Spring 5-14-2022

Modeling Framework and Solution Methodologies for On-Demand Mobility Services With Ridesharing and Transfer Options

Sepide Lotfi
Southern Methodist University, slotfi@smu.edu

MODELING FRAMEWORK AND SOLUTION METHODOLOGIES FOR

ON-DEMAND MOBILITY SERVICES WITH RIDESHARING AND

TRANSFER OPTIONS

Approved by:

____
Dr. Khaled Abdelghany (Advisor)

____
Dr. Mohammed Khodayar

____
Dr. Barbara Minsker

____
Dr. Eli Olinick

____
Dr. Janille Smith-Colin

MODELING FRAMEWORK AND SOLUTION METHODOLOGIES FOR ON-DEMAND MOBILITY SERVICES WITH RIDESHARING AND TRANSFER OPTIONS

A Dissertation Presented to the Graduate Faculty of

Lyle School of Engineering

Southern Methodist University

in

Partial Fulfillment of the Requirements

for the degree of

Doctor of Philosophy

with a

Major in Civil and Environmental Engineering

by

Sepide Lotfi

B.Sc., Amirkabir University of Technology, Tehran, Iran
M.Sc., Amirkabir University of Technology, Tehran, Iran

Lotfi, Sepide B.Sc., Amirkabir University of Technology, Tehran, Iran, 2008
M.Sc., Amirkabir University of Technology, Tehran, Iran, 2012

Modeling Framework and Solution Methodologies for On-Demand Mobility Services with Ridesharing and Transfer Options

Advisor: Professor Khaled Abdelghany

Doctor of Philosophy conferred May 14, 2022
Dissertation completed April 5, 2022

The growing complexity of the urban travel pattern and its related traffic congestion, along with the extensive usage of mobile phones, invigorated On-Demand Mobility Services (ODMS) and opened the door to the emergence of Transportation Network Companies (TNC). By adopting the shared economy paradigm, TNCs enable private car owners to provide transportation services to passengers by providing user-friendly mobile phone applications that efficiently match passengers to service providers. Considering the high level of flexibility, convenience, and reliability of ODMS, compared to those offered by traditional public transportation systems, many metropolitan areas in the United States and abroad have reported rapid growth of such services.

This dissertation presents a modeling framework to study the operation of on-demand mobility services (ODMS) in urban areas. The framework can analyze the operation of ODMS while representing emerging services such as ridesharing and transfer. The problem is formulated as a mixed-integer program and an efficient decomposition-based methodology is developed for its solution. This solution methodology aims at solving the offline version of the problem, in which the passengers' demand is assumed to be known

for the entire planning horizon. The presented approach adopts a modified column generation algorithm, which integrates iterative decomposition and network augmentation techniques to analyze networks with moderate size.

Besides, a novel methodology for integrated ride-matching and vehicle routing for dynamic (online) ODMS with ridesharing and transfer options is developed to solve the problem in real-time. The methodology adopts a hybrid heuristic approach, which enables solving large problem instances in near real-time, where the passengers' demand is not known a priori. The heuristic allows to (1) promptly respond to individual ride requests and (2) periodically re-evaluate the generated solutions and recommend modifications to enhance the overall solution quality by increasing the number of served passengers and total profit of the system.

The outcomes of experiments considering hypothetical and real-world networks are presented. The results show that the modified column generation approach provides a good quality solution in less computation time than the CPLEX solver. Additionally, the heuristic approach can provide an efficient solution for large networks while satisfying the real-time execution requirements.

Additionally, investigation of the results of the experiments shows that increasing the number of passengers willing to rideshare and/or transfer increases the general performance of ODMS by increasing the number of served passengers and associated revenue and reducing the number of needed vehicles.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

INTRODUCTION

1-1. Background

The increasing complexity of the urban travel pattern and associated traffic congestion, and the widespread usage of mobile phones invigorated On-Demand Mobility Services (ODMS) and opened the door to the emergence of Transportation Network Companies (TNC). For private car users, ODMS provides alternative door-to-door service without the burden of driving and searching for parking, especially in congested situations. While the ODMS is relatively more expensive for transit users, the service is more convenient and reliable than conventional transit services. Many metropolitan areas in the United States have reported rapid growth of such services. For instance, major United States metropolitan areas such as San Francisco, Dallas, and Washington, D.C. have reported that Uber served about 71%, 50%, and 49% of their expensed rides in 2016, respectively (Fischer, 2017). The number of trips given by Uber has jumped from 3.79 billion in 2017 to 4.98 billion in 2020 (Iqbal, 2022). In addition, Lyft announced that its annual rides reached over $8.1 billion, accounting for 39% of the market share in the United States. (Carson, 2022).

The concept of ridesharing has first emerged in the 1950s in the form of carpooling, where a group of commuters shares the exact origin, destination, and time windows for pickup and drop-off. The cost of the trip is usually split equally among the group members. The concept has then evolved, allowing a driver of a privately-owned vehicle to pick up and drop off passengers at any location along the driver's route towards her/his destination. With the emergence of TNCs, ODMS with a ridesharing option is offered at a discounted rate. Efficient vehicle tours are constructed to visit the passengers' pickup and drop-off locations at pre-specified time windows (Cozza, 2019).

Fiedler et al. (2018) quantified the potential of ridesharing in an ODMS system in Prague. Their simulation results show that the average occupancy of a vehicle will increase to 2.7 passengers compared to the system without ridesharing. Accordingly, the mileage traveled by vehicles will decrease to 35% of the amount in the ODMS system without ridesharing and to 60% of the number of private car trips.

In a similar study, Soza-Parra et al. (2021) presented a set of experiments to identify the benefits of ridesharing and concluded that compared to private rides, a carpooling service could reduce the vehicle travel time by 18-59%, assuming a fixed demand level and depending on the concentration of travel destinations and the trip length distribution. Their experiments indicate that the vehicle occupancy rate in a carpooling service varies from 1.25 to 1.74.

The effort is underway by most ODMS providers to develop new services to increase their market share while maintaining the first-rate customer experience. For instance, TNCs have introduced new services, which provide passengers the option to rideshare in return for discounted fares (e.g., UberPool). Furthermore, providing passengers the option to transfer between vehicles or existing transit services is under investigation (Campbell, 2017; Ma et al., 2017, Lotfi et al., 2019). If a passenger accepts the transfer option, the TNC offers the service at a discounted rate to compensate for the inconvenience associated with transferring. A parallel effort is proceeding by urban transportation planners to understand the short- and long-term impacts of ODMS on the travel pattern in their regions as many policies, planning, and operation questions related to these services are rapidly arising (DuPuis et al., 2015).

Generally, “ridesharing” is used when drivers give rides to travelers for profit. In this case, drivers do not have a specific destination. Instead, they are moving in the network to find new passengers. On the other hand, “carpooling” is used when a driver and passengers have the same destination and share the car to reduce the trip cost. The United States Congress, in June 2012, signed into law the "Moving Ahead for Progress in the 21st Century (MAP-21) Transportation Act” and included online ridesharing in the definition of carpooling to make ridesharing eligible for federal funds that were previously dedicated for carpooling.

Previous studies in several cities imply that ride-hailing services, which are individual customer-focused, may lead to higher congestion levels, primarily due to the addition of empty miles that vehicles are traveling to reach the origin of the passengers. In a study by Tirachini et al. (2020), the effect of a ridesharing system on traffic congestion that offers shared-ride services with a car, van, or bus in Mexico City was investigated. They found that the effect of ridesharing on vehicle miles traveled (VMT) depends on many factors, including the availability of public transportation services as a replacement for ridesharing and the occupancy distribution in ridesharing vehicles. Their sensitivity analysis denotes that offering hailing services may increase VMT (in the range of 4.3 to 6.2 mile/passenger), while shared vans can decrease VMT (around −0.1 to −0.7 mile/passenger), whereas buses are estimated to increase VMT (0.2 to 0.7 mile/passenger). They argued that these differences between VMTs are due to the tradeoff between increasing the occupancy rates per vehicle and decreasing the attractiveness of the service for passengers for larger vehicles. One significant result of their study is the relevance of ridesharing with more occupancy rate in decreasing the VMT and congestion on the roads.

In a similar study, Oh et al. (2021) presented a simulation framework to evaluate the impacts of Automated Mobility on-Demand (AMOD) on network traffic, congestion, energy consumption, and vehicle emissions. Their study showed that AMOD services might induce additional traffic on the network resulting in more congestion. This congestion is due to the demand patterns, dead-heading, and empty trips for operational purposes.

Many factors could cause motivation or reluctance of passengers to share the ride with others. Alonso-Gonzalez et al. (2020) investigated these elements and stated that addition to the travel time due to ridesharing is the primary concern for travelers when they are considering this service. Their simulation experiments showed that willingness to share the ride for most passengers depends on the tradeoffs between travel time and cost. This result emphasizes the importance of considering proper ridesharing discounts when TNCs offer this type of service.

Nonetheless, the success of these efforts requires the availability of adequate tools that can be used to evaluate new services and answer arising policy, planning, and operation questions. For example, ODMS providers are interested in predicting passengers' demand in response to changes in the service configuration, estimating the number of required drivers and their working schedules, determining pricing schemes and driver payments, and estimating operation costs and revenues, to name a few. Similarly, transportation planners are interested in predicting the travel demand induced by ODMS, possible changes in the modal split and route assignment patterns, and the impact of ODMS on existing transit services and parking facilities. A tool that can address these questions should be able to represent (I) passengers' key characteristics, including distribution over time and space, and flexibility of time windows for pickup and drop-off; (II) service availability and pricing schemes; and (III) emerging services such as ridesharing and transferring while taking into consideration passengers' preferences on these services.

Considering transfer services would enable ODMS providers to provide service in situations where there is a shortage in the vehicle supply. For instance, TNCs might not be able to recruit an adequate number of drivers in some locations or during specific periods to serve the anticipated passengers' demand. Although enabling the transfer service may cause an increase in passengers' travel time, it would allow TNCs to serve more customers using fewer vehicles. Besides, offering discounted services such as ridesharing and transfer would allow TNCs to diversify their services to attract demand from low-income groups. Furthermore, the operating cost of serving a transferring passenger is minimal compared to that of other passengers. Thus, designing efficient networks that accommodate more transferring passengers could significantly enhance network profitability.

Moreover, the main requirement in offering ODMS is to match ride seekers to vehicles and confirm service availability promptly. In addition, vehicle tours need to be optimally designed to ensure that the service is offered efficiently. An efficient service will serve more passengers with a limited number of vehicles without violating passengers' preferences on ridesharing and transfer and their time window constraints. Several challenges complicate satisfying these requirements. For example, the problem can be viewed as an extension of the traditional vehicle routing problem with time window constraints, which is an NP-hard problem as its execution time grows exponentially with the problem size defined in terms of the number of vehicles and number of passengers (Cordeau et al., 2007). In addition, the demand for ODMS is not known a priori. As such, the problem is typically solved myopically upon the arrival of a new ride request. The inability to foresee the demand for an extended horizon is expected to result in less-efficient solutions as vehicles might not be optimally assigned or routed.

This research describes a modeling framework for ODMS operation that addresses these requirements. The framework presents a mixed integer programming of the problem and provides an efficient methodology for its solution. The optimization-based solution methodology integrates iterative decomposition and network augmentation techniques to analyze networks of moderate sizes. In addition, a novel heuristic-based methodology for integrated ride-matching and vehicle routing for ODMS with ridesharing and transfer options is presented. The methodology adopts a hybrid heuristic approach, which enables solving large problem instances in near real-time. The methodology enables ODMS providers to (1) promptly respond to individual ride requests and (2) periodically re-evaluate the generated solutions and recommend modifications to enhance overall efficiency.

The framework evaluates the network performance from the operator and passenger’s perspectives by providing different operational and financial performance measures. The framework is also ready for integration within the conventional transportation planning processes to enable transportation planners to study the impact of ODMS on the travel pattern in their regions.

1-2. Research Motivation

Significant growth of ODMS has been reported over the past decade in many cities worldwide. This growth is due to the high convenience of these services that reduce dependence on private cars and eliminate challenges associated with driving and finding parking spaces in congested areas and/or during bad weather conditions. ODMS services are also getting more popular by offering a variety of personal mobility choices ranging from simple rideshare to a business class service.

Soon, we will see new business models that transform many traditional industries into services-on-wheels; therefore, ODMS will play an essential role in all businesses. For instance, Toyota introduced the e-Palette Concept Vehicle to satisfy multi-function transportation and business demands and provide different transportation experiences based on the purpose of the trip, which could include a wide range of services such as transporting people and delivering packages, serving as shopping centers, restaurants, offices, and hotel rooms (Bakutyte, 2017). This will cause a rapid change away from personal vehicle ownership to a shared, on-demand mobility model. This shift in passengers' behavior encourages transportation planners to develop models to study ODMS performance, investigate the challenges associated with ODMS growth, and study the effect of ODMS on different industries.

This research is inspired by the necessity of developing a framework for analysis of ODMS operation in urban regions while representing emerging services such as ridesharing and transfer. The framework captures the spatial-temporal interactions between service requests and routes the vehicles to maximize service coverage/revenue. Furthermore, the developed model could be adopted by many industries/services. In the following, the economic and social effects of ODMS on different industries are discussed.

1-2-1. ODMS and Food Delivery

The food delivery business from famous chain restaurants is not new; however, many companies have recently been founded, like DoorDash, Postmates, GrubHub, and UberEats, to offer food delivery services from local restaurants. For instance, DoorDash, a company that started business in 2012, attracted investors to raise more than $700 million, created thousands of jobs for drivers, and valued at $28 billion in 2022 (Curry, 2022). These companies have developed a new business model to serve customers who are willing to pay for the food delivered to their homes instead of going to the restaurant. This platform will also allow restaurants to increase their customers without adding new resources to serve them. Revenue from the online food delivery market is estimated to be more than $16 billion, with more than 78 million active users across the United States (Kohl, 2018).

1-2-2. On-Demand Delivery Services

Logistics companies can ship and deliver large quantities of products with low shipping costs. However, the efficient last-mile delivery to the customer’s door is still a challenge for retailers and delivery companies. Recently, technology companies have introduced new services to solve this problem, rapidly gaining a considerable market share. These companies are referred to as on-demand delivery providers. Studies show that the total transaction value of the on-demand delivery services was $171 billion at the end of 2020 (Mazareanu, 2021). Recently, Target, one of America's biggest retailers, purchased a delivery company, Shipt, in a $550 million agreement (Shieber, 2017). This agreement shows the response of traditional warehouses and retailers to the Amazon business model, which is delivering products to customers in a one-hour window. It should be noted that coining the pickup and drop-off of people and packages delivery is not an easy process for ODMS providers. There are many challenges like the waiting time of a passenger in the vehicle during pickup or drop-off of a package.

1-2-3. ODMS and Autonomous Mobility

Autonomous Mobility-as-a-Service (AMaaS) is the future of transportation. Autonomous taxi networks can offer ODMS with a cost of $0.35 per mile due to high utilization rates and eliminating the cost of a driver, which is almost half of the cost of driving private cars (Keeney, 2017). In addition, autonomous electric trucks and drones will deliver products at lower costs than traditional delivery methods. Due to this low-cost model, travelers will stop driving personal cars and use the autonomous mobility service. In addition, autonomous vehicles will provide better access to transportation to non-driving populations, such as people with disabilities and the elderly. Shortly, one might expect such service to be the first option for trips in urban areas or delivery services.

1-2-4. ODMS and Health Services

Recent studies show that 30% of hospitals' medical appointments are being canceled in the United States, which imposes over $150 billion costs on the healthcare system each year (Gadam, 2018). Shortage of reliable transportation is one of the leading causes of these missed appointments, especially for senior people. Uber recently launched UberHealth that provides reliable, comfortable, and safe ODMS to healthcare organizations for patients. Lyft provides the same service via Lyft Concierge and a new partnership with Allscripts, a large electronic health record company. These companies provide healthcare organizations with a platform to schedule rides for patients going to and from hospitals. These on-demand services enable people with limited mobility to get to healthcare organizations without facing the challenge of transportation.

1-2-5. ODMS and Public Transit

Travelers, traffic managers, and governments consider a new transportation model that integrates ODMS and public transit to solve the first/last mile problem and connect travelers to the transit network from/to areas with no transit coverage. A simulation study by Narayan et al. (2020) showed that an integrated fixed-route public transportation system with a flexible-route ridesharing service could decrease the average waiting time for travelers and cover 30% more of the demand compared to the traditional public transportation systems. Cats et al. (2021) studied Uber trip data in six cities in the U.S. and Europe and identified the best public transportation alternative for each ride. Comparing the Uber data and public transportation alternatives for each trip showed that Uber trips have shorter out-of-vehicle and in-vehicle travel times for most passengers than their fastest public transport counterparts. For 13-36% of the Uber trips, the travel time associated with ridesharing was at least twice as fast. Moreover, for 0-7% of the Uber trips, no alternative public transportation was found within a reasonable walking distance.

Particularly in congested urban areas, instead of adding new bus or subway lines, planners turn to the shared economy to expand urban transportation networks and provide access to public transit to more people (Elliott, 2017). The City of Arlington, Texas, is among the first cities in the United States to offer ODMS as a public transportation solution. The “Via” rideshare program provides affordable transportation to entertainment, shopping and dining options, work or school, and medical appointments. Customers also can connect to the Trinity Railway Express (TRE) Station, where they can catch a train to Dallas or Fort Worth (Schrock, 2017).

Moreover, Masabi, a technology company that is the leader in mobile ticketing, has announced a partnership with Uber to add the option of purchasing public transit tickets into the Uber application, which allows users to transfer from TNCs to public transit services for convenient and affordable multi-modal trips (Gooch, 2018).

1-2-6. ODMS, Transportation Equity, and Job Accessibility

Transportation equity is an essential factor to assure everyone can access opportunities such as jobs, healthcare, and services. Transportation planners face challenges ensuring transportation equity due to geographic, economic, and socio-demographic diversity. Shared mobility, including bike-sharing, carsharing, and TNCs that provide users affordable on-demand access to transportation, has been considered as a solution to these challenges (Shaheen et al., 2017). Shared mobility is a good travel option in a multi-modal transportation system, particularly in low population areas and during off-peak hours, but still, there are many challenges in attracting low-income communities, minorities, and the elderly.

1-3. Research Objectives

Several objectives are considered for this research. First, an inclusive literature review will be conducted to review current research works related to the ridesharing systems and their solution procedures. The second objective pertains to developing a modeling framework for the ODMS with the transfer option. The problem is formulated in the form of a mathematical programming formulation to determine the most optimal routes of vehicles to maximize revenue. Lastly, the third objective is to develop efficient solution methodologies to solve large-size problems. We present two solution methodologies that target different applications. The first methodology focus on ODMS offline planning applications in which the passengers' demand is known for the entire horizon of interest. The second methodology is more suitable for online applications in which the service is scheduled in real-time to meet the received ride requests.

1-4. Research Contribution

This research contributes to the current literature in several ways. First, to the authors' understating, it is among the first efforts to provide a comprehensive methodology for the operation of ODMS considering a variety of service options, including single-ride, ridesharing, and transfer. While considerable research work considered ODMS with ridesharing option, to our knowledge, the research effort that focuses on developing methodologies that combine ridesharing and transfer options for real-time ODMS dispatch is still in its infancy. As these services evolve, rides with a transfer are expected to grow, especially when integrated with existing public transportation services to support first/last-mile transportation.

Second, existing models assume homogenous passengers in terms of their service preferences. For instance, most models assume passengers are identical in terms of the flexibility of their pickup and drop-off time windows, willingness to carpool or transfer, sensitivity to ride with a particular gender, and inclination to travel through or transfer in certain parts of the region. Capturing the heterogeneity of customers' preferences while quantifying the effect of different passengers' preferences on the number of served passengers, total profit, travel time, and total mileage traveled by vehicles and passengers enhances the fidelity of these tools in representing the demand and supply interactions.

Third, the framework represents the vehicle routes using the existing roadway network; thus, enabling it to capture the effect of congestion on routes used by the drivers and helping to map their vehicles as part of the traffic in the roadway network.

Fourth, this research presents a novel solution methodology, which adopts a modified version of the column generation methodology to account for transfer trips. The methodology allows solving problems of moderate sizes compared to other similar problems reported in the literature.

Fifth, this research proposes a heuristic solution methodology explicitly designed to suit the real-time nature of ODMS operation. Most existing methodologies adopt computationally-demanding optimization frameworks that fail to solve large-scale problems while meeting the real-time execution requirement. We propose a rollback approach to periodically re-optimize the generated vehicle routing solutions to overcome the problem of not foreseeing the ODMS demand for an extended horizon.

Finally, this research demonstrates the framework's application considering a real-world network. A network representing the downtown area of the City of Dallas is used to examine the performance of the developed solution methodology considering different combinations of passengers' demand and the number of vehicles available to provide ODMS.

1-5. Dissertation Organization

This dissertation consists of eight chapters. Chapter 2 offers a revision of different methods given in the existing literature to formulate and solve the ODMS problem. Chapter 3 defines the ODMS with ridesharing and transfer options and provides a mathematical formulation describing the problem's decision variables, objective function, and constraints. Chapter 4 describes the optimization-based solution methodology used for solving the static version of the problem. Chapter 5 describes the heuristic-based solution methodology to solve the online version of the problem. Chapter 6 presents the results of applying the optimization-based solution methodology, while chapter 7 presents the results of the heuristic-based solution methodology. Finally, chapter 8 provides a summary and concluding comments on the research tasks that have been completed.

Chapter 2

BACKGROUND REVIEW

2-1. Introduction

The vehicle routing problem (VRP) is a well-studied optimization problem that pertains to determining the optimal routes and schedule of a set of vehicles to serve passengers or deliver packages. The problem was first introduced by Dantzig and Ramser (1959), which was considered a generalization of the traveling salesman problem (TSP). The solution methodologies developed to solve the TSP determine the shortest route to visit all customers, where the routes start and end at a depot. The problem has been extensively studied since then and researchers focused on different aspects of this problem (Balinski and Quandt, 1964; Fisher and Jaikumar, 1978). Good coverage of advances in formulating and solving vehicle routing problem classes can be found in Cordeau et al. (2007), Agatz et al. (2012), Furuhata et al. (2013), Braekers et al. (2016), and Vidal et al. (2019).

The Vehicle Routing Problem is a classical operations research problem with numerous applications. The problem is an NP-hard problem as the computation time required to obtain its optimal solution grows exponentially with the problem size (Golden et al., 2008). Many factors contribute to the complexity of the vehicle routing problem. For example, in the context of MOD services, the problem involves many decision variables, including vehicles' and passengers' routing decision variables and time windows variables. Besides, for problems that include ridesharing and transfer, more variables will be added to the problem representing passengers' preferences on ridesharing and transfer. Therefore, the execution time required to obtain the optimal solution grows exponentially as the problem size increases.

Moreover, the decision variables involved in this problem are not independent; thus, the problem cannot be optimized separately for passengers and vehicles. For example, the arrival and departure time of a passenger to the transfer node depend on the arrival time of the first vehicle which has the passenger onboard, and the departure time of the second vehicle which will pick up passengers from the transfer node.

Developing a modeling framework for studying ODMS operation in urban areas can be viewed as a generalization of the multi-vehicle Dial-a-Ride-Problem (DARP) in which a ride is requested by indicating passengers' pickup and drop-off locations and time windows for the earliest pickup and the latest drop-off times. Two general versions of the DARP are studied in the literature: the carpooling problem (Giuliano et al., 1990; Calvo et al., 2004) and the taxi-sharing problem (Braekers et al., 2014; Alonso-Mora et al., 2017; and Ho et al., 2018). In the carpooling problem, drivers define their destination and time window to reach that destination and move from their current location to the destination regardless of whether they find a passenger to share the ride with or not. On the other hand, the drivers do not have a specific destination in the taxi-sharing problem and are willing to change their routes to serve more passengers.

The problem can also be considered as a version of the Vehicle Routing Problem with Time Window (VRPTW), which is extensively studied in the literature (Cordeau et al., 2007; Eksioglu et al. 2009; Desaulniers et al., 2014; Braekers et al. 2014; Toro et al. 2016; and Dixit et al. 2019).

This chapter reviews the vehicle routing problem and its extensions related to the ODMS problem. Section 2-2 provides a review of the classical single-vehicle routing problem, while section 2-3 reviews the literature related to the multi-vehicle routing problem. Section 2-4 presents the different formulations and solution methodologies for routing problems with online demand. Section 2-5 describes an essential extension of the problem, which includes transfer. Research works reviewed in this section consider the transfer of packages and passengers in the vehicle routing problem. Finally, section 2-6 concludes this background review and highlights the main research gaps identified in the literature.

2-2. The Single-Vehicle Routing Problem with Time Window

Early research works have focused on the single-vehicle routing problem with time windows. Examples of the exact algorithms to solve this problem include: set partitioning and column generation (Balinski and Quandt, 1964), dynamic programming (Eilon et al., 1971), and branch-and-bound algorithm (Laporte et al., 1986).

Exact algorithms are developed to obtain optimal solutions for the problem. However, these algorithms can only be applied to problems with a small number of vehicles and passengers because they require high computation time, even for a small problem. In one of the first research works to develop an exact algorithm, Psaraftis (1980) presented a branch-and-bound approach with the objective to minimize the weighted combination of the time to serve all customers and the dissatisfaction experienced by passengers. They defined a linear function to measure this dissatisfaction, which correlates each customer's waiting and riding times with their level of dissatisfaction. Their research focused on solving both static and dynamic versions of the problem. In the static problem, new requests that may appear during the execution of the problem are not considered. On the other hand, when the dynamic version of the problem is solved, new requests are considered into the problem at the time of their arrival. Furthermore, they extended the developed algorithm and presented a framework to solve problems with specified pickup and drop-off time windows for each passenger (Psaraftis, 1983).

Another example of developing exact algorithms for the single-vehicle routing problem with time windows can be found in Sexton et al. (1985a, 1985b). They implemented Benders' decomposition technique to solve the single-vehicle VRPTW. The final solution to their problem is the order of pickup and drop-off for requests, as well as the times of pickup and drop-off of each request. Similar to the Psaraftis (1980), the objective of the problem was to minimize the total customer inconvenience.

Many researchers started developing heuristic algorithms to generate good quality near-optimal solutions and reduce computation time compared to exact methods to solve large-size problems in a reasonable computational time window. One of the first heuristic algorithms for solving the VRP is the famous method of Clarke and Wright (1964). The algorithm is developed based on the saving concept to find one route with the highest saving cost by merging two routes. Similar to all other heuristic methods, Clarke and Wright's algorithm is designed in a way to provide a near-optimal solution, but there is no guarantee to find the optimal solution (Clarke and Wright, 1964).

Many other researchers have focused on improving the solution quality and the computation time of the Clarke and Wright algorithm. For example, Wark and Holt

(1994) introduced a matching heuristic using Clarke and Wright algorithm and investigated the use of parallel computing to reduce the running time of their heuristics. Reimann et al. (2004) presented a heuristic algorithm that implemented the Clarke and Wright saving approach and decomposition method that results in small subproblems and solved them using an Ant System process.

Hosny and Mumford (2010) compared different heuristic approaches and presented a heuristic-based solution methodology that implements intelligent neighborhood moves. Their objective function minimizes the total route duration and the degree of infeasibility in capacity and time window constraints. The constraints in their problem are considered soft, which means an infeasible solution that violates the capacity, and/or the time windows constraints will be penalized by adding a term in the objective function.

Unlike the heuristic approaches, the metaheuristics algorithms may even accept a temporary deterioration of the solution (moves that worsen the objective function value) which allows them to explore more thoroughly the solution space and, therefore, to get a hopefully better solution (that sometimes will coincide with the global optimum). Similar to heuristic methods, metaheuristics do not guarantee to find the optimal solution. Examples of metaheuristics used to solve the VRP include tabu search (Glover, 1986), simulated annealing (Corana et al., 1987), genetic algorithms (Goldberg, 1989), and ant search algorithms (Bullnheimer et al., 1997).

2-3. The Multi-Vehicle Routing Problem with Time Window

A significant number of studies focused on the more complicated version of the problem in which multiple vehicles are routed, namely, the multi-vehicle routing problem. The problem has been considered for dial-and-ride services as well as the package pickup and delivery services. Jaw et al. (1986) were among the first to consider the multi-vehicle DARP with time window constraints and presented a heuristic-based solution for this problem. To consider a specific time window for passengers, they defined two elements: (I) the amount of delay added to the customer's desired pickup or delivery time; (II) the riding time that a customer can spend in a vehicle. Their two-step sequential insertion procedure first orders customers by the earliest pickup time and then inserts them into the vehicles' routes considering the cheapest feasible insertion criterion.

Dumas et al. (1991) presented an exact algorithm, which solves the pickup and delivery problem for package pickup and delivery considering multiple depots. They were the first to use column generation for solving VRPTW. Their algorithm uses a column generation scheme with a constrained shortest path as a sub-problem. They considered heterogeneous vehicles, time windows, and multiple depots. The constrained shortest path problems in sub-problems are solved utilizing a forward dynamic programming algorithm.

A common approach to solve VRPTW is branch-and-price, a variant of the branch and bound in which the nodes are processed by solving linear-programming relaxations via column generation. Barnhart et al. (1998) adopted a branch and bound scheme in which lower bounds are computed by column generation.

Baldacci et al. (2004) studied the problem of offering carpooling services for company employees and proposed an exact solution methodology based on Lagrangian decomposition and column generation. Cordeau (2006) and Ropke et al. (2007) proposed a branch-and-cut approach with valid inequalities and efficiently solved small to medium-size problem instances.

Baldacci et al. (2011) proposed an exact algorithm that integrates set partitioning and column generation procedures for the VRPTW in which a set of identical vehicles are located at a central depot. The objective function minimizes the total travel cost plus the vehicle fixed cost. Hosni et al. (2014) proposed a mixed-integer program to solve the multi-vehicle version of the VRPTW. A Lagrangian decomposition approach was proposed, which decomposed the problem into a master problem and smaller single-vehicle sub-problems that are solved separately.

Santi et al. (2014) introduced the concept of shareability networks which allows modeling the benefits of sharing a taxi as a function of passenger inconvenience. They showed that although ridesharing will increase passenger discomfort, the average travel distance for passengers would decrease by 40% or more. Alonso-Mora et al. (2017) applied the concept of the shareability network to an extensive data set representing the taxi services in New York City to quantify the benefits of providing ridesharing services. Their solution algorithm starts from a greedy assignment and improves this passenger-vehicle assignment through a constrained optimization, quickly returning solutions of good quality and converging to the optimal assignment over time.

Mahmoudi and Zhou (2016) proposed a time-discretized multi-commodity network flow model based on the integration of vehicles' carrying states within space-time transportation networks. Their three-dimensional state–space-time network enumerates possible transportation states at a given time along with the vehicle's space-time paths and further allows a forward dynamic programming solution algorithm to solve the single-vehicle VRPTW problem. Tong et al. (2017) also used a similar space-time transportation network model to customize the design of bus services and developed a solution algorithm based on the Lagrangian decomposition for solving the problem.

Several metaheuristic approaches have been developed to solve the multi-vehicle routing problem with time windows. Huang (2016) presented an integer programming model for the carpooling problem and implemented the Tabu search algorithm to solve large instances of problems. The objective function of their problem was to minimize the total operating cost and assign the passengers to their nearest driver.

In a more recent study, Ma et al. (2017) proposed a genetic algorithm to find an efficient solution for a multi-vehicle routing problem with time windows. The presented model is limited to carpooling services in which a taxi collects all passengers from their origins and then travels to a predefined destination. In addition, the problem assumed a homogenous passenger population in terms of their service preferences. Moreover, the model does not allow the transfer of passengers between vehicles. Finally, the performance of their solution methodologies is evaluated using a relatively small network (24 nodes, nine passengers, and three taxis), which limits the validity of the obtained results for real-world operations.

Liang et al. (2019) proposed an integer programming model that optimizes the routing of the automated taxis intending to maximize the profit. They considered the effect of automated taxis' flow on travel times in the network. They developed a solution approach based on a customized Lagrangian relaxation algorithm, which identifies a near-optimal solution for this problem.

2-4. The Routing Problem with Online Demand

The reviewed studies above assume all passengers' demands to be known for the entire planning horizon. One of the challenging aspects of the DARP is to solve the online version of the problem in which passengers' demand is not known a priori. The problem entails determining whether to accept new ride requests upon their arrivals in real-time. If a ride request is accepted, a new vehicle is dispatched, or the route of one of the vehicles currently in the network is modified to serve this new request. Considering the complexity of the problem and the requirement to solve it in near real-time, existing solution methodologies mainly adopt heuristic-based approaches that solve for near-optimal solutions.

For example, Attanasio et al. (2004) implemented parallel tabu search heuristics for the online DARP to insert new passengers into planned routes with the objective to maximize the number of served requests. The algorithm developed for the problem works as follows: an initial solution is obtained by randomly assigning requests to routes while satisfying constraints. Starting from the initial solution S_0, the algorithm moves at iteration t from S_t to the best solution in a neighborhood $N(S_t)$ of S_t. Solutions that have some attributes of recently visited solutions are forbidden, or tabu, for several iterations to avoid cycling. Kirchler and Calvo (2013) also described a tabu search heuristic for DARP to design a set of least-cost vehicle routes capable of serving all requests. Recent examples of using Tabu search algorithm for solving dynamic VRP can be found in Li et al. (2020) and Arockia et al. (2021).

Gendreau et al. (2006) proposed neighborhood search heuristics to optimize the already planned routes when new requests occur in real-time. The presented numerical results showed the benefits of their procedures for online applications. Coslovich et al. (2006) developed an efficient insertion algorithm, which examined the validity of inserting new requests into the already computed routes. They developed a two-phase insertion algorithm using route perturbations. The first phase creates a feasible neighborhood of the current route. The second phase, which runs in real-time at the time of each new request, inserts the pickup and delivery stop of the new customer in the current route. Parragh and Schmid (2013) presented a solution methodology that integrated variable neighborhood search into a column generation algorithm, which produced high-quality solutions.

Agatz et al. (2011) developed optimization-based approaches that minimize the total system-wide vehicle miles incurred by system users and their travel costs and used travel demand data from metropolitan Atlanta for their simulation. They compared the performance of a greedy heuristic to that of a rolling horizon modeling framework. Berbeglia et al. (2012) introduced a hybrid algorithm that combined an exact constraint programming algorithm and a tabu search heuristic. These two optimization modules are set to run in parallel and continuously optimize the newly arrived requests. The new request is accepted if any of the two modules identify a feasible solution.

Hyland and Mahmassani (2018) compared six different strategies to assign new ride requests to vehicles for single rides. Allowing vehicles to change their ride assignments and considering solving the problem simultaneously for multiple passengers outperforms the simplistic assignment of travelers to the nearest vehicle or to the vehicle that is idle for a longer time.

Several metaheuristics were also proposed for the problem, which provided high-quality solutions. Jorgensen et al. (2007) presented a Genetic Algorithm (GA) for solving the DARP. The algorithm is based on the cluster-first, route-second approach, where it assigns customers to vehicles using a GA and solves separate routing problems for the vehicles. In a recent study, Cheikh et al. (2017) presented a model to optimize taxi carpooling services. Their problem can be viewed as a simplified version of the problem presented in this research. They developed a GA to solve the problem; however, they ignored heterogeneity of customer preferences and assumed a homogenous passenger population in terms of their service option preferences.

Parragh et al. (2010) solved the DARP using a Variable Neighborhood Search (VNS) heuristic. Schilde et al. (2011) studied the problem of transporting patients from home to a hospital or back home from the hospital and modeled the problem as a dynamic stochastic dial-a-ride problem. They proposed different metaheuristic solutions for this problem, including Variable Neighborhood Search (VNS) and stochastic VNS.

2-5. The Vehicle Routing Problem with Transfer

New mobility services have the option to transfer passengers or packages between multiple vehicles. Early research works have focused on applying transfer location in the package delivery system. The transfer allows a request to be served by two vehicles: one vehicle collects them at the pickup location, drops it at the transfer point, and another vehicle carries the load to the delivery location. Several studies have considered the vehicle routing problem with transfer for freight transportation in multi-modal networks.

For example, Mitrovic´-Minic´ and Laporte (2006) were among the first to prove the benefits of adding transfer points in the operation of package delivery companies. For a case study of a company located in San Francisco, they proposed a heuristic solution for freight transportation and showed that by adding the transfer option, the availability and range of the service are expected to increase, and the total miles traveled by all vehicles in the network will reduce. In addition, the transfer option enables serving larger geographic areas and more trips with a limited number of vehicles and reduces total operating costs.

Kerivin et al. (2008) considered the case in which shipments can be transferred from one vehicle to another at any node in the network and developed a branch-and-cut algorithm to solve the problem. The problem ignored the time windows constraint for pickup and delivery.

Qu and Bard (2012) combined Adaptive Large Neighborhood Search (ALNS) and greedy randomized adaptive search procedure (GRASP) to solve a problem with one transfer location. The algorithm is evaluated on instances with up to 25 shipment requests and one transfer point. Masson et al. (2013) proposed an ALNS method, which assumed that transfers could only happen at predefined sets of transfer nodes. The heuristic was applied to the problem instances presented in Mitrovic´-Minic´ and Laporte (2006) and to real-world instances of 33 transfer points and 193 requests.

Fugenschuh (2009) studied the transfer application in the passenger transportation system. They presented an integer programming model to coordinate the public bus services and included the transfer option in a school bus model. Their model assumed a predefined set of bus routes. Hence, their problem is reduced to minimize the number of buses needed to cover this set of predefined routes and find the optimal schedule for buses. They proposed a branch-and-cut algorithm to solve the problem and stated that allowing transfer would decrease the number of buses needed to serve the demand.

Cortes et al. (2010) presented a new formulation of the classical pickup and delivery problem and added the flexibility for passengers to transfer from one vehicle to another at specific locations. They concluded that allowing transfers results in better performance of the system and good quality optimal solutions for most of the cases. They proposed a solution method based on Benders decomposition to solve the problem.

Hou et al. (2012) stated that similar to delivering packages using a one-hop freight system, which usually performs worse than allowing multi-hop systems, carpooling platforms without transfer cannot fully utilize the vehicles' available capacity. They proposed a carpooling paradigm with the option of transfer and the objective to maximize the number of served passengers. Their analysis of this system showed that the proposed system could significantly improve the number of served passengers (by 35% to 60%) compared to the traditional carpooling system without transfer. Besides, they conclude that allowing one transfer improves the performance of the carpooling system, while allowing more than one transfer does not result in more improvement.

In another study, Coltin and Veloso (2014) studied the benefit of transferring passengers between multiple drivers in a ridesharing system. They considered a ridesharing system without time window constraints and proposed a greedy heuristic insertion algorithm to solve the problem and schedule rideshare routes with transfers.

They conclude that by allowing transfers, the availability and coverage range of the ridesharing service will be increased. In addition, they showed that in the case of ridesharing with the transfer, the total mileage traveled by vehicles will be reduced by nearly 30%.

Masson et al. (2014) proposed three mathematical models for integrating TNC services into shuttle services. The problem requires the transportation of passengers from a large set of pickup locations to a limited set of delivery locations. The branch-and-cut technique is used to develop an efficient solution methodology for the problem.

Nam et al. (2018) developed a simulation platform that included ridesharing, transit, bike-sharing, and walking. This research included a case study of the operation of the multi-modal system that includes ridesharing participants, the Los Angeles Metro Redline subway rail, and the Los Angeles downtown bike-share system. The results indicate that a multi-modal network with the option of transfer between ridesharing and transit services expands public transit coverage and that ride- and bike-sharing could feed transit systems effectively when properly designed and integrated into the transit system.

Nonetheless, the enabling of the transfer option adds to the complexity of the problem. Thus, most existing solution methodologies adopt heuristics and metaheuristics approaches to solve the problem. Herbawi and Weber (2011) considered a multi-hop ridesharing problem where drivers have fixed routes and schedules. In this system, the ridesharing system with a fixed route and schedule act as a feeder to the public transit system. They solved this problem using the genetic algorithm to assign passengers to a route that minimizes costs, time, and number of transfers. Their experiments show that the multi-hop ride-matching system increases the number of matched requests while increasing the total travel time for the system's users. The problem ignored heterogeneity of the passengers' willingness to rideshare and/or transfer. Furthermore, the methodology's performance is presented using small networks compared to the one presented in this research.

Masoud and Jayakrishnan (2017) developed a decomposition algorithm to solve the multi-hop ride-matching problem. They provided a heuristic insertion algorithm that adopts a pre-processing procedure to reduce the size of the input sets and decomposes the problem into smaller problems that are iteratively solved.

Andini et al. (2019) considered a ridesharing system with fixed transfer locations and used an insertion heuristic method to obtain a solution for the online version of the problem. The objective of the problem was to minimize the operational cost of the TNC by reducing the travel distance of each vehicle. Based on the result of the simulation experiment, their proposed system can serve more requests, up to 16.7%, compared to the conventional ridesharing systems.

Ma et al. (2019) proposed a ridesharing system with integrated transit in which a private TNC provider may drop off passengers at the transit station or pick up passengers from a transit station. They created a discrete event simulation scheme to solve the problem and conducted several experiments to test the effectiveness of this integrated system, the influence of different model parameters, and measure the benefit of such cooperation. Their results suggest that vehicle travel time can reduce by 40–60% when the demand is high. A case study of Long Island commuters to New York City (NYC) suggests that this proposed operating strategy can reduce passengers’ travel times and travel costs by up to 54% and 60%, respectively.

Singh et al. (2019) proposed a multi-hop ridesharing algorithm with the option of transfer and developed a deep reinforcement learning algorithm to find optimal vehicle dispatch and matching decisions. Their experiment showed that by allowing customers to transfer between vehicles, travel costs would decrease by 30%, and utilization of fleets would increase by 20% compared to the conventional ridesharing algorithms.

In a recent study, Lu et al. (2021) proposed a vehicle-space-time network model for optimizing the integration of a ridesharing system in short-notice evacuations and solved the problem using the Lagrangian relaxation approach. Their proposed method is suitable for offline applications. Their analysis showed that transfer option could improve the capacity of evacuation corridors, reduce traffic congestion, and alleviate the fuel shortage problem.

2-6. Summary

While the existing literature captures essential aspects of ODMS operation, the developed methodologies lack essential capabilities that preclude their adoption in real-world applications. For example, existing models cannot represent both ridesharing and rides with transfer in one comprehensive model. As mentioned above, ODMS is rapidly growing with more drivers offering their services and more customers getting familiar with the service. As the system grows and matures technically (developing user-friendly applications with more capabilities) and behaviorally (drivers and customers are more familiar with the service), one expects more services to be introduced to maximize service coverage and reduce the cost for customers. Similar to introducing shared-ride services, one can anticipate transfer options to be included as part of the offered services.

In addition, existing models ignored the heterogeneity of passengers' preferences, especially with respect to their willingness to rideshare and/or transfer, which affects the fidelity of these models in representing the demand-supply interactions.

Furthermore, unlike the traditional DARP, where the entire demand must be served, ODMS providers can decline ride requests if they are not profitable. Examining the profitability of ODMS requires a modeling framework that can identify profitable trips based on the temporal-spatial distributions of passengers and vehicles in the network. Finally, despite the previous effort to develop efficient solution methodologies, they are still short of modeling ODMS in networks of moderate sizes.

Chapter 3

PROBLEM DEFINITION AND FORMULATION

3-1. Introduction

As mentioned earlier, this research is motivated by the need to develop a modeling framework for the planning and operation of ODMS with transfer option and time window constraints. This chapter presents a formal definition of the problem in the form of a mathematical program formulation. In this mathematical program, the objective function maximizes the total ODMS profit, which is a function of total revenue collected from all served passengers considering discounts offered to passengers who rideshare or transfer and the operation cost of the vehicles. A set of constraints is presented to satisfy passengers' time window constraints and rideshare and/or transfer preferences and ensure path continuity for vehicles and passengers. This chapter is organized as follows. Section 3-2 describes the problem and presents the list of variables and other notations used to formulate it. Section 3-3 presents the mathematical formulation of the problem. Finally, section 3-4 gives a summary of the chapter.

3-2. Problem Definition

The following notation is used to describe data sets, model parameters, and decision variables to develop a modeling framework for ODMS operation. This notation builds on the one given in Hosni et al. (2014) to facilitate model cross-reference.

Notation:

G: Directed graph

N: Set of nodes

A: Set of links

V: Set of all vehicles ($|V| = m$)

v: Vehicle index ($v \in V$)

P: Set of all passengers ($O \cup S$)

O: Set of all onboard passengers

S: Set of all ride seeking passengers

p: Passenger index ($p \in P$)

Q^v: The capacity of vehicle v

R_p: Revenue of serving passenger p

T_p: Request time of passenger p

e_p: Earliest pickup time of passenger p from the origin

l_p: Latest drop-off time of passenger p at destination

D_p: Maximum trip duration for each passenger p

r_p: = 1 if passenger p is willing to rideshare, and 0 otherwise

t_p: = 1 if passenger p is willing to transfer, and 0 otherwise

ω_p: A weight for each passenger which is equal to Q^v if $r_p = 0$ and 1 if $r_p = 1$

(i, j): Index of the link between adjacent nodes i and j

T_{ij}: Travel time from node i to j

C_{ij}: Travel cost from node i to j

m: Total number of vehicles

N_1: The subset of nodes associated with the drop-off nodes of onboard passengers

N_2: The subset of nodes associated with the pickup nodes of ride seeking passengers

N_3: The subset of nodes associated with the drop-off nodes of ride seeking passengers

N_4: The subset of nodes where the transfer is prohibited

$N(v)$: Current location of vehicle v

$v(p)$: The vehicle that is currently transporting onboard passenger p

$p(p)$: Pickup location of passenger p

$d(p)$: Drop-off location of passenger p

α: Fare discount rate associated with ridesharing in any part of the trip

β: Fare discount associated with each transfer along the trip

M: Very large number

Decision variables:

X_{ij}^{pv}: = 1 if passenger p traverses link (i, j) on vehicle v, and 0 otherwise

Y_{ij}^{v}: = 1 if vehicle v traverses link (i, j), and 0 otherwise

d_{pv}: = 1 if passenger p is picked up by vehicle v, and 0 otherwise

n_i^p: = 1 if node i is used as a transfer node for passenger p, and 0 otherwise

a_p: Number of transfers along the route of passenger p

u_{vi}^a: Time at which vehicle v arrives at node i

u_{vi}^d: Time at which vehicle v departs from node i

z_{pi}^a: Time at which passenger p arrives at node i

z_{pi}^d: Time at which passenger p departs from node i

Here, we consider a transportation network of a typical metropolitan area which is represented by a directed graph $G(N, A)$, where N is the set of nodes and A is the set of links. The time-varying travel time T_{ij} and travel cost C_{ij} are specified for each link (i, j), which could capture the congestion pattern in the network. Consider an ODMS provider operating a set of vehicles V in this metropolitan area. Each vehicle $v \in V$ is defined in terms of its maximum capacity Q^v, starting location, and number of onboard passengers at the start of the horizon of interest.

These vehicles serve a set of passengers P with heterogeneous characteristics consisting of the two groups O and S, where O is the set of all onboard passengers at the start of the operation horizon, and S is the set of all ride seekers during this horizon. Passengers are assumed to differ in terms of their characteristics. Each passenger $p \in P$ requests a ride at time T_p and is associated with a set of attributes including pickup location $p(p)$ and drop-off location $d(p)$, the earliest time for pickup e_p and latest time for drop-off l_p, and maximum trip duration D_p. In addition, two binary parameters, r_p and t_p, are defined for each passenger to specify her/his willingness to rideshare and transfer, respectively.

Each served passenger $p \in P$ is assumed to yield a revenue R_p. In this formulation, this revenue is assumed as a function of the shortest distance between the passenger's pickup and drop-off locations. Of course, other revenue accounting methods can be used as adopted by the ODMS provider under consideration. Passengers can transfer between vehicles at a predetermined set of nodes in the network. The variable a_p tracks the number of transfers made by each passenger. We assume that each passenger receives discount percentages of α and β for ridesharing and transferring, respectively.

To track the itinerary of passenger p, we use $v(p)$ to denote the vehicle(s) transporting this passenger, z_{pi}^{a} to denote her/his arrival time at node i, and z_{pi}^{d} to denote the passenger's departure time from node i. Similarly, to track each vehicle, we use u_{vi}^{a} and u_{vi}^{d} to denote the arrival and departure times of vehicle v at node i, respectively. Here, without loss of generality, considering that both u_{vi}^{a} and u_{vi}^{d} are formulated as scalar variables, no vehicle is assumed to visit any of the nodes along its route more than one time. This assumption prevents route circuity, which is negatively perceived by passengers using ridesharing services.

The problem is to assign vehicles to ride seekers and to determine the optimal route for each vehicle such that the total profit of the ODMS is maximized. Each route starts from the vehicle's current location assigned to that route. Following the starting node, the route includes a sequence of nodes representing the pickup and drop-off locations of the passengers served by this vehicle. The problem entails satisfying all customer constraints related to pickup and drop-off time windows, maximum trip time, and rideshare and/or transfer preferences. The proposed model maximizes the total network profit, which is computed as the total revenue of all served passengers. Ride seekers that are determined to be unprofitable or causing problem infeasibility, receive no service.

Two flavors of this problem are considered in this dissertation. The first problem pertains to solving one operation stage of a predefined horizon (e.g., one to two hours) with predefined demand. In other words, the problem assumes that the demand for ride requests is known a priori for the entire horizon; that is the time $T_p \; \forall \, p \in P$ at which passengers requested the service is always before the time at which the problem is solved.

As needed, the problem can be solved at any point in time (e.g., the arrival of new requests).

The problem also entails determining an efficient route for each vehicle to serve all assigned passengers while satisfying their time window constraints and their preferences with respect to ridesharing and transfer. This version of the problem is more suitable in the situation where TNCs provide riders the options to schedule their trips ahead of time. The problem is also encountered in operation planning applications, in which TNCs examine the system performance under different possible operational conditions (e.g., demand levels, number of vehicles, pricing schemes, etc.) derived from historical data.

The second type is the online version of the first problem. It entails matching each received ride request to one of the vehicles in the network in real-time (i.e., upon the service is requested). If no single vehicle can be found to offer the service, the ride request is matched to two vehicles considering a transfer. Once a passenger requests a ride, this passenger is expected to promptly receive a confirmation message from the TNC operator, typically through a mobile application, on the service availability. The confirmation message sent to a passenger includes the vehicle identification information, expected pickup time, and the route to the destination, including information on the transfer location, if any (i.e., trip itinerary). If the generated itinerary does not satisfy the passenger's trip constraints and preferences, the passenger is declined the service, and she/he is free to request the service from another TNC. We assume that the TNC operator can communicate with the passenger later and check if she/he is still seeking a ride.

3-3. Mathematical Formulation

This problem is formulated in the form of a mixed-integer program (MIP-1) as follows.

MIP-1:

$$\textbf{Max } \sum_{p\in P} R_p\left(1-\alpha.r_p\right)\sum_{v\in V} d_{pv} - \sum_{p\in P} R_p.(1+\beta).a_p\,.t_p - \sum_{i,j\in N}\sum_{v\in V} C_{ij}.Y_{ij}^{v} \quad (1)$$

$$\sum_{j\in N} X_{N(v(p))\,j}^{p\,v(p)} = 1 \qquad \forall\, p \in O \quad (2)$$

$$\sum_{j\in N} X_{j\,d(p)}^{pv} = 1 \qquad \forall\, p \in O, \forall\, v \in V \quad (3)$$

$$\sum_{j\in N} X_{j\,p(p)}^{p\,v} = \sum_{j\in N} X_{p(p)j}^{p\,v} - (\sum_{v\in V} d_{pv} - a_p\,.t_p) \qquad \forall\, p \in S, \forall\, v \in V \quad (4)$$

$$\sum_{j\in N} X_{j\,d(p)}^{p\,v} = \sum_{j\in N} X_{d(p)j}^{p\,v} + (\sum_{v\in V} d_{pv} - a_p\,.t_p) \qquad \forall\, p \in S, \forall\, v \in V \quad (5)$$

$$\sum_{v\in V}\sum_{j\in N} X_{j\,i}^{p\,v} = \sum_{v\in V}\sum_{j\in N} X_{ij}^{p\,v} \qquad \forall\, p \in P, \forall\, i \in N - \{N(v), p(p), d(p)\} \quad (6)$$

$$\sum_{p\in P} \omega_p X_{ij}^{p\,v} \leq Q^v\, Y_{i\,j}^{v} \qquad \forall\, v \in V, \forall\, i \in N, \forall\, j \in N \quad (7)$$

$$X_{ij}^{p\,v} \leq d_{p\,v} \qquad \forall\, p \in P, \forall\, v \in V, \forall\, (i,j) \in A \quad (8)$$

$$\sum_{v\in V} d_{pv} \leq a_p\,.t_p + 1 \qquad \forall\, p \in P \quad (9)$$

$$\sum_{j\in N} Y_{i\,j}^{v} \leq 1 \qquad \forall\, v \in V, \forall\, i \in N \quad (10)$$

$$\sum_{j\in N} Y_{j\,i}^{v} \leq 1 \qquad \forall\, v \in V, \forall\, i \in N \quad (11)$$

$$\sum_{j\in N} Y_{i\,j}^{v} \leq \sum_{j\in N} Y_{j\,i}^{v} \qquad \forall\, v \in V, \forall\, i \in \{(N_1, (N_2 - N(v)), N_3\} \quad (12)$$

$$\sum_{j\in N} Y_{i\,j}^{v} \geq \sum_{j\in N} Y_{j\,i}^{v} \qquad \forall\, v \in V, \forall\, i \in \{(N_2 \cap N(v))\} \quad (13)$$

$$\sum_{j\in N} Y_{ij}^{v} = \sum_{j\in N} Y_{ji}^{v} \qquad \forall\, v \in V, \forall\, i \in N - \{N_1, N_2, N_3, N(v)\} \qquad (14)$$

$$\sum_{j\in N} Y_{N(v)\,j}^{v} \geq d_{p\,v} \qquad \forall\, v \in V, \forall\, p \in P \qquad (15)$$

$$u_{vj}^{a} - u_{vi}^{d} \geq T_{i\,j} - M\left(1 - Y_{i\,j}^{v}\right) \qquad \forall\, v \in V, \forall\, i \in N, \forall\, j \in N \qquad (16)$$

$$u_{vp(p)}^{d} \geq e_p - M(1 - d_{p\,v}) \qquad \forall\, p \in S, \forall v \in V \qquad (17)$$

$$u_{vd(p)}^{a} \leq l_p + M(1 - d_{p\,v}) \qquad \forall\, p \in S, \forall v \in V \qquad (18)$$

$$u_{v\,d(p)} - u_{v\,p(p)} \leq D_p + M\left(1 - d_{p\,v}\right) \qquad \forall\, p \in P, \forall\, v \in V \qquad (19)$$

$$u_{vi}^{d} \geq u_{vi}^{a} \qquad \forall\, v \in V, \forall\, i \in N \qquad (20)$$

$$u_{vN(v)}^{a} = Start\ time \qquad \forall\, v \in V \qquad (21)$$

$$z_{pi}^{a} \leq u_{vi}^{d} + M\left(1 - \sum_{j\in N} X_{i\,j}^{p\,v}\right) \qquad \forall\, p \in P, \forall\, v \in V, \forall\, i \in N - \{p(p)\} \qquad (22)$$

$$z_{pi}^{d} \geq u_{vi}^{d} - M\left(1 - \sum_{j\in N} X_{i\,j}^{p\,v}\right) \qquad \forall\, p \in P, \forall\, v \in V, \forall\, i \in N \qquad (23)$$

$$z_{pj}^{a} - z_{pi}^{d} \geq T_{i\,j} - M\left(1 - \sum_{j\in N} X_{i\,j}^{p\,v}\right) \qquad \forall\, p \in P, \forall\, v \in V, \forall\, i \in N \qquad (24)$$

$$z_{pp(p)}^{a} = e_p \qquad \forall\, p \in S \qquad (25)$$

$$z_{pN(v)}^{a} = Start\ time \qquad \forall\, p \in O \qquad (26)$$

$$z_{pi}^{d} \geq z_{pi}^{a} \qquad \forall\, p \in P, \forall\, i \in N \qquad (27)$$

$$z_{pd(p)}^{a} \leq u_{vd(p)}^{a} - M\left(1 - \sum_{j\in N} X_{d(p)\,j}^{p\,v}\right) \qquad \forall\, p \in P, \forall\, v \in V \qquad (28)$$

$$\sum_{i\in N} n_i^{p} = a_p \,.\, t_p \qquad \forall\, p \in P \qquad (29)$$

$$\sum_{p\in P} n_i^{p} = 0 \qquad \forall\, i \in N_4 \qquad (30)$$

The objective function given in (1) maximizes the total ODMS profit, which is computed as the total revenue collected from all served passengers minus the operation cost of the vehicles. The objective function accounts for discounts offered to passengers who rideshare or transfer. Different mechanisms can be used to account for these discounts. For instance, they could be granted only if the scheduled ride requires ridesharing or transferring. On the other hand, they could be granted as long as the customer indicates her/his willingness to rideshare or transfer, even if the scheduled ride does not include ridesharing or transferring. In this formulation, a ridesharing discount is assumed to be granted to all passengers as long as they indicate their willingness to rideshare, while transfer discount is granted only for passengers who make a transfer along with their trips. In this version of the trip, the ridesharing discount is given for each leg in the transfer trip. Also, the formulation allows an unlimited number of transfers for each passenger. However, we limit the solution to one transfer.

While in this version of the problem we adopt an objective function that focuses on maximizing the profit, the framework is flexible to consider other objective functions that explicitly incorporate passenger-oriented performance measures such as passengers' waiting time. However, one should note that a profit-oriented objective function implicitly reduces the passengers' waiting time. Maximizing the profit requires maximizing the utilization of the vehicles by serving as many passengers as possible. Therefore, the vehicle routes push to pick up passengers as soon as their pickup time windows open, ultimately reducing their waiting times.

Constraints (2) and (3) ensure that all onboard passengers continue their trips to reach their destinations. Constraints (2) ensure that these passengers leave their current location, while constraints (3) ensure that onboard passengers arrive at their destinations. Similarly, constraints (4) and (5) ensure those ride seekers are picked up from their pickup locations and dropped off at their destinations, respectively. Constraints (6) are for the conservation of flow. Constraints (7) ensure that the number of passengers assigned to a vehicle is less than the vehicle's capacity. The weight ω_p ensures that passengers who are not willing to rideshare get no other passengers in their vehicles.

Constraints (8) relate the different vehicle-passenger matching variables used in the formulation. Constraints (9) map the number of transfers to the number of vehicles used to serve each passenger. Passengers who are willing to transfer ($t_p = 1$) could be served by more than one vehicle. Constraints (10) ensure that a vehicle cannot simultaneously leave its location to two different destination nodes from a specific source node. Similarly, Constraints (11) ensure that a vehicle cannot reach a destination node from two different source nodes.

Constraints (12) and (13) ensure the path continuity for the vehicle from the passenger's pickup and to the drop-off location, respectively. Constraints (14) ensure path continuity for vehicles along with all other nodes. Constraints (15) ensure that the vehicle moves from its location if a passenger is assigned to it. Constraints (16) define the vehicle arrival time at a downstream node in terms of its departure time from its current link's upstream node and the travel time on that link. Constraints (17) and (18) ensure that passengers can only be picked up after the specified earliest pickup time and dropped off before the specified latest drop-off time, respectively.

Constraints (19) ensure that the passenger's time on the road is less than the maximum ride time. Constraints (20) ensure that the departure time from a node is greater than the arrival time at this node for all vehicles. Constraints (21) define the arrival time of a vehicle at its current location at the beginning of the planning horizon. Constraints (22) and (23) map a passenger's arrival and departure times at a node to those of the vehicle serving this passenger. Constraints (24) are similar to constraints (16); they define the arrival time for passengers at each node along her/his route. Constraints (25) define the time at which ride seekers are ready for pickup, while constraints (26) define the start time of onboard passengers.

Constraints (27) ensure that a passenger's departure time from a node is greater than her/his arrival time at that node. Constraints (28) define the arrival time of each passenger at her/his destination. Constraints (29) define the total number of transfers along each passenger's trip. Finally, constraints (30) prevent the transfer from occurring at any node where the transfer is prohibited (e.g., for safety considerations).

3-4. Summary

This chapter presented a formal definition of the ODMS with transfer option and time window constraints. The problem is formulated in the form of a mixed-integer mathematical program. In this mathematical program, the objective function maximizes the total profit of passengers. The profit is calculated as the difference between (a) the sum of the fares collected from passengers considering discounts for ridesharing and transfer options and (b) the operation cost of the service. A set of constraints is presented to satisfy passengers' time windows, rideshare, and/or transfer preferences and ensure path continuity for vehicles and passengers.

Chapter 4

SOLUTION METHODOLOGY FOR OFFLINE PROBLEMS WITH PREDEFINED DEMAND

4-1. Introduction

This chapter presents the solution methodology for solving the offline version of the ODMS with ridesharing/transfer options and time window constraints. As mentioned above, this problem assumes that the passengers' demand is known a priori for the entire horizon of interest. The developed solution methodology is a modified version of the column generation proposed to solve the classical vehicle routing problem to consider the transfer option. The methodology integrates network decomposition and augmentation approaches to allow solving problems of reasonable sizes. This chapter is organized as follows. Section 4-2 describes the general idea of modified column generation. Section 4-3 and 4-4 present the mathematical formulation of the master problem and sub-problems, respectively. Section 4-5 explains the network augmentation method used to solve the sub-problems. Finally, section 4-6 gives a summary of the chapter.

4-2. An Overview of the Solution Methodology

The solution methodology adopts a modified version of the column generation (CG) technique to account for rides with the transfer. CG is one of the most commonly used methods to solve the vehicle routing problem with time windows constraints (Desaulniers et al., 2006). This method implements a decomposition approach leading to smaller sub-problems that can be solved separately. In most implementations, the constraints that ensure each passenger is assigned to at most one vehicle are relaxed leading to m single-vehicle problems that are solved separately (Desaulniers et al., 2006). The technique iterates to determine the upper and lower bounds of the problem until these bounds coincide and yield the optimal solution. For example, in a maximization problem, the optimal solution of the relaxed problem presents an upper bound on the primary problem. Solving the master problem at each iteration finds a new cut for the relaxed problem. The relaxed problem is solved using this new cut to provide a new lower bound for the problem. The algorithm ends when the upper and lower bounds coincide, or the difference between these bounds is within a pre-specified threshold.

Most CG implementations for the vehicle routing problem represent the sub-problems in the form of a constrained shortest path problem. Thus, at each iteration, the master problem finds new multipliers, which are used by sub-problems to find new routes (columns) to send back to the master problem. Nonetheless, this approach is suitable only for problems in which the transfer option is not considered.

Figure 4-1 shows the general idea of the application of column generation to solve the conventional vehicle routing problem. At each iteration of solving the master problem, a set of primal optimal solutions and a set of dual solutions are obtained. More details about these primal and dual solutions are given in section 4-3. The dual solutions will be introduced to sub-problems to generate a new set of routes (columns) in the next iteration. These new columns will be given to the master problem, and the master problem picks the set of most profitable routes such that each customer is served in at most one route (infeasibility elimination).

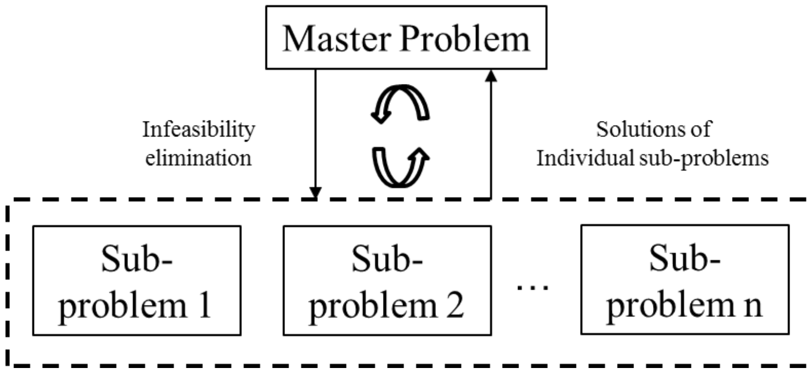
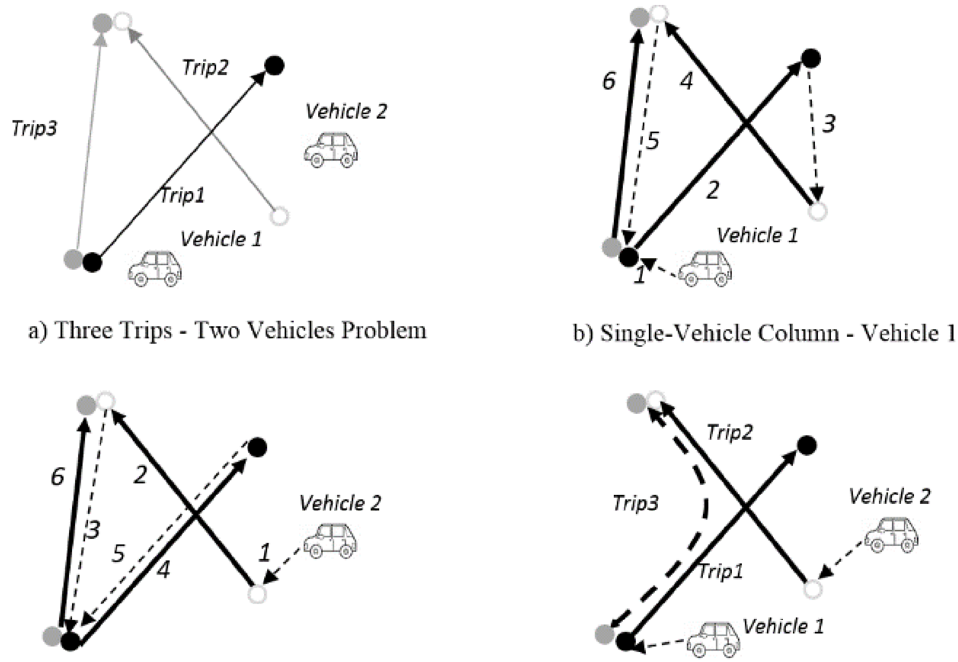


Figure 4-1: Application of column generation method in the conventional vehicle routing problem

Accounting for the transfer option adds to the complexity of the problem as the search for the optimal solution must examine vehicle combinations that can jointly offer a service to ride seekers who are willing to transfer. It means that if the original problem is divided into single-vehicle sub-problems, no solution can be found in which a passenger is served with two different vehicles via a transfer point in the middle of the route. As such, the CG methodology needs to be modified to allow the sub-problems to generate: a) columns that define the routes of single vehicles serving passengers with no transfer and b) hybrid columns that define the itineraries of multiple vehicles serving transferring passengers.

Figure 4-2 demonstrates the structure of the columns generated for a hypothetical problem of three trips and two vehicles. In this problem, only the ride seeker of Trip 3 is willing to transfer. Figures 4-2b and 4-1c illustrate the structure of the columns generated for Vehicle 1 and Vehicle 2, respectively. The figures show the route used by each vehicle to serve the three trips. For instance, Vehicle 1 serves Trip 1, Trip 2, and Trip 3. Vehicle 2 serves Trip 2, Trip 1, and then Trip 3. Figure 4-2d provides an example of a hybrid column for the routes of Vehicle 1 and Vehicle 2. In this hybrid column, Vehicle 1 serves Trip 1, and Vehicle 2 serves Trip 2. In addition, Trip 3 is served via a transfer between the two vehicles. This solution is shared with the master problem in the form of three columns: I) a column for Vehicle 1; II) a column for Vehicle 2, and III) a column for a virtual vehicle that is created to represent the case in which Vehicle 1 and Vehicle 2 jointly serve Trip 3. The actual vehicles (Vehicle 1 and Vehicle 2) serving Trip 3 are mapped to this virtual vehicle. While no cost is assigned to this virtual vehicle, the fare collected from Trip 3 is considered the virtual vehicle's revenue.

The overall framework of the solution methodology is described in Figure 4-3. As shown in the figure, the framework consists of two main loops. The outer loop iterates between a master problem and sub-problems following the structure of the conventional CG algorithm. More details on these problems are given in sections 4-4 and 4-5. An inner loop is set up for each sub-problem to determine the optimal route for each vehicle or a combination of vehicles that provide a transfer option.



a) Three Trips - Two Vehicles Problem

b) Single-Vehicle Column - Vehicle 1

c) Single-Vehicle Column - Vehicle 2

d) Hybrid Column – Vehicles 1 & 2

Figure 4-2: Columns generated for a hypothetical problem of two-vehicle and three passengers considering the transfer option

Considering the example in Figure 4-2, three sub-problems are solved. In the first sub-problem, an optimal route is constructed for Vehicle 1, aiming to serve all three passengers. In the second sub-problem, a route for Vehicle 2 aiming to serve all three passengers is constructed. The third sub-problem includes both Vehicle 1 and Vehicle 2 and the three passengers. The first and second sub-problems generate two columns for Vehicle 1 and Vehicle 2, respectively, as shown in Figures 4-2b and 4-1c. The third sub-problem generates three columns: one route for Vehicle 1, one route for Vehicle 2, and one route for a virtual vehicle. The columns generated for Vehicle 1 and Vehicle 2 in the third sub-problem could be different from those generated in the first and the second sub-problems. The route of the virtual vehicle matches the route of the transferring passenger who is served by transferring between Vehicle 1 and Vehicle 2.

The methodology starts by generating an initial feasible route for each vehicle. In the simplest implementation, the initial route for the vehicle is generated such that it serves one passenger. Given the initial set of feasible routes for all vehicles, the master problem described in section 4-3 is solved. The master problem picks the set of most profitable routes such that each customer is served in one route only. The duals of the master problem are made available to the sub-problems.

Unlike the conventional CG algorithm, the proposed methodology allows sub-problems to include multiple vehicles to serve transfer trips. Thus, a module to set up these sub-problems is activated. This module implements a set of feasibility checks and profitability rules, as described in more detail in section 4-4, to determine vehicle and customer combinations considered in each sub-problem. This module prevents setting up sub-problems that are either infeasible or expected to generate columns with low profits. We assume ride seekers are unlikely to make more than one transfer in the current implementation. Thus, the number of vehicles considered in each sub-problem is limited to a maximum of two vehicles.

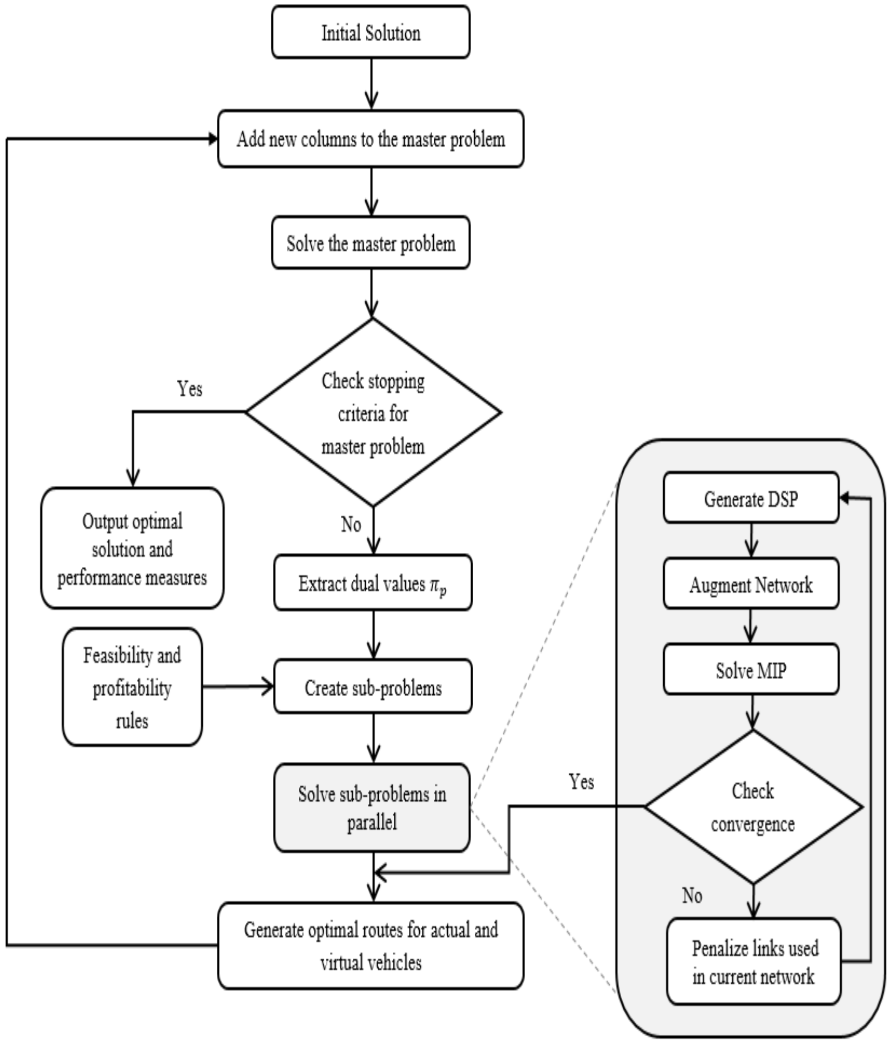


Figure 4-3: Overall modeling framework

Each sub-problem is solved iteratively. The smallest possible sub-network is extracted from the original roadway network in the first iteration. This sub-network includes links most likely used to route the vehicles while serving the customers specified in this sub-problem. A K-Distinct Shortest Path Algorithm (KDSPA) is used to determine this set of links. More details on selecting these links are given below.

Given the reduced-size sub-problems in terms of the number of vehicles, the number of customers, and the size of the routing network, the MIP-1 described above is used to solve these problems considering the dual values obtained from the master problem. Following the inner loop iterations for each sub-problem, the size of the sub-network is incrementally increased by adding more links using the KDSPA.

Each sub-problem is again solved using the new augmented sub-network. If the sub-problem solution stabilizes in two successive iterations, the inner loop terminates, and the generated routes for each actual and virtual vehicle are saved to be sent back to the master problem. The master problem is again solved using the newly generated columns from the sub-problems. New dual values are generated and shared with the sub-problems. The outer loop terminates when the gap in the objective function for two successive iterations is less than a predefined threshold.

Algorithm: Modified Column Generation
Input: Network topology, set of passengers P, and set of vehicles V
begin
Initialization
$i_m = 1 \leftarrow$ Iteration number of the master problem
$i_s = 1 \leftarrow$ Iteration number of sub-problem s
$e_m = +\infty \leftarrow$ error for master problem
$e_s = +\infty \leftarrow$ error for sub-problem s
$\xi_m \leftarrow$ error threshold for the master problem objective function
$\xi_s \leftarrow$ error threshold for the sub-problem objective function
$\pi_p \leftarrow$ Dual variable for passenger p
$S \leftarrow$ Number of sub-problems
$R_p \leftarrow$ Revenues associated with serving passenger p
${O_m}^{i_m} \leftarrow$ Objective value of the master problem in the iteration i_m
${O_s}^{i_s} \leftarrow$ Objective value of sub-problem s in the iteration i_s
Setup master problem and sub-problems
//original problem is decomposed to a master problem and sub-problems of single and multi-vehicle considering feasibility and profitability rules. The initial set of routes (columns) is obtained.
while $e_m > \xi$ ***do***
Add columns to master problem
Solve the master problem and save ${O_m}^{i_m}$
$e_m \leftarrow \dfrac{{O_m}^{i_m} - {O_m}^{i_{m-1}}}{{O_m}^{i_{m-1}}}$
Save dual variables $\pi_p \ \forall \ p$
    for $s \coloneqq 1$ to S, ***do***
        Update the sub-problem objective function
        //Update of sub-problem's objective functions using ${R_p}^{i_m}$
        ${R_p}^{i_m} \leftarrow R_p - \pi_p \ \forall \ p$
        while $e_s > \xi_s$ ***do***
            Generate DSP and augment network
            Solve MIP-1 and save ${O_s}^{i_s}$
            $e_s \leftarrow \frac{{O_s}^{i_s} - {O_s}^{i_{s-1}}}{{O_s}^{i_{s-1}}}$
            Penalize links in the augmented network
            $i_s + 1$
        end
    Generate optimal routes for actual and virtual vehicles
    end
Save all new columns
$i_m + 1$
end
Solve the integer master problem
Output: Final set of vehicle routes and a list of served passengers

Figure 4-4: The modified column generation algorithm

4-3. The Master Problem

The master problem is formulated as a set partitioning problem represented in the form of an integer mathematical program as presented in (31)-(34). Given the set of routes generated for all actual and virtual vehicles, the master problem determines the most optimal route for each vehicle such that each passenger is served by at most one vehicle. The parameter a_{pr}^{v} defines which route-vehicle combinations serve passenger p, based on the solution of the sub-problems. Constraints (32) and (33) assign each passenger and vehicle to at most one route, respectively. Constraint (34) ensures that the route of a virtual vehicle is selected as part of the optimal solution only if the routes of two actual vehicles associated with this virtual vehicle are also selected.

$$\textbf{Max} \sum_{r \in R^v} \sum_{v \in V} (RVN_r^v - CST_r^v) \cdot Z_r^v \tag{31}$$

$$\sum_{r \in R^v} \sum_{v \in V} a_{pr}^v \cdot Z_r^v \leq 1 \qquad \forall\, p \in P \tag{32}$$

$$\sum_{r \in R^v} Z_r^v \leq 1 \qquad \forall\, v \in V \tag{33}$$

$$(n(v') + 1) \cdot Z_r^{v'} = \sum_{i \in n(v')} Z_r^{veh_i(v')} \qquad \forall\, v' \in V', \forall\, r \in R^v \tag{34}$$

where,

V: Set of all vehicles (actual and virtual)

V': Set of virtual vehicles $V' \subset V$

v: Index of vehicles

v': Index of virtual vehicles such that $v' \in V'$

$veh_i(v')$: Index of the i^{th} actual vehicle serving the transfer trip of virtual vehicle v'

$n(v')$: Number of transfers of passenger that is served with actual vehicles (here equal to zero or 1)

R^v: The set of routes generated for vehicle v

r: Index for routes in R^v

RVN_r^v: The revenue of route r served by vehicle v

CST_r^v: The cost of route r served by vehicle v

a_{pr}^v: = 1 if passenger p is served by vehicle v traveling route r, and 0 otherwise

Z_r^v: = 1 if vehicle v is assigned to route r, and 0 otherwise

The solution of the linear relaxation of the master problem includes the primal optimal solution Z_r^v and the dual solution $\pi = (\pi_1, \pi_2, \pi_p, \dots, \pi_{|P|} \mid \forall\, p \in P)$ for the current iteration. Where π includes the values of the dual variables of constraints. These dual values are shared with the sub-problems to determine a new set of columns. At convergence, the optimal integer solution is obtained for the master problem.

4-4. Setting-up the Lower Level Problems

As stated earlier, the CG technique relaxes the problem by removing the constraints that prevent the assignment of each passenger to more than one vehicle. For services with no transfer option, this relaxation allows solving the problem for every single vehicle separately using a constrained shortest path algorithm (Desaulniers et al., 2006). Nonetheless, with the integration of the transfer option, this approach will not be proper to solve the sub-problems.

Thus, we use the mathematical program MIP-1 to solve each sub-problem. Each sub-problem is defined in terms of a maximum of two vehicles, a set of customers that can be efficiently served by the specified vehicle(s), and a reduced-size network that defines links that are most likely used to route these vehicles. As explained above, two types of sub-problems are considered. In the first type, a sub-problem is set for every single vehicle. In the second type, two vehicles are combined to examine if they can jointly offer service to customers willing to transfer.

Several rules are considered to determine efficient vehicle combinations. For example, as shown in Figure 4-5a, two vehicles are included in one sub-problem only if the distance between their current locations is less than a certain threshold. This distance is typically double the maximum distance a driver is willing to drive to arrive at a customer's location (e.g., 10 to 15 miles). If the shortest travel distance between two vehicles is greater than the defined threshold (e.g., 30 miles in large networks), there is no possibility that the routes of these two vehicles intersect to serve one transferring passenger.

An additional set of feasibility checks and profitability rules are used to determine the subset of customers that can be included in the sub-problem of each vehicle or a combination of vehicles. The first rule excludes passengers if the vehicle(s) violates their time window constraints. The shortest path algorithm is used to determine the fastest route to serve a passenger by each vehicle allocated to a sub-problem. As shown in Figure 4-5b, this route consists of the segment from the vehicle’s current location to the passenger’s origin and the segment from the passenger’s origin to her/his destination. If none of the vehicles arrives at the passenger's destination before her/his latest drop-off time, this passenger is excluded from this sub-problem.

The second rule considers the profitability of a trip. This rule checks the travel distance from the vehicle's current location to the passenger's origin. Because drivers usually prefer to pick up passengers in their vicinity, a passenger is excluded if the travel distance to the location of this passenger exceeds a predefined limit (e.g., 10 to 15 miles). Finally, other rules related to the behavior of passengers can be incorporated. For example, for safety concerns, a passenger possibly will agree to share the ride only if other passengers are diverse in terms of gender and/or ethnicity. Consequently, if such a condition is not satisfied, this passenger is excluded from this sub-problem.

As such, two types of sub-problems are created. Sub-problems that determine the optimal route of a single vehicle, and sub-problems that determine the routes of two vehicles. If the solution of the two-vehicle sub-problem results in serving any passengers via transfer, the routes of the two vehicles and the route of a virtual vehicle are created. The route of the virtual vehicle represents the route of the transferring passenger.

These sub-problems are solved using a modified version of MIP-1. Similar to the objective function presented for MIP-1, the objective function given in (35) maximizes the total ODMS profit, which is computed as the total revenue collected from all served passengers minus the operation cost of the vehicle (for the single-vehicle sub-problem) or the operation cost of the two vehicles (for the two vehicles sub-problems). The objective function accounts for discounts offered to passengers who rideshare or transfer. However, as presented in (35), the objective function is modified to incorporate the values of the dual variables π_p obtained from the master problem. The revenue of each passenger will be reduced by π_p to make passengers that were served in more than one route in the previous iteration less attractive in the current iteration.

$$\textbf{Max} \sum_{p\in P}((1-\alpha.r_p)R_p - \pi_p)\sum_{v\in V} d_{pv} - \sum_{p\in P}(R_p - \pi_p).(1+\beta).a_p.t_p - \sum_{i,j\in N}\sum_{v\in V} C_{ij}.X_{ij}^v \qquad (35)$$

If the master problem yields $\pi_p > 0$, it implies that passenger p appears in more than one route. To avoid infeasibility, the master problem picks the route that maximizes its objective function. Then, to prevent this passenger from appearing in multiple routes in the next iteration, the revenue of this passenger is reduced. Thus, in the next iteration of solving the sub-problems, this passenger appears only in routes that remain profitable despite her/his reduced revenue.

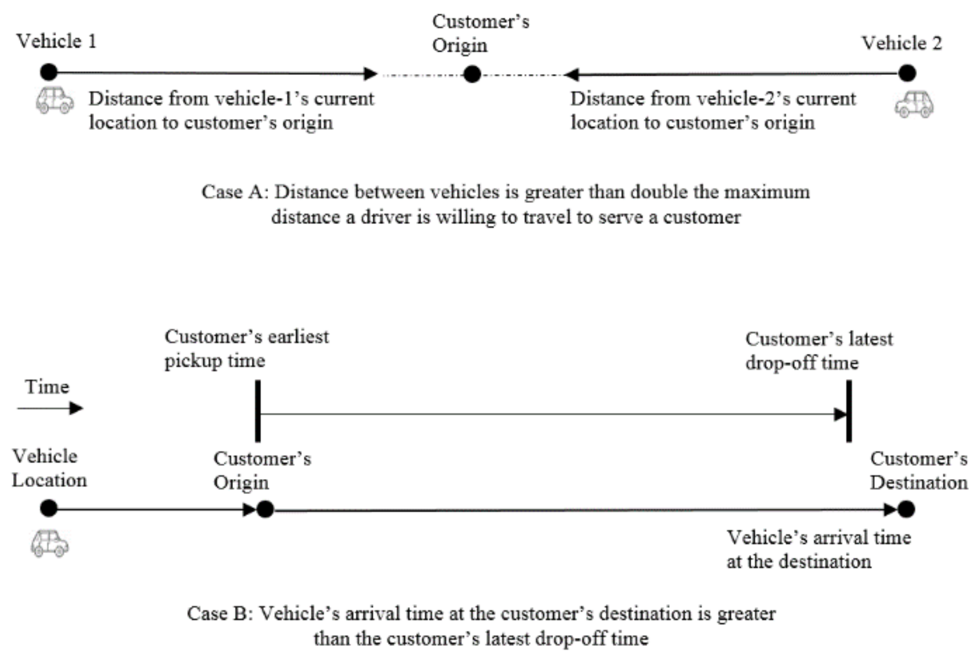


Figure 4-5: Illustration of feasibility and profitability checking rules

4-5. Network Augmentation

As mentioned earlier, each sub-problem is solved iteratively. Determining the vehicles and passengers belonging to each sub-problem, the KDSPA is applied to specify the network used to solve each sub-problem. In each iteration, as illustrated in the gray box in Figure 4-3, a set of new distinct shortest routes is generated, and their links are added to augment the network of each sub-problem. The augmented network includes a subset of nodes and links of the original network.

For each sub-problem, the KDSPA is activated to determine the shortest paths between three sets of origin-destination (OD) pairs. As illustrated in Figure 4-6, the first set is from the vehicles' current locations to the origins of all passengers. The second set is from the origin to the destination of each passenger. The third set is from each passenger's destination to the origin of all other passengers in the sub-problem. For each sub-problem, we set the parameter K as a large number. The links generated from the first iteration of the KDSPA give the basic sub-network used to solve this sub-problem.

Penalizing all links in the basic sub-network, the KDSPA is again activated to determine a new set of paths for all OD pairs mentioned above. The links of these paths are added to all links determined in the previous iteration(s), forming a larger sub-network. The sub-problem is solved again using the new augmented sub-network. The process continues for each sub-problem until no change is recorded in the optimal routes generated for all vehicles in the sub-problem.

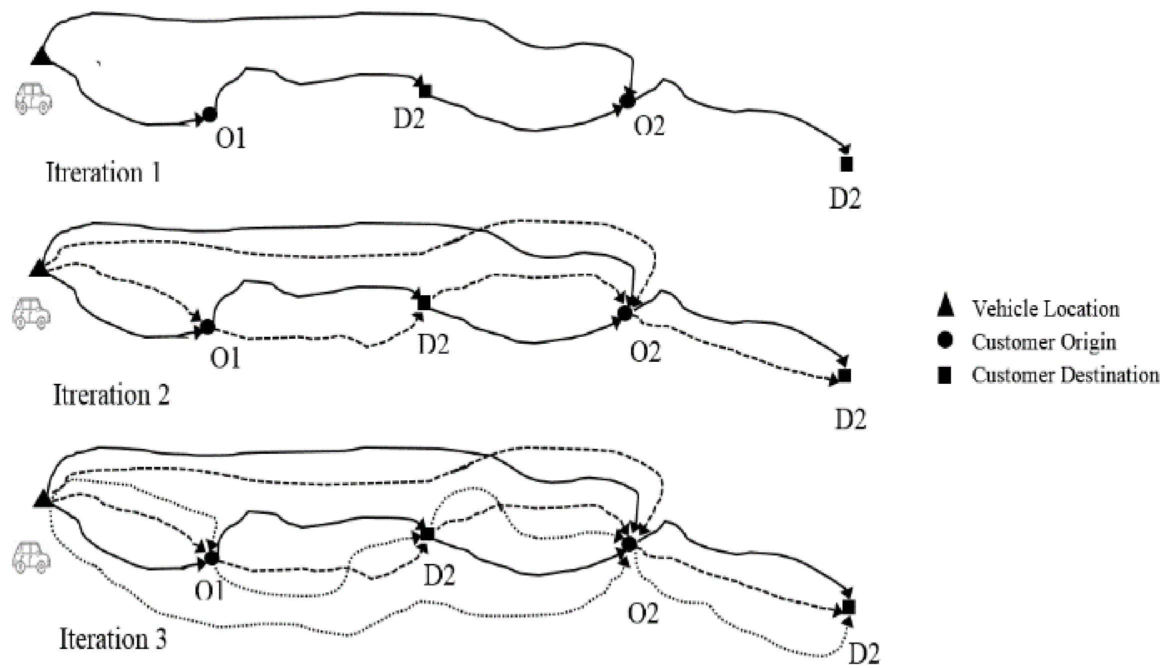


Figure 4-6: Network augmentation using the KDSPA for a network used to route one vehicle and two customers

4-6. Summary

This chapter presented the solution methodology for solving the offline version of the ODMS with ridesharing/transfer options and time window constraints. As mentioned above, this problem assumes that the passengers' demand is known a priori for the entire horizon of interest. The solution methodology adopts a version of column generation that can create hybrid columns, including passengers served by one vehicle and those served via a transfer between two vehicles. The original problem is solved as a bi-level optimization problem with a master problem and a set of sub-problems. Each sub-problem is solved iteratively. In each iteration, several distinct (non-overlapping) shortest routes are generated, and their links are added to create an augmented network of each sub-problem. Hence, the augmented network includes a subset of nodes and links within the original network. When there is no change in the solutions of the sub-problems obtained in two consecutive iterations, the results are introduced to the master problem. The master problem picks columns that maximize the objective function.

Chapter 5

SOLUTION METHODOLOGY FOR PROBLEMS WITH ONLINE DEMAND

5-1. Introduction

This chapter describes an innovative methodology for integrated ride-matching and vehicle routing for ODMS with ridesharing and transfer options with online demand. Unlike the problem presented in the previous chapter, this problem assumes that the demand is not known a priori. Therefore, when a new ride request arrives, the problem is solved to assign a ride to this new customer. The methodology implements a hybrid heuristic approach, which can solve large problem instances in near real-time. From now, this problem is referred to as the dynamic (online) ODMS. The heuristic is capable of (1) promptly responding to individual ride requests and (2) periodically re-evaluating the generated solutions and recommending modifications to enhance the overall solution quality.

This chapter is organized as follows. Section 5-2 describes the general idea of the solution methodology. Section 5-3 describes the method to create the service area of each vehicle. Section 5-4 explains the network augmentation technique for fast shortest path computation. Section 5-5 describes how candidate vehicles to offer a service to a new ride request are determined, while section 5-6 describes the mechanism used to update the routes of these vehicles. Section 5-7 describes the rollback strategy used to enhance the overall quality of the solution. Finally, section 5-8 provides a summary of the chapter.

5-2. An Overview of the Solution Methodology for Online Problems

Consider the directed network described in chapter 3. Once a passenger requests a ride, this passenger is expected to promptly receive a confirmation message from the TNC operator, typically through a mobile application, to inform the passenger about service availability. The confirmation message sent to a passenger includes the vehicle identification information, expected pickup time, and the route to the destination, including information on the transfer location, if any (i.e., trip itinerary).

The problem entails matching each received ride request to one of the vehicles in the network. In case no single vehicle can be found to offer the service, the ride request is matched to two vehicles considering a transfer. The problem also entails determining an efficient route for each vehicle to serve all assigned passengers while satisfying their time window constraints and their preferences with respect to ridesharing and transfer. If the generated itinerary does not satisfy the passenger's trip constraints and preferences, the passenger is declined the service, and she/he is free to request the service from another TNC. We assume that the TNC operator can communicate with the passenger later and check if she/he is still seeking a ride.

The methodology adopts a hybrid heuristic-based approach that integrates (a) a myopic-based heuristic to allow fast response to individual calls following the First-Come-First-Served (FCFS) rule, which is activated upon receiving a new ride request (i.e., event-based activation); and (b) a greedy-based heuristic to re-optimize previous service commitments which is activated periodically at predefined activation intervals (i.e., time-based activation).

The framework generates ride-matching and vehicle routing solutions based on the latest information available on ride requests and vehicle locations and schedules. If no passenger is assigned to a vehicle, they are assumed to stay in their current location while waiting for ride-matching. As shown in Figure 5-1, the methodology starts by defining a potential service (catchment) area for each vehicle, which is dynamically updated based on the current location of the vehicle, the time windows of all passengers currently assigned to the vehicle, and prevailing traffic congestion in vehicle's vicinity.

Upon the arrival of a new ride request, the methodology determines candidate vehicle(s) that can serve the passenger(s) associated with this request. More details about determining the candidate vehicles are given in section 5-5. If a passenger could be served in multiple vehicle tours, these tours are evaluated considering a pre-specified criterion, and the most efficient tour is selected. Once a vehicle(s) is selected, the service area of the vehicle(s) is updated to include new nodes along with their feasible visiting time windows. A new node is added if the vehicle can reach it without violating the time window constraints of any of the passengers already assigned to it.

As no information on future ride requests is available, there might be a situation in which a future more-profitable ride request is denied service if all vehicles are currently busy serving less-profitable requests. The rollback procedure, which is described in more detail in section 5-7, is designed to reduce such revenue losses by re-assigning vehicles and updating passenger itineraries, if feasible. The rollback process is activated periodically by considering a past horizon of a pre-determined length. The procedure re-evaluates the passenger-vehicle assignments made in this past horizon. The procedure determines if a passenger could be assigned to a different vehicle or if a service can be offered to a passenger(s) who was previously denied service due to vehicle unavailability. The following sections describe the main procedures constituting the solution methodology given in Figure 5-1.

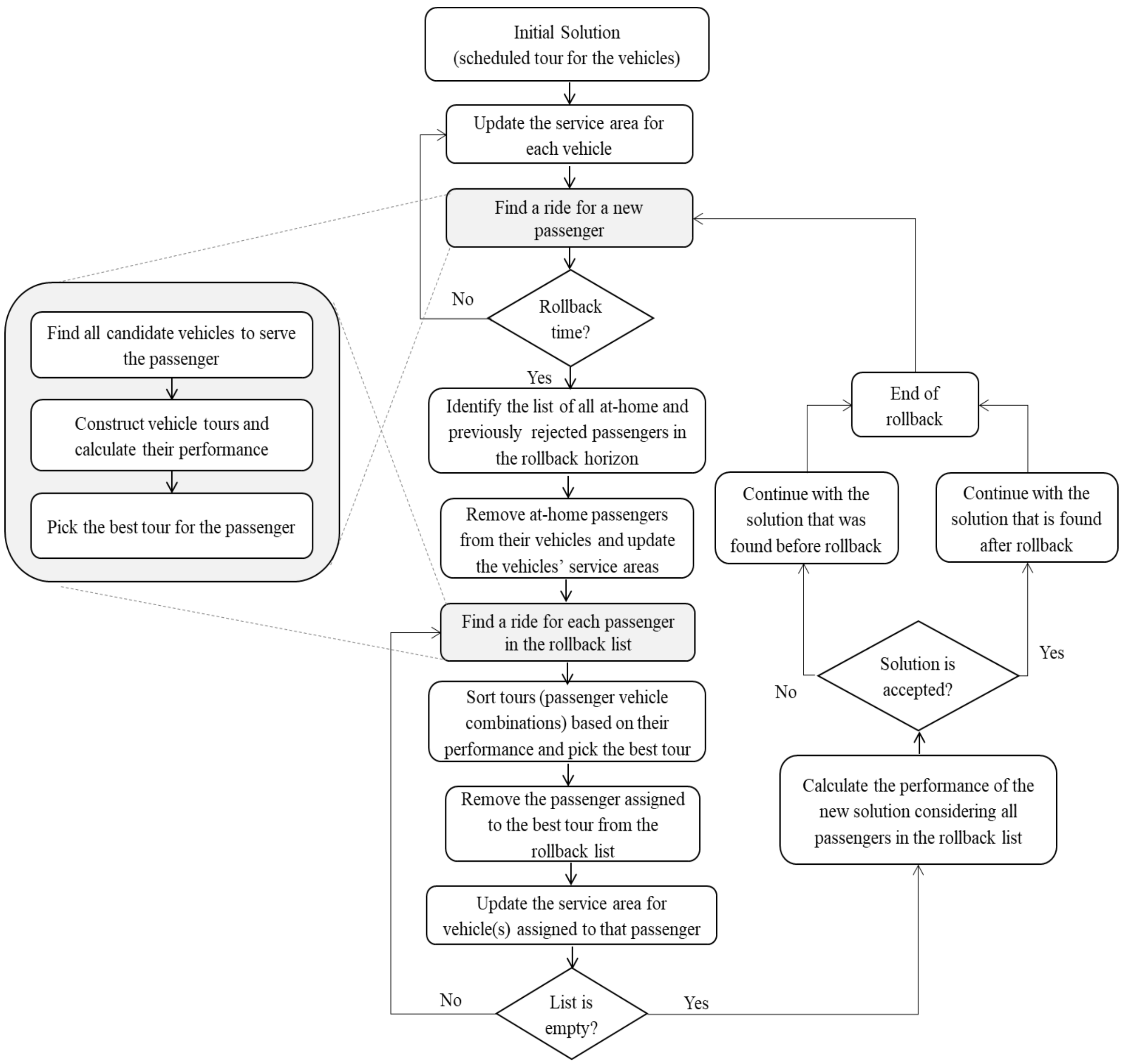


Figure 5-1: Overall framework of the solution methodology

5-3. Creating the Service Area of Each Vehicle

The service area of a vehicle is a sub-network that includes all nodes that can be reached by the vehicle considering the time window constraints of all passengers currently assigned to that vehicle and the prevailing traffic congestion along with the roadway network. To illustrate how the service area of a vehicle is determined, consider a vehicle that is assigned to serve one passenger with a given earliest pickup time and latest drop-off times. In this example, there are three nodes in the route of the vehicle, 1) the current location of the vehicle, 2) the origin of the passenger, 3) the destination of the passenger. Consider the earliest time of visiting the current location of vehicle equal to the current time, the earliest time of visiting of origin node equal to passenger's earliest pickup, and latest visiting time of destination equal to passenger's latest drop-off time.

The shortest path algorithm is used to determine the earliest and latest times (a_i, b_i) at which each node i in the vehicle tour can be reached without violating the pickup and drop-off time windows defined by the passenger. Given the vehicle tour and the earliest and latest time to visit each node, other nodes in the network are checked to be included in the vehicle's service area. As illustrated in Figure 5-2, node k can be included in the vehicle's service area between nodes i and j in the vehicle tour if the condition in (36) is satisfied.

$$t_{ik} + t_{kj} \leq b_j - a_i \tag{36}$$

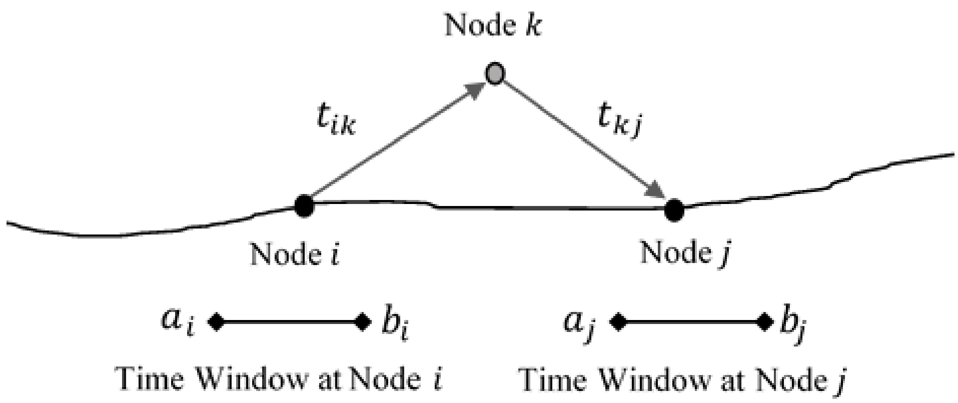


Figure 5-2: Tour Divergence to Add a New Node to the Vehicle's Service Area

Where t_{ik} and t_{kj} are the travel time between node pairs $i - k$ and $k - j$, respectively, a_i is the earliest pickup time at node i, and b_j is the latest drop-off time at node j. Satisfying this condition implies that the vehicle can deviate from its current tour to visit node k after node i and return to node j without violating the time window constraint of node j. The earliest and latest times to access node k are determined as explained in Equation (37) and Equation (38), respectively. This step is repeated to determine all nodes that can be included in the service area of the vehicle, along with their earliest and latest access times.

$$a_k = a_i + t_{ik} \tag{37}$$

$$b_k = b_j - t_{kj} \tag{38}$$

Reaching the last node in the tour (dropping off the last passengers at a destination or a transfer node), the vehicle could reach any node in the network with no constraints. Equation (37) is used to determine the earliest time that a node could be visited from the last node in the tour. One should note that a node could be included multiple times in the service area of a vehicle with different earliest and latest access times. This allows a node to be visited multiple times by the vehicle to construct efficient vehicle tours. Such a feature overcomes the limitation of most existing vehicle routing methodologies in which each node is visited once (e.g., frameworks presented in Hosni et al., 2014 and Mahmoudi and Zhou, 2016).

5-4. Network Augmentation for Efficient Shortest Route Computation

As mentioned above, constructing the service area of each vehicle requires information on the shortest route between the vehicle's already scheduled stops and potential nodes to be included in the service area. To reduce the running time associated with such computation, we adopt the Ellipsoid Spatiotemporal Accessibility Method (ESTAM) as a network augmentation method (Masoud and Jayakrishnan, 2017). According to ESTAM, the region in the network accessible by the vehicle is assumed to be restricted to the inside and on the circumference of an ellipse. All nodes outside the ellipse are assumed to be outside the service area of the vehicle. The vehicle's origin and destination are the two focal points of this ellipse. For ellipse with two focal points i and j, the transverse diameter D is the summation of the distance from the first foci i and node k on the circumference of the ellipse plus the distance from node k and the second foci j. The diameter, D, is computed as a function of the maximum time available for the vehicle to travel between i and j, assuming a conservative average traveling speed, s, as given in (39). Thus, node k falls inside or on the circumference of the ellipse if it satisfies the inequality given in (40), where d measures the Euclidean distance between two nodes.

$$D = (b_j - a_i) \cdot s \tag{39}$$

$$d(i,k) + d(k,j) \leq D \tag{40}$$

As shown in Figure 5-3, each pair of consecutive nodes (origin, destination, and transfer) in the vehicle tour are considered as focal points of an ellipse. Hence, multiple ellipses are constructed to form the augmented network for the vehicle. It should be noted that not all nodes in the ellipse are accessible to the vehicle as the actual travel time in the network is higher than the travel time of the straight line between nodes, which is also computed using a conservative traveling speed.

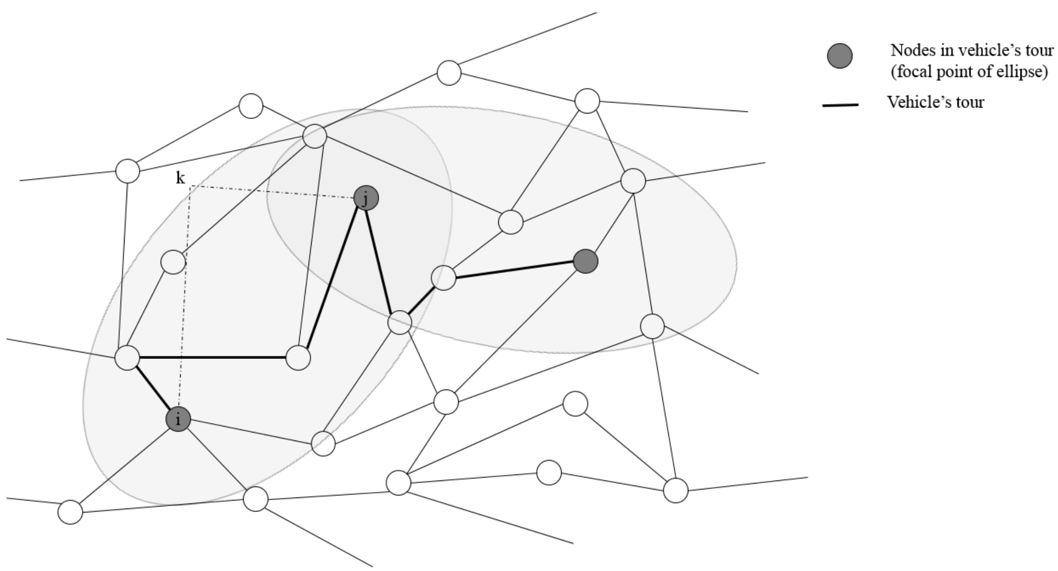


Figure 5-3: Network augmentation to determine the service area of each vehicle

5-5. Determining Candidate Vehicles to Serve a New Ride Request

A new ride request is defined with the passenger's origin-destination pair OD, earliest pickup time e_p, and latest drop-off time l_p. Given the travel time from the origin to the destination t_{OD}, the latest feasible pickup time from origin b_O, and the earliest feasible drop-off time at destination a_D, are computed in Equation (41) and Equation (42), respectively.

$$b_O = l_p - t_{OD} \tag{41}$$

$$a_D = e_p + t_{OD} \tag{42}$$

Thus, the passenger’s pickup time window extends from e_p to b_O, and her/his drop-off time window extends from a_D to l_p. The ride information is then compared against the service areas of all vehicles in the network to find candidate vehicles to serve the passenger. A vehicle is a candidate to serve this ride with no transfer if both the origin and the destination of the trip are falling in the service area of that vehicle and they are reached within the time windows specified by the passenger.

If the transfer option is considered, two vehicles are determined such that the origin and the destination of the trip belong to the service areas of the first and the second vehicles, respectively. Again, the time windows at which these two vehicles are available at the origin and destination must overlap with the passenger's pickup and drop-off time windows. In addition, the service area of these two vehicles must include a transfer node such that the time windows scheduled for the two vehicles to reach that transfer node overlap to enable the transfer to occur. If no single vehicle or two vehicles are available, the passenger is marked as unserved at the current time. We assume that the TNC operator can communicate with this passenger later to offer the service as it becomes available, and the passenger is still seeking it.

5-6. Vehicle Tour Construction and Selection of the Best Tour

As described earlier, candidate vehicles are selected by checking the feasibility of inserting one node at a time in the current tour of the vehicle. Thus, there is no guarantee of the feasibility of a newly constructed tour if two new nodes (e.g., origin-destination, origin-transfer, or transfer-destination) are simultaneously inserted in the current tour. Therefore, a step is required to determine the set of feasible vehicle tours after inserting the two new nodes associated with the new ride request. Inserting the two new nodes could result in different tours for the vehicle, as shown in Figure 5-4. In this step, taking advantage of the limited number of stops in any tour, all feasible tours are generated after inserting the origin and the passenger's destination in the original tour of the vehicle(s). This step is implemented in parallel following a multi-threading technique to expedite the generation of these tours. Any tours that do not meet the passenger's preferences regarding ridesharing and transfer are eliminated. The feasible tours then are ranked based on the performance criterion of interest. For example, ride requests could be assigned to vehicles such that the TNC profit is maximized considering any paid discounts for ridesharing and transfer. Alternatively, the ride requests are assigned to vehicles that minimize the passengers' travel time. After selecting the best tour and determining the vehicle(s) serving this new ride, the vehicle(s)' service area is updated to reflect the time window constraints of all passengers served by this vehicle, including the newly added passenger.

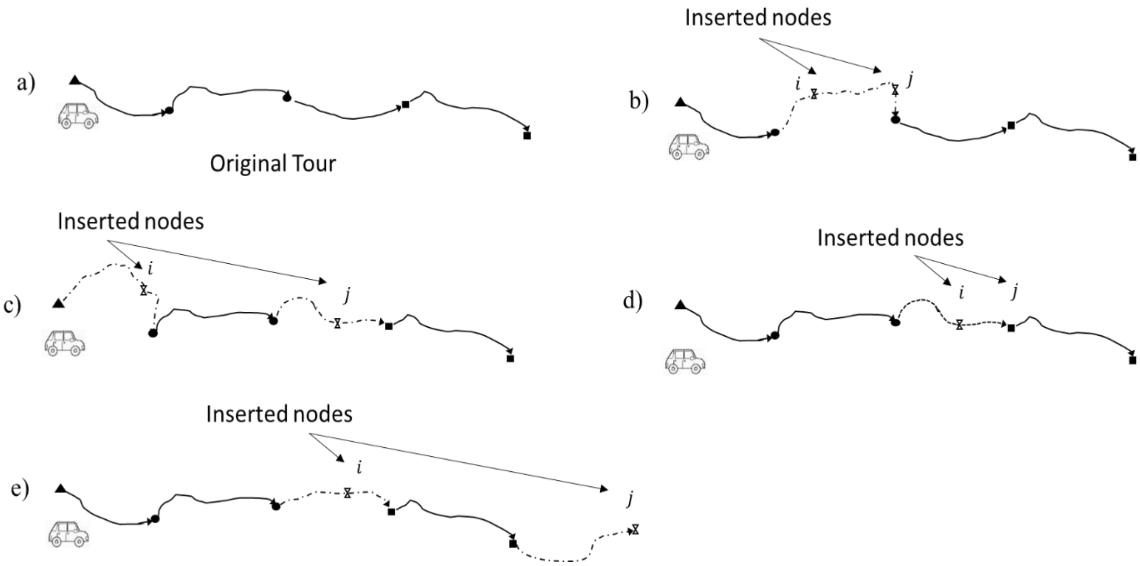


Figure 5-4: Different insertion strategies for new passenger

5-7. Activation of the Rollback Procedure

The methodology periodically activates a greedy-based heuristic to examine if a solution generated during a predefined past horizon can be improved considering updated information on the ride requests and the tours scheduled for all vehicles. A list of passengers is constructed at each activation, including passengers already assigned rides but have not yet started their trips, and passengers previously denied the service due to vehicle unavailability when they initially requested the service. Figure 5-5 illustrates the logic of the procedure. The steps of the greedy-based heuristic implemented at each activation are also given in Figure 5-1.

The heuristic starts by removing any passenger previously assigned to a vehicle from the tour of that vehicle and updating the service area of that vehicle accordingly. Then, all ride requests in the rollback list are grouped. The ride-matching algorithm is activated to determine a match for each passenger in the list, as described above. The passenger-vehicle(s) combination with the highest performance is selected following a greedy strategy. The service area of the vehicle(s) matched to this passenger is updated to consider the time window constraints of this new passenger. This passenger is removed from the list, and the process is repeated until all passengers in the list are matched to vehicles or marked as passengers who cannot be served. At the end of the process of matching all passengers in the list, the performance of this new solution is compared to the solution found before activating the rollback procedure. The new solution is adopted if it has a higher performance and it does not deny the service to any passengers who were previously assigned a service.

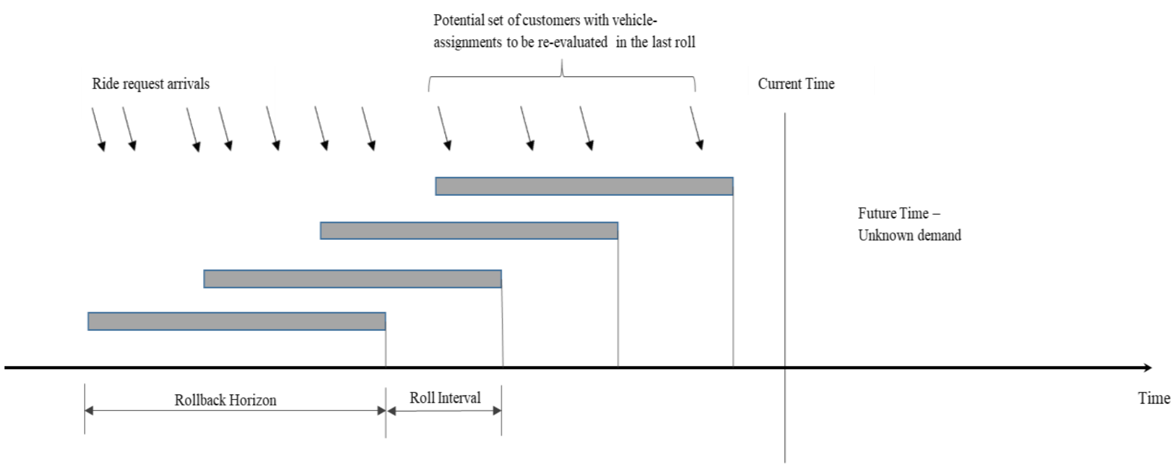


Figure 5-5: Implementation of the rollback procedure

5-8. Summary

This chapter presented the solution methodology for solving the online version of the ODMS with ridesharing/transfer options and time window constraints. In this version of the problem, the passengers' demand is not known a priori for the entire horizon of interest; therefore, at the arrival of a new ride request, the problem is solved to assign a ride to this new customer. The methodology adopts a hybrid heuristic approach, which enables solving large problem instances in near real-time. The heuristic allows to (1) promptly respond to individual ride requests and (2) periodically re-evaluate the generated solutions and recommend modifications to enhance the overall solution quality.

The solution methodology starts with creating each vehicle's service area, which is updated dynamically through the horizon. Then, at the arrival of each ride request, candidate vehicles to offer a service to the new ride request are determined. Different criteria could be defined to choose the best route among these candidate routes. After selecting the best vehicle to serve this passenger, the route(s) and service area(s) of the vehicle(s) will be updated. A network augmentation technique is used to compute the shortest paths in real-time, which adopts the Ellipsoid Spatiotemporal Accessibility Method (ESTAM). Finally, a rollback strategy which is a greedy-based heuristic, is activated periodically to examine if a solution generated during a predefined past horizon can be improved considering updated information on the ride requests and the tours scheduled for all vehicles.

Chapter 6

RESULTS AND ANALYSIS: THE OFFLINE SOLUTION METHODOLOGY

6-1. Introduction

This section applies the modified column generation solution methodology to solve hypothetical and real-world problems. The experiments examine the performance of the solution methodology with respect to different factors, including network size, the number of passengers and vehicles, and the percentage of passengers willing to rideshare and/or transfer. This chapter is organized as follows. Section 6-2 describes the setup of the experiments. Section 6-3 compares the performance of the proposed solution methodology with that obtained using the CPLEX solver. Section 6-4 describes the convergence pattern of the solution methodology. Section 6-5 presents vehicle occupancy distribution for one test problem. Section 6-6 shows the effect of passengers' preferences on the overall performance of the ODMS. Finally, section 6-7 presents a summary of this chapter.

6-2. Experimental Setup

To demonstrate the capabilities of the modeling framework and to assess the performance of the developed solution methodology, a set of experiments is conducted. These experiments examine the effect of different features, including network size, number of passengers, number of vehicles, and percentage of passengers willing to rideshare or transfer. Java programming language and the CPLEX 12.6.1 as solver are used to implement the algorithm. Experiments were executed on a Dell workstation with 512 GB of Memory and 72 processors of 2.3 GHz.

The results are recorded for two grid networks of 25 (5x5) and 100 (10x10) nodes. The results are also recorded for a real-world network representing the downtown area of the City of Dallas, which consists of 2,424 nodes and 5,947 links (see Figure 6-1). To evaluate the performance of the proposed algorithm, these problems are solved using the CPLEX solver using the default settings of the solver. For the grid network, the initial location of each vehicle and the OD pair of each passenger are selected randomly.

For the Dallas network, the passengers are randomly picked such that they resemble the region's OD demand matrix for the morning peak hour (North Central Texas Council of Government, Transportation Department, 2009). The earliest pickup time for each passenger is randomly selected within the operation horizon. To set the latest drop-off time for a passenger, a randomly generated time interval between 15 and 30 minutes plus the travel time of the shortest path between the passenger's OD pair is added to the passenger's earliest pickup time. Without loss of generality, we assume identical vehicles with a maximum capacity of four passengers and an average operating cost of $0.50/mile. A fare value of $2.0/mile is assumed to be collected from each passenger with a 10% discount for passengers who rideshare and transfer.

Figure 6-1: The Dallas downtown network used in the analysis

6-3. Benchmarking Against CPLEX

As given in Table 6-1, several performance measures are recorded for each problem instance, including Lower Bounds and Upper Bounds (LB, UB) for the best solution found by the CPLEX solver, the percentage gap of the CPLEX solution, the value of the objective function obtained by the modified CG methodology, and the execution time of both methods. The network augmentation step is not considered for the first ten problem instances (5x5 grid networks) as the network is already small. However, as described above, the network augmentation step is activated for both the 10x10 grid and the Dallas networks.

Table 6-1: Computation results of the modified column generation methodology

				CPLEX			Modified Column Generation	
Test No.	**Network**	**No. of Vehicles**	**No. of Passengers**	**Objective Function ($)**	**CPU Time (Sec.)**	**Gap (%)**	**Objective Function ($)**	**CPU Time (Sec.)**
1	5x5 Grid Network	2	8	45	53	0.0	45.0	130
2		2	10	68.0	195	0.0	64.0	320
3		3	6	33.5	1,756	0.0	33.0	760
4		3	10	60.0	2,280	0.0	60.0	730
5		3	12	(68.9,92.0)	>10,800	4.0	66.5	785
6		4	8	(43.2,49.2)	>10,800	14.0	39.0	825
7		4	10	(54.4,60.6)	>10,800	12.0	54.4	885
8		4	12	(67.3,74.3)	>10,800	10.0	59.9	1,035
9		5	14	(75.9,84.1)	>10,800	11.0	60.4	3,100
10		5	20	(72.5,102.0)	>10,800	41.0	75.3	7,500
11	10x10 Grid Network	3	4	60.0	1,680	0.0	60.0	510
12		3	10	(112.0,169.8)	>10,800	51.0	140.0	585
13		4	13	(128.0,219.4)	>10,800	71.4	155.6	565
14		4	26	-	>10,800	-	286.8	805
15		5	20	-	>10,800	-	252.3	950
16		5	30	-	>10,800	-	334.4	1,150
17		6	25	-	>10,800	-	382.0	1,248
18		6	40	-	>10,800	-	444.5	1,330
19		7	45	-	>10,800	-	475.5	1,410
20		7	50	-	>10,800	-	505.5	1,480
21	Dallas Network	10	40	-	>10,800	-	594.0	7080
22		15	50	-	>10,800	-	758.0	8,350
23		18	60	-	>10,800	-	913.5	9,200
24		22	65	-	>10,800	-	983.0	12,150 (> 3 Hr.)
25		25	70	-	>10,800	-	1,058.0	13,980 (> 3 Hr.)

The results show that the CPLEX solver can solve only small problem instances. As the problem size increases, the solver fails to provide the solution within the three-hour limit defined for the execution time. On the other hand, the modified CG algorithm could generate near-optimal solutions in reasonable execution times. It should be noted that for some problems, the CG methodology finds a solution with an objective function that is less than the LB obtained by the CPLEX. The main reason for obtaining a lower objective function is the value of the gap used as a stopping criterion for the master and sub-problems (10% gap). For example, considering problem 6, when the gap is reduced to 0%, the objective function increases to 46.0, which is higher than the LB obtained by the CPLEX.

While the CPLEX solver recorded less execution time than the CG algorithm for small problem sizes (e.g., problem no. 1 and problem no. 2), the CG algorithm obtained the optimal or near-optimal solutions for all other problem instances in much fewer execution times. For example, considering problem 11, which includes three vehicles and four passengers, the CPLEX solver obtained the optimal solution in 1,680 seconds compared to 510 seconds recorded by the modified CG. For problem 13, which includes four vehicles and 13 passengers, the CPLEX solver provided lower and upper bounds of $128.0 and $219.5 within three hours. The modified CG algorithm obtained an objective value of $155.6 in 565 seconds. No solution was obtained for any of the Dallas network's problem instances using the CPLEX solver within the three-hour execution time, as shown in Table 6-1.

6-4. Convergence Pattern

Figure 6-2 shows an example of the convergence pattern of the modified CG algorithm. This pattern is recorded for a problem instance that utilizes the Dallas downtown network with 18 vehicles and 60 passengers (Problem 23 in Table 6-1). The convergence pattern is recorded for the mater problem and four different sub-problems. Two of the sub-problems are selected to include one vehicle only, while the other two sub-problems include two vehicles. As shown in the figure, the master problem converged in six iterations. In addition, the sub-problems converge in five to six iterations. Sub-problems with a single-vehicle tend to converge in slightly fewer iterations compared to sub-problems with two vehicles. These results demonstrate the fast convergence pattern of the algorithm, including its embedded sub-problems.

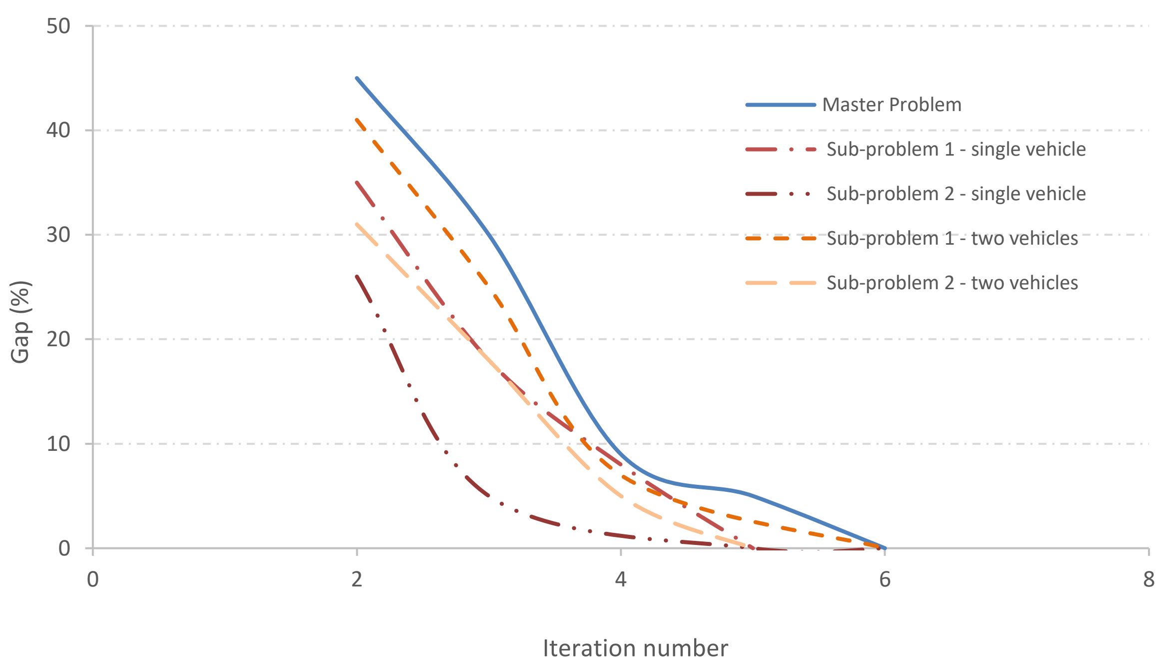


Figure 6-2: Convergence pattern of the solution algorithm

6-5. Vehicle Occupancy Distribution

Table 6-2 provides the occupancy distribution of the vehicles used to serve the 60 passengers. For example, the first vehicle was not occupied by any passengers for 16% of the time. The same vehicle was occupied by one, two, and three passengers for 37%, 29%, and 18% of the time, respectively. The vehicle was not observed to serve four passengers at any time on the horizon. The table also shows that only 12 vehicles are used in the solution. The occupancy of other vehicles is equal to zero for the entire horizon. The average percentage of time in which vehicles were idle is about 45%. Excluding the six unused vehicles, this percentage drops to 16.7%. Fifteen customers were denied service according to the obtained solution. Careful examinations of these customers reveal that these customers are either not profitable or have tight time windows that were infeasible to satisfy considering the locations of the vehicles. As explained earlier, if a vehicle's travel cost to arrive at the customer's origin is greater than the revenue generated by serving the customer, this vehicle is not assigned to this customer. Similarly, if the vehicle's expected arrival time at the customer's origin plus the travel time to her/his destination is beyond the latest drop-off time of this passenger, the customer is marked as infeasible to be served by this vehicle.

Table 6-2: Vehicle occupancy rate for a problem instance of 18 vehicles and 60 passengers in Dallas's downtown network

Vehicle	Percentage of time vehicle occupancy is:				
	0	**1**	**2**	**3**	**4**
1	16	37	29	18	0
2	19	48	33	0	0
3	15	14	39	19	13
4	25	34	26	15	0
5	13	38	22	19	8
6	19	27	42	12	0
7	18	82	0	0	0
8	16	34	23	15	12
9	21	28	35	16	0
10	12	25	63	0	0
11	3	39	30	18	10
12	23	21	39	17	0
13	100	0	0	0	0
Average	44.6	23.5	21.1	8.2	2.3
Average (without vehicle 13 to 18)	16.7	35.6	31.8	12.4	3.6

6-6. Passenger Preferences

The last set of experiments examines the effect of passengers' preferences on the performance of the ODMS. The experiments show the effect of serving demand populations with different percentages of passengers willing to rideshare or transfer. These experiments are conducted using Dallas' downtown network with 18 vehicles and 60 passengers. The results of these experiments are presented in Table 6-3 and Table 6-4, respectively. Table 6-3 gives the value of the objective function, the number of used vehicles, and the number of served passengers considering different percentages of passengers willing to transfer. For example, in the scenario with no transferring passengers, the ODMS utilizes 16 vehicles and serves only 34 customers. The ODMS' total profit under this scenario is $688.1.

On the other hand, in the scenario in which all passengers are assumed to be willing to transfer, the ODMS' profit jumps to $913.5 by serving 45 passengers using 12 vehicles. Twelve passengers transfer in this scenario with an average waiting time of 4.5 minutes, which explains the increase in the average passenger travel time. Serving more passengers with a smaller number of vehicles led to increasing the average miles traveled by vehicles from 16.5 to 23.5 miles.

Table 6-3: The effect of the percentage of passengers willing to transfer on the ODMS performance

(Percentage of passengers willing to ride share = %100)

Test No.	Percentage of passengers willing to transfer %	No. of transferred passengers	No. of served passengers	Objective Function ($)	No. of used vehicles	Average miles traveled per vehicle (M)	Average Passenger travel time (Sec.)	Average passenger waiting time at transfer nodes (Sec.)
1	0	0	34	688.1	16	16.5	840	0
2	25	5	37	733.5	14	18.7	936	252
3	50	7	38	762.6	14	19.3	1,044	318
4	75	10	40	818.0	13	20.9	1,242	360
5	100	12	45	913.5	12	23.5	1,176	270

Table 6-4 gives the results of the experiments in which different percentages of passengers are willing to rideshare. As shown in the table, serving demand populations with more passengers willing to rideshare could significantly impact the ODMS profitability. For example, if none of the passengers is willing to rideshare, a total profit of $354.0 is recorded as a result of serving 20 passengers. The profit almost tripled when this percentage increased to 100% serving 45 passengers. One can also notice that fewer vehicles are used as the percentage of passengers willing to rideshare increases. While 16 vehicles are used to serve 20 passengers in the scenario with no ridesharing, 45 passengers are served by only 12 vehicles when all passengers are assumed to be willing to rideshare.

As more passengers are served through ridesharing, the average passenger travel time and average miles traveled by vehicles also increase. The average travel time increases from 12 minutes to about 20 minutes when all passengers are assumed to be willing to rideshare. One can notice that the solutions in some problem instances include unutilized vehicles and unserved passengers as it could be infeasible and/or unprofitable for these unutilized vehicles to serve these passengers given their locations and pickup/drop-off time windows. This set of experiments shows that the attitude of passengers towards ridesharing and transfer services may have a substantial impact on the ODMS profitability. Accordingly, introducing these services demand careful studying of passengers' preferences, which could vary significantly across metropolitan areas.

Table 6-4: The effect of the percentage of passengers willing to rideshare on the ODMS performance

(Percentage of passengers willing to transfer = %100)

Test No.	Percentage of Passengers Willing to Rideshare %	No. of Served Passengers	Objective Function ($)	No. of Used vehicles	Average miles traveled per vehicle (M)	Average Passenger travel time (Sec.)
1	0	20	354.0	16	16.2	726
2	25	26	465.5	15	17.3	780
3	50	31	559.0	15	19.6	990
4	75	34	727.0	14	21.3	1,068
5	100	45	913.5	12	23.5	1,176

6-7. Summary

The chapter presents the results of a set of experiments that are performed to show the capabilities of the modeling framework and to evaluate the performance of the developed solution methodology. Different problems are solved using the CPLEX solver and the proposed methodology. Results show that CPLEX can solve only small-size problems; however, the proposed method generates good quality solutions for problems with reasonable sizes. Moreover, experiments are conducted to show the effect of serving demand with different percentages of passengers willing to rideshare or transfer. Increasing the percentage of passengers willing to transfer and/or rideshare enables the system to serve more passengers using fewer vehicles. This set of experiments shows that passengers' preferences towards ridesharing and transfer services could substantially impact the performance and profitability of the ODMS.

Chapter 7

RESULTS AND ANALYSIS: THE ONLINE SOLUTION METHODOLOGY

7-1. Introduction

In this chapter, we present the outcomes of numerous experiments that are performed to demonstrate the potentials of the heuristic-based solution methodology presented in chapter 5. The first set of experiments evaluates the effect of the criteria used to match passengers to vehicles on the overall solution quality. The second set of experiments examines the quality of the solution considering different passenger populations that vary in terms of the percentage of passengers willing to rideshare and/or transfer.

The third set of experiments investigates the effect of applying the rollback procedure on improving the overall solution quality. Finally, the last set of experiments pertains to benchmarking the solution quality obtained using the solution methodology presented in chapter 5 against those of the methodology presented in chapter 4 and the framework developed by Cheikh et al. (2017).

The methodology is implemented using the Java programming language, and all runs are carried out on a Dell workstation with 72 logical processors of 2.3 GHz and 512 GB Memory. Two different networks are used in this analysis, considering an operation period of three hours. The first network is a hypothetical grid network of 400 (20 × 20) nodes. The second network, presented in Figure 6-1, is a real-world network representing the core area of the City of Dallas, consisting of 2,424 nodes and 5,947 links. The initial location of each vehicle and the OD pair of each passenger are selected randomly for the grid network. For the Dallas network, passengers are randomly picked such that they resemble the region's OD demand matrix for the morning peak hour. The service request time of each passenger is randomly generated over a two-hour horizon.

A randomly generated number between 10 and 20 minutes is added to the service request time to set the earliest pickup time for each passenger. Similarly, to define the latest drop-off time for a passenger, a randomly generated time interval between 15 and 30 minutes plus the travel time of the shortest path between the passenger’s OD pair is added to the passenger’s earliest pickup time. In addition, a randomly generated passenger is assigned to each vehicle at the beginning of the horizon to construct the initial tour and the service area of each vehicle. Without loss of generality, we assume identical vehicles with a maximum capacity of four passengers and an average operating cost of $0.50/mile. A fare rate value of $2.50/mile multiplied by the shortest route distance between passenger’s OD pair is assumed to be collected from each passenger with a 10% discount for ridesharing and transferring passengers.

7-2. Effect of Adopting Different Passenger-Vehicle Matching Criteria

Two different criteria for matching passengers-vehicles are assessed. In the first criterion (C-I), a passenger is matched to a vehicle that maximizes the TNC total profit, which is computed as the difference between the ride fare minus the extra cost of re-routing the vehicle to serve the new request. The second criterion (C-II) focuses on enhancing the passengers’ travel experience. Options without transfer are first examined, and the vehicle that offers minimum travel time is matched to the passenger. If no route without transfer is available, the passenger is matched to two vehicles that offer the minimum travel time route with the transfer. The experiment is conducted considering the grid network with 20 vehicles and 100 passengers and the Dallas network with 50 vehicles and 200 passengers. It is assumed that all passengers are willing to share the ride and to have a transfer.

Table 7-1 compares the solution quality obtained under these two criteria. Several performance measures are recorded, including the number of served passengers, the number of transferred passengers, total profit, average miles traveled per vehicle, average passenger travel distance, and the average time that is taken to respond to a ride request. As shown in the table, solutions that adopt C-I record more profit for both networks. For example, 20.5% and 18.4% more profits are recorded for the grid and Dallas networks, respectively, compared to their corresponding solutions in which C-II is considered. Furthermore, solutions that adopt C-I serve more passengers than those of C-II. For example, 159 passengers are served under C-I compared to 138 passengers served under C-II for the Dallas network. On the other hand, solutions that adopt C-II enhance the passengers' travel experience as indicated by the average passengers' travel distance and the total number of passengers served with the transfer.

As shown for the Dallas network, the average passenger travel distance is reduced by 23.8% under C-II compared to C-I. Similarly, the total number of transfers dropped from 29 to 10 for these two solutions. It should be noted that the average time taken to respond to a new ride request (i.e., the execution time of the methodology) does not exceed 48.0 seconds in any of the tested cases.

Table 7-1: Effect of the ride-vehicle matching criteria

Matching criteria	Network	No. of served passengers	No. of transferred passengers	Total profit ($)	Average miles traveled per vehicle (miles)	Average miles traveled per passenger (miles)	Average time to respond to a ride request (sec.)
Increasing Profitability (C-I)	20 x 20 grid network	73	15	1456.3	18.4	8.7	13.0
	Dallas network	159	29	3859.5	25.3	11.9	48.0
Enhancing Travel Experience (C-II)	20 x 20 grid network	60	4	1208.4	16.2	7.9	11.0
	Dallas network	138	10	3256.7	23.8	9.6	42.0

7-3. Effect of Passengers' Willingness to Ridesharing and Transfer

This set of experiments evaluates the effect of having passenger populations with different preferences on ridesharing and/or transfer. These experiments are conducted using the Dallas network with 50 vehicles and 200 passengers. The passenger-vehicle matching criteria C-I is adopted in this set of experiments. Three cases are considered. In the first case, 50% of passengers are assumed to be willing to rideshare, with the percentage of passengers willing to transfer being 0%, 10%, 50%, and 100%. In the second case, all passengers are assumed to be willing to rideshare while the percentages of passengers willing to transfer are 0%, 10%, 50%, and 100%. Finally, in the third case, all passengers are assumed to be willing to transfer, with the percentages of passengers willing to rideshare being 0%, 10%, 50%, and 100%.

The outcomes of these experiments are given in Table 7-2. As shown in the table, increasing the percentage of passengers willing to rideshare and/or transfer increases the profitability and improves other performance measurements of the ODMS. In addition, having passengers willing to rideshare is shown to have a more positive impact on the ODMS operation than having passengers willing to transfer. Comparing the solutions of the case of 50% ridesharing and 10% transfer with the case of 100% ridesharing and 10% transfer, the number of served passengers increased by 22.9% resulting in a 12.5% increase in the profit. Comparing the solutions of the case of 100% ridesharing and 10% transfer with the case of 100% transfer and 10% ridesharing, the number of served passengers increased by 21.5% resulting in a 25.1% increase in the profit. This set of experiments illustrates that passenger preferences regarding the ridesharing and transfer options could play an essential role in ODMS profitability. Therefore, introducing such services requires carefully studying passengers' preferences, which vary significantly across communities.

Table 7-2: Effect of passenger preferences on the performance of ODMS

Passenger population		No. of served passengers	No. of transferred passengers	Total profit ($)	Average ride cost for each passenger ($)	Average miles traveled per vehicle (miles)	Average miles traveled per passenger (miles)
Transfer (50% of passengers are willing to rideshare)	0	82	0	2658.4	23.8	17.5	7.5
	10	87	3	2743.2	22.7	17.9	7.8
	50	93	8	3079.2	20.9	19.7	8.1
	100	111	12	3375.8	17.3	21.2	8.3
Transfer (All passengers are willing to rideshare)	0	105	0	3040.2	18.9	19.1	9.5
	10	107	2	3087.5	18.9	19.5	9.9
	50	128	12	3689.0	17.7	21.4	10.5
	100	159	29	3859.5	15.3	25.3	11.9
Rideshare (All passengers are willing to transfer)	0	81	9	2239.7	20.5	19.8	8.0
	10	88	11 15	2467.0	19.9	20.6	8.4
	50	114	29	3489.7	18.1	22.7	9.8
	100	159	9	3859.5	15.3	25.3	11.9

7-4. Effect of Activating the Rollback procedure

In this set of experiments, the performance of the ODMS is compared for cases with and without activating the rollback procedure. The experiment is conducted considering the Dallas network. Four instances of the problem are created with different numbers of vehicles and passengers. It is assumed that all passengers are willing to rideshare and transfer. In addition, the passenger-vehicle matching criteria C-I is adopted in this set of experiments. The rollback is activated every 15 minutes covering a rollback horizon of 5, 15, and 30 minutes, respectively. The results of these experiments are presented in Table 7-3. As shown in the table, activating the rollback procedure enhances the quality of the solution in terms of TNC profitability and the number of served passengers. For example, for the case with 5 minutes rollback horizon, the number of served passengers and total profit improved by 4.6% and 3.9%, respectively. When the 15 minutes rollback horizon is considered, the average improvements in the number of served passengers and total profit are 14.2% and 12.7%, respectively.

One can also observe the additional improvements achieved by further extending the rollback horizon; however, improvements are not as significant as those with 15 minutes rollback horizon. These average improvements are increased to 15.1% and 13.3% when the rollback horizon is extended to 30 minutes. Moreover, the rollback procedure will allow re-assign passengers who have been previously assigned a vehicle but have not started their trip yet (at-home passengers). The results of this set of experiments did not show any trend in change to the overall travel time and trip cost due to re-assigning these passengers.

Table 7-3: Effect of applying rollback procedure on ODMS performance

Length of rollback horizon (Min)	Test No.	No. of vehicles	No. of passengers	Without rollback		With rollback	
				No. of served passengers	Profit ($)	No. of served passengers	Profit ($)
5	1	40	100	71	1705.2	74	1812.5
	2	40	150	119	2904.8	126	3045.7
	3	50	150	125	3078.9	130	3164.8
	4	50	200	159	3859.5	163	3976.0
15	5	40	100	71	1705.2	82	1923.1
	6	40	150	119	2904.8	138	3245.5
	7	50	150	125	3078.9	141	3367.9
	8	50	200	159	3859.5	178	4258.0
30	9	40	100	71	1705.2	84	1966.0
	10	40	150	119	2904.8	138	3245.5
	11	50	150	125	3078.9	142	3379.0
	12	50	200	159	3859.5	178	4258.0

7-5. Benchmarking the Solution Quality

This section compares the methodology's performance presented in this chapter with those obtained by two other ridesharing frameworks in the literature. The first framework is presented in chapter 4, and the second methodology is presented by Cheikh et al. (2017). In chapter 4, the problem is modeled in the form of a Mixed Integer Program (MIP) and a solution methodology that integrates a modified version of the CG algorithm, and a network augmentation technique is used to obtain a near-optimal solution of the problem. This CG methodology assumes that information on all ride requests is available at the start of the operation horizon. In this set of experiments, three problem instances with the different number of vehicles and passengers are considered for the Dallas network, as presented in Table 7-4. The CG methodology was able to solve problems with up to 20 vehicles and 60 passengers in a pre-specified three-hour execution time threshold. Thus, no problem instances of larger sizes are considered in this set of experiments. All passengers are assumed to be willing to rideshare and transfer. Both methodologies are set to maximize the TNC profit. In addition, the rollback procedure is activated considering a rollback horizon of 15 minutes.

As shown in Table 7-4, the proposed ride-matching and vehicle routing methodology outperforms the modified CG methodology regarding the number of served passengers and total profit in two out of the three tested problem instances. Higher profitability of 4% and 2.9% are recorded for problem instances 1 and 3, respectively. In addition, the number of served passengers increased by 6.4% and 4% for these two problems. The average time to respond to the ride requests does not exceed 42 seconds. Although the CG provides a good quality solution, it had an execution time of 7,450 and

9,430 seconds for the first and third problems, limiting its applicability considering the real-time nature of the ODMS operation.

Table 7-4: Comparing solution of the proposed framework and the modified column generation

			Modified column generation			Ride-matching and vehicle routing methodology		
Test No.	**No. of vehicles**	**No. of passengers**	**No. of served passengers**	**Total profit ($)**	**Execution time (sec.)**	**No. of served passengers**	**Total profit ($)**	**Average time to respond to a ride request (sec.)**
1	10	40	31	986.6	7,450	33	1025.3	40.0
2	15	50	41	893.0	8,160	39	849.0	40.0
3	20	60	49	1043.5	9,430	51	1065.0	42.0

In the following, we compare the performance of our solution methodology with the one presented in Cheikh et al. (2017) to solve a dynamic ride-matching problem with the transfer. Cheikh et al. (2017) considered a carpooling problem in which drivers have pre-determined origins and destinations as well as specified departure and arrival times. This assumption is expected to significantly reduce the complexity of the problem compared to the general settings considered for the ODMS presented here. In addition, they assume all passengers are willing to share the ride with others and have transfer in their routes. The problem is solved using a meta-heuristic approach that adopts a Controlled Genetic Operators (MACGeO). Table 7-5 summarizes the results of a set of experiments that compare the performance of the solution methodology presented above against that of MACGeO. Two scenarios are considered as presented in Cheikh et al. (2017).

The first scenario assumes 20 vehicles and 36 service requests, while the second scenario assumes 75 vehicles and 120 service requests. The first scenario results show that our approach used 11 vehicles to serve all passengers (36 passengers), while MACGeO serves only 34 passengers using 12 vehicles. Because drivers do not have pre-determined origins, destinations, and specified departure and arrival times in our framework, more passengers are served using a smaller number of vehicles. Our methodology is also shown to be more superior in terms of offering more convenient services to passengers as only 20% of the passengers are served by a transfer option compared to 26.5% in the solution of MACGeO. A similar pattern is observed for the second scenario. According to the MACGeO solution, 36 vehicles are used to serve 115 passengers, with about 26% of the passengers being served with a transfer. Our methodology offered service to 119 passengers using 34 vehicles, where only 22% of the passengers are served by a transfer option. The results show the superiority of our methodology as it provides services to more passengers and reduces the number of trips with transfers, which enhances the passengers' travel experience and overall service satisfaction. One should note that the result of some presented experiments indicates that not all passengers are served. This result pertains to the limited number of vehicles used in the setting of these experiments to compare our algorithm's performance with that of other methods. As illustrated, increasing the number of vehicles allows serving more passengers and increasing the overall profitability.

Table 7-5: Comparing solution of the proposed framework and the MACGeO

Methodology	No. of vehicles	No. of passengers	No. of used vehicles	No. of served passengers	% of transferred passengers
MACGeO	20	36	12	34	26.5
	75	120	36	115	25.6
Ride-matching and vehicle routing	20	36	11	36	20.0
	75	120	34	119	22.0

7-6. Summary

This chapter presented the results of experiments that are performed to illustrate the capabilities of the heuristic-based solution methodology to solve the online version of the problem. The first set of experiments evaluates the effect of the criteria used to match passengers to vehicles on the overall solution quality. In the first criterion (C-I), a passenger is matched to a vehicle that maximizes the TNC profit. The second criterion (C-II) focuses on enhancing the passengers' travel experience. Results show that solutions that adopt C-I record more profit and more served passengers.

On the other hand, solutions that adopt C-II enhance the passengers' travel experience as indicated by the average passengers' travel distance and the total number of passengers served with the transfer. The second set of experiments examines the quality of the solution considering different passenger populations that vary in terms of the percentage of passengers willing to rideshare and/or transfer. As shown, increasing the percentage of passengers willing to rideshare and/or transfer increases the profitability and enhances other performance measures of the ODMS.

The third set of experiments investigates the effect of applying the rollback procedure on improving the overall solution quality. Activating the rollback procedure will enhance the quality of the solution in terms of TNC profitability and the number of served passengers. Finally, the last set of experiments pertains to benchmarking the solution quality obtained using the solution methodology presented in chapter 5 against those obtained by two other ridesharing frameworks in literature. The first framework is presented in chapter 4, and the second methodology is presented by Cheikh et al. (2017).

Chapter 8

SUMMARY

This dissertation presents a modeling framework for ODMS operation in metropolitan areas. The framework represents ridesharing and transfer services while capturing the passengers' critical characteristics, including the willingness to rideshare and transfer. The problem pertains to routing vehicles efficiently to serve passengers who specify their desirable pickup and drop-off times and their willingness to rideshare and/or transfer in return for fare discounts. Previous studies show that considering transfer services would enable ODMS providers to offer service in situations where there is a shortage in the vehicle supply. For instance, TNCs might not be able to recruit an adequate number of drivers in some locations or during specific periods to serve the anticipated passengers' demand. Although enabling the transfer service may cause an increase in passengers' travel time, it would allow TNCs to serve more customers using fewer vehicles.

In chapter 2, an inclusive review of the vehicle routing problem and its extensions related to the ODMS problem was presented. Problems reviewed in this chapter include (I) single-vehicle and multi-vehicle static routing problems with time window, (II) the vehicle routing problem with dynamic (online) demand, and (III) the vehicle routing problem with the transfer.

Reviewing the previous research reveals that while the existing literature captures essential aspects of ODMS operation, the developed methodologies lack essential capabilities that prevent their adoption in real-world applications. For example, existing models cannot represent both ridesharing and transfer in one comprehensive model. In addition, existing models ignored the heterogeneity of passengers' preferences, especially with respect to their willingness to rideshare and/or transfer. Furthermore, ODMS providers can decline ride requests if they are not profitable. Examining the profitability of ODMS requires a modeling framework that can identify profitable trips based on the temporal-spatial distributions of passengers and vehicles in the network. Finally, despite the previous effort to develop efficient solution methodologies, they are still short of modeling ODMS in networks of moderate sizes.

Chapter 3 described the formal definition of the problem in the form of a mathematical program formulation. In this mathematical program, the objective function maximizes the total ODMS profit, which is a function of total revenue collected from all served passengers considering discounts offered to passengers who rideshare or transfer and the operation cost of the vehicles. A set of constraints is presented to satisfy passengers' time window constraints and rideshare and/or transfer preferences and ensure path continuity for vehicles and passengers.

Chapters 4 and 5 provided the solution methodologies to solve the problem's offline and online versions, respectively. In chapter 4, a modified version of the column generation that can create hybrid columns is proposed to solve the classical vehicle routing problem to consider the transfer option. The methodology integrates network decomposition and augmentation approaches to solve problems of reasonable sizes. The original problem is solved as a bi-level optimization problem with a master problem and a set of sub-problems. Each sub-problem is solved iteratively. In each iteration, several distinct (non-overlapping) shortest routes are generated, and their links are added to create an augmented network of each sub-problem. When there is no change in the solutions of the sub-problems obtained in two consecutive iterations, the results are introduced to the master problem. The master problem picks columns that maximize the objective function.

Chapter 5 presented a novel methodology that adopts a hybrid heuristic approach, enabling solving large problem instances in near real-time. The methodology adopts a hybrid heuristic-based approach that integrates (a) a myopic-based heuristic to allow fast response to individual calls following the First-Come-First-Served (FCFS) rule, which is activated upon receiving a new ride request (i.e., event-based activation); and (b) a greedy-based heuristic to re-optimize previous service commitments which is activated periodically at predefined activation intervals (i.e., time-based activation). The framework generates ride-matching and vehicle routing solutions based on the latest information available on ride requests and vehicle locations and schedules. The methodology starts by defining a potential service (catchment) area for each vehicle, which is dynamically updated based on the vehicle's current location, the time windows of all passengers currently assigned to the vehicle, and prevailing traffic congestion in the vehicle's vicinity. Upon the arrival of a new ride request, the methodology determines candidate vehicle(s) that can serve the passenger(s) associated with this request. Once a vehicle(s) is selected to serve the new passenger, the service area of the vehicle(s) is updated to include new nodes along with their feasible visiting time windows. The rollback process is activated periodically by considering a past horizon of a pre-determined length. The procedure re-evaluates the passenger-vehicle assignments made in this past horizon. The procedure determines if a passenger could be assigned to a different vehicle or if a service can be offered to a passenger(s) who was previously denied service due to vehicle unavailability.

A set of experiments is conducted using the grid and real-world networks to illustrate the capabilities of the modeling framework and the performance of the developed solution methodologies. The results are presented in chapters 6 and 7 for offline and online solution methodologies, respectively.

In chapter 6, different problem instances were solved using the CPLEX solver and the proposed methodology. Results show that the CPLEX could solve only small-size problems; however, the proposed method generates good quality solutions for problems with reasonable sizes. Moreover, experiments are conducted to show the effect of serving demand with different percentages of passengers willing to rideshare or transfer. Increasing the percentage of passengers that are willing to transfer and/or rideshare enables the system to serve more passengers using fewer vehicles. This set of tests explains that passengers' preferences towards these new services could significantly affect the performance and profitability of the service.

Chapter 7 presented the results of experiments that are performed to illustrate the capabilities of the heuristic-based solution methodology. Two different criteria were used to match passengers to vehicles in this experiment. In the first criterion (C-I), a passenger is matched to a vehicle that maximizes the TNC profit. The second criterion (C-II) focuses on enhancing the passengers’ travel experience. Results show that solutions that adopt C-I record more profit and more served passengers. On the other hand, solutions that adopt C-II enhance the passengers' travel experience as indicated by the average passengers' travel distance and the total number of passengers served with the transfer.

In addition, the results of experiments show that increasing the percentage of passengers who are willing to rideshare and/or transfer increases the profitability and enhances other performance measures of the ODMS. Comparing the solutions of the case of 100% ridesharing and 10% transfer with the case of 100% transfer and 10% ridesharing resulted in a 25.1% increase in the profit. Moreover, activating the rollback procedure will result in enhancing the quality of the solution in terms of TNC profitability and the number of served passengers.

Three main extensions could be considered for this research work. For example, the framework could be extended to consider ODMS integration with public transportation services. In this case, ODMS could play a vital role as a feeder to the public transportation system and increase public transportation ridership. However, coordination of this integrated system is challenging. In addition, integrating the above methodology in demand forecasting studies to capture the effect of ODMS modal split and route assignment is another extension.

Moreover, studying the effect of the congestion dynamics in the traffic network on ODMS operation is another possible extension. Considering congestion in the network and its effect on the travel time will improve the solution quality and reliability of the ODMS. Although the problem instances considered in this research are large compared to the problems studied in the literature, larger problem instances with higher demand levels, more vehicles, and a bigger coverage area could be considered in future works.

BIBLIOGRAPHY

Agatz, N., Erera, A., Savelsbergh, M., and Wang, X. (2011), Dynamic ride sharing: A simulation study in metro Atlanta, *Procedia-Social and Behavioral Sciences*, 17(1), 532-550.

Agatz, N., Erera, A., Savelsbergh, M., and Wang, X. (2012), Optimization for dynamic ride sharing: A review, *European Journal of Operational Research*, 223(2), 295-303.

Alonso-González, M., Cats, O., Van Ort, N., Hoogendoorn-Lanser, S., and Hoogendoorn, S. (2020), What are the determinants of the willingness to share rides in pooled on-demand services. *Transportation*, 1-33.

Alonso-Mora, J., Samaranayake, S., Wallar, A., Frazzoli, E., and Rus, D. (2017), On-demand high-capacity ridesharing via dynamic trip-vehicle assignment. *Proceedings of the National Academy of Sciences*, 114(3), 462-467.

Andini, M., Satria, Y., and Burhan, H. (2019), Dynamic pickup and delivery problem with the transfer in ridesharing to reduce congestion. In *Journal of Physics: Conference Series,* 1218(1).

Arslan, A., Agatz, N., Kroon, L., and Zuidwijk, R. (2016), Crowdsourced delivery: A dynamic pickup and delivery problem with ad-hoc drivers. *Transportation Science*, 12(2), 23-34.

Arockia, A., Lochbrunner, M., Hanne, T., and Dornberger, R. (2021). Benchmarking Tabu Search and Simulated Annealing for the Capacitated Vehicle Routing Problem. *In 2021 The 4th International Conference on Computers in Management and Business*, 118-124.

Asper, D. (2017), The timely guide to on-demand delivery. https://shopify.com/guides/on-demand-delivery. Accessed April 30, 2019.

Attanasio, A., Cordeau, F., Ghiani, G., and Laporte, G. (2004), Parallel tabu search heuristics for the dynamic multi-vehicle dial-a-ride problem. *Parallel Computing*, 30(3), 377-387.

Azi N., Gendreau M., Potvin J. (2012), A dynamic vehicle routing problem with multiple delivery routes. *Operations Research*, 199(1), 103–112.

Bakutyte, J. (2017), Toyota's is developing a self-driving store with amazon, Uber and pizza hut. http://konbini.com/us/entertainment/toyota-e-palette-self-driving-vehicle-uber-amazon-pizza-hut. Accessed November 5, 2018.

Baldacci, R., Vittorio M., and Aristide M. (2004), An exact method for the carpooling problem based on Lagrangean column generation, *Operations Research*, 52(3), 422-439.

Baldacci, R., Bartolini, E., and Mingozzi, A. (2011), An exact algorithm for the pickup and delivery problem with time windows. *Operations Research*, 59(2), 414-426.

Balinski, M., and Quandt, R. (1964), On an integer program for a delivery problem. *Operations Research*, 12, 300-304.

Barnhart, C., Johnson, E., Nemhauser, G., Savelsbergh, M., and Vance, P. (1998), Branch-and-price: Column generation for solving huge integer programs, *Operations research*, 46(3), 316-329.

Berbeglia, G., Cordeau, J., Gribkovskaia, I., and Laporte, G. (2007), Static pickup and delivery problems: a classification scheme and survey, *Top* 15(1), 1-31.

Berbeglia, G., Cordeau, J., and Laporte, G. (2012), Hybrid tabu search and constraint programming algorithm for the dynamic dial-a-ride problem, *INFORMS Journal on Computing*, 24(3), 343-355.

Braekers, K., Ramaekers, K., and Nieuwenhuysec, I. (2014), The vehicle routing problem: state of the art classification and review. *Computers and Industrial Engineering*, 99, 300-313.

Braekers, K., Caris, A., and Janssens, K. (2016), Exact and meta-heuristic approach for a general heterogeneous dial-a-ride problem with multiple depots, *Transportation Research Part B: Methodological*, 67, 166-186.

Bullnheimer, B., Hartl, R., and Strauss, C. (1997), Applying the Ant System to the Vehicle Routing Problem. *2nd International Conference on Metaheuristics.* Sophia-Antipolis, France.

Calvo, W., De-Luigi, F., Haastrup, P., and Maniezzo, V. (2004), A distributed geographic information system for the daily carpooling problem, *Computers and Operations Research*, 31(13), 2263-2278.

Campbell, H. (2016), How are Uber and Lyft working with public transportation authorities, www.theridesharegguy.com/how-are-uber-and-lyft-working-with-public-transportation-authorities. Accessed June 30, 2018.

Carson, B. (2022), 20+ Lyft Statistics-Rides, Market Share, and Revenue (2022 Edition), January 2022. www.https://carsurance.net/insights/lyft-statistics/. Accessed February, 2022.

Cats, O., Kucharski, R., Danda, S., and Yap, M. (2021), Beyond the Dichotomy: How Ride-hailing Competes with and Complements Public Transport. *arXiv preprint arXiv:2104.04208*.

Cheikh, B., Tahon, C., and Hammadi, S. (2017), An evolutionary approach to solve the dynamic multihop ridematching problem. *Simulation*, 93(1), 3-19.

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

Coltin, B., and Veloso, M. (2014), Ridesharing with passenger transfers. *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*.

Corana, A., Marchesi, M., Martini, C., and Ridella, S. (1987), Minimizing multimodal functions of continuous variables with the simulated annealing algorithm. *ACM Transactions on Mathematical Software (TOMS)*, 13(3). 262-280.

Cordeau, F. (2006), A branch-and-cut algorithm for the dial-a-ride problem. *Operations Research*, 54(3), 573-586.

Cordeau, F., Laporte, G., Savelsbergh, M., and Vigo, D. (2007), Vehicle routing. *Handbooks in operations research and management science*, 367-428.

Cortes, E., Matamala, M., and Contardo, C. (2010), The pickup and delivery problem with transfers: Formulation and a branch-and-cut solution method, *European Journal of Operational Research*, 200(3), 711-724.

Coslovich, L., Pesenti, R., and Ukovich, W. (2006), A two-phase insertion technique of unexpected customers for a dynamic dial-a-ride problem. *European Journal of Operational Research*, 175(3), 1605-1615.

Cozza, J. (2019), The History of Carpooling. www.shareable.net/the-history-of-carpooling-from-jitneys-to-ridesharing. Accessed July 22, 2019.

Curry, D. (2022), DoorDash Revenue and Usage Statistics (2022), https://www.businessofapps.com/data/doordash-statistics/. Accessed January 23, 2022.

Dantzig, G., and Ramser, J. (1959), The truck dispatching problem. *Management Science*, 80-91.

Desaulniers, G., Desrosiers, J., and Solomon, M. (2006), *Column generation*. Vol. 5. Springer Science and Business Media.

Desaulniers, G., Madsen, O., and Ropke, S. (2014), The vehicle routing problem with time windows. *Vehicle routing: Problems, methods, and applications*, 18(1), 119-159.

Desrochers, M., Desrosiers, J., and Solomon, M. (1992), A new optimization algorithm for the vehicle routing problem with time windows. *Operations research*, 40(2), 342-354.

Dixit, A., Mishra, A., and Shukla, A. (2019), Vehicle routing problem with time windows using meta-heuristic algorithms: a survey. *Harmony Search and Nature Inspired Optimization Algorithms*, 539-546.

Dumas, Y., Desrosiers, J., and Soumis. F. (1991), The pickup and delivery problem with time windows. *European Journal of Operations Research*, 54(1), 7-22.

DuPuis, N., Martin, C., and Rainwater, B. (2015), *City of the Future: Technology and Mobility*, Center for City Solutions and Applied Research, National League of Cities.

Eilon, S., Watson-Gandy, C., and Christofides, N. (1971), Distribution Management. *Mathematical Modelling and Practical.* Griffin, London.

Eksioglu, B., Vuralb, A., and Reismanc, A. (2009), The vehicle routing problem: A taxonomic review. *Computers and Industrial Engineering*, 57(4), 1472-1483.

Elliott, C. (2017), Mass transit gets boost from ridesharing, https://usatoday.com/story/travel/advice/2017/01/22/uber-lyft-ridesharing-public-transportation/96830100. Accessed April 30, 2018.

Fiedler, D., Čertický, M., Alonso-Mora, J., and Čáp, M. (2018), The impact of ridesharing in mobility-on-demand systems: Simulation case study in Prague. *21st International Conference on Intelligent Transportation Systems (ITSC),* 1173-1178.

Fischer, B. (2017), In Uber vs. taxi cab fight, expense reports offer telling barometer, www.bizjournals.com/newyork/blog/techflash/2015/04/uber-taxi-expense-report-certify-study.html. Accessed June 30, 2017.

Fisher, M., and Jaikumar, R., (1978), A decomposition algorithm for large scale vehicle routing, working paper 78-11-05, Wharton Department of Decision Science, University of Pennsylvania.

Fugenschuh, A. (2009), Solving a school bus scheduling problem with integer programming. *European Journal of Operational Research*, 193(3), 867-884.

Furuhata, M., Dessouky, M., Ordóñez, F., Brunet, M., Wang, X., and Koenig, S. (2013), Ridesharing: The state-of-the-art and future directions. *Transportation Research Part B: Methodological*, 57(1), 28-46.

Gendreau, M., Guertin, F., Potvin, Y., and Séguin, R. (2006), Neighborhood search heuristics for a dynamic vehicle dispatching problem with pickups and deliveries. *Transportation Research Part C: Emerging Technologies*, 14(3), 157-174.

Giuliano, G., Levine, D., and Teal, R. (1990), Impact of high occupancy vehicle lanes on carpooling behavior, *Transportation*, 17(2), 159-177.

Glover, F. (1986), Future Paths for Integer Programming and Links to Artificial Intelligence. *Computers and Operations Research*, 13(5): 533–549.

Goldberg, D. (1989), Genetic Algorithms in Search, Optimization, and Machine Learning. *Reading*.

Golden, B., Raghavan, S., and Wasil, E. (2008), The vehicle routing problem: latest advances and new challenges. *Operations Research/Computer Science Interface Series*, 43.

Gooch, J. (2018), Masabi and Uber announce first-of-its-kind ridesharing and public transit ticketing partnership, http://masabi.com/2018/04/11/masabi-and-uber-announce-first-of-its-kind-ride-sharing-and-public-transit-ticketing-partnership/. Accessed April 30, 2018.

Herbawi, W., and Weber, M. (2011), Evolutionary multi-objective route planning in dynamic multi-hop ridesharing, *European conference on evolutionary computation in combinatorial optimization*, 2011. Springer, Berlin, Heidelberg.

Ho, S., Szeto, W., Kuo, Y., Leung, J., Petering, M., and Tou, T. (2018), A survey of dial-a-ride problems: Literature review and recent developments. *Transportation Research Part B: Methodological*, *111*, 395-421.

Hosni, H., Naoum-Sawaya, J., and Artail, H. (2014), The shared-taxi problem: Formulation and solution methods. *Transportation Research Part B: Methodological*, 70(1), 303-318.

Hosny, M., and Mumford, C. (2010), The single vehicle pickup and delivery problem with time windows: intelligent operators for heuristic and metaheuristic algorithms, *Journal of Heuristics*, 16(1), 417-439.

Hou, Y., Li, X., and Qiao, C. (2012), TicTac: From transfer-incapable carpooling to transfer-allowed carpooling. In *2012 IEEE Global Communications Conference (GLOBECOM)*, 268-273.

Huang, C., Zhang, D., Si, Y., and Leung, S. (2016), Tabu search for the real-world carpooling problem. *Journal of Combinatorial Optimization*, 32(2), 492-512.

Hyland, M., and Mahmassani, H. (2018), Dynamic autonomous vehicle fleet operations: Optimization-based strategies to assign AVs to immediate traveler demand requests. *Transportation Research Part C: Emerging Technologies*, 92, 278-297.

Iqbal, M. (2022), Uber Revenue and Usage Statistics. www.businessofapps.com/data/uber-statistics. Accessed February 6, 2022.

Jaw, J., Odoni, A., Psaraftis, H., and Wilson, N. (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.

Jorgensen, R., Larsen, J., and Bergvinsdottir, K. (2007), Solving the dial-a-ride problem using genetic algorithms. *Journal of the operational research society*, 58(10), 1321-1331.

Keeney, T. (2017), The future of transport is autonomous Mobility-as-a-Service, https://ark-invest.com/research/autonomous-mobility-as-a-service. Accessed April 30, 2018.

Kerivin, H., Lacroix, M., Mahjoub, A., and Quilliot, A. (2008), The splittable pickup and delivery problem with reloads. *European Journal of Industrial Engineering*, 2(2), 112-133.

Keyes, D. (2017), E-Commerce will make up 17% of all US retail sales by 2022 and one company is the main reason, www.businessinsider.com/e-commerce-retail-sales-2022-amazon-2017-8. Accessed April 30, 2018.

Kirchler, D., and Calvo, R. (2013), A granular tabu search algorithm for the dial-a-ride problem. *Transportation Research Part B: Methodological*, 56(1), 120-135.

Klapp, M., Erera, A., and Toriello, A. (2016), The one-dimensional dynamic dispatch waves problem. *Transportation Science*, 52(2), 402-415.

Kohl, A. (2018), Online Food Delivery, https:/statista.com/outlook/374/109/online-food-delivery/united-states. Accessed October 30, 2018.

Laporte, G., Mercure, H., and Nobert, Y. (1986). An exact algorithm for the asymmetrical capacitated vehicle routing problem. *Networks*, 16, 33-46.

Laporte, G. (2009), Fifty years of vehicle routing. *Transportation Science* 43(4), 408-416.

Li, L., Chen, Y., and Meng, J. (2020). Improved Tabu search algorithm for solving the vehicle routing problem with soft time windows in B2C environment. *In 2020 Chinese Control And Decision Conference (CCDC),* 3629-3633.

Liang, X., Corriea, G., An, K., and Arem, B. (2019), Automated taxis' dial-a-ride problem with ridesharing considering congestion-based dynamic travel times. *Transportation Research Part C: Emerging Technologies*, 112, 260-281.

Lienert, P., Somerville, H., and Sage, A. (2016), Silicon Valley VCs are growing wary of on-demand delivery, https://reuters.com/article/us-delivery-startups-analysis/silicon-valley-vcs-are-growing-wary-of-on-demand-delivery. Accessed April 30, 2018.

Lotfi, S., Abdelghany, K., and Hashemi, H., (2019), Modeling Framework and Decomposition Scheme for On-Demand Mobility Services with Ridesharing and Transfer, *Computer-Aided Civil and Infrastructure Engineering*, 34(1), 21-37.

Lu, W., Liu, L., Wang, F., Zhou, X., and Hu, G. (2021). Two-phase optimization model for ride-sharing with transfers in short-notice evacuations. *Transportation research part C: emerging technologies*, 114, 272-296.

Ma, A., Urbanek, M., Pardo, M., Chow, J., and Lai, X. (2017), Spatial welfare effects of shared taxi operating policies for first mile airport access. *International Journal of Transportation Science and Technology*.

Ma, T., Rasulkhani, S., Chow, J., and Klein, S. (2019), A dynamic ridesharing dispatch and idle vehicle repositioning strategy with integrated transit transfers. *Transportation Research Part E: Logistics and Transportation Review*, *128*, 417-442.

Mahmoudi, M., and Zhou, X. (2016), Finding optimal solutions for vehicle routing problem with pickup and delivery services with time windows: A dynamic programming approach based on state–space–time network representations. *Transportation Research Part B: Methodological*, 89, 19-42.

Masson, R., Lehuédé, F., and Péton, O. (2013), An adaptive large neighborhood search for the pickup and delivery problem with transfers. *Transportation Science*, 47(3), 344-355.

Masson, R., Ropke, S., Lehuédé, F., and Péton, O. (2014), A branch-and-cut-and-price approach for the pickup and delivery problem with shuttle routes. *European Journal of Operational Research*, 236(3), 849-862.

Masoud, N., and Jayakrishnan, R. (2017), A decomposition algorithm to solve the multi-hop Peer-to-Peer ride matching problem. *Transportation Research Part B: Methodological*, 99, 1-29.

Mitrović-Minić, S., and Laporte, G. (2006), The pickup and delivery problem with time windows and transshipment. *INFOR: Information Systems and Operational Research*, 44(3), 217-227.

Mazareanu, E. (2021), U.S. courier, express and parcel market - statistics & facts, https://www.statista.com/topics/4063/courier-express-and-parcels-cep-market-in-the-united-states/#topicHeader__wrapper. Accessed January 23, 2022.

Nam, D., Yang, D., An, S., Yu, G., Jayakrishnan, R., and Masoud, N. (2018), Designing a Transit-Feeder System using Multiple Sustainable Modes: Peer-to-Peer (P2P) Ridesharing, Bike Sharing, and Walking. *Transportation Research Record*, 2672(8), 754-763.

Narayan, J., Cats, O., Van Oort, N., and Hoogendoorn, S. (2020), Integrated route choice and assignment model for fixed and flexible public transport systems. *Transportation Research Part C: Emerging Technologies*, 115, 102631.

North Central Texas Council of Government, Transportation Department. (2009), Dallas-Fort Worth regional travel model: Model Description.

Oh, S., Lentzakis, A., Seshadri, R., and Ben-Akiva, M., (2021), Impacts of Automated Mobility-on-Demand on traffic dynamics, energy and emissions: A case study of Singapore. *Simulation Modelling Practice and Theory*, 102327.

Parragh, S., Doerner, K., and Hartl, F. (2008), A survey on pickup and delivery problems. *Journal für Betriebswirtschaft*, 58(1), 21-51.

Parragh, S., Doerner, K., and Hartl, R. (2008), A survey on pickup and delivery problems, part ii: Transportation between pickup and delivery locations. *Journal für Betriebswirtschaft*, 58(2), 81-117.

Parragh, S., Doerner, K., and Hartl, R. (2010), Variable neighborhood search for the dial-a-ride problem. *Computers and Operations Research*, 37(6), 1129-1138.

Parragh, S., and Schmid, V. (2013), Hybrid column generation and large neighborhood search for the dial-a-ride problem. *Computers and Operations Research*, 40(1), 490-497.

Psaraftis, H. (1980), A dynamic programming solution to the single vehicle many-to-many immediate request dial-a-ride problem. *Transportation Science*, 14(2), 130-154.

Psaraftis, H. (1983), An exact algorithm for the single vehicle many-to-many dial-a-ride problem with time windows. *Transportation Science*, 17(3), 351-357.

Psaraftis, H. (1988), Vehicle routing: Methods and studies. *Dynamic Vehicle Routing Problems*. North Holland, Amsterdam, Netherlands, 223-248.

Qu, Y., and Bard, J. (2012), A GRASP with adaptive large neighborhood search for pickup and delivery problems with transshipment. *Computers and Operations Research*, 39(10), 2439-2456.

Reimann, M., Doerner, K., and Hartl, R. (2004). D-Ants: Savings Based Ants divide and conquer the vehicle routing problem. *Computers and Operations Research*, 31(4), 563-591.

Ropke, S., Cordeau, J., and Laporte G. (2007), Models and branch-and-cut algorithms for pickup and delivery problems with time windows. *Networks*, 49(4), 258-272.

Santi, P., Resta, G., Szell, M., Sobolevsky, S., Strogatz, S., and Ratti, C. (2014), Quantifying the benefits of vehicle pooling with shareability networks. *Proceedings of the National Academy of Sciences*, 111(37), 13290-13294.

Schilde, M., Doerner, K., and Hartl, R. (2011), Metaheuristics for the dynamic stochastic dial-a-ride problem with expected return transports. *Computers and operations research*, 38(12), 1719-1730.

Schrock, S. (2017), Via, city of Arlington launch innovative, on-demand rideshare pilot program, https:// arlington-tx.gov/news/2017/12/11/via-city-arlington-launch-innovative-demand-rideshare-pilot-program/. Accessed April 30, 2018.

Sexton, T., and Bodin, L. (1985), Optimizing single vehicle many-to-many operations with desired delivery times: I. Scheduling. *Transportation Science,* 19(4), 378-410.

Sexton, T., and Bodin, L. (1985), Optimizing single vehicle many-to-many operations with desired delivery times: II. Routing. *Transportation Science*, 19(4), 411-435.

Shaheen, S., Bell, C., Cohen, A., and Yelchuru, B. (2017), *Travel Behavior: Shared Mobility and Transportation Equity,* No. PL-18-007.

Shieber, J. (2017), Target is buying Alabama-based, same-day delivery service Shipt for $550 million, https://techcrunch.com/2017/12/13/target-is-buying-alabama-based-same-day-delivery-service-shipt-for-550-million/. Accessed April 30, 2018.

Singh, A., Alabbasi, A., and Aggarwal, V. (2019), A Distributed Model-Free Algorithm for Multi-hop Ride-sharing using Deep Reinforcement Learning. *arXiv preprint arXiv:1910.14002*.

Soza-Parra, J., Kucharski, R., and Cats, O. (2021), Ride-pooling potential under alternative spatial demand patterns. *arXiv preprint arXiv:2104.04209*.

Tirachini, A., Chaniotakis, E., Abouelela, M., and Antoniou, C. (2020). The sustainability of shared mobility: Can a platform for shared rides reduce motorized traffic in cities? *Transportation Research Part C: Emerging Technologies*, 117, 102707.

Tong, L., Zhou, L., Liu, J., and Zhou, X. (2017). Customized bus service design for jointly optimizing passenger-to-vehicle assignment and vehicle routing. *Transportation Research Part C: Emerging Technologies*, *85*, 451-475.

Toro, O., Eliana, M., Escobar, Z., Antonio, H., and Granada, E. (2016). Literature review on the vehicle routing problem in the green transportation context. *Luna Azul*, (42), 362-387.

Ulmer, W., Barrett, W., and Dirk, C. (2016), Preemptive depot returns for a dynamic same-day delivery problem. Technical Report, Technische UniversitÃ; t Braunschweig, 1-39.

Ulmer, W., Mattfeld, C., and Köster, F. (2017), Budgeting time for dynamic vehicle routing with stochastic customer requests. *Transportation Science*, 52(1), 20-37.

Voccia, A., Campbell, A., and Thomas, B. (2017), The same-day delivery problem for online purchases. *Transportation Science*, 12(1), 42-52.

Vidal, T., Laporte, G., and Matl, P. (2019). A concise guide to existing and emerging vehicle routing problem variants. *European Journal of Operational Research*.

Wark, P., and Holt, J. (1994). A Repeated Matching Heuristic for the Vehicle Routing Problem. *Journal of the Operational Research Society*, 45(10), 1156-1167.