability of being late decreases as a customer is visited later in his pickup time window and thus also depends on the probability of skipping preceding nodes. Some customers may be scheduled early in their time window in order to serve the majority of users as late as possible. This may generate considerable savings over a deterministic optimum with expected delay costs, even if the scheduling procedure is adapted. Deflorio et al. (2002) discuss lateness of both customers and drivers, the latter due to time-dependent variability of travel times and unforeseen waiting times. Decreasing the variance on how long drivers decide to wait for late customers increases the number of requests met.

Chapter 3. ALGORITHMS

Our algorithm set up is based on the one proposed by Lois (2016), "*On the Online Dial-a-Ride Problem*. LAP LAMBERT Academic Publishing". This section gives a brief overview of the algorithms used, but more details can be found in his dissertation. The proposed algorithms were implemented in C++.

3.1 Static Dial-a-Ride Algorithms | The Insertion Algorithm InsertionH

1. Basic concept

The insertion algorithm that has been developed is a variation of the one presented by Jaw (1986). The basic function of this insertion algorithm is to make an initial assignment for each trip. This solution can be refined later by other optimization heuristics.

2. Nomenclature

***V* = {1,2,3,.....,|*V*|} =** *Set of Vehicles*

$\boldsymbol{R}_{v\{v=1,2,\ldots,|V|\}}$ = *is the route schedule for every* $v \in V$

$\boldsymbol{RC}_{v\{v=1,2,\ldots,|V|\}}$ = *is the route cost for the for every* $v \in V$

N = *is the number of trip requests. Each trip request is consisted by one pickup and one deliver node.*

$\boldsymbol{TR}_{i\{i=1,2\ldots,N\}}$ = *is the sequence of trip demands.*

$\boldsymbol{EPTi}_{\{i=1,2,\ldots,N\}}$ = *Earlier pickup time of trip request i*

$\mathbf{BIP}_{iv}$ = *the best insertion position of i-th trip request* TR_i *to Route schedule* R_v

$\mathbf{MinCostBIP}_{iv}$ = *is the cost of the best insertion position of the trip request* TR_i *to Route schedule* R_v

3. Algorithm description

In a preprocessing step the algorithm creates an empty route for every vehicle. Then for every trip request the algorithm finds the best position -in terms of cost- to insert the trip by searching all vehicle routes. If this position is found, then the trip request is inserted; otherwise, if it is not possible to find at least one position where the request could be inserted, the trip request is rejected. At the end of the execution, specific trips have been assigned to specific vehicles. The main drawback of this algorithm is that it always tends to load the first vehicle with more demands. That is why we implemented a variation where the search for each demand starts with a different vehicle each time. A pseudo code description of InsertionH algorithm follows:

Step0: for every ***v*** *in* ***V*** *build an empty* $\boldsymbol{R_v}$
Step1: Sort $\boldsymbol{R_v}$ *in descending order according to* $\boldsymbol{RC_v}$
Step2: Sort $\boldsymbol{TR_i}$ *in ascending order according to demands* $\boldsymbol{EPT_i}$
Step3: for every $\boldsymbol{TR_i}$ ***{i=1,2,..,N}***
  Step3.1: Sort $\boldsymbol{R_v}$ *in ascending order according to* $\boldsymbol{RC_v}$
  Step3.2:for every $\boldsymbol{R_v}$ ***{v=1,2,,,,|V|}*** *do*
    find the $\boldsymbol{BIP_{iv}}$
    Calculate $\boldsymbol{MinCostBIP_{iv}}$
  Step3.3: If no $\boldsymbol{BIP_{iv}}$ ***{v=1,2,,,,|V|}*** *found go to step 3.6*
  Step3.4: From all $\boldsymbol{MinCostBIP_{iv}}$ *select the minimum one*
Step3.5: Assign that $\boldsymbol{TR_i}$ *to the appropriate* $\boldsymbol{R_v}$ *and go to step3*
Step3.6: Reject that trip request and repeat step3
end for

4. Algorithm computational effort

Computational effort of the InsertionH algorithm can be simply calculated by the number of possible insertions that the algorithm should check in order to find the best position in terms of solution cost (given that this is the objective). The worst scenario is the case that only one vehicle is available. The best scenario is the case that the number of vehicles is equal to the number of trip requests and each vehicle has one trip request assigned.

Given a number of n demands – not including the first demand - and one available vehicle, the total number of possible searches for the insertion algorithm can be calculated as the following sum:

$$1+\sum_{i=1}^{n}\sum_{k=1}^{2*i+1}k \quad .$$

The closed form of this expression is

$$1+\frac{1}{6}n(17+15n+4n^2) \quad .$$

The algorithm complexity is $O(n^3)$. For a total number of 1000 demands the worst scenario (one vehicle) gives a total number of 669,169,501 possible trip combinations.

3.2 Static Dial-a-Ride Algorithms | The Static Regret based heuristic algorithm RegretH

1. Basic concept

The Regret based Heuristic algorithm aims to reduce the chances to get trapped in a severely suboptimal solution. It does this by considering the opportunity which is defined as the difference between the cost actually calculated by the algorithm (total distance in our case) and the cost calculated for a better position that could be obtained if a "different" course of action had been chosen. In our case the term "different" course can be defined as the calculation of the best position to which an already assigned demand can be reassigned to produce a better solution in terms of cost. This reassignment could be intra-route or inter-route. The basic idea is to use a fast algorithm in order to quickly get a feasible suboptimal solution. Usually this is an insertion algorithm. The insertion algorithm gives us a myopic sub-optimal solution based on the positions of the previously assigned demands. Then, based on the existing solution, we try to find a better solution by reassigning the most expensive demands. The reallocation which provides the largest gain in the system is chosen. Unlike the tabu-search algorithm, regret algorithm never explores non-feasible solutions. The advantage of the regret algorithm is that it has much less computational burden. However, this algorithm, like any heuristic algorithm, is prone to getting trapped in local minima.

2. Nomenclature

$V = \{1,2,3,\ldots,|V|\}$ = Set of Vehicles

$R_{V\{v=1,2,\ldots,|V|\}}$ = is the route schedule for every $v \in V$

$|R_v|$ = size of route schedule defined as the number of demands included in this route

$RC_{v\{v=1,2,\ldots,|V|\}}$ = is the route cost

$TRC = \sum_{1}^{|V|} RC_v$ is the total cost for all routes

$DR_{kv\{k=1,2,\ldots,|Rv|\},\{v=1,2,\ldots,|V|\}}$ = the assigned k demand belongs to route v

$DMaxR_{v\{v=1,2,\ldots,|V|\}}$ = the maximum demand cost concerning route v

$RCostDMaxR_{vm\{v=1,2,\ldots,|V|\},\{l=1,2,\ldots,|V|\},v<>l}$ = is the extra cost incurred in route m, if we move the demand (with the maximum cost) that belongs to route v to the route m

$CostMatrix[v,m]_{\{m=1,2,\ldots,|V|\},\{v=1,2,\ldots,|V|\}}$ = is 2-d matrix cost for the Regret Algorithm. Diagonal positions CostMatrix[v,v] contains $DMaxR_v$ while other positions $CostMatrix[v,m]_{\{m<>v\}\{m=1,2,\ldots,|V|\},\{v=1,2,\ldots,|V|\}}$ contains $RCostDMaxR_{vm}$

$RegretCostMatrix[v]_{\{v=1,2,\ldots,|V|\}}$ = is the one dimensional matrix containing the maximum value of the difference between the element with the least $RCostDMaxR_{vm\{m=1,2,\ldots,|V|\}}$ and the corresponding $DMaxR_v$.

3. Algorithm description

The main objective is the minimization of the total cost for all vehicles. Our regret algorithm uses the original regret concept where the absolute difference between the “best” lower cost and the second “best” lower cost alternative is used as a metric for guiding problem solving.

Our regret algorithm uses a fast heuristic in order to produce an initial solution. The fast heuristic we use is the InsertionH heuristic mentioned before. Then we start the regret process. We build the regret matrix.

Each row represents the route produced by the initial heuristic.

Each diagonal item in the 2-d matrix represents the cost of the most expensive demand of the route described by the corresponding row.

Each element of a column (except the one on the diagonal) represents the cost that will be produced if we insert the aforementioned most expensive demand to the route represented by the corresponding row.

If the insertion is not feasible, then we set the insertion cost to an arbitrarily large number.

By using this matrix, we define the “profit” we gain if we move demands from one route to another. This can be done by using the following rule: for every column calculate the differences of diagonal element minus every other element. If at least one difference is positive select the greatest one. Then move the corresponding demand to the appropriate row (route in our case).

The Algorithm pseudo code follows:

Step0: Run the InsertionH Algorithm

*Calculate **TRC***

NewTRC= TRC

Step 1:While (TRC-NewTRC>0)

TRC= NewTRC

*Step 1.1: for every **v** in **V***

for every $\boldsymbol{DR_{kv\ \{k=1,2,\ldots|RV|\}}}$

find $\boldsymbol{DMaxR_v}$

$\boldsymbol{CostMatrix[v,v] = DMaxR_v}$

Step 1.2: for every $\boldsymbol{DMaxR_{v\ \{v=1,2,\ldots|V|\}}}$

*for every **m** in **V***

Find $\boldsymbol{RCostDMaxR_{v,m\{v=1,2,,,,|V|\},\{m=1,2,\ldots|V|\},\ v<>m}}$

$\boldsymbol{CostMatrix[v,m]= RCostDMaxR_{vm}}$

*Step 1.3: for every v in **V***

$\boldsymbol{RegretCostMatrix[v]= positive\ Max(DMaxR_v - CostMatrix[v,m])_{\{m=1,2,\ldots|V|\}}}$

the greatest positive difference.

Move demand $\boldsymbol{DMaxR_v}$ *from route v to route m*

Step 1.4: Calculate new TRC

4. Algorithm computational effort

Computational effort of the RegretH, given n demands, is the sum of the computational effort of:

1. InsertionH algorithm used in the initial phase. This computational effort has already been calculated as $O(n^3)$.

2. The procedure that gives the most expensive demand of any route. This computational effort is obviously $O(n)$. This procedure removes temporarily one by one every demand and then calculates the solution cost produced in the absence of this demand. The difference between the previous solution cost and current solution cost is the cost of every demand in the solution.

3. The procedure that gives the additional cost that will be produced if we insert the most expensive demand of a route to another route. Consider a feasible solution produced by the InsertionH algorithm for n trip demands and m vehicle routes. The computational effort for this solution is the sum of all possible searches for the best point to insert the most expensive trip demand of every vehicle route to another vehicle routes. The number of possible searches is given by formula

$$\sum_{k=1}^{2(n-1)+1} k = (2n^2 - n)$$

. For all m vehicle routes the number of possible searches – the worst case - is $m(2n^2 - n)$. Consequently, the computational effort is $O(mn^2)$.

The two aforementioned procedures are executed repetitively while the solution is optimized. The number of repetitions is unknown but larger than one. Taking r, the number of repetitions, the final computational load is $O(n^3)+rO(mn^2)$. The order of r as a function of m,n wasn't calculated and is therefore considered unknown. The total computational effort is a function of number of repetitions and number of demands. The main factor affecting the computational effort is the number of demands.

Chapter 4. STATISTICAL DATA ANALYSIS

In this section we present our statistical analysis concerning a large sized data set obtained on the city of Volos, Greece.

4.1 Data Set – Data Provider

In order to test the proposed model, contacts with the only taxi company in the city of Volos, Greece, have taken place. The VOLOSTAXI company has a fleet of 200 taxis out of 250 taxis that drive around in the city of Volos. In consequence of our contacts, a big set of data is available for use. This set consists of 959780 requests for transport in the region of Volos for the last 4 years (2016 - 2019). Moreover, the geographically related data has been displayed on Google maps in order to be clearer and more intuitive. In this way, the distribution of data in each different region is visual (see Figure 1).

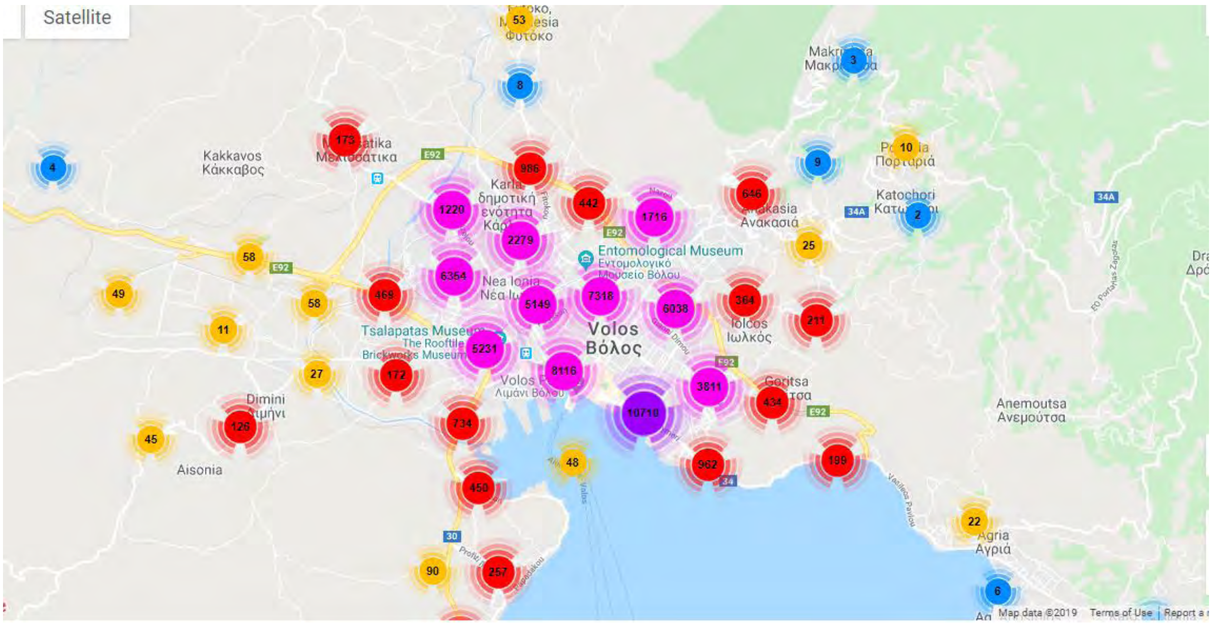

Figure 1. Distribution of data in each different region of Volos

More precisely, for the purposes of our research, we selected a region to study based on where most requests occurred (see Figure 2).

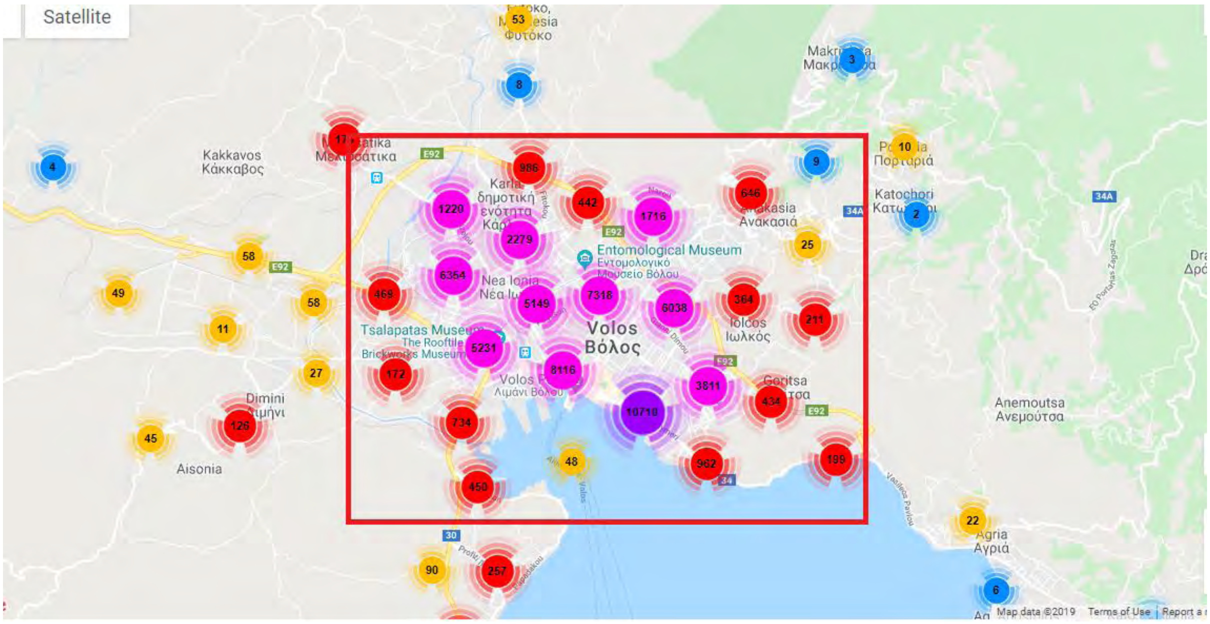

Figure 2. Selected region of Volos where most requests occurred.

After that, we divided the selected region into 4x4=16 equal and discrete subregions. Each subregion's longitude and latitude coordinates (diagonal coordinates – topleft, bottomright) and its centroid's coordinates are given in Table 1. Furthermore, the centroids have been displayed on Google maps for the sake of convenience (see Figure 3).

Region	Lon(top_left)	Lat(top_left)	Lon*(bottom_right)	Lat*(bottom_right)	Diagonal	Lon_Centroid	Lat_Centroid
1	22.908294	39.392861	22.929046	39.380934		22.91867	39.386898
2	22.908294	39.380934	22.929046	39.369007		22.91867	39.374971
3	22.908294	39.369007	22.929046	39.35708		22.91867	39.363044
4	22.908294	39.35708	22.929046	39.345153		22.91867	39.351117
5	22.929046	39.392861	22.949798	39.380934		22.939422	39.386898
6	22.929046	39.380934	22.949798	39.369007		22.939422	39.374971
7	22.929046	39.369007	22.949798	39.35708		22.939422	39.363044
8	22.929046	39.35708	22.949798	39.345153		22.939422	39.351117
9	22.949798	39.392861	22.970551	39.380934		22.960175	39.386898
10	22.949798	39.380934	22.970551	39.369007		22.960175	39.374971
11	22.949798	39.369007	22.970551	39.35708		22.960175	39.363044
12	22.949798	39.35708	22.970551	39.345153		22.960175	39.351117
13	22.970551	39.392861	22.991303	39.380934		22.980927	39.386898
14	22.970551	39.380934	22.991303	39.369007		22.980927	39.374971
15	22.970551	39.369007	22.991303	39.35708		22.980927	39.363044
16	22.970551	39.35708	22.991303	39.345153		22.980927	39.351117

Table 1. Longitude and latitude coordinates for each different subregion and its centroid

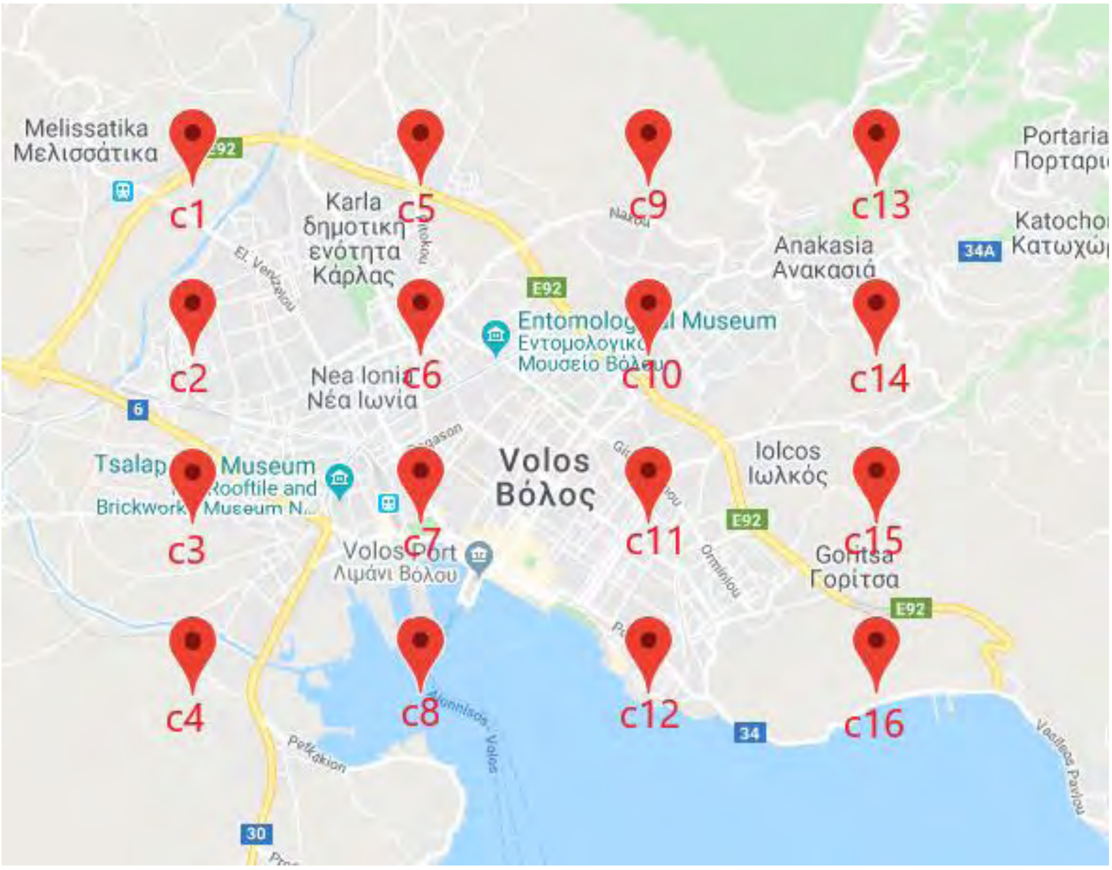

Figure 3. The centroid of each discrete subregion

4.2 Statistical Analysis

1. Subregion based analysis concerning the total amount of requests

We stored our data in an Access database in order to manage and analyze all the available information. Additionally, we wanted to know how many requests occurred in each different subregion.

One can see that concerning the total amount of requests (959780 requests over the last 4 years), only a small proportion (2.71%) is not included in any of the 16 subregions (see Table 2). Also, it can be obviously seen that the vast majority of the total requests occurred in subregion 7 and 11 – 20.57% and 22.70% respectively. Last but not least, subregions 2, 6 and 10 have attained considerable proportions – 9.72%, 14.23% and 9.41% respectively (see Table 2 in conjunction with Figure 3 and 4).

area	counter
Area-	
Summary for 'area' = Area->01_22.908294_39.392861_22.929046_39.380934 (11189 detail records)	
Sum	11189
Standard	1.17%
Area-	
Summary for 'area' = Area->02_22.908294_39.380934_22.929046_39.369007 (93319 detail records)	
Sum	93319
Standard	9.72%
Area-	
Summary for 'area' = Area->03_22.908294_39.369007_22.929046_39.35708 (45136 detail records)	
Sum	45136
Standard	4.70%
Area-	
Summary for 'area' = Area->04_22.908294_39.35708_22.929046_39.345153 (12948 detail records)	
Sum	12948
Standard	1.35%
Area-	
Summary for 'area' = Area->05_22.929046_39.392861_22.949798_39.380934 (27977 detail records)	
Sum	27977
Standard	2.91%
Area-	
Summary for 'area' = Area->06_22.929046_39.380934_22.949798_39.369007 (136551 detail records)	
Sum	136551
Standard	14.23%
Area-	
Summary for 'area' = Area->07_22.929046_39.369007_22.949798_39.35708 (197390 detail records)	
Sum	197390
Standard	20.57%
Area-	
Summary for 'area' = Area->08_22.929046_39.35708_22.949798_39.345153 (1768 detail records)	
Sum	1768
Standard	0.18%
Area-	
Summary for 'area' = Area->09_22.949798_39.392861_22.970551_39.380934 (12832 detail records)	
Sum	12832
Standard	1.34%
Area-	
Summary for 'area' = Area->10_22.949798_39.380934_22.970551_39.369007 (90274 detail records)	
Sum	90274
Standard	9.41%
Area-	
area	counter
Summary for 'area' = Area->11_22.949798_39.369007_22.970551_39.35708 (217907 detail records)	
Sum	217907
Standard	22.70%
Area-	
Summary for 'area' = Area->12_22.949798_39.35708_22.970551_39.345153 (62866 detail records)	
Sum	62866
Standard	6.55%
Area-	
Summary for 'area' = Area->13_22.970551_39.392861_22.991303_39.380934 (4360 detail records)	
Sum	4360
Standard	0.45%

Area-	
Summary for 'area' = Area->14_22.970551_39.380934_22.991303_39.369007 (6329 detail records)	
Sum	6329
Standard	0.66%
Area-	
Summary for 'area' = Area->15_22.970551_39.369007_22.991303_39.35708 (8734 detail records)	
Sum	8734
Standard	0.91%
Area-	
Summary for 'area' = Area->16_22.970551_39.35708_22.991303_39.345153 (4163 detail records)	
Sum	4163
Standard	0.43%
N/A	
Summary for 'area' = N/A (26037 detail records)	
Sum	26037
Standard	2.71%
Grand Total	959780

Table 2. The proportion of the total amount of requests in each different subregion

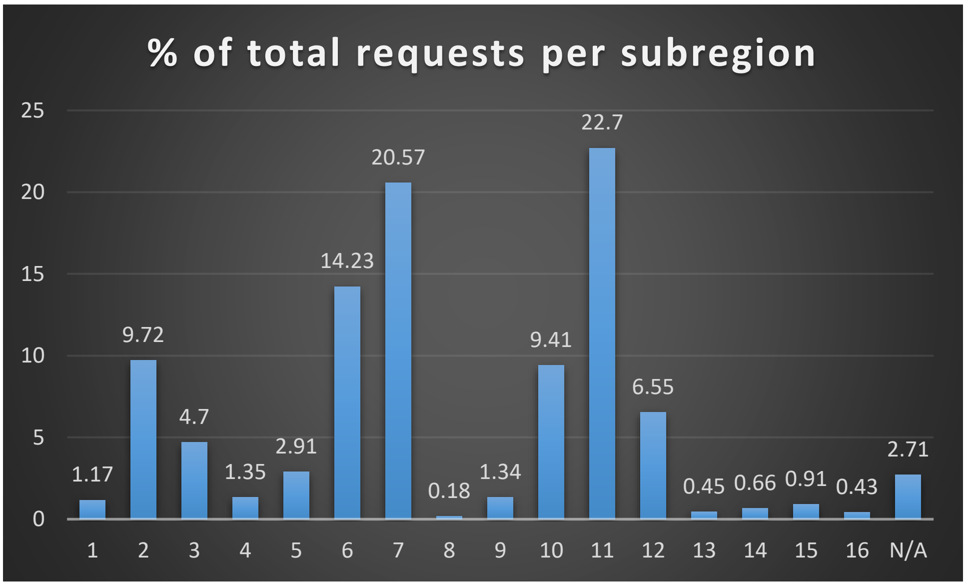

Figure 4. The percentage of the total amount of requests per subregion

2. *Subregion based analysis concerning the weekday*

Another interesting point of view is to examine the distribution of data in each different subregion regarding the weekday. Detailed information about each different day is as follows:

a) Sundays (see Table 3, Figure 5)
Total amount of requests: 115234
Subregions with the largest proportions: 7 (20.64%), 11 (21.29%)
Following up: 2 (9.74%), 6 (14.13%), 10 (9.31%)

area	counter
Area-	
Summary for 'area' = Area->01_22.908294_39.392861_22.929046_39.380934 (1022 detail records)	
Sum	1022
Standard	0.89%
Area-	
Summary for 'area' = Area->02_22.908294_39.380934_22.929046_39.369007 (11227 detail records)	
Sum	11227
Standard	9.74%
Area-	
Summary for 'area' = Area->03_22.908294_39.369007_22.929046_39.35708 (5190 detail records)	
Sum	5190
Standard	4.50%
Area-	
Summary for 'area' = Area->04_22.908294_39.35708_22.929046_39.345153 (1101 detail records)	
Sum	1101
Standard	0.96%
Area-	
Summary for 'area' = Area->05_22.929046_39.392861_22.949798_39.380934 (3249 detail records)	
Sum	3249
Standard	2.82%
Area-	
Summary for 'area' = Area->06_22.929046_39.380934_22.949798_39.369007 (16284 detail records)	
Sum	16284
Standard	14.13%
Area-	
Summary for 'area' = Area->07_22.929046_39.369007_22.949798_39.35708 (23787 detail records)	
Sum	23787
Standard	20.64%
Area-	
Summary for 'area' = Area->08_22.929046_39.35708_22.949798_39.345153 (451 detail records)	
Sum	451
Standard	0.39%
Area-	
Summary for 'area' = Area->09_22.949798_39.392861_22.970551_39.380934 (1599 detail records)	
Sum	1599
Standard	1.39%
Area-	
Summary for 'area' = Area->10_22.949798_39.380934_22.970551_39.369007 (10732 detail records)	
Sum	10732
Standard	9.31%
Area-	
Summary for 'area' = Area->11_22.949798_39.369007_22.970551_39.35708 (24537 detail records)	
Sum	24537
Standard	21.29%
Area-	
Summary for 'area' = Area->12_22.949798_39.35708_22.970551_39.345153 (8827 detail records)	
Sum	8827
Standard	7.66%
Area-	
Summary for 'area' = Area->13_22.970551_39.392861_22.991303_39.380934 (676 detail records)	
Sum	676
Standard	0.59%

Area-	
Summary for 'area' = Area->14_22.970551_39.380934_22.991303_39.369007 (751 detail records)	
Sum	751
Standard	0.65%
Area-	
Summary for 'area' = Area->15_22.970551_39.369007_22.991303_39.35708 (926 detail records)	
Sum	926
Standard	0.80%
Area-	
Summary for 'area' = Area->16_22.970551_39.35708_22.991303_39.345153 (483 detail records)	
Sum	483
Standard	0.42%
N/A	
Summary for 'area' = N/A (4392 detail records)	
Sum	4392
Standard	3.81%
Grand Total	115234

Table 3. The proportion of the total amount of requests that occurred on Sundays in each different subregion

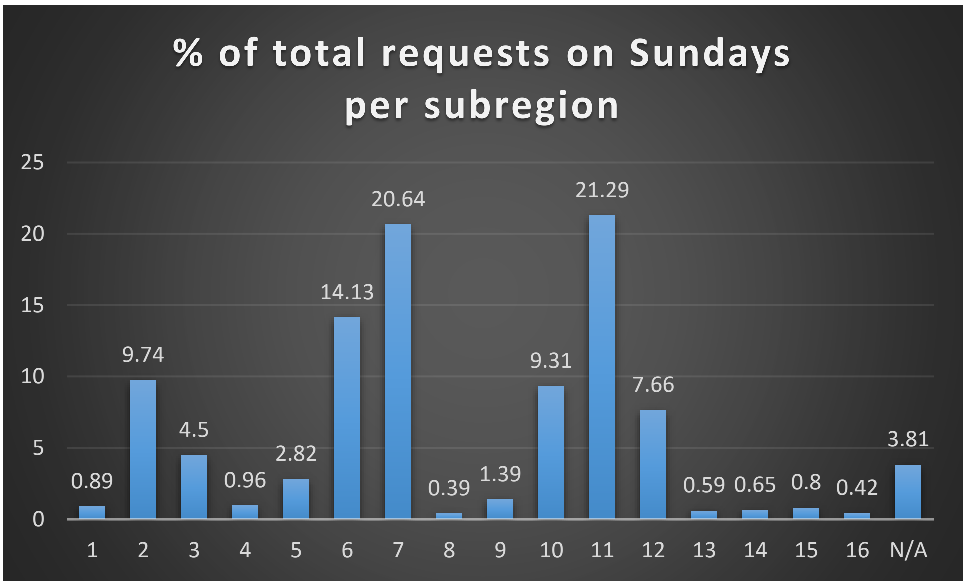

Figure 5. The percentage of the total amount of requests on Sundays per subregion

b) <u>Mondays</u> (see Table 4, Figure 6)
Total amount of requests: 137143
Subregions with the largest proportions: 7 (20.64%), 11 (22.56%)
Following up: 2 (9.57%), 6 (13.94%), 10 (9.47%)

area	counter
Area-	
Summary for 'area' = Area->01_22.908294_39.392861_22.929046_39.380934 (1736 detail records)	
Sum	1736
Standard	1.27%
Area-	
Summary for 'area' = Area->02_22.908294_39.380934_22.929046_39.369007 (13131 detail records)	
Sum	13131
Standard	9.57%
Area-	
Summary for 'area' = Area->03_22.908294_39.369007_22.929046_39.35708 (7021 detail records)	
Sum	7021
Standard	5.12%
Area-	
Summary for 'area' = Area->04_22.908294_39.35708_22.929046_39.345153 (2190 detail records)	
Sum	2190
Standard	1.60%
Area-	
Summary for 'area' = Area->05_22.929046_39.392861_22.949798_39.380934 (4231 detail records)	
Sum	4231
Standard	3.09%
Area-	
Summary for 'area' = Area->06_22.929046_39.380934_22.949798_39.369007 (19113 detail records)	
Sum	19113
Standard	13.94%
Area-	
Summary for 'area' = Area->07_22.929046_39.369007_22.949798_39.35708 (28302 detail records)	
Sum	28302
Standard	20.64%
Area-	
Summary for 'area' = Area->08_22.929046_39.35708_22.949798_39.345153 (261 detail records)	
Sum	261
Standard	0.19%
Area-	
Summary for 'area' = Area->09_22.949798_39.392861_22.970551_39.380934 (1813 detail records)	
Sum	1813
Standard	1.32%
Area-	
Summary for 'area' = Area->10_22.949798_39.380934_22.970551_39.369007 (12988 detail records)	
Sum	12988
Standard	9.47%
Area-	
Summary for 'area' = Area->11_22.949798_39.369007_22.970551_39.35708 (30944 detail records)	
Sum	30944
Standard	22.56%
Area-	
Summary for 'area' = Area->12_22.949798_39.35708_22.970551_39.345153 (8377 detail records)	
Sum	8377
Standard	6.11%
Area-	
Summary for 'area' = Area->13_22.970551_39.392861_22.991303_39.380934 (631 detail records)	
Sum	631
Standard	0.46%

Area-	
Summary for 'area' = Area->14_22.970551_39.380934_22.991303_39.369007 (881 detail records)	
Sum	881
Standard	0.64%
Area-	
Summary for 'area' = Area->15_22.970551_39.369007_22.991303_39.35708 (1229 detail records)	
Sum	1229
Standard	0.90%
Area-	
Summary for 'area' = Area->16_22.970551_39.35708_22.991303_39.345153 (601 detail records)	
Sum	601
Standard	0.44%
N/A	
Summary for 'area' = N/A (3694 detail records)	
Sum	3694
Standard	2.69%
Grand Total	137143

Table 4. The proportion of the total amount of requests that occurred on Mondays in each different subregion

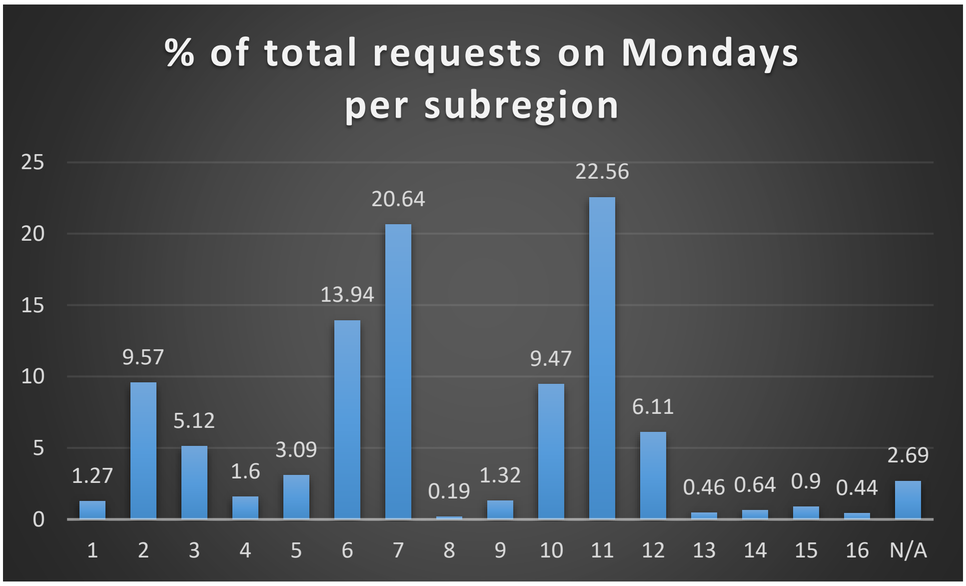

Figure 6. The percentage of the total amount of requests on Mondays per subregion

c) <u>Tuesdays</u> (see Table 5, Figure 7)
Total amount of requests: 138399
Subregions with the largest proportions: 7 (20.16%), 11 (23.03%)
Following up: 2 (9.99%), 6 (14.25%), 10 (9.48%)

area	counter
Area-	
Summary for 'area' = Area->01_22.908294_39.392861_22.929046_39.380934 (1715 detail records)	
Sum	1715
Standard	1.24%
Area-	
Summary for 'area' = Area->02_22.908294_39.380934_22.929046_39.369007 (13824 detail records)	
Sum	13824
Standard	9.99%
Area-	
Summary for 'area' = Area->03_22.908294_39.369007_22.929046_39.35708 (6794 detail records)	
Sum	6794
Standard	4.91%
Area-	
Summary for 'area' = Area->04_22.908294_39.35708_22.929046_39.345153 (2087 detail records)	
Sum	2087
Standard	1.51%
Area-	
Summary for 'area' = Area->05_22.929046_39.392861_22.949798_39.380934 (4045 detail records)	
Sum	4045
Standard	2.92%
Area-	
Summary for 'area' = Area->06_22.929046_39.380934_22.949798_39.369007 (19718 detail records)	
Sum	19718
Standard	14.25%
Area-	
Summary for 'area' = Area->07_22.929046_39.369007_22.949798_39.35708 (27908 detail records)	
Sum	27908
Standard	20.16%
Area-	
Summary for 'area' = Area->08_22.929046_39.35708_22.949798_39.345153 (176 detail records)	
Sum	176
Standard	0.13%
Area-	
Summary for 'area' = Area->09_22.949798_39.392861_22.970551_39.380934 (1824 detail records)	
Sum	1824
Standard	1.32%
Area-	
Summary for 'area' = Area->10_22.949798_39.380934_22.970551_39.369007 (13122 detail records)	
Sum	13122
Standard	9.48%
Area-	
area	counter
Summary for 'area' = Area->11_22.949798_39.369007_22.970551_39.35708 (31876 detail records)	
Sum	31876
Standard	23.03%
Area-	
Summary for 'area' = Area->12_22.949798_39.35708_22.970551_39.345153 (8442 detail records)	
Sum	8442
Standard	6.10%
Area-	
Summary for 'area' = Area->13_22.970551_39.392861_22.991303_39.380934 (581 detail records)	
Sum	581
Standard	0.42%

Area-	
Summary for 'area' = Area->14_22.970551_39.380934_22.991303_39.369007 (904 detail records)	
Sum	904
Standard	0.65%
Area-	
Summary for 'area' = Area->15_22.970551_39.369007_22.991303_39.35708 (1353 detail records)	
Sum	1353
Standard	0.98%
Area-	
Summary for 'area' = Area->16_22.970551_39.35708_22.991303_39.345153 (618 detail records)	
Sum	618
Standard	0.45%
N/A	
Summary for 'area' = N/A (3412 detail records)	
Sum	3412
Standard	2.47%
Grand Total	138399

Table 5. The proportion of the total amount of requests that occurred on Tuesdays in each different subregion

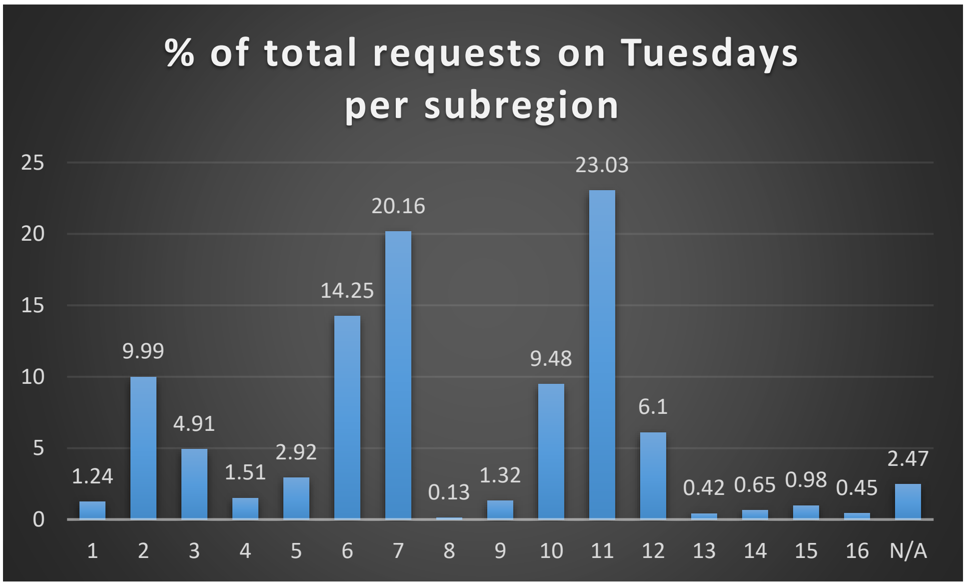

Figure 7. The percentage of the total amount of requests on Tuesdays per subregion

d) Wednesdays (see Table 6, Figure 8)
Total amount of requests: 135813
Subregions with the largest proportions: 7 (20.19%), 11 (23.06%)
Following up: 2 (9.50%), 6 (14.24%), 10 (9.52%)

area counter

Area-

Summary for 'area' = Area->01_22.908294_39.392861_22.929046_39.380934 (1681 detail records)

Sum	1681
Standard	1.24%

Area-

Summary for 'area' = Area->02_22.908294_39.380934_22.929046_39.369007 (12909 detail records)

Sum	12909
Standard	9.50%

Area-

Summary for 'area' = Area->03_22.908294_39.369007_22.929046_39.35708 (6545 detail records)

Sum	6545
Standard	4.82%

Area-

Summary for 'area' = Area->04_22.908294_39.35708_22.929046_39.345153 (1989 detail records)

Sum	1989
Standard	1.46%

Area-

Summary for 'area' = Area->05_22.929046_39.392861_22.949798_39.380934 (4088 detail records)

Sum	4088
Standard	3.01%

Area-

Summary for 'area' = Area->06_22.929046_39.380934_22.949798_39.369007 (19339 detail records)

Sum	19339
Standard	14.24%

Area-

Summary for 'area' = Area->07_22.929046_39.369007_22.949798_39.35708 (27419 detail records)

Sum	27419
Standard	20.19%

Area-

Summary for 'area' = Area->08_22.929046_39.35708_22.949798_39.345153 (224 detail records)

Sum	224
Standard	0.16%

Area-

Summary for 'area' = Area->09_22.949798_39.392861_22.970551_39.380934 (1810 detail records)

Sum	1810
Standard	1.33%

Area-

Summary for 'area' = Area->10_22.949798_39.380934_22.970551_39.369007 (12936 detail records)

Sum	12936
Standard	9.52%

Area-

area counter

Summary for 'area' = Area->11_22.949798_39.369007_22.970551_39.35708 (31316 detail records)

Sum	31316
Standard	23.06%

Area-

Summary for 'area' = Area->12_22.949798_39.35708_22.970551_39.345153 (8661 detail records)

Sum	8661
Standard	6.38%

Area-

Summary for 'area' = Area->13_22.970551_39.392861_22.991303_39.380934 (600 detail records)

Sum	600
Standard	0.44%

Area-	
Summary for 'area' = Area->14_22.970551_39.380934_22.991303_39.369007 (868 detail records)	
Sum	868
Standard	0.64%
Area-	
Summary for 'area' = Area->15_22.970551_39.369007_22.991303_39.35708 (1320 detail records)	
Sum	1320
Standard	0.97%
Area-	
Summary for 'area' = Area->16_22.970551_39.35708_22.991303_39.345153 (591 detail records)	
Sum	591
Standard	0.44%
N/A	
Summary for 'area' = N/A (3517 detail records)	
Sum	3517
Standard	2.59%
Grand Total	135813

Table 6. The proportion of the total amount of requests that occurred on Wednesdays in each different subregion

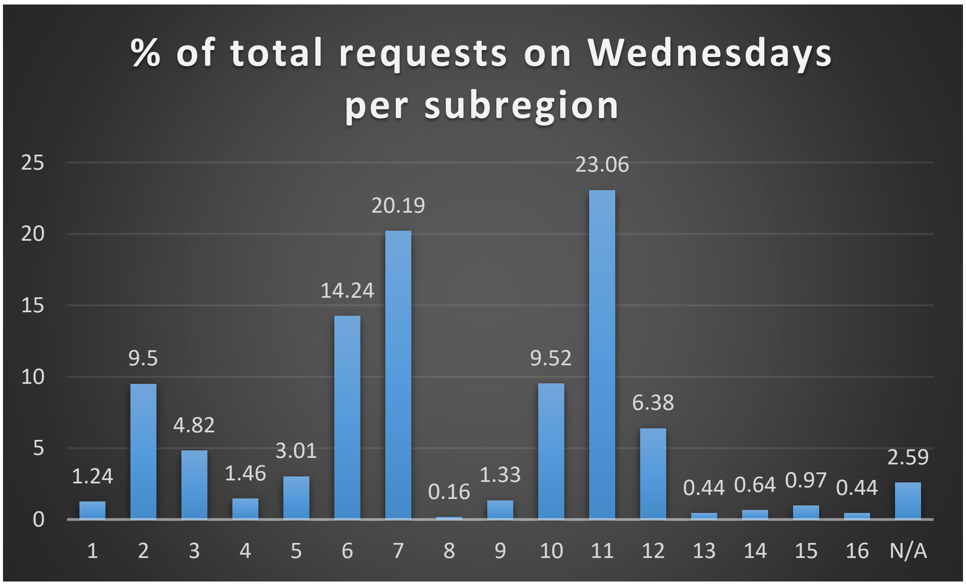

Figure 8. The percentage of the total amount of requests on Wednesdays per subregion

e) <u>Thursdays</u> (see Table 7, Figure 9)
Total amount of requests: 145761
Subregions with the largest proportions: 7 (20.86%), 11 (23.37%)
Following up: 2 (9.55%), 6 (14.16%), 10 (9.29%)

area	counter
Area-	
Summary for 'area' = Area->01_22.908294_39.392861_22.929046_39.380934 (1768 detail records)	
Sum	1768
Standard	1.21%
Area-	
Summary for 'area' = Area->02_22.908294_39.380934_22.929046_39.369007 (13923 detail records)	
Sum	13923
Standard	9.55%
Area-	
Summary for 'area' = Area->03_22.908294_39.369007_22.929046_39.35708 (6948 detail records)	
Sum	6948
Standard	4.77%
Area-	
Summary for 'area' = Area->04_22.908294_39.35708_22.929046_39.345153 (2227 detail records)	
Sum	2227
Standard	1.53%
Area-	
Summary for 'area' = Area->05_22.929046_39.392861_22.949798_39.380934 (4141 detail records)	
Sum	4141
Standard	2.84%
Area-	
Summary for 'area' = Area->06_22.929046_39.380934_22.949798_39.369007 (20644 detail records)	
Sum	20644
Standard	14.16%
Area-	
Summary for 'area' = Area->07_22.929046_39.369007_22.949798_39.35708 (30400 detail records)	
Sum	30400
Standard	20.86%
Area-	
Summary for 'area' = Area->08_22.929046_39.35708_22.949798_39.345153 (161 detail records)	
Sum	161
Standard	0.11%
Area-	
Summary for 'area' = Area->09_22.949798_39.392861_22.970551_39.380934 (1822 detail records)	
Sum	1822
Standard	1.25%
Area-	
Summary for 'area' = Area->10_22.949798_39.380934_22.970551_39.369007 (13548 detail records)	
Sum	13548
Standard	9.29%
Area-	
Summary for 'area' = Area->11_22.949798_39.369007_22.970551_39.35708 (34060 detail records)	
Sum	34060
Standard	23.37%
Area-	
Summary for 'area' = Area->12_22.949798_39.35708_22.970551_39.345153 (9172 detail records)	
Sum	9172
Standard	6.29%
Area-	
Summary for 'area' = Area->13_22.970551_39.392861_22.991303_39.380934 (595 detail records)	
Sum	595
Standard	0.41%

Area-	
Summary for 'area' = Area->14_22.970551_39.380934_22.991303_39.369007 (950 detail records)	
Sum	950
Standard	0.65%
Area-	
Summary for 'area' = Area->15_22.970551_39.369007_22.991303_39.35708 (1330 detail records)	
Sum	1330
Standard	0.91%
Area-	
Summary for 'area' = Area->16_22.970551_39.35708_22.991303_39.345153 (638 detail records)	
Sum	638
Standard	0.44%
N/A	
Summary for 'area' = N/A (3434 detail records)	
Sum	3434
Standard	2.36%
Grand Total	145761

Table 7. The proportion of the total amount of requests that occurred on Thursdays in each different subregion

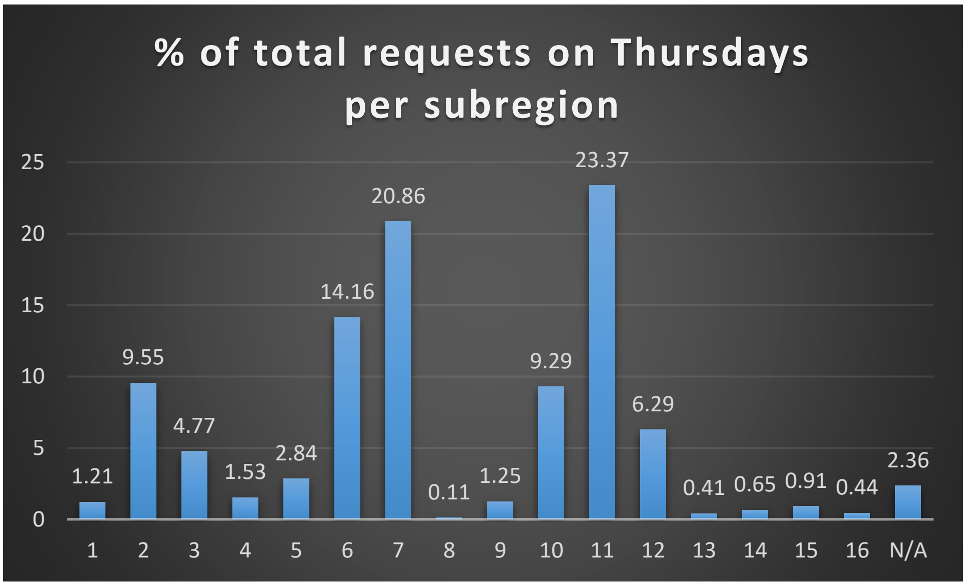

Figure 9. The percentage of the total amount of requests on Thursdays per subregion

f) Fridays (see Table 8, Figure 10)
Total amount of requests: 157278
Subregions with the largest proportions: 7 (20.34%), 11 (23.55%)
Following up: 2 (9.77%), 6 (14.42%), 10 (9.27%)

area	counter
Area-	
Summary for 'area' = Area->01_22.908294_39.392861_22.929046_39.380934 (1957 detail records)	
Sum	1957
Standard	1.24%
Area-	
Summary for 'area' = Area->02_22.908294_39.380934_22.929046_39.369007 (15367 detail records)	
Sum	15367
Standard	9.77%
Area-	
Summary for 'area' = Area->03_22.908294_39.369007_22.929046_39.35708 (7092 detail records)	
Sum	7092
Standard	4.51%
Area-	
Summary for 'area' = Area->04_22.908294_39.35708_22.929046_39.345153 (2009 detail records)	
Sum	2009
Standard	1.28%
Area-	
Summary for 'area' = Area->05_22.929046_39.392861_22.949798_39.380934 (4511 detail records)	
Sum	4511
Standard	2.87%
Area-	
Summary for 'area' = Area->06_22.929046_39.380934_22.949798_39.369007 (22678 detail records)	
Sum	22678
Standard	14.42%
Area-	
Summary for 'area' = Area->07_22.929046_39.369007_22.949798_39.35708 (31992 detail records)	
Sum	31992
Standard	20.34%
Area-	
Summary for 'area' = Area->08_22.929046_39.35708_22.949798_39.345153 (226 detail records)	
Sum	226
Standard	0.14%
Area-	
Summary for 'area' = Area->09_22.949798_39.392861_22.970551_39.380934 (2068 detail records)	
Sum	2068
Standard	1.31%
Area-	
Summary for 'area' = Area->10_22.949798_39.380934_22.970551_39.369007 (14574 detail records)	
Sum	14574
Standard	9.27%
Area-	
Summary for 'area' = Area->11_22.949798_39.369007_22.970551_39.35708 (37032 detail records)	
Sum	37032
Standard	23.55%
Area-	
Summary for 'area' = Area->12_22.949798_39.35708_22.970551_39.345153 (10180 detail records)	
Sum	10180
Standard	6.47%
Area-	
Summary for 'area' = Area->13_22.970551_39.392861_22.991303_39.380934 (664 detail records)	
Sum	664
Standard	0.42%

Area-		
Summary for 'area' = Area->14_22.970551_39.380934_22.991303_39.369007 (1039 detail records)		
	Sum	1039
	Standard	0.66%
Area-		
Summary for 'area' = Area->15_22.970551_39.369007_22.991303_39.35708 (1443 detail records)		
	Sum	1443
	Standard	0.92%
Area-		
Summary for 'area' = Area->16_22.970551_39.35708_22.991303_39.345153 (668 detail records)		
	Sum	668
	Standard	0.42%
N/A		
Summary for 'area' = N/A (3778 detail records)		
	Sum	3778
	Standard	2.40%
Grand Total		157278

Table 8. The proportion of the total amount of requests that occurred on Fridays in each different subregion

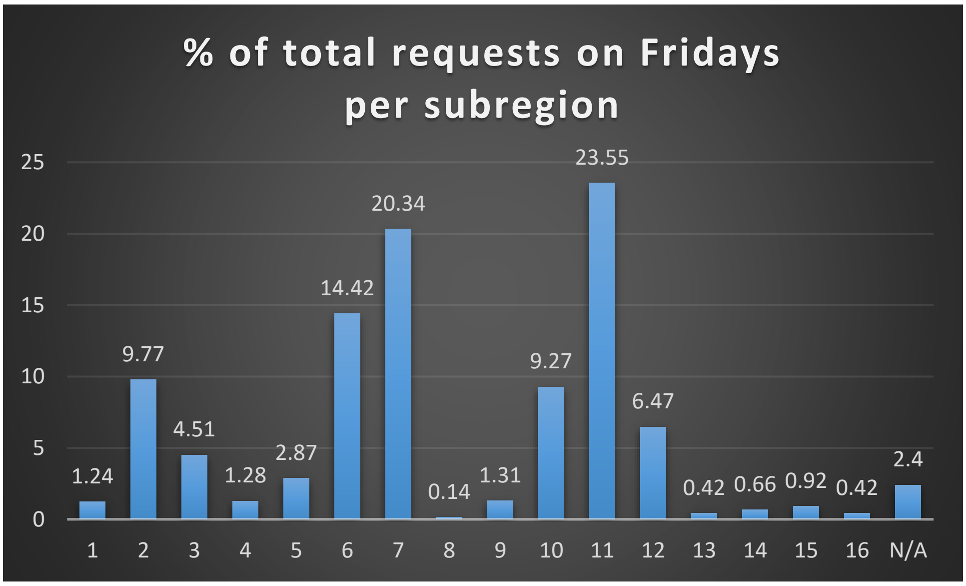

Figure 10. The percentage of the total amount of requests on Fridays per subregion

g) Saturdays (see Table 9, Figure 11)
Total amount of requests: 130152
Subregions with the largest proportions: 7 (21.19%), 11 (21.62%)
Following up: 2 (9.94%), 6 (14.43%), 10 (9.51%)

area	counter
Area-	
Summary for 'area' = Area->01_22.908294_39.392861_22.929046_39.380934 (1310 detail records)	
Sum	1310
Standard	1.01%
Area-	
Summary for 'area' = Area->02_22.908294_39.380934_22.929046_39.369007 (12938 detail records)	
Sum	12938
Standard	9.94%
Area-	
Summary for 'area' = Area->03_22.908294_39.369007_22.929046_39.35708 (5546 detail records)	
Sum	5546
Standard	4.26%
Area-	
Summary for 'area' = Area->04_22.908294_39.35708_22.929046_39.345153 (1345 detail records)	
Sum	1345
Standard	1.03%
Area-	
Summary for 'area' = Area->05_22.929046_39.392861_22.949798_39.380934 (3712 detail records)	
Sum	3712
Standard	2.85%
Area-	
Summary for 'area' = Area->06_22.929046_39.380934_22.949798_39.369007 (18775 detail records)	
Sum	18775
Standard	14.43%
Area-	
Summary for 'area' = Area->07_22.929046_39.369007_22.949798_39.35708 (27582 detail records)	
Sum	27582
Standard	21.19%
Area-	
Summary for 'area' = Area->08_22.929046_39.35708_22.949798_39.345153 (269 detail records)	
Sum	269
Standard	0.21%
Area-	
Summary for 'area' = Area->09_22.949798_39.392861_22.970551_39.380934 (1896 detail records)	
Sum	1896
Standard	1.46%
Area-	
Summary for 'area' = Area->10_22.949798_39.380934_22.970551_39.369007 (12374 detail records)	
Sum	12374
Standard	9.51%
Area-	
Summary for 'area' = Area->11_22.949798_39.369007_22.970551_39.35708 (28142 detail records)	
Sum	28142
Standard	21.62%
Area-	
Summary for 'area' = Area->12_22.949798_39.35708_22.970551_39.345153 (9207 detail records)	
Sum	9207
Standard	7.07%
Area-	
Summary for 'area' = Area->13_22.970551_39.392861_22.991303_39.380934 (613 detail records)	
Sum	613
Standard	0.47%

Area-

Summary for 'area' = Area->14_22.970551_39.380934_22.991303_39.369007 (936 detail records)

Sum	936
Standard	0.72%

Area-

Summary for 'area' = Area->15_22.970551_39.369007_22.991303_39.35708 (1133 detail records)

Sum	1133
Standard	0.87%

Area-

Summary for 'area' = Area->16_22.970551_39.35708_22.991303_39.345153 (564 detail records)

Sum	564
Standard	0.43%

N/A

Summary for 'area' = N/A (3810 detail records)

Sum	3810
Standard	2.93%
Grand Total	130152

Table 9. The proportion of the total amount of requests that occurred on Saturdays in each different subregion

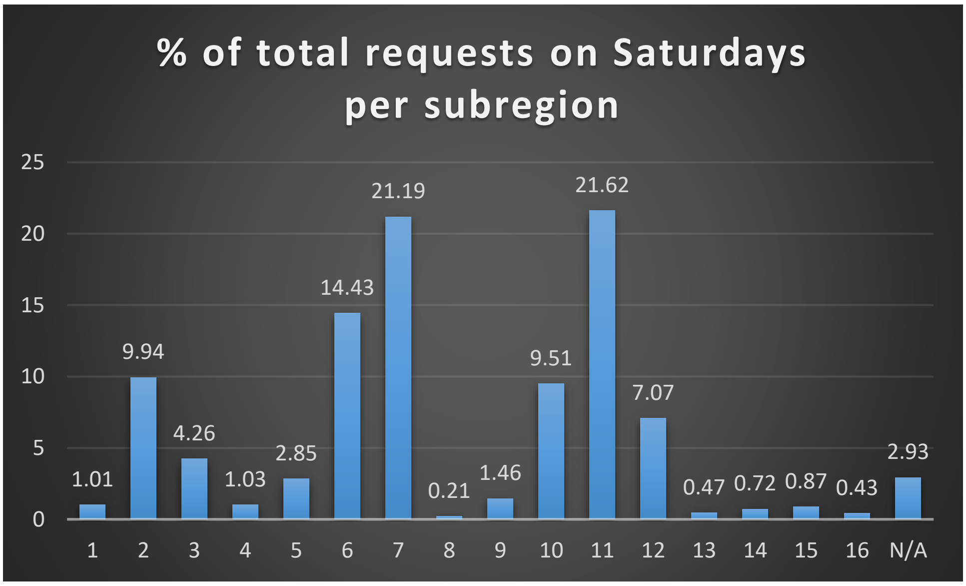

Figure 11. The percentage of the total amount of requests on Saturdays per subregion

Taking an overall look at the previous results, it follows that subregions 7 and 11 have the largest proportions on any weekday. Moreover, subregions 2, 6 and 10 maintain the next highest proportions on any weekday.

In addition, in regard to each subregion's percentages, we conclude that there is no large deviation among them in each different weekday. Taking this fact into account, we could say that each subregion's need for transportation does not vary too much from day to day. For example, concerning subregions 6 and 7, in Figure 12 and 13 respectively, one could see how their percentages vary from day to day.

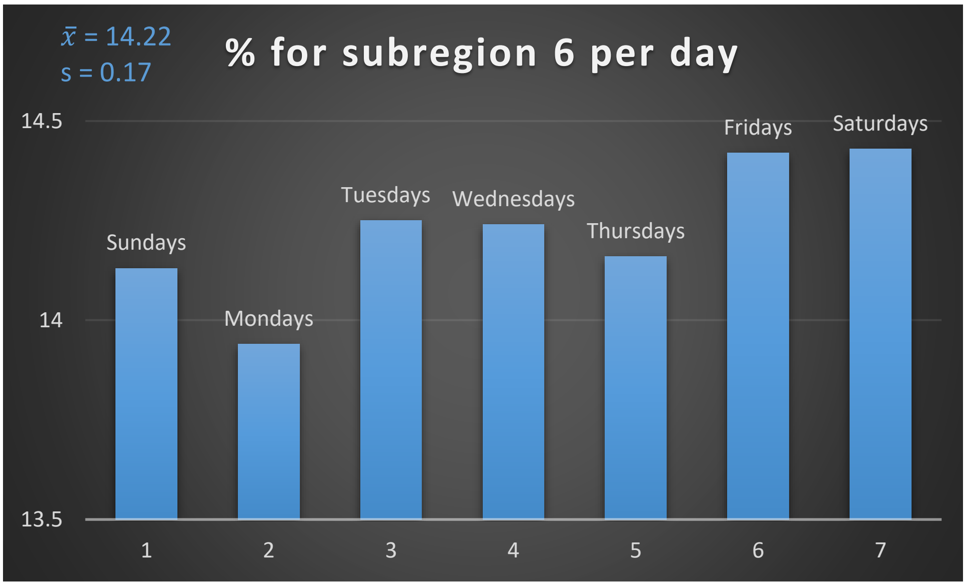

Figure 12. Percentages of the total amount of requests per day for subregion 6

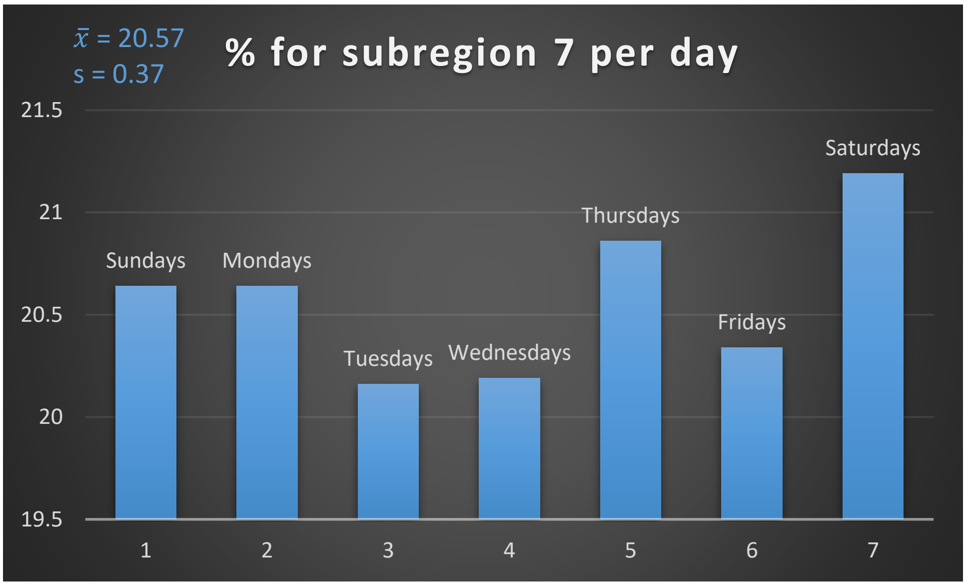

Figure 13. Percentages of the total amount of requests per day for subregion 7

3. Weekday based analysis concerning the total amount of requests

Further to the previous analysis, we can also compare the total amount of requests that occurred during different days in order to estimate which day of the week is busiest.

In Figure 14 we see a bar chart concerning the percentage of the total amount of requests per day.

It is obviously seen that there is no large difference among the percentages. This fact means that the total requests that occur per day do not depend much on the weekday. More precisely, Friday seems to be the busiest day of the week which is indicated by the highest percentage of total requests (16.4% - 157278 out of 959780 requests). Moreover, the smallest percentage of total requests is found on Sunday (12% - 115234 out of 959780 requests). Finally, regarding the other days of the week, they seem to be quite similar.

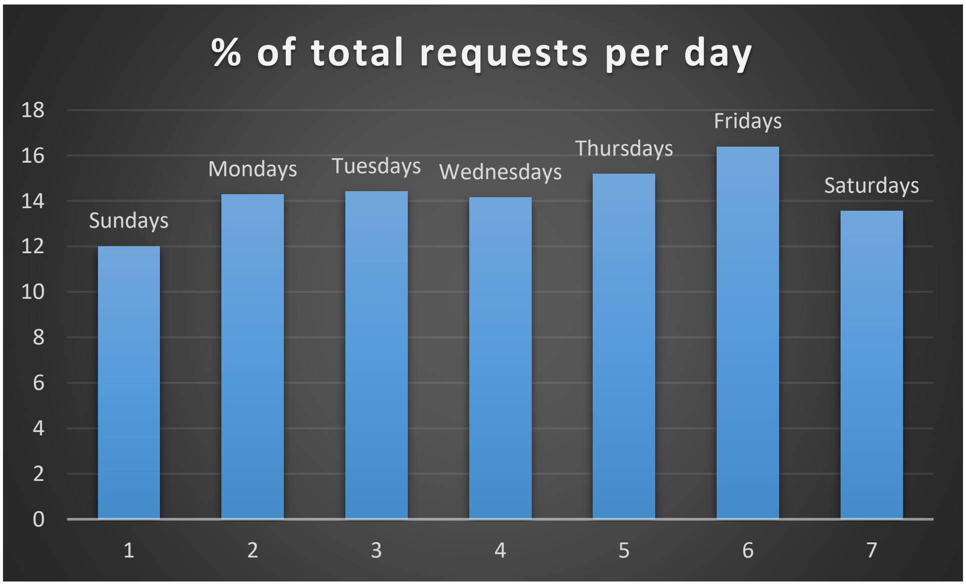

Figure 14. The percentage of the total amount of requests per day

4. Hour based analysis concerning the total amount of requests

Apart from the spatial information about each different subregion and the weekday statistics, we also analyzed the total amount of requests regarding the time that each request occurred.

Taking a look at the percentages of Table 10, it can be obviously seen that the vast majority of the total requests occurred between 8.00 and 12.00 (7.10%, 7.98%, 8.00%, 7.62% respectively). Additionally, considerable proportions have been attained between 7.00 and 8.00 (5.15%), 12.00 and 13.00 (5.98%), 13.00 and 14.00 (5.69%), 17.00 and 18.00 (5.20%), 18.00 and 19.00 (5.23%).

For the sake of convenience, in Figure 15 there is a bar chart about the percentage of the total amount of requests per hour.

byhour

hournum	counter
Summary for 'hournum' = 1 (19487 detail records)	
Sum	19487
Standard	2.03%
Summary for 'hournum' = 2 (17310 detail records)	
Sum	17310
Standard	1.80%
Summary for 'hournum' = 3 (11674 detail records)	
Sum	11674
Standard	1.22%
Summary for 'hournum' = 4 (11402 detail records)	
Sum	11402
Standard	1.19%
Summary for 'hournum' = 5 (13204 detail records)	
Sum	13204
Standard	1.38%
Summary for 'hournum' = 6 (20896 detail records)	
Sum	20896
Standard	2.18%
Summary for 'hournum' = 7 (28334 detail records)	
Sum	28334
Standard	2.95%
Summary for 'hournum' = 8 (49444 detail records)	
Sum	49444
Standard	5.15%
Summary for 'hournum' = 9 (68187 detail records)	
Sum	68187
Standard	7.10%
Summary for 'hournum' = 10 (76626 detail records)	
Sum	76626
Standard	7.98%
Summary for 'hournum' = 11 (76736 detail records)	
Sum	76736
Standard	8.00%
Summary for 'hournum' = 12 (73089 detail records)	
Sum	73089
Standard	7.62%
Summary for 'hournum' = 13 (57432 detail records)	
Sum	57432
Standard	5.98%
Summary for 'hournum' = 14 (54626 detail records)	
Sum	54626
Standard	5.69%
Summary for 'hournum' = 15 (46899 detail records)	
Sum	46899
Standard	4.89%
Summary for 'hournum' = 16 (36411 detail records)	
Sum	36411
Standard	3.79%
Summary for 'hournum' = 17 (37910 detail records)	
Sum	37910
Standard	3.95%

Summary for 'hournum' = 18 (49900 detail records)	
Sum	49900
Standard	5.20%
Summary for 'hournum' = 19 (50161 detail records)	
Sum	50161
Standard	5.23%
Summary for 'hournum' = 20 (43410 detail records)	
Sum	43410
Standard	4.52%
Summary for 'hournum' = 21 (36797 detail records)	
Sum	36797
Standard	3.83%
Summary for 'hournum' = 22 (32076 detail records)	
Sum	32076
Standard	3.34%
Summary for 'hournum' = 23 (25609 detail records)	
Sum	25609
Standard	2.67%
Summary for 'hournum' = 24 (22160 detail records)	
Sum	22160
Standard	2.31%
Grand Total	959780

Table 10. The proportion of the total amount of requests for each hour of the day

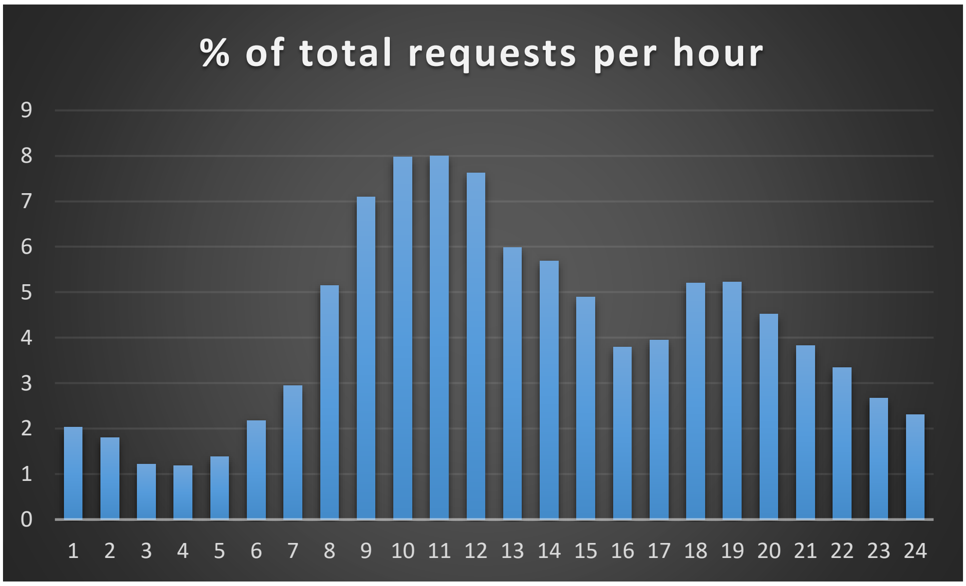

Figure 15. The percentage of the total amount of requests per hour

Chapter 5. COMPUTATIONAL TESTS

In this section we present the construction of our experiments and how we have taken into account probabilistic information. Their computational results are also contained in this section.

5.1 How to insert probabilistic information

In the previous sections we have described in detail the structure of our algorithms. Also, we have provided an extensive statistical analysis regarding our data set. Now, based on the statistical results and the taxi expert's opinion, we effort to insert probabilistic information to our system. More precisely, we have attempted to place probability in three levels. First of all, in accordance with the proportions given on the statistical reports we have constructed the cumulative probabilities concerning the subregion's centroid (pick up point) and the hour in which a request occurs. Moreover, we asked the taxi expert to estimate the delivery points in the city of Volos and each proportion of the total requests for transportation that goes to them. Then, we calculated the cumulative probability for the delivery points. In the following Tables (11, 12, 13), one can see the proportions and the cumulative probabilities regarding each subregion's centroid (pick up point), hour and delivery point.

FROM_CENTROID	%_stat_report	Pr(cumulative)	[	]
1	0.0117	0.0117	0.0000	0.0116
2	0.0972	0.1089	0.0117	0.1088
3	0.0470	0.1559	0.1089	0.1558
4	0.0135	0.1694	0.1559	0.1693
5	0.0291	0.1985	0.1694	0.1984
6	0.1423	0.3408	0.1985	0.3407
7	0.2057	0.5465	0.3408	0.5464
8	0.0018	0.5483	0.5465	0.5482
9	0.0134	0.5617	0.5483	0.5616
10	0.0941	0.6558	0.5617	0.6557
11	0.2270	0.8828	0.6558	0.8827
12	0.0655	0.9483	0.8828	0.9482
13	0.0045	0.9528	0.9483	0.9527
14	0.0066	0.9594	0.9528	0.9593
15	0.0091	0.9685	0.9594	0.9684
16	0.0043	0.9728	0.9685	0.9727
N/A			0.9728	1.0000

Table 11. The proportion and the cumulative probability regarding each subregion's centroid – pick up point.

HOUR	%_stat_report	Pr(cumulative)	[	]
1	0.0203	0.0203	0.0000	0.0202
2	0.0180	0.0383	0.0203	0.0382
3	0.0122	0.0505	0.0383	0.0504
4	0.0119	0.0624	0.0505	0.0623
5	0.0138	0.0762	0.0624	0.0761
6	0.0218	0.0980	0.0762	0.0979
7	0.0295	0.1275	0.0980	0.1274
8	0.0515	0.1790	0.1275	0.1789
9	0.0710	0.2500	0.1790	0.2499
10	0.0798	0.3298	0.2500	0.3297
11	0.0800	0.4098	0.3298	0.4097
12	0.0762	0.4860	0.4098	0.4859
13	0.0598	0.5458	0.4860	0.5457
14	0.0569	0.6027	0.5458	0.6026
15	0.0489	0.6516	0.6027	0.6515
16	0.0379	0.6895	0.6516	0.6894
17	0.0395	0.7290	0.6895	0.7289
18	0.0520	0.7810	0.7290	0.7809
19	0.0523	0.8333	0.7810	0.8332
20	0.0452	0.8785	0.8333	0.8784
21	0.0383	0.9168	0.8785	0.9167
22	0.0334	0.9502	0.9168	0.9501
23	0.0267	0.9769	0.9502	0.9768
24	0.0231	1.0000	0.9769	0.9999
			1.0000	

Table 12. The proportion and the cumulative probability regarding each hour.

TO_DELIVERY_POINT	%_taxi_expert	Pr(cumulative)	[	]
1	0.0300	0.0300	0.0000	0.0299
2	0.0700	0.1000	0.0300	0.0999
3	0.1500	0.2500	0.1000	0.2499
4	0.0500	0.3000	0.2500	0.2999
5	0.2500	0.5500	0.3000	0.5499
6	0.0300	0.5800	0.5500	0.5799
7	0.2000	0.7800	0.5800	0.7799
8	0.0300	0.8100	0.7800	0.8099
9	0.0200	0.8300	0.8100	0.8299
10	0.1000	0.9300	0.8300	0.9299
11	0.0100	0.9400	0.9300	0.9399
12	0.0100	0.9500	0.9400	0.9499
13	0.0500	1.0000	0.9500	0.9999
			1.0000	

Table 13. The proportion and the cumulative probability regarding each delivery point.

5.2 Experiments construction

The cumulative probabilities mentioned before enabled us to develop a code that constructs our experiments. More precisely, a random generator picks a choice from our lists of centroids-pickup points, hour and delivery points respectively, in accordance with the cumulative probabilities. In this way, we create demands for transportation by one person at a time, based on probabilistic information about pickup and delivery points and the hour that they occur. In this context, it has to be mentioned that our demands may be static, which means that they are known beforehand, but they are completely based on probabilistic information, which is available through the previous extensive statistical analysis for our data set.

Our code was implemented in C++. At first, we constructed 3 sets of experiments. Each set consists of 500 experiments, as it is a good number in order to achieve reliable enough results. Each experiment consists of 500 demands for transportation. We ended up to this number because the average daily demand for the VOLOSTAXI company is about 1000 demands per day. We excluded the rest 500 demands from our on-demand cab-sharing transportation system because of the company's need to operate as a pure taxi company. Additionally, for the same reason we exclude 110 out of 200 company's taxis in order to serve the rest requests and the rejected requests that may occur during the run of our algorithm. Furthermore, the fleet of 90 taxis is equally distributed among the taxi stations – depots that already exist in various places around the city of Volos. In order to be more specific, there are 15 different taxi stations in the city and as a result, each of them has a capacity of 6 taxis. In the 1st set of experiments, the 6 taxis of each taxi station – depot are available during the whole day. On the other hand, regarding the 2nd and the 3rd set of experiments, we divided the 24hour period into three 8hour shifts for the vehicles-drivers: 00:00-08:00, 08:00-16:00 and 16:00-24:00. About the 2nd set of experiments, the 6 taxis are equally available in each 8hour period (2 taxis per 8hour period). In contrast, concerning the 3rd set of experiments, the 6 taxis follow the demand distribution in each 8hour period (1 taxi the 1st, 3 taxis the 2nd, 2 taxis the 3rd – see the demand distribution in Figure 15 on statistical analysis section).

To sum up, each set consists of 500 experiments and each experiment contains 500 demands for transportation. Each demand is described by its pick up point (1 out of 16 subregions' centroids), its timestamp (in day minutes) and its delivery point (1 out of 13 delivery points). We have excluded from our experiments the small proportion (2.72%) of the demands that did not occur in any of the 16 subregions. Also, based on the taxi expert's experience and the size of the city of Volos, we believe that the 13 delivery points are satisfactory. Additionally, for these 44 points (16 pick-up points, 13 delivery points, 15 taxi stations - depots) that constitute our network, we calculated our cost matrix (shortest path distance and shortest path time units). Moreover, the maximum number of passengers was 4 persons and the time window 15 minutes. The maximum ride time was 1.6 times the absolute shortest path time for that specific distance or the absolute shortest path time + 10 minutes for shortest path time less than 15 minutes. For the sake of reader's convenience, in the following Figures (16, 17, 18) we have displayed on Google maps the pickup and delivery points of our network and the existing taxi stations – depots, too.

Moreover, we also constructed an experiment which consists of 5000 requests for transportation (10 times bigger than the previous requests), using the same fleet of 90 taxis that is equally distributed among the taxi stations – depots (6 per taxi depot) and in accordance with the demand distribution in each 8hour period (1 taxi the 1st, 3 taxis the 2nd, 2 taxis the 3rd). The maximum number of passengers, the maximum ride time and the time window remained the same.

Last but not least, we tried to further investigate concepts with different parameters. For this purpose, we ran 3 more sets of experiments by changing a parameter at a time or a combination of them. Thus, each of the 5th and 6th set of tests consists of 10 experiments. Each experiment contains 500 demands for transportation using the same fleet of 90 taxis that is equally distributed among the taxi stations – depots (6 per taxi depot) and in accordance with the demand distribution in each 8hour period (1 taxi the 1st, 3 taxis the 2nd, 2 taxis the 3rd). The maximum number of passengers remained the same (4 persons). Concerning the 5th set of experiments we changed the maximum ride time from 1.6 times the absolute shortest path time to 2 times the absolute shortest path time for that specific distance or the absolute shortest path time + 10 minutes for shortest path time less than 15 minutes. The time window remained the same (15 minutes). In contrast, regarding the 6th set, the maximum ride time remained 1.6 times the absolute shortest path time, but we changed the time window from 15 minutes to 30 minutes. Moreover, the 7th set consists of 4 experiments which use the same fleet of 90 taxis that is equally distributed among the taxi stations – depots (6 per taxi depot) and in accordance with the demand distribution in each 8hour period (1 taxi the 1st, 3 taxis the 2nd, 2 taxis the 3rd). The difference among the 7th set and the other sets is the fact that we changed at the same time the time window (30 minutes), the max ride time (2 times the absolute shortest path time) and the demands for transportation (1500 demands - 3 times bigger than the previous requests). At last, we also constructed 4 more experiments in order to have a clear understanding of how time windows and maximum ride time (in terms of absolute shortest path) affect our results. Thus, concerning the 8th and 9th experiment we changed time windows to 45 and 60 minutes, respectively. Finally, regarding the 10th and 11th experiment maximum ride time was set as 2.5 times and 3 times the absolute shortest path time for that specific distance or the absolute shortest path time + 10 minutes for shortest path time less than 15 minutes, respectively. All the other parameters remained the same.

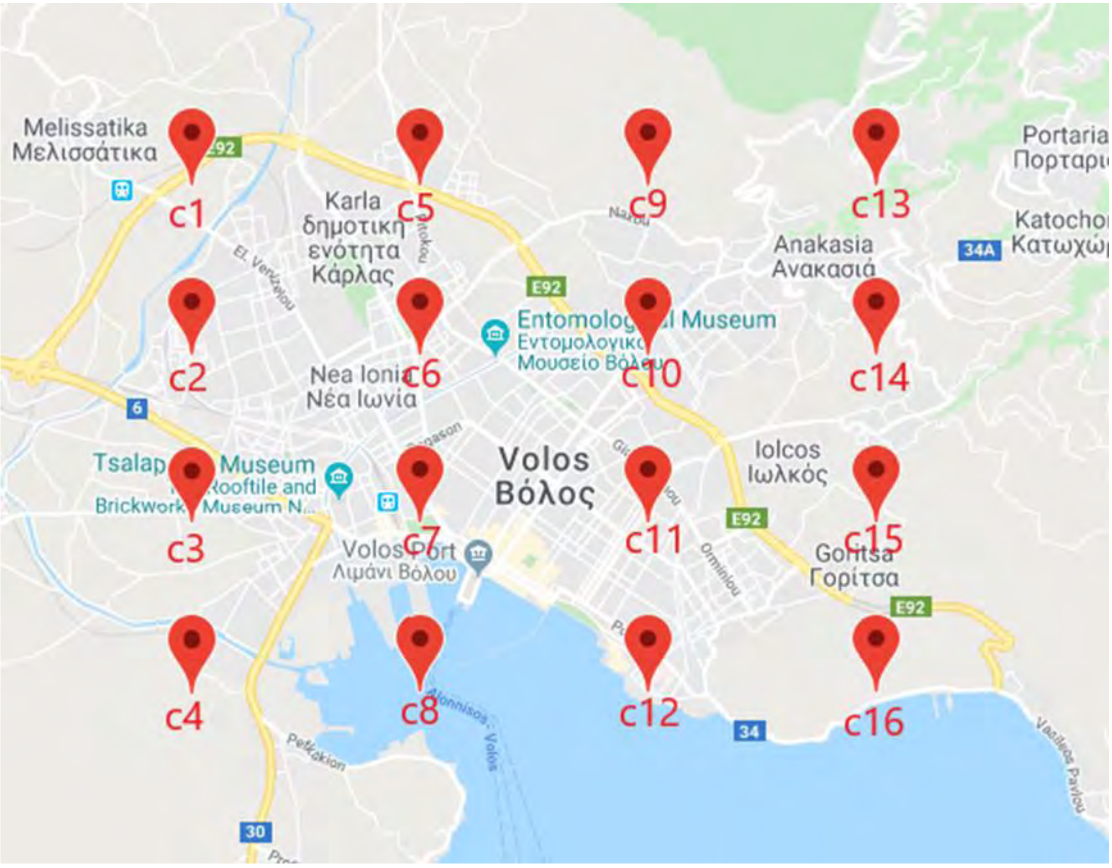

Figure 16. The pickup points of our network (subregions' centroids)

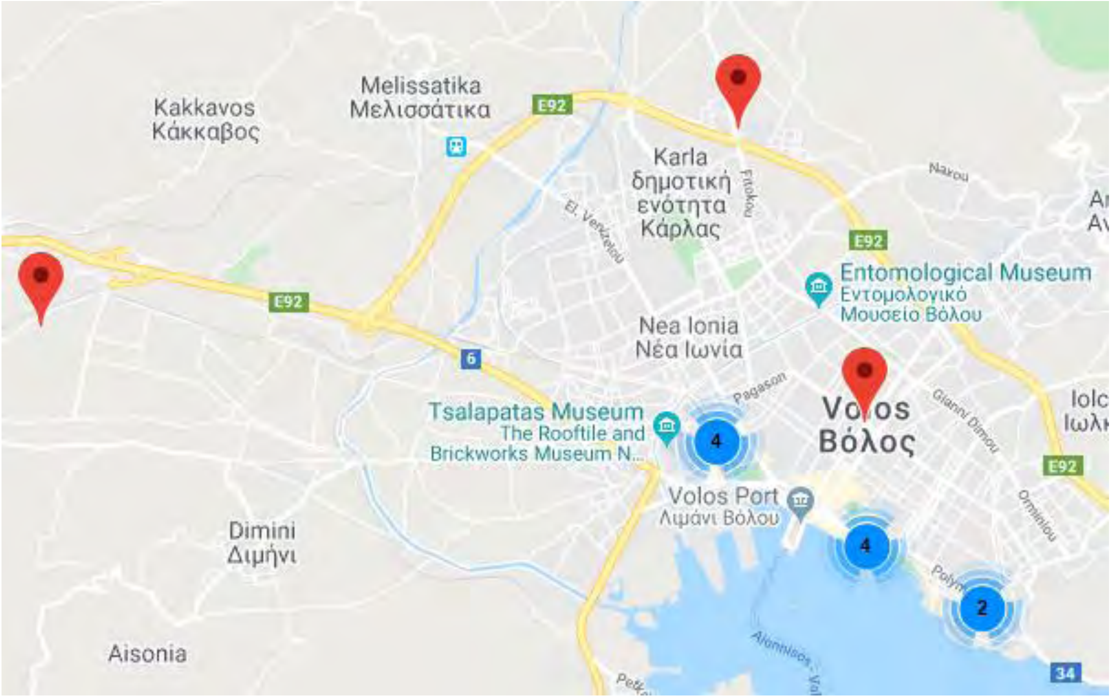

Figure 17. The delivery points of our network

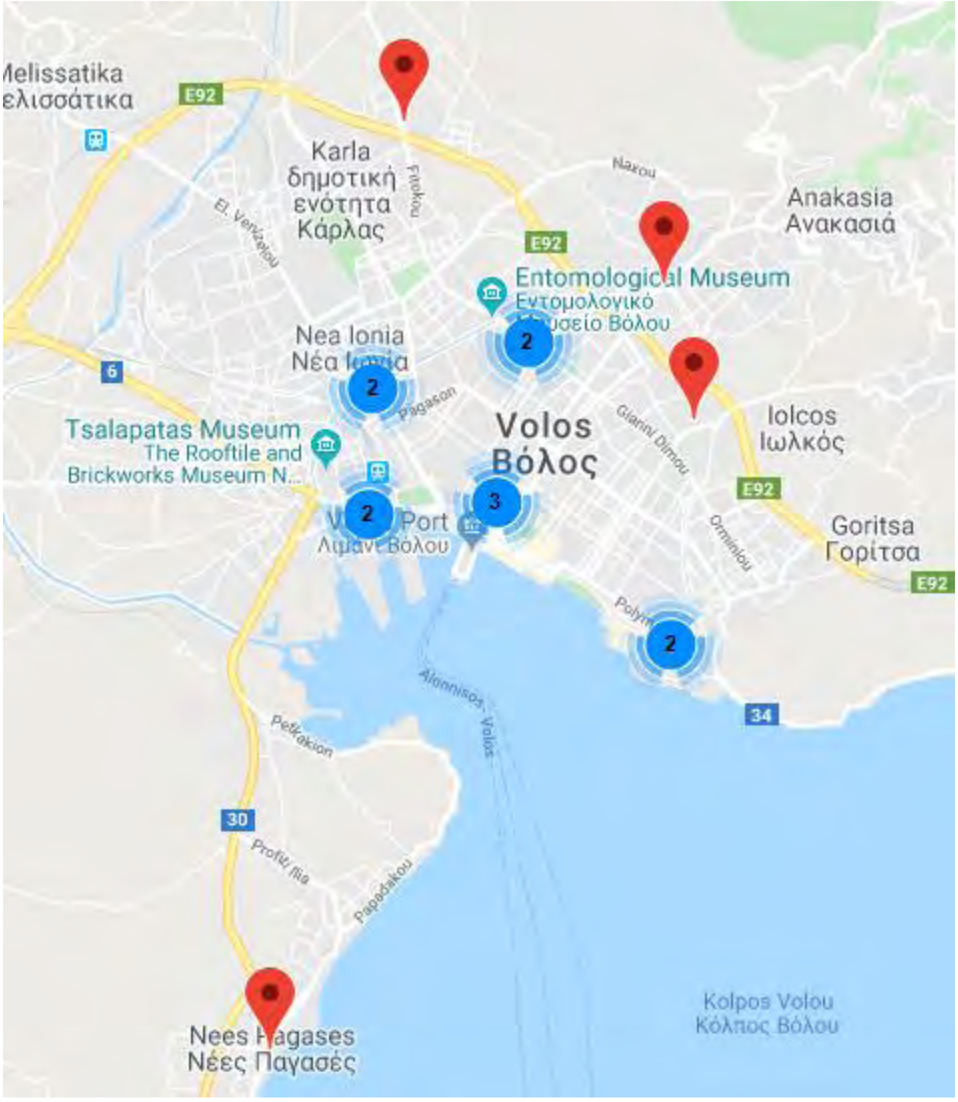

Figure 18. The existing taxi stations – depots

5.3 Experiments results

Following up the implementation of our sets of experiments, in this section we provide significant measurements about major components of our problem regarding each different set of tests.

- 1st set of experiments (500 experiments)

Each experiment's components: 500 demands for transportation, maximum number of passengers: 4 persons, time window: 15 minutes, maximum ride time: 1.6 times the absolute shortest path time for that specific distance or the absolute shortest path time + 10 minutes for shortest path time less than 15 minutes, fleet: 90 taxis equally distributed among 15 taxi depots - 6 taxis per taxi depot - *available during the whole day 00:00-24:00*

Amount of transportation requests: 500

Not allocated requests: avg: 0.886, min: 0, max: 5

% of not allocated requests: avg: 0.18%

Needed taxis: avg: 28.798, min: 22, max: 37

Average passengers / distance(km): avg: 0.788, min: 0.742, max: 0.835

Total distance units (m): avg: 2090685.36, min: 1900040, max: 2312280

- 2nd set of experiments (500 experiments)

Each experiment's components: 500 demands for transportation, maximum number of passengers: 4 persons, time window: 15 minutes, maximum ride time: 1.6 times the absolute shortest path time for that specific distance or the absolute shortest path time + 10 minutes for shortest path time less than 15 minutes, fleet: 90 taxis equally distributed among 15 taxi depots - 6 taxis per taxi depot - *equally available in each 8hour period (2 taxis per 8hour period)*

Amount of transportation requests: 500

Not allocated requests: avg: 3.78, min: 0, max: 10

% of not allocated requests: avg: 0.76%

Needed taxis: avg: 47.354, min: 41, max: 55

Average passengers / distance(km): avg: 0.793, min: 0.750, max: 0.845

Total distance units (m): avg: 2121223.48, min: 1950520, max: 2280310

- 3rd set of experiments (500 experiments)

Each experiment’s components: 500 demands for transportation, maximum number of passengers: 4 persons, time window: 15 minutes, maximum ride time: 1.6 times the absolute shortest path time for that specific distance or the absolute shortest path time + 10 minutes for shortest path time less than 15 minutes, fleet: 90 taxis equally distributed among 15 taxi depots - 6 taxis per taxi depot – *following the demand distribution in each 8hour period (00:00-08:00_1 taxi, 08:00-16:00_3 taxis, 16:00-24:00_2 taxis)*

Amount of transportation requests: 500

Not allocated requests: avg: 3.806, min: 0, max: 11

% of not allocated requests: avg: 0.76%

Needed taxis: avg: 47.87, min: 41, max: 55

Average passengers / distance(km): avg: 0.794, min: 0.756, max: 0.836

Total distance units (m): avg: 2114251.88, min: 1935360, max: 2319140

- 4th experiment

Experiment’s components: *5000 demands* for transportation, maximum number of passengers: 4 persons, time window: 15 minutes, maximum ride time: 1.6 times the absolute shortest path time for that specific distance or the absolute shortest path time + 10 minutes for shortest path time less than 15 minutes, fleet: 90 taxis equally distributed among 15 taxi depots - 6 taxis per taxi depot – following the demand distribution in each 8hour period (00:00-08:00_1 taxi, 08:00-16:00_3 taxis, 16:00-24:00_2 taxis)

Amount of transportation requests: 5000

Not allocated requests: 1580

% of not allocated requests: 31.6%

Needed taxis: 90

Average passengers / distance(km): 1.182

Total distance units (m): 8880430

- <u>5th set of experiments (10 experiments)</u>

<u>Each experiment's components</u>: 500 demands for transportation, maximum number of passengers: 4 persons, time window: 15 minutes, *maximum ride time: 2 times the absolute shortest path time* for that specific distance or the absolute shortest path time + 10 minutes for shortest path time less than 15 minutes, fleet: 90 taxis equally distributed among 15 taxi depots - 6 taxis per taxi depot – following the demand distribution in each 8hour period (00:00-08:00_1 taxi, 08:00-16:00_3 taxis, 16:00-24:00_2 taxis)

Amount of transportation requests: 500

Not allocated requests: avg: 3.1, min: 1, max: 7

% of not allocated requests: avg: 0.62%

Needed taxis: avg: 48.4, min: 44, max: 53

Average passengers / distance(km): avg: 0.82, min: 0.804, max: 0.839

Total distance units (m): avg: 2066591, min: 1981000, max: 2154060

- <u>6th set of experiments (10 experiments)</u>

<u>Each experiment's components</u>: 500 demands for transportation, maximum number of passengers: 4 persons, *time window: 30 minutes*, maximum ride time: 1.6 times the absolute shortest path time for that specific distance or the absolute shortest path time + 10 minutes for shortest path time less than 15 minutes, fleet: 90 taxis equally distributed among 15 taxi depots - 6 taxis per taxi depot – following the demand distribution in each 8hour period (00:00-08:00_1 taxi, 08:00-16:00_3 taxis, 16:00-24:00_2 taxis)

Amount of transportation requests: 500

Not allocated requests: avg: 0.3, min: 0, max: 2

% of not allocated requests: avg: 0.06%

Needed taxis: avg: 42.9, min: 39, max: 46

Average passengers / distance(km): avg: 0.849, min: 0.831, max: 0.862

Total distance units (m): avg: 1999087, min: 1933920, max: 2102370

➢ <u>7th set of experiments (4 experiments)</u>

<u>Each experiment's components</u>: *1500 demands for transportation*, maximum number of passengers: 4 persons, *time window: 30 minutes, maximum ride time: 2 times the absolute shortest path time* for that specific distance or the absolute shortest path time + 10 minutes for shortest path time less than 15 minutes, fleet: 90 taxis equally distributed among 15 taxi depots - 6 taxis per taxi depot – following the demand distribution in each 8hour period (00:00-08:00_1 taxi, 08:00-16:00_3 taxis, 16:00-24:00_2 taxis)

Amount of transportation requests: 1500

Not allocated requests: avg: 2.5, min: 0, max:4

% of not allocated requests: avg: 0.16%

Needed taxis: avg: 75, min: 71, max: 77

Average passengers / distance(km): avg: 1.02, min: 1.002, max: 1.04

Total distance units (m): avg: 4972077.5, min: 4900440, max: 5082760

➢ <u>8th experiment</u>

<u>Experiment's components</u>: 500 demands for transportation, maximum number of passengers: 4 persons, *time window: 45 minutes*, maximum ride time: 1.6 times the absolute shortest path time for that specific distance or the absolute shortest path time + 10 minutes for shortest path time less than 15 minutes, fleet: 90 taxis equally distributed among 15 taxi depots - 6 taxis per taxi depot – following the demand distribution in each 8hour period (00:00-08:00_1 taxi, 08:00-16:00_3 taxis, 16:00-24:00_2 taxis)

Amount of transportation requests: 500

Not allocated requests: 0

% of not allocated requests: 0%

Needed taxis: 41

Average passengers / distance(km): 0.865

Total distance units (m): 2045080

- 9th experiment

Experiment's components: 500 demands for transportation, maximum number of passengers: 4 persons, *time window: 60 minutes*, maximum ride time: 1.6 times the absolute shortest path time for that specific distance or the absolute shortest path time + 10 minutes for shortest path time less than 15 minutes, fleet: 90 taxis equally distributed among 15 taxi depots - 6 taxis per taxi depot – following the demand distribution in each 8hour period (00:00-08:00_1 taxi, 08:00-16:00_3 taxis, 16:00-24:00_2 taxis)

Amount of transportation requests: 500
Not allocated requests: 0
% of not allocated requests: 0%
Needed taxis: 34
Average passengers / distance(km): 0.924
Total distance units (m): 1820440

- 10th experiment

Experiment's components: 500 demands for transportation, maximum number of passengers: 4 persons, time window: 15 minutes, *maximum ride time: 2.5 times the absolute shortest path time* for that specific distance or the absolute shortest path time + 10 minutes for shortest path time less than 15 minutes, fleet: 90 taxis equally distributed among 15 taxi depots - 6 taxis per taxi depot – following the demand distribution in each 8hour period (00:00-08:00_1 taxi, 08:00-16:00_3 taxis, 16:00-24:00_2 taxis)

Amount of transportation requests: 500
Not allocated requests: 0
% of not allocated requests: 0%
Needed taxis: 44
Average passengers / distance(km): 0.85
Total distance units (m): 2091670

- 11th experiment

Experiment's components: 500 demands for transportation, maximum number of passengers: 4 persons, time window: 15 minutes, *maximum ride time: 3 times the absolute shortest path time* for that specific distance or the absolute shortest path time + 10 minutes for shortest path time less than 15 minutes, fleet: 90 taxis equally distributed among 15 taxi depots - 6 taxis per taxi depot – following the demand distribution in each 8hour period (00:00-08:00_1 taxi, 08:00-16:00_3 taxis, 16:00-24:00_2 taxis)

Amount of transportation requests: 500
Not allocated requests: 3
% of not allocated requests: 0.6%
Needed taxis: 52
Average passengers / distance(km): 0.839
Total distance units (m): 2066590

Taking an overall look at the previous results, it follows that the percentage of not allocated requests is not significant in any set of experiments except for the 4th experiment, where the demand was 10 times bigger (5000 requests) and the fleet of 90 taxis managed to service only 68.4% of the total requests. (see Figure 19). However, the rejected requests can be serviced by the rest -not in use- taxis.

Also, concerning the taxis needed in order to meet demand, in Figure 20 one can see the fleet needed for each set of experiments. Regarding the sets that contain 500 requests for transportation, it is worth noting that there is a need of less taxis for less strict time windows. In Figures 23 and 24 it can be seen how time windows and maximum ride time (in terms of shortest path) affect the fleet needed. Moreover, about the 7th set of experiments which consists of 1500 requests with less strict maximum ride time and time windows at the same time, a fleet of 75 taxis achieved to meet demand.

In addition, the average passengers/distance(km) for each different set of tests is presented in Figure 21. For the sets that contain 500 requests, it goes up for less strict maximum ride time and time windows (see Figures 25, 26). Also, there is a more significant increase for the sets with higher demand. Therefore, in any set of experiments the average passengers/distance(km) is more than 0.5 that applies for a pure taxi transportation.

Last but not least, in Figure 22 one can see the total distance units(km) for each different set of experiments. It is noteworthy that concerning the sets which consist of 500 demands, in Figures 27 and 28 it is shown how different values of maximum ride time (in terms of shortest path) and time windows have an effect on the total distance units(km).

In conclusion, the measurements about major KPIs of our problem look very promising. However, further experimental investigations by changing different parameters and market characteristics are also suggested.

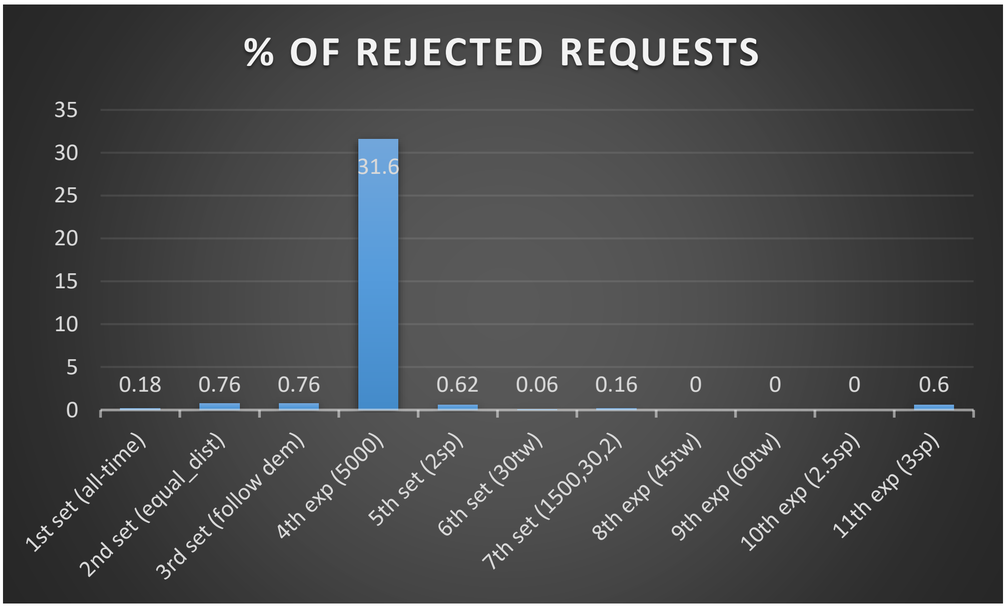

Figure 19. The percentage of rejected requests for each set of experiments

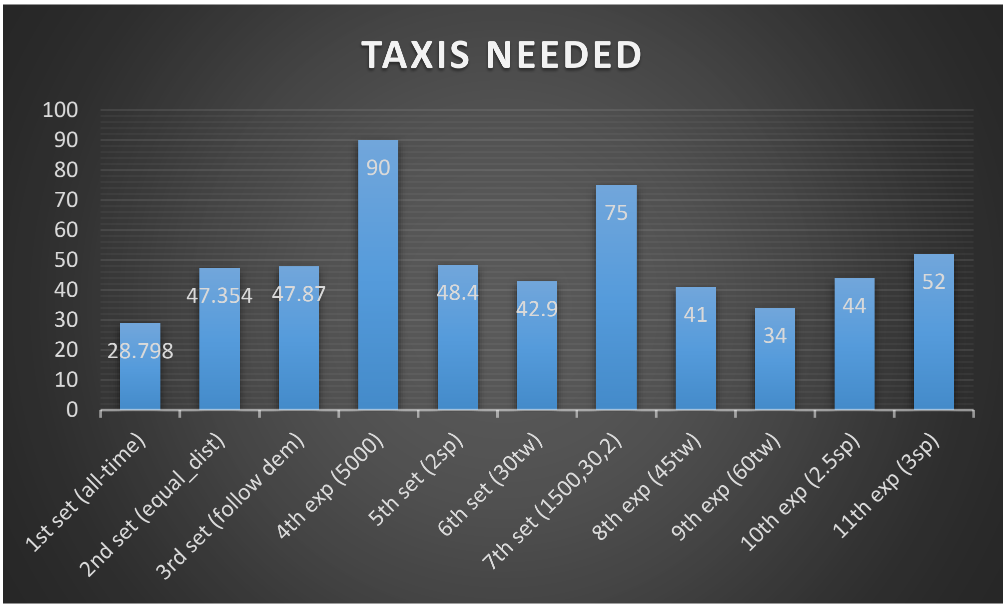

Figure 20. Taxis needed to meet demand for each set of experiments

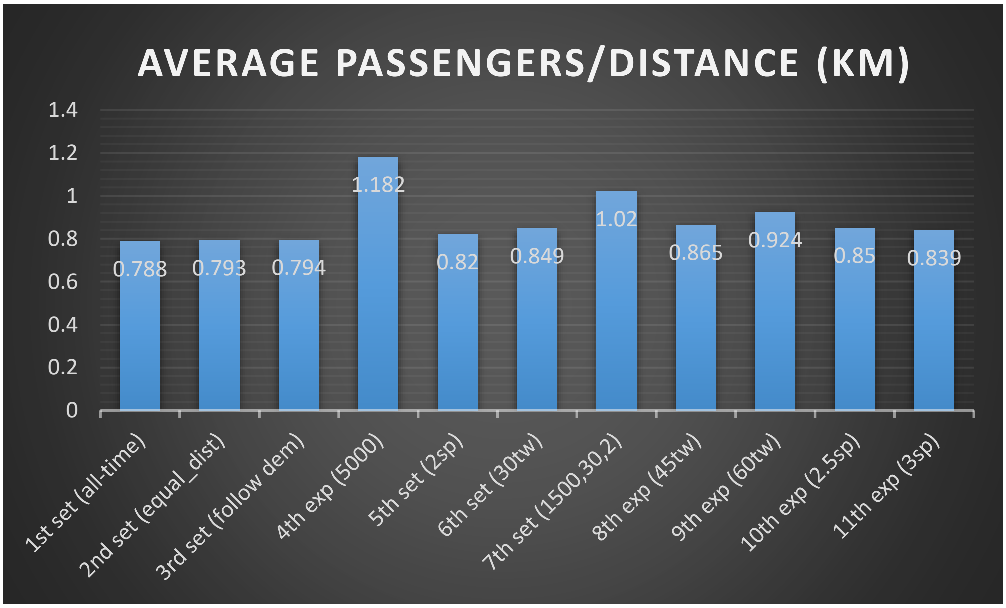

Figure 21. Average passengers / distance (km) for each set of experiments

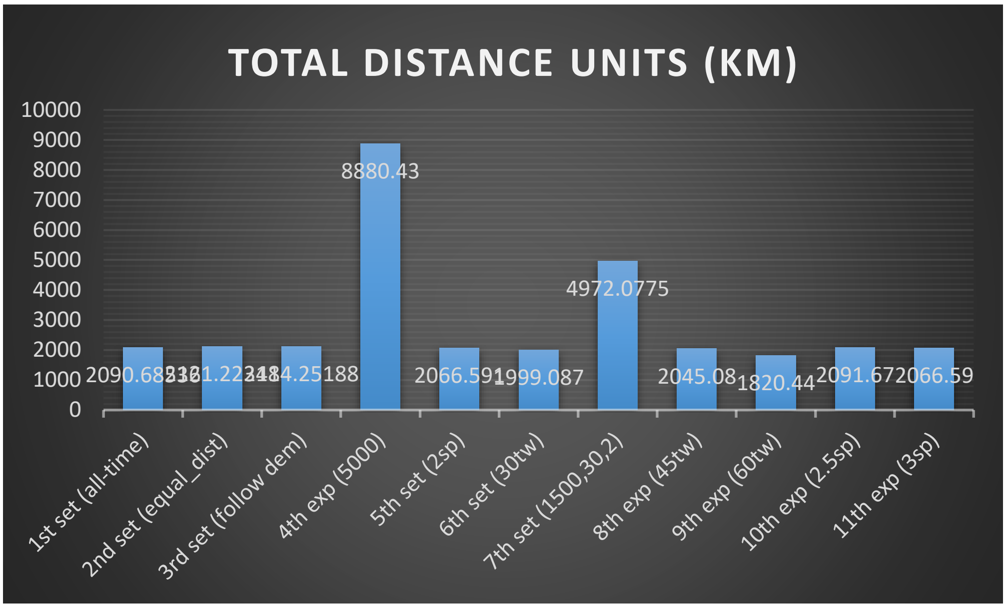

Figure 22. Total distance units (km) for each set of experiments

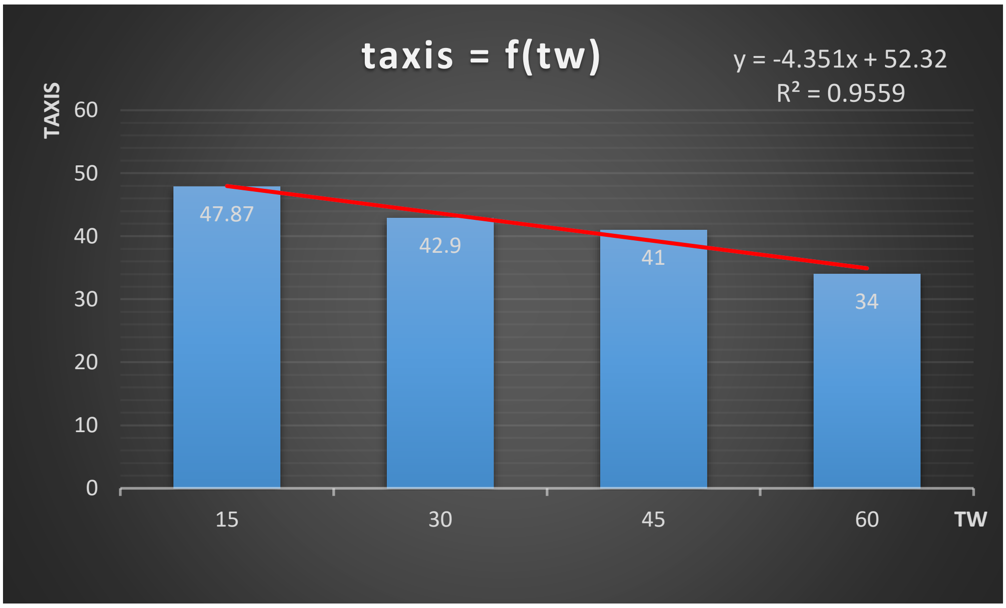

Figure 23. Taxis needed as a function of time windows

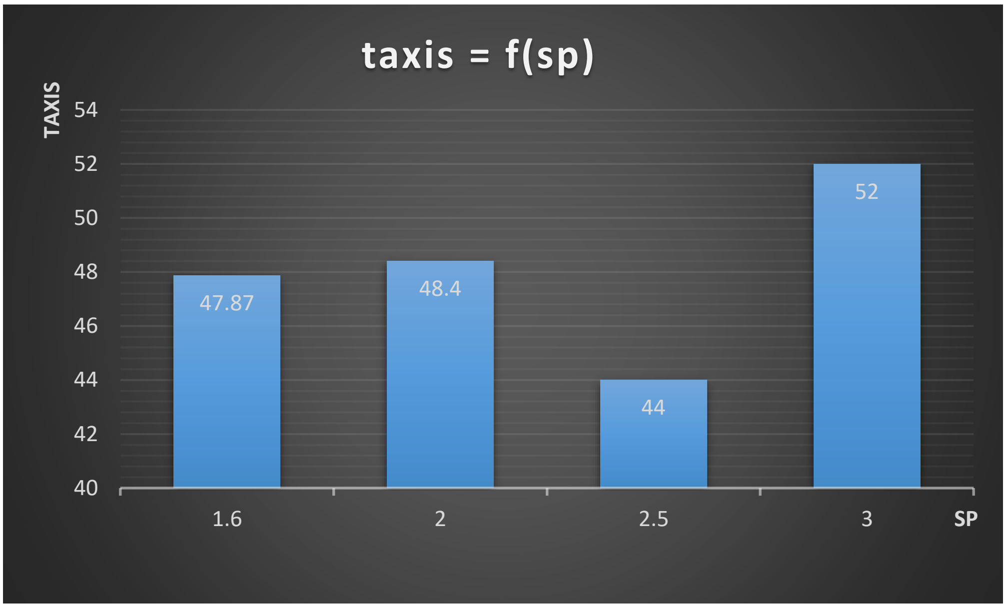

Figure 24. Taxis needed as a function of shortest path

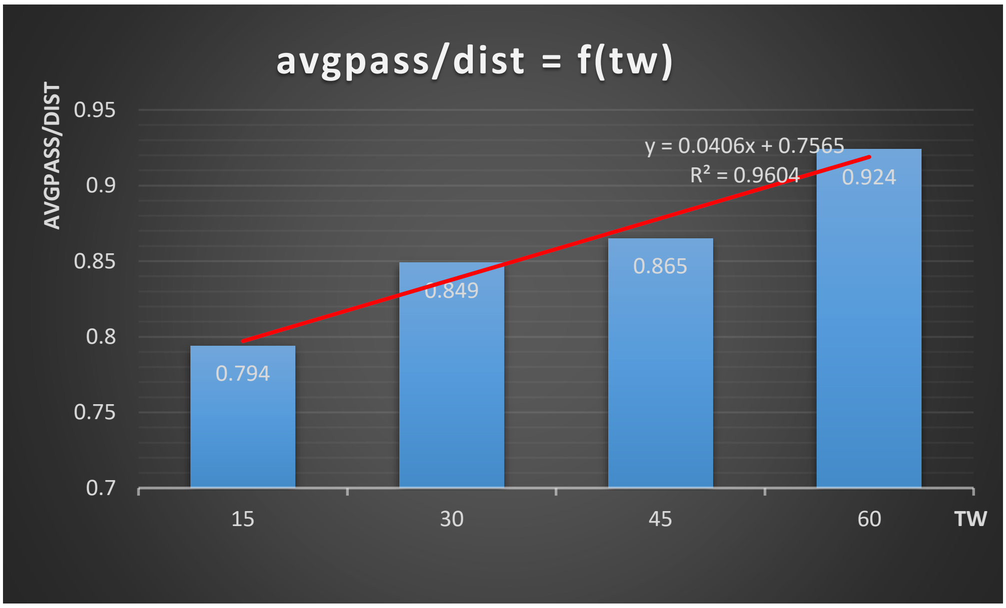

Figure 25. Average passengers/distance(km) as a function of time windows

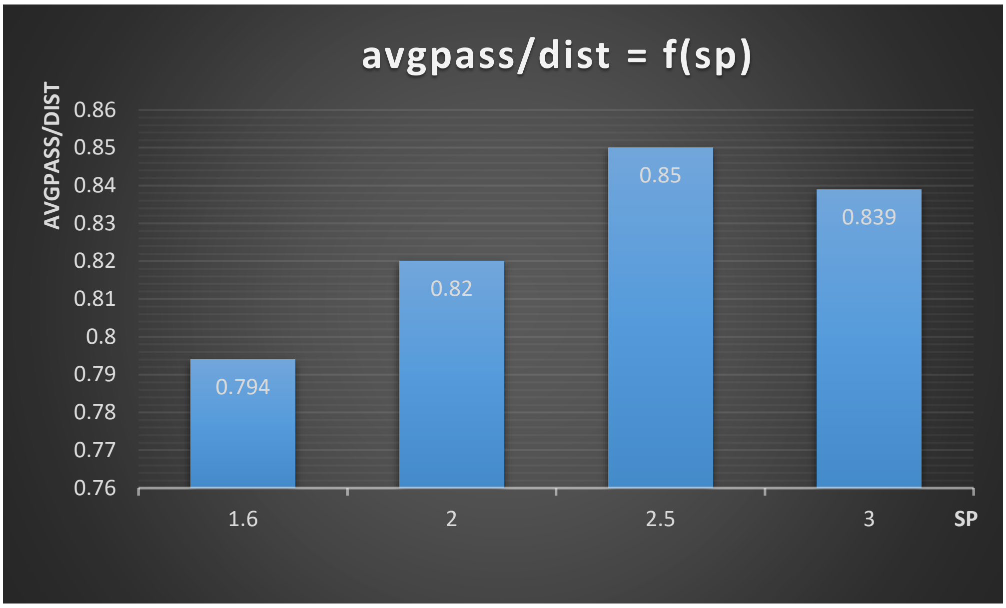

Figure 26. Average passengers/distance(km) as a function of shortest path

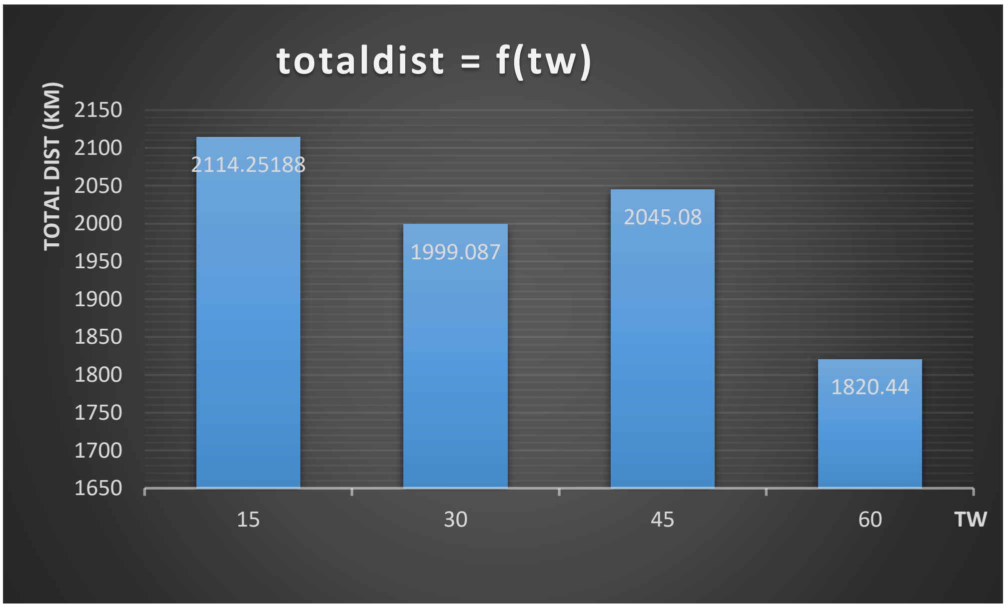

Figure 27. Total distance units (km) as a function of time windows

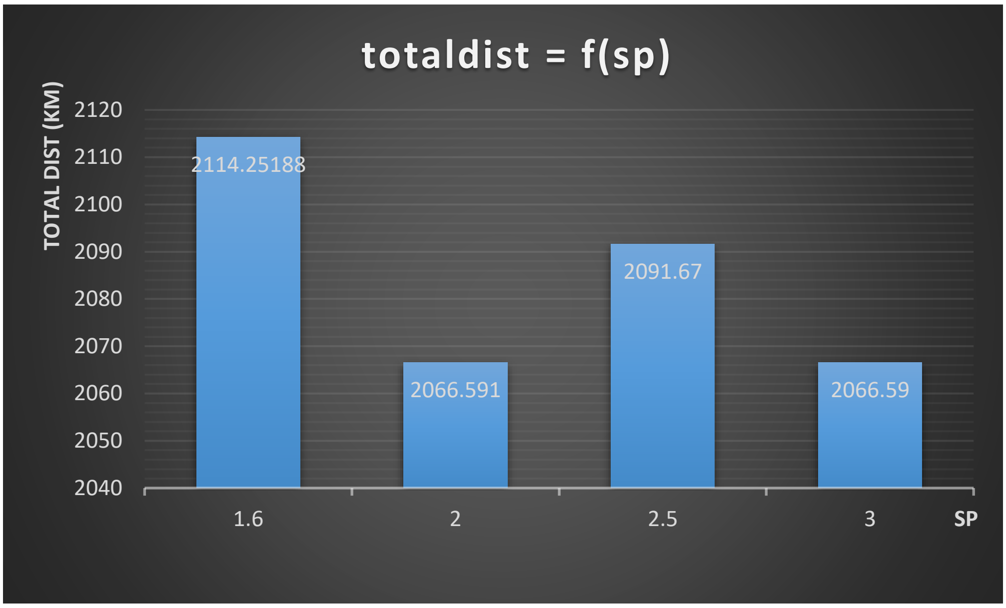

Figure 28. Total distance units (km) as a function of shortest path

Chapter 6. CONCLUSIONS – DIRECTIONS

In this thesis we focused on the Dial-a-Ride problem in its static version but also taking into account probabilistic information. Probability was added to the dial-a-ride problem in three different ways, by considering probabilistic pickup and delivery points and probabilistic times in which each request occurs. Probabilistic information became available through an extensive statistical analysis on historical data concerning the city of Volos, Greece. We constructed different test cases in accordance with our market's characteristics and then we used an insertion algorithm at first and a Regret based Heuristic algorithm at second (both proposed by Lois (2016)), in order to solve the Dial-a-Ride problem.

Our basic aim was to define the steps needed to design a data driven on-demand transportation system from scratch. Our work has led us to conclude that the required steps are the following:

Required steps for a data driven design of a DRT system

1. *<u>Collect data regarding your market</u>*

   In our case: city of Volos, Greece

   Data provider: VOLOSTAXI company

   Company's fleet: 200 taxis

   Data set: 959780 requests for transportation in the region of Volos for the last 4 years (2016 - 2019)

2. *<u>Display data on map</u>*

   In our case: distribution of data in each different region of Volos

3. *<u>Select area for research</u>*

   In our case: selected region of Volos where the most requests occurred

4. *<u>Divide into subregions</u>*

   In our case: we defined the centroids of each of the 16 discrete subregions

5. *Move to statistical data analysis*

In our case: a) Subregion based analysis concerning the total amount of requests, b) Subregion based analysis concerning the weekday, c) Weekday based analysis concerning the total amount of requests, d) Hour based analysis concerning the total amount of requests

6. *Run tests in accordance with your market's characteristics*

- Based on your previous statistical analysis and taxi expert's opinion, insert probabilistic information through cumulative probabilities
- Construct your experiments: create demands for transportation by one person at a time, based on probabilistic information about pickup and delivery points and the hour that they occur
- Construct your network (in our case: 44 points: 16 pick-up points, 13 delivery points, 15 taxi stations – depots)
- Calculate your cost matrix for your network (shortest path distance and shortest path time units).
- Set your parameters for each set of tests, for example: tw:15min, capacity: 4 passengers etc.
- Select your algorithms to solve the dial-a-ride problem and then run different set of experiments (e.g. change demand, fleet, distribution of the fleet in each time period in accordance with demand distribution, etc.)

7. *Analyze your results – Define strategies*

- Examine closely the experiments' results and state what these findings revealed about crucial factors of your problem (in our case: not allocated requests for transportation, needed taxis, average passengers / distance(km), etc.)
- Define strategies based on your results. For example, describe the fleet needed in order to meet your market's demand. Also, outline the benefits of a DRT service and a pricing policy. Furthermore, make a comparison with public transport and a potential redesign of bus routes or taxi stations, as it is described as follows.

Benefits of a DRT service for different stakeholders

- For Users (with different age, financial situation, mobility needs, trip purpose, ownership of private means of transport, etc.):
  - ✓ Improves mobility for all travelers including elderly, disabled, students, retired and commuters
  - ✓ Increases the level of service in terms of quality and cost
  - ✓ Transportation solution in a lower cost than a taxi or a private mean of transport and quite similar to a bus (especially when the fare is distributed among passengers)
  - ✓ Lower waiting time than bus pick-up points and less stress than driving and searching for a parking spot

- For Operators and Drivers:
  - ✓ Improved communications and dispatch system
  - ✓ Reduced operating cost
  - ✓ Higher demand and as a consequence higher profits based on the pricing policy

- For Society:
  - ✓ Better people’s mobility especially for the old and disadvantaged that have mobility problems
  - ✓ Less cars and traffic and consequently less pollution
  - ✓ Eco-friendly especially using electric cars
  - ✓ Demand satisfaction for transportation where the existing infrastructure cannot satisfy the requests

Pricing policy – Could an on-demand transportation system be profitable?

At this point, it is interesting to examine the profits of an on-demand transportation system. Concerning Greece, taxis determine the final price for the passenger by calculating the cost of the distance units (0.74€/km) plus a fixed cost of 1.29€ regardless the route distance. By taking a look at the previous results and taking into account the cost parameters, we could show that an on-demand transportation system could be profitable. For example, regarding the 7th set of experiments, the average total distance units that required for the allocated demands is 4972.0775 km. Also, the average for the average passengers/distance(km) is 1.02 in contrast with the 0.5 passengers/distance(km) that applies for a pure taxi transportation. In other words, if a new on demand transportation system prices only the distance units (e.g. 0.74€/km as a pure taxi) without the fixed cost gains more than operating as a pure taxi without the fixed cost. (e.g. 4972.0775km*0.74€/km*1.02 ≈ 3752.93€ > 4972.0775km*0.74€/km*0.5 ≈ 1839.67€)

Could a demand responsive transportation system answer to insufficient public transport?

From another point of view, it is of paramount importance to investigate if the existing routes of public transport are efficient. For example, in our case concerning the city of Volos let us take a look at subregions 2 and 6, which attain together a considerable proportion of the total demand for transportation (24%). These subregions are about the wider area of the city of N.Ionia. Concerning this surrounding area, the bus provider of the city (ASTIKOVOLOU) offers two routes (of the 13 available) from different starting points and the same destination point. Although the starting and the intermediate stations may differ, it seems for the routes to be inconvenient as it is indicated by the percentage of 24% of the total requests for transportation from this area to the taxi provider of the city. On the other hand, a demand responsive transportation system could fill the gap that now exists in a lower cost than a taxi and quite similar to a bus. Moreover, this issue could be also taken into account by the bus provider in order to redesign the existing routes based on the people's needs. In a city of the future, transport systems should adapt to the people's needs for transportation not people adapt to existing routes which in many cases have been designed a long time ago.

A potential redesign of the existing taxi stations - depots

Another issue arises regarding the redesign of the existing taxi stations – depots in the city of Volos. These spots have been established many years before and since then the demand for transportation in the area has been utterly changed. For example, as mentioned above, in subregion 2 which attains approximately 10% of the total requests for transportation (see Figure 29), there is no taxi station – depot and it is served by nearby taxi – stations (see Figure 30). As a result, a new taxi station – depot in subregion 2 should be considered. For this purpose, we investigated the 1-median problem of the facilities location problems. Thus, 4 potential taxi stations – depots in subregion 2 were spotted, in accordance with subregion's spatial characteristics and taxi expert's opinion (in main roads and close to parks, schools, cafes, bus stops, etc.) (see Figure 31). The goal was to find where the new taxi station – depot should be located in subregion 2, in order to minimize the average travel distance to and from it for all passengers (800 points in subregion 2 were considered). The results shown that among the 4 potential taxi stations, the minimum average travel distance was found in point 2 – about 0.5km. Point 3 came next with 0.59km approximately. Moreover, each of the existing nearby taxi stations has an average travel distance of 1.94km, 1.32km and 1.13km. By putting in place the new taxi station there will be a saving of distance about 1.4km, 0.8km and 0.6km respectively for a one-way trip. It is noteworthy that as subregion 2 attains approximately 10% of the total requests for transportation, which means about 100 requests per day, there will be at least a saving of 140km, 80km and 60km respectively per day. The establishment of the new taxi station – depot in subregion 2 depends on an agreement between the authorities and the taxi company. One could further investigate new potential taxi station – depots all over the city.

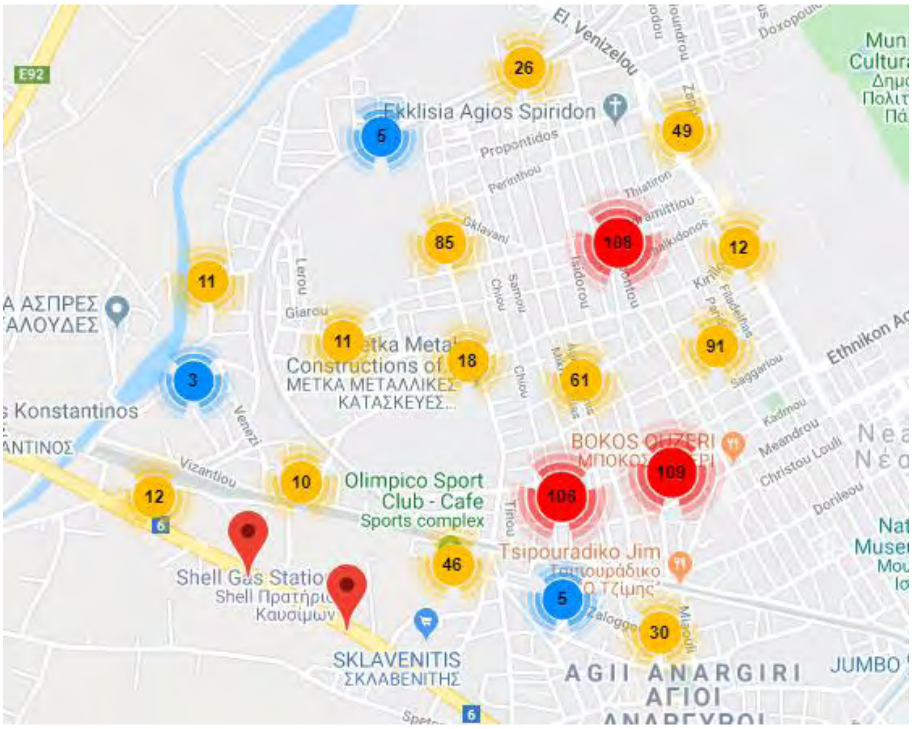

Figure 29. Requests for transportation in subregion 2

Figure 30. Nearby taxi stations – depots for subregion 2

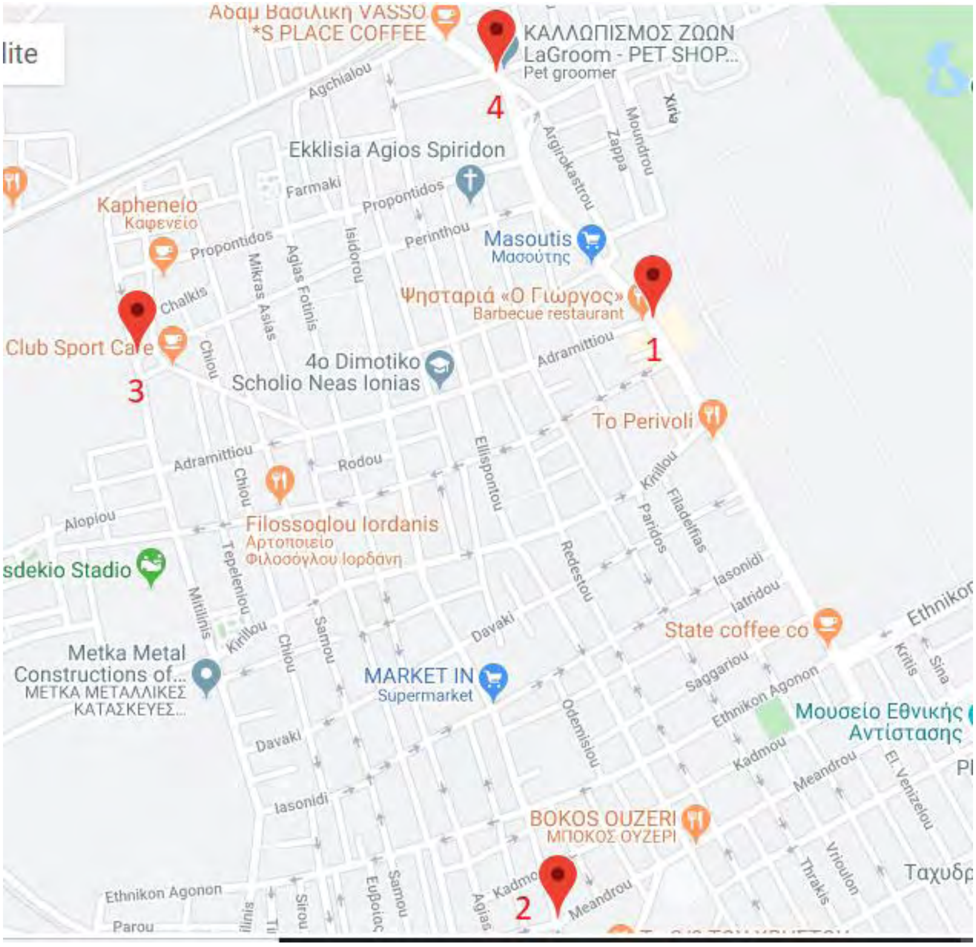

Figure 31. Potential taxi stations – depots for subregion 2

In conclusion, our computational results on randomly generated instances (mentioned in previous sections) look very promising, both in terms of meeting demand and in terms of computational efficiency. The running time for each experiment was quite similar and not more than 10 minutes for 500 requests in any case, about 1 hour for 1500 requests and 15 hours for 5000 requests. Planners can use the proposed methodology to design DRT services that meet demand and quality of service and promise plenty of benefits for all the stakeholders as mentioned above.

REFERENCES

Bent, R. W., & Van Hentenryck, P. (2004). Scenario-based planning for partially dynamic vehicle routing with stochastic customers. *Operations Research*, *52*(6), 977-987.

Birge, J. R., & Louveaux, F. (2011). *Introduction to stochastic programming*. Springer Science & Business Media.

Diana, M., & Dessouky, M. M. (2004). A new regret insertion heuristic for solving large-scale dial-a-ride problems with time windows. *Transportation Research Part B: Methodological*, *38*(6), 539-557.

Gomes, R., de Sousa, J. P., & Dias, T. G. (2014). A grasp-based approach for demand responsive transportation. *International Journal of Transportation*, *2*(1), 21-32.

Ho, S. C., & Haugland, D. (2011). Local search heuristics for the probabilistic dial-a-ride problem. *Or Spectrum*, *33*(4), 961-988.

Jaw, J. J., Odoni, A. R., Psaraftis, H. N., & Wilson, N. H. (1986). A heuristic algorithm for the multi-vehicle advance request dial-a-ride problem with time windows. *Transportation Research Part B: Methodological*, *20*(3), 243-257.

Li, B. (2016). Optimization of people and freight transportation: pickup and delivery problem variants.

Lois, A. (2016). *On the Online Dial-a-Ride Problem*. LAP LAMBERT Academic Publishing.

Máhr, T. (2011). Vehicle routing under uncertainty.

Molenbruch, Y., Braekers, K., & Caris, A. (2017). Typology and literature review for dial-a-ride problems. *Annals of Operations Research*, *259*(1-2), 295-325.

Powell, W. B., Jaillet, P., & Odoni, A. (1995). Stochastic and dynamic networks and routing. *Handbooks in operations research and management science*, *8*, 141-295.

Ritzinger, U., Puchinger, J., & Hartl, R. F. (2016). A survey on dynamic and stochastic vehicle routing problems. *International Journal of Production Research*, *54*(1), 215-231.

Schilde, M., Doerner, K. F., & Hartl, R. F. (2014). Integrating stochastic time-dependent travel speed in solution methods for the dynamic dial-a-ride problem. *European journal of operational research*, *238*(1), 18-30.

Srour, F. J., Agatz, N., & Oppen, J. (2016). Strategies for handling temporal uncertainty in pickup and delivery problems with time windows. *Transportation Science*, *52*(1), 3-19.

Tensen, I., UT, R. B., Hoeksma, I. B., & UT, B. M. (2015). *Stochastic optimization of the dial-a-ride problem* (Doctoral dissertation, Master thesis, University of Twente).

Thomas, B. W. (2007). Waiting strategies for anticipating service requests from known customer locations. *Transportation Science*, *41*(3), 319-331.

Toth, P., & Vigo, D. (Eds.). (2014). *Vehicle routing: problems, methods, and applications*. Society for Industrial and Applied Mathematics.