Title
Planning and Operation of a Crowdsourced Package Delivery System: Models, Algorithms and Applications

Author
Yang, Dingtong

Publication Date
2021

UNIVERSITY OF CALIFORNIA, IRVINE

Planning and Operation of a Crowdsourced Package Delivery System: Models, Algorithms and Applications

DISSERTATION

submitted in partial satisfaction of the requirements for the degree of

DOCTOR OF PHILOSOPHY

in Civil and Environment Engineering

by

Dingtong Yang

Dissertation Committee:

Assistant Professor Michael F. Hyland, Chair

Professor R. Jayakrishnan

Professor Stephen Ritchie

2021

DEDICATION

To my parents, Prof. Weiping Yang and Ms. Min Ding.

To my grandmother, Ms. Yulan Ding

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

At the very beginning, I would like to thank my parents for raising me, educating me and supporting me for this PhD study. I would also like to thank my grandmother, a middle school biology teacher, for cultivating my interest in natural science in my early childhood.

I would like to express my thank to my chair supervisor, Prof. Michael F. Hyland. I should have asked him the full form of “F” before writing this acknowledgement, but here I will take it as “fantastic” since he is more than a fantastic advisor. He is a great mentor and a good friend. Mike started to supervise me in my fourth year of PhD. It was also the first year that he came to UCI as an assistant professor. I really appreciate his help in research questions discussion, financial support and paper writing.

I would also like to express my thank to my co-advisor, Prof. Radhakrishnan Jayakrishnan Nair. He is a smart free-thinker and cares about students. I like his way of teaching and appreciate his help in multiple aspects throughout my PhD study.

I would like to thank my MPhil supervisor Dr. Tsz Leung Yip. He is the one who guide me to start my career in research. Without his help and encouragement, I could not be a PhD today.

I would like to thank Prof. Stephen Ritchie for being my research committee member and dedicating his energy to the entire UCI ITS as the director. I also need to thank both Prof. Wilfred Recker and Prof. John Tuner for their teaching and being my qualifying exam committee members. I would also thank Prof. Wen-Long Jin, Prof. Rick So and Prof. Jan Brueckner for their teaching and help in my PhD study.

I could not survive my PhD study without the support of my friends. On the top of the list, I would like to thank Dr. Dongshu Liu, Chang Liu, Enguo Zhang and Dr. Rongheng Li for their help in different issues in these years. I would like to thank my friend Navjyoth Jayashanka Sarma Shoba for coopering in different research projects with me.

My colleagues from the research group also provided a lot of help and I would like to acknowledge their contributions here. I would like to thank my senior students Dr. Neda Masoud, Dr. Roger Lloret, Dr. Jiangbo Yu, Dr. Daisik Nam for helping me in my study and research. I would also like to thank my colleagues Sunghi An, Eduardo Malino, Marjan Mosslemi, Pengyuan Sun and Negin Shariat for their support in different research projects.

I would also like to thank the support of the Civil Engineering Department and the Institute of Transportation Studies. I would like to thank Dr. Zhe Jared Sun and Dr. Andre Tok for their support in facility management. I would like to thank my friends in the department for spending time together. I would like to thank Dr. Yiqiao Li, Koti Reddy Allu, Yue Yu, Dr. Youngeun Bae, Lu Xu (and her husband Dr. Jin Yang), Arash Ghaffar, the Casebolt family (Chenying, Brian and little Thomas), Ximeng Fan, Guoliang Feng and Siwei Hu. I would also like to thank my senior students who graduated and left UCI. They are Dr. Xuting Wang, Dr. Qinglong Yan, Dr. Sheng-Hsiang Peng, Dr. Suman Kumar Mitra (the first person I know in UCI), Dr. Felipe de Souza, Dr. Karina Hermawan, Dr. Xiaoxia Shi, Dr. Jun Hyeong Park, Dr. Yunwen Feng and Dr. Riju Lavanya. Wish them all the best after PhD.

I would like to express my sincere thanks to Shirley Hsiao and Gabriella Marquez from Long Beach Transit. Thanks for your support in research projects and my intern. I would also thank my friends who provide a lot of remote love and help. I would like to thank Danfeng Guo and Dr. Di Wu for helping me in different computer science and machine learning questions. I would like to thank Feier Chen, Yuhang Lai and Chenge Jia for discussing different issues about life and brings me a lot of joy. I would also like to thank my cousin, Yanni Liu, for her support during these years. I need to thank my friend Dr. Xun Tong, Dr. Yefei Yang and Dr. Sida Luo for their help and support. Wish all of them the best in life.

Finally, I would like to thank myself for making the decision of pursuing a PhD in life. It is always enjoyable for me to learn knowledge. If I were not a PhD in transportation engineering, I would have attempted for linguistics.

-Written on the Thanksgiving Day of 2021 in Irvine-

CURRICULUM VITAE

Dingtong Yang

EDUCATION

PhD in Civil and Environmental Engineering 2021
University of California, Irvine *Irvine, California, USA*

MPhil in Business Administration 2016
The Hong Kong Polytechnic University *Hong Kong SAR, China*

MSc in Industrial Engineering and Logistics Management 2013
The University of Hong Kong *Hong Kong SAR, China*

BBA in Supply Chain Management *(First Class Honors)* 2012
The Hong Kong Polytechnic University *Hong Kong SAR, China*

REFERRED JOURNAL PUBLICATIONS

Dynamic modeling and real-time management of a system of EV fast-charging stations 2021
Transportation Research Part C
D Yang, NJS Sarma, MF Hyland, R Jayakrishnan

Eco-driving algorithm with a moving bottleneck on a single-lane road 2020
P Sun, D Yang, WL Jin *Transportation Research Record*

Designing a transit-feeder system using multiple sustainable modes: Peer-to-peer (P2P) ridesharing, bike sharing, and walking 2018
Transportation Research Record
D Nam, D Yang, S An, JG Yu, R Jayakrishnan, N Masoud

REFERRED CONFERENCE PROCEEDINGS

Effective and Efficient Fleet Dispatching Strategies for Dynamically Matching AVs to Travelers in Large-scale Transportation Systems — 2021, *IEEE 23rd ITSC*

NJS Sarma, D Nam, MF Hyland, F de Souza, D Yang, A Ghaffar, I Verbas

A Practical Data-Driven Approach to Analyze Inter-agency Passenger Transfer with GTFS Data: A Case Study of the City of Long Beach — 2019, *TRB 98th Annual Meeting*

D Yang, R Jayakrishnan

An Implementation-Ready Approach for Multiple-Van Multi-Criteria Dynamic Demand Rebalancing at Bike-Share Stations — 2018, *TRB 97th Annual Meeting*

JG Yu, D Yang, D Nam, S An, R Jayakrishnan

ABSTRACT OF THE DISSERTATION

Planning and Operation of a Crowdsourced Package Delivery System:

Models, Algorithms and Applications

by

Dingtong Yang

Doctor of Philosophy in Civil and Environmental Engineering

University of California, Irvine, 2021

Professor Michael F. Hyland, Chair

Online shopping has increased steadily over the past decade that has led to a dramatic increase in the demand for urban package deliveries. Crowdsourced delivery, or crowd shipping, has been proposed and implemented by logistics companies in response to the growth in package delivery business. Crowdsourced delivery is a delivery service in which logistics service providers contract delivery services from the public (i.e., non-employees), instead of providing delivery services exclusively with an in-house logistics workforce.

This dissertation studies different types of urban last-mile crowdsourced delivery services and provides a taxonomy for crowdsourced package delivery. Urban package crowdsourced delivery can be categorized in terms of the way packages are delivered and the role/tasks of crowdsourced drivers. Given these two dimensions, this study identifies three types of urban package crowdsourced delivery, namely, crowdsourced time-based delivery, crowdsourced trip-based delivery, and crowdsourced shared-trip delivery. Crowdsourced time-based delivery drivers are paid for their idle time and work as sub-contractors. Crowdsourced trip-based delivery matches drivers with individual tasks and utilizes the drivers for specific delivery trips. The last type,

crowdsourced shared-trip delivery utilizes the common segments of a crowdsourced personal vehicle trip to deliver packages. In this type, the package shares part of the driver's trip.

The literature formulates the crowdsourced delivery problem as a Vehicle Routing Problem (VRP) and proposes a variety of solution approaches. However, all the solution algorithms are limited to relatively small-scale problems. In addition, the factors that impact the efficiency and effectiveness of crowdsourced delivery have not been thoroughly analyzed. To bridge the gap in crowdsourced delivery and urban freight logistics, this dissertation provides an alternative formulation for the static crowdsourced shared-trip delivery problem and proposes a novel decomposition heuristic to solve the problem.

The alternative formulation is based on the set partitioning problem. The novel decomposition heuristic handles packages that are served by shared personal vehicles (SPVs) and dedicated vehicles (DVs), separately. After that, the algorithm deploys a package switch procedure, which rearranges packages between SPVs and DVs. The dissertation discusses various algorithms employed to solve different sub-problems, such as the *budgeted k-shortest path, large scale bi-partite matching, decision of package switching,* and *vehicle routing.*

To validate the models and algorithms, this dissertation presents a numerical case study that uses the network of the City of Irvine, CA, USA. The results of the numerical study unveil interesting results that are valuable to both researchers and industrial practitioners. The results indicate that crowdsourced shared-trip delivery service can reduce total delivery costs by between 20% to 50%, compared to a delivery service that exclusively uses its own dedicated vehicles and drivers. However, the results show that dedicated vehicles are still required since the shared vehicles are not able to serve all packages even with a considerably large set of candidate shared vehicles. Vehicle Miles Traveled (VMT) savings depend on the crowdsourced driver selection and their trip origins. The dissertation also analyzes and discusses important factors that impact the effectiveness of crowdsourced delivery. In particular, the dissertation includes sensitivity analysis results with respect to changes in the depot location and the willingness of shared vehicles to detour.

Chapter 1 Introduction

1.1 Overview of crowdsourced freight delivery

This dissertation explores a novel way of conducting urban freight delivery service, crowdsourced delivery, which is sometimes named as “crowd shipping” or “delivery with ad hoc drivers”. The term “crowdsourced” indicates that vehicles, and more importantly, driver-hours are sourced from the general public.

Crowdsourcing might not be a brand-new idea in the field of transportation studies and is definitely not a new concept in the human history. Two hundred years ago, in the financial industry, this idea was applied for mutual fund raising and insurance purchasing. Moreover, when the website Wikipedia was first launched in 2001, the website sourced information and knowledge from the general public.

In the transportation service sector, the concept of crowdsourcing become prominent with the introduction and growth of transportation network companies (TNCs). TNCs source passenger transportation service to essentially anyone with a vehicle and a smartphone. TNC services are similar to traditional carpooling and ride-sharing services wherein, in the latter services, a driver utilizes the extra capacity (i.e., seats) in their vehicle to transport passengers whose trips originate and terminate at locations that do not require the driver to detour too much from their original route. However, while traditional ride-sharing and carpooling services often require drivers and riders to agree on pickup location and pickup time at least an hour or two before the driver began their trip, the ubiquity of smartphones allows ride-sharing services to dynamically match potential riders and drivers who are traveling in a similar direction. This dissertation extends the concept of dynamic ride-sharing for people to ride-sharing or ride-crowdsourcing for parcels.

Crowdsourced delivery or crowd-shipping has been defined in various ways by researchers. Punel & Stathopoulos (2017) define crowd-shipping as "*a goods delivery service that is outsourced to occasional carriers drawn from the public of private travelers and is coordinated by a technical platform to achieve benefits for the involved stakeholders*." Rai et al. (2017) defines crowd-shipping as *"(an informative connectivity enabled concept that) matches supply and demand for logistics services with an undefined and external crowd that has free capacity with regards to time and/or space, participates on a voluntary basis and is compensated accordingly*".

From these definitions, the dissertation summarizes three key factors associated with crowdsourced delivery:

1. Supply of mobility (shipping/delivering source): undefined/external/general public/crowd
2. Matching of demand and supply: Either smart algorithms considering time and space constraints or spontaneous search and bargain on a platform.
3. Medium of information exchange (Platform of ordering and matching): information technology supported platforms (Mobile phone applications/mobile apps, websites)

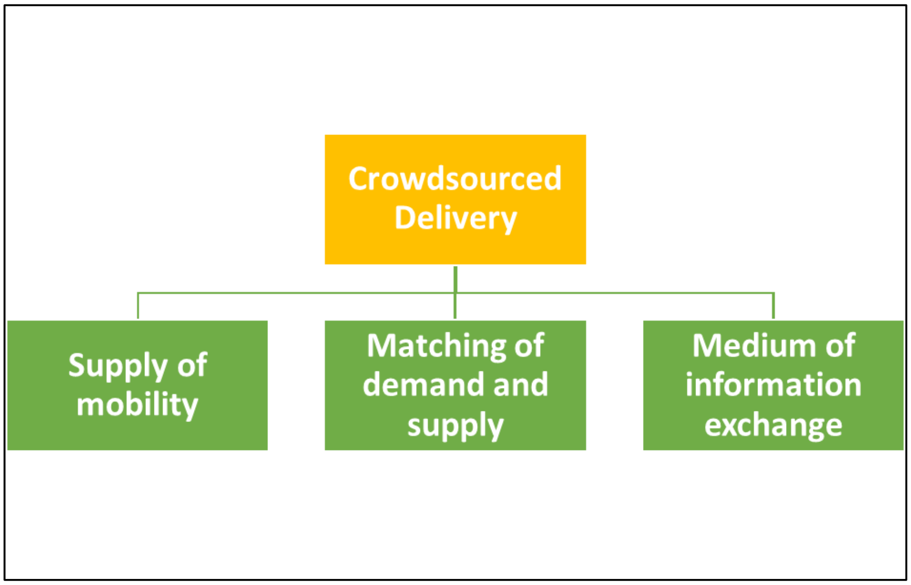

Figure 1.1 Three Key Factors of Crowdsourced Delivery

The three factors represent the information flow and goods flow in the crowdsourced delivery process. In the following chapters, when analyzing a crowdsourced delivery case, the dissertation applies the three-factor definition. Out of the three, the first factor is the most important one. Without sufficient supply of crowdsourced drivers, the entire system effectiveness and efficiency becomes severely degraded. The second factor requires an algorithm to determine eligibility and feasibility as well as perform the matching. The third factor works as an infrastructure for this type of service, it provides timely information to both the demand and supply parties.

Following the three-factor definitions, the dissertation lists out several crowdsourced delivery examples:

1. Walmart asks in-store customers to delivery packages for online order customers.
2. Amazon asks drivers with idle time to sign up for a short-term package delivery task.
3. Roadie seeks drivers for inter-state package delivery.
4. Uber Eats seeks drivers for meal delivery.

There are many other examples of crowdsourced delivery. If the mode of service is not restricted to road/land vehicles, asking people that are flying to another city to carry goods/items is also a type of crowdsource delivery. To categorize the service, the dissertation first groups the service by travel distance. Crowdsource delivery can be done at both inter-city level (long distance) and intra-city level (short distance). The majority of crowdsourced delivery these days involves intra-city travel, since most deliveries include food, groceries and emergency document same-day delivery.

For intra-city delivery, the supply of mobility mainly involves city level commuter trips or people's daily shopping trips. The matching of demand and supply usually relies on smart algorithms that consider spatial and temporal dimensions of vehicles and packages. In this type of matching, usually the drivers cannot choose a specific item that they would like to deliver, rather, they follow the results of matching algorithms. The only option that the drivers may have is to accept or reject the task order. The third factor is a smartphone app, where the end-customer orders and the driver receive delivery information. Meal, grocery, and one-day delivery of packages and documents all belong to this category.

1.2 Motivations of the dissertation

Then the next question to ask is: *why freight sharing is needed in our society*? One may understand this question from the most basic two factors of modern economics, demand and supply.

In recent decade, the demand for package delivery and reverse logistics services has increased significantly due to online shopping. E-commerce in the U.S grew at a rate of 16% from 2017 to 2018 (Figure 1.2, (Statista, 2021)); this growth rate would double packages delivery volume every 5 years at (Ivanov, 2018). However, the COVID pandemic, which rooted people indoor, resulted in a huge increase in demand for package delivery to households (Bhatti et al., 2020). Ultimately, package delivery requests need to be delivered to households and businesses and doing so requires a large number of trucks, vans or other last mile delivery vehicles.

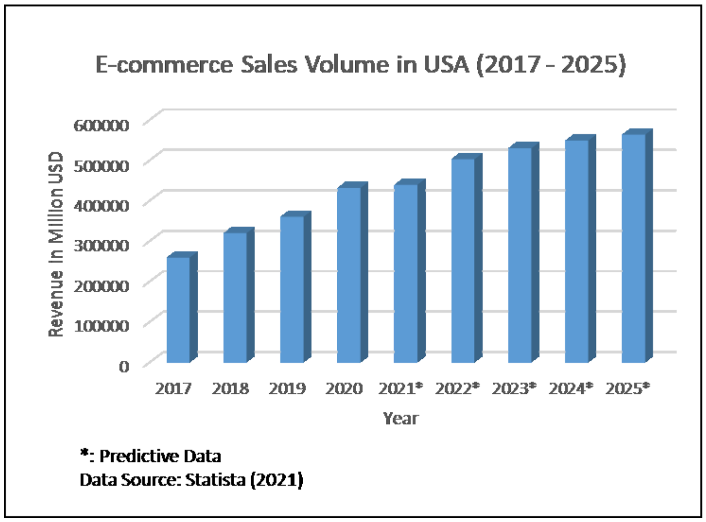

Figure 1.2 E-commerce Volume in USA (2017 – 2025)

On one hand, logistics companies, who are the major service providers of freight delivery, are facing challenges related to distributing large volumes of parcels and they need to make planning decisions on future facility investments and their vehicle fleet based on uncertainty about future demand. Given the uncertainty about future demand, logistics companies are looking for alternatives to making large capital investments in facilities and vehicle fleets, and crowdsourcing has emerged as a flexible option.

On the other hand, from the social perspective, additional package deliveries may result in increasing vehicle trips. Growth in vehicle trips for pickup and delivery in urban/suburban areas presents the following major challenges to our society.

1. Additional vehicle trips will contribute to traffic congestion. Additional vehicle trips and the traffic congestion they generate may lead to more energy consumption and carbon emissions.
2. While handling parcels, delivery vehicles require parking space, such as curb space, loading zones, and alleys, which are already of scarcity in urban areas. Ivanov (2018) show that Commercial Vehicle Load Zones (CVLZs) almost reach full occupancy during workdays in Seattle.

Designing new logistics systems in urban areas can both save cost for logistics companies and reduce carbon emissions (Huang et al., 2018). Currently, most packages or freight items are delivered by designated vehicles and personnel. However, other alternatives based on crowdsourcing deliveries are possible.

Large numbers of people travel everyday with idle space in their personal vehicles. The empty space can be leveraged for freight delivery. If these people do not have to detour far to pick up and deliver parcels to houses or businesses, then the cost to logistics providers (or shippers) are likely to be low and the extra vehicle trips, mileage, and congestion from increased package delivery are likely to be low. Hence, this form of crowdsourced package delivery is an option that could benefit both logistics companies and the society.

Benefits of crowdsourced delivery include improved economic, social, and environmental sustainability (Rai et al., 2017). From an economic perspective, first, for logistic companies, crowdsource delivery helps reduce the fleet size and dedicated human resources involved in delivery. As a result, crowdsourcing can reduce capital and labor cost for logistics company (Qi et al., 2018). From a social perspective, crowdsourced delivery could potentially lower traffic generated from freight delivery, which reduces congestion and improves transportation efficiency (Archetti et al., 2016; Arslan et al., 2019). Crowdsourced delivery also enhances the opportunity for same-day delivery, including both packages and food (Ulmer, Thomas, Campbell, & Woyak, 2021; Voccia et al., 2019), which improves quality-of-life in urban areas. Last but not least, from an environmental perspective, reducing freight delivery traffic can decrease fuel consumption and harmful emissions (S. Lee et al., 2016a).

Motivated by the needs of establishing a new urban freight delivery system and the benefit that the crowdsourced delivery system can bring, this dissertation attempts to develop a modeling and analysis framework to improve understanding of and evaluate crowdsourced delivery systems. This dissertation reviews the current practice and studies related to crowdsource delivery, while at the same time, creates a new algorithm which is capable of solving crowdsourced problem in larger scale scenarios.

1.3 Problem statements, research questions and contributions

Previous research related to crowdsourced logistics has been wide-ranging in terms of research methodology. This section reviews crowdsourced logistics research that employ (i) empirical methods to model crowdsourced delivery behavior and demand (Punel et al., 2018; J. Rougès & Montreuil, 2014), (ii) optimization methods to model, design, and analyze crowdsourced logistics systems/services (Archetti et al., 2016; Arslan et al., 2019), and (iii) other methods including analytical models and simulations (P. Chen & Chankov, 2018).

Previous studies formulate the crowdsourced delivery under various assumptions. These assumptions jeopardize the generality of modelling a crowdsourced delivery problem. Besides, the problem instance sizes addressed in literature are relatively small, and in some real-world scenarios, existing models and solution algorithms are hard to apply. In addition, the details of crowdsourced delivery operation are not fully understood. *To bridge the gap and improve understanding of and knowledge related to crowdsourced delivery, this dissertation aims to provide a general formulation of the crowdsourced delivery problem, design a novel solution algorithm that is capable of solving real-world scale problems, and reveal critical service design factors and model parameters that impact the operation of crowdsourced delivery.*

Additionally, this dissertation aims to answer research questions covering both planning level and operational level concerns. Addressing these research questions, as this dissertation does, should provide valuable information for both academia and industry.

(1) Planning Questions:

- What vehicles could be utilized for crowdsourced delivery?
- Are dedicated vehicles still required for logistics companies?
- How many dedicated vehicles should be reserved for one distribution center?
- What are the potential savings of crowdsourced delivery in terms of monetized cost, vehicle miles travelled (VMT), and emissions?

(2) Operational Questions:

- What packages should be handled by shared vehicles?
- What packages should be handled by dedicated vehicles?
- How many incentives should be offered to trip-sharing (i.e., crowdsourcing) drivers?
- How will time window and capacity constraints impact the performance of a combined dedicated vehicle and shared vehicle fleet?
- How will time window and capacity constraints be included and considered in the problems?

The major contributions of this dissertation include the following. First, the dissertation provides a comprehensive overview of the different types of crowdsourced delivery systems and summarizes the unique features of each type. Second, this dissertation provides a comprehensive review of the literature related to the development of crowdsourced delivery systems. Third, the dissertation proposes an urban freight delivery system that combines shared-use and dedicated vehicles (crowdsourced shared-trip delivery). Specifically, the proposed delivery system leverages the considerably large volume of trips made each day by persons in their personal vehicles for package delivery in urban areas. Even if a small percentage of these vehicles opt into a crowdsourcing delivery system, the shared personal vehicles (SPVs) can deliver a large volume of packages with minimal detour time, distances, or cost. Fourth, this study models the proposed urban freight delivery system and the underlying operational problem as both a mixed integer program based on the Vehicle Routing Problem (VRP) and a set covering problem, while summarizing and comparing the solution techniques that could be used. Fifth, based on the set covering formulation, the paper introduces a decomposition heuristic that is capable of solving relatively large-scale problem instances. In addition, the dissertation, through the models, solution approach, and case studies, provides valuable insights into the design of the proposed urban delivery system with shared and dedicated vehicles. The relevant design aspects include the dedicated and shared vehicle fleet sizes, as well as the impact of parameters such as the cost of DVs, the cost of SPVs, the maximum detour distance for SPV, etc.

1.4 Dissertation outline

This dissertation is organized as the follows.

Chapter 2 provides the necessary background information, conceptual framework, and a literature review of the research topic. This chapter describes in detail the operations of various crowdsourced delivery services.

Chapter 3 consists of different mathematical formulations of the operational problem. The chapter explains the rationale of different formulations and the meaning of the equations. This dissertation applies a set-partitioning type of formulation that will be used by the decomposition heuristic developed in this dissertation.

Chapter 4 presents a novel solution approach for the static crowdsourcing package delivery problem. The solution method is a decomposition heuristic that separately considers the shared vehicle routes and dedicated truck routes, and then jointly decides the package split between the two vehicle types. At the end of the chapter, the dissertation presents a comparison in terms of computation time between the decomposition heuristic and traditional Vehicle Routing Problem (VRP) problem.

Chapter 5 is the application of models and algorithms. The dissertation uses the City of Irvine, CA as the study area and compares performance metrics including cost of using SPV, total cost of delivery, SPV/truck VMT, total VMT and shared vehicle used. This chapter also conducts sensitivity analyses by varying the detour willingness of drivers and choosing an alternative depot. Significant findings about cost and VMT of crowdsourced shared-trip delivery are unveiled.

Chapter 6 presents different operating policies and incentive strategies for crowdsourced delivery. The chapter presents a Pickup and Delivery Problem (PDP) based formulation to find the optimal operation policy for logistics companies. The chapter also lists out different incentive/compensation policy for crowdsourced drivers.

Chapter 7 summarizes the dissertation and discusses further research.

.

Chapter 2 Conceptual Framework and Literature Review

This chapter presents the conceptual framework of urban crowdsourced delivery, compares three types of crowdsourced delivery services, identifies the crowdsourced delivery service that this dissertation focuses, and reviews related literature.

2.1 Urban freight crowdsourced delivery

As described in Chapter 1, crowdsourced delivery could be intra- or inter-city level. This dissertation narrows down to intra-city level (urban level) crowdsourced delivery. In urban level crowdsourced delivery, three major categories could be identified (Table 2.1). The separation is based on whether the driver is an "amateur" service provider or a semi-professional ("semi-pro") service provider, and whether the source of mobility is a trip-based sourcing or time-based sourcing. The three different ways of crowdsourced delivery include sourcing a semi-pro driver for trips, sourcing a semi-pro driver for time, and sourcing an amateur driver for shared trips. In the section, the three types of crowdsourced delivery are explained in detail.

Table 2.1 Types of Urban Crowdsourced Delivery and Examples

Working Type	Sourcing Type	
	Time	Trips
Amateur	-	Crowdsourced shared-trip Delivery (e.g., Store Customer Delivery)
Semi-pro	Crowdsourced Time-based Delivery (e.g., Amazon Flex)	Crowdsourced Trip-based Delivery (e.g., Uber Eats)

Though in the above table, Uber Eats type of business is categorized as crowdsourced trip-based delivery, in real-world operations, the business could be a mixture of crowdsourced trip-based delivery and trip-shared delivery depending on the driver types. In food delivery sector, the coordinating platform may contract with a group of "committed drivers", who are committed to conducting the service in a designated time slot, then the service that is provided by these drivers are more likely to be in the category of crowdsourced trip-based or time-based delivery depending on the ways of compensation. If a driver happens to carry a delivery order on the way home, then the service is a trip-shared delivery.

Crowdsourced time-based delivery

The first type of urban crowdsourced delivery is a delivery type which logistics companies crowdsource drivers who are willing to work for a relatively long period as a delivery person. It usually requires drivers to commit 3 to 4 consecutive hours to deliver a set of packages (usually 60 to 80). An example is the program of Amazon Flex package delivery. The process is as follows. First, after registering as a time-shared driver, a driver will be provided with a list of delivery tasks with different time requirements and number of packages. The potential list that a specific driver could receive depends on the location of the driver. The compensation for completing different tasks is different and the calculation is usually based on the number of packages in the task. Then the driver could select a task that fits him/her the best. The last step is travelling to the depot/distribution center, pick up all packages and deliver them.

To understand the service better, one could use the three-factor analysis for the time-based crowdsource delivery. The major supply of mobility is drivers with long idle time. It is worth noticing that though idle time is the key component for mobility supply, an idle vehicle and empty space are also underlying components. The matching of demand and supply is supplier selection based. However, the bundling of packages and pricing of tasks are done by the logistics company. The packages are bundled by their delivery locations and the route of delivery is optimized. Once receiving the tasks, drivers will have a time limit to finish delivering all packages. During the working time, the driver is quite similar to profession delivery personnel. The final factor, media of communication, is a mobile phone app.

The dissertation names this type of crowdsourced delivery as crowdsourced time-based delivery, since the type of service requires more than a single trip or a short period of time. It is closer to the type of "hiring occasional drivers and their vehicles for a period of time". Crowdsourced time-based delivery has multiple benefits over the traditional truck delivery service. From the cost perspective, the logistics companies save cost of purchasing additional trucks, cost of hiring professional delivery personnel and related administrative/miscellaneous cost. For VMT, it seems drivers may incur additional VMT by travelling to the distribution center. However, their delivery service replaced the heavy truck/vehicle delivery. Therefore, although the total VMT may increase, but the actual pollution brought by truck delivery may exceed the crowdsourced delivery. For drivers, they could earn additional income in their idle time.

Crowdsourced trip-based delivery

The second type of service is commonly seen in meal delivery or some other same day delivery examples (Ulmer, Thomas, Campbell, Woyak, et al., 2021). Uber Eats and DoorDash are both examples of this type of service. I name it "trip-based" because drivers are usually required to complete certain trips in order to receive compensation. The crowdsourced trip-based delivery is similar as an Uber/Lyft type of ride sourcing service, or a ride share service. When packages are ready to be delivered, the delivering platform searches nearby drivers and matches them with orders. Then drivers pick up packages and deliver them to designated locations. If a driver picks up and deliver one order at a time, it is similar as a non-share ride matching problem. If multiple packages are picked up simultaneously or sequentially, and then delivered sequentially, the shipping process is close to a rideshare of people transportation. It is worth noticing that the drivers are usually paid per order, and compensation per order is either fixed or based on distance travelled (Yildiz & Savelsbergh, 2019).

The three-factor analysis in Chapter 1 could be applied again for this type of service. First, the supply mobility is the trips of drivers who are willing to deliver. The matching of packages and drivers are conducted by optimization schemes of the matching plat form. Drivers may reject unwilling-to-deliver packages. The third factor, communication media, is a mobile app.

The dissertation hereby compares the crowdsourced time-based delivery and the crowdsourced trip-based delivery. The first difference is that the former requires drivers to work for consecutive hours for a set of inseparable tasks, while for the second one, the drivers usually have choices to accept or reject individual tasks and the working time is based on the number of tasks that the driver has accepted. The second difference is that the crowdsourced time-based delivery is closer to a static Vehicle Routing Problem (VRP), while the second one is closer to a dynamic pickup and delivery problem (PDP). The third difference is that the goods delivered by crowdsourced trip-based delivery is usually more urgently needed than the goods delivered by crowdsource time-based delivery. Therefore, the trip-based one is usually used in food, medicine or same-day delivery. The third difference also leads to the fourth one that the per package delivery cost of crowdsourced trip-based delivery is considerably higher than the per package delivery cost of crowdsourced time-based delivery. The common feature of the two is that drivers are semi-professional, which means they are not involved in other tasks or trips with other purposes at the meantime.

Freight share-a-trip delivery/Crowdsourced shared-trip delivery

The last type of urban crowdsourced delivery is the one that this dissertation concentrates on. This type of service is usually conducted by an amateur driver who shares part of his/her trips to deliver goods. In literature, it is called as VRP with occasional drivers (Archetti et al., 2016), crowdsourced delivery (Arslan et al., 2019), same-day crowd-shipping(Dayarian & Savelsbergh, 2020). Also in literature, usually this type of service is restricted as "using in-store customers for the delivery". In this dissertation, instead of restricting the supply of mobility to in-store customers, the dissertation applies a more general description for the crowdsourced shared-trip delivery.

Assume one distribution center (DC) is responsible for package delivery in a service area. The task of distribution center is to deliver packages/freight to specific locations and not violate the time window constraint. Private vehicle drivers who are willing to participate in delivering small to medium size package in urban/sub-urban areas register their *trip* information to the depot. Once they are matched with a package, they come to the depot and carry the package on their way to destinations. The trip sharing vehicles are called shared-trip vehicles (named "shared personal vehicles", "shared vehicles", or abbreviated as SPVs). To successfully complete a delivery, SPV drivers may need to detour for package pickup and delivery. Each driver has a time window of travel, which is described by the difference between latest arrival time (LAT) at their destination and their earliest departure time (EDT) from their origin. The detour time needs to be with the time window of travel. Each package also has a loose time window of delivery. The distribution center also has a number of dedicated vehicles (DVs) available, which are dedicating to package delivery. Each DV has a capacity limit for carrying packages but no time constraint for returning to the depot. Since in this setting, the packages are delivered by shared trips of SPVs, I name this type as freight share-a-trip delivery or crowdsourced shared-trip delivery. Figure 2.1demonstrates the crowdsourced shared-trip delivery.

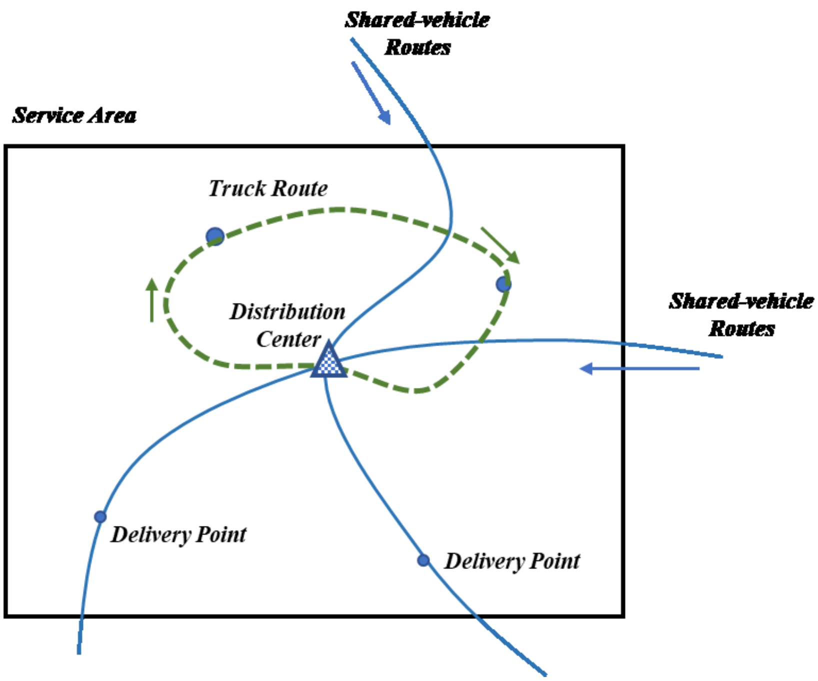

Figure 2.1 Freight share-a-trip delivery

The three-factor analysis could be again used on this type of crowdsourced delivery. The mobility supply of this service is the trips of personal car drivers who are willing to share. In addition, DV will be used if SPVs are not able to complete the task. The matching between the demand and supply are conducted by optimization schemes. The media of communication is also mobile apps. The major difference between crowdsourced trip-based delivery and crowdsourced shared-trip delivery is whether the drivers' primary purpose of a trip is a delivery trip or not. If the driver's primary purpose of a trip is not delivery goods, but shopping trips or working trips etc., the delivery is considered as crowdsourced shared-trip delivery. Otherwise, it is a crowdsourced trip-based delivery.

Comparison between crowdsourced delivery types and traditional truck delivery

This subsection compares the differences between all urban freight delivery methods. The following is a summary table of three crowdsourced delivery types and traditional truck delivery.

Table 2.2 Comparison between urban crowdsourced delivery and traditional delivery

Delivery Type	***Crowdsourced Time-based Delivery***	***Crowdsourced Trip-based Delivery***	***Crowdsourced shared-trip Delivery***	***Traditional Truck Delivery***
Supply of Mobility	Private vehicle drivers to complete specific trips	Private vehicle drivers to complete specific trips	Daily traveler trips and vehicle extra capacity + Truck	In-house logistics or 3rd Party logistics
Matching of Supply and Demand	Driver selection	Optimization	Optimization	Vehicle Routing Problem
Handling Personnel	Private vehicle drivers, Semi-pro	Private vehicle drivers, Semi-pro	Private vehicle drivers, amateur	Professional delivery workers
Compensation	Package bundles and distance	Fixed per trip or distance base	Fixed per deliver + detour	Wages

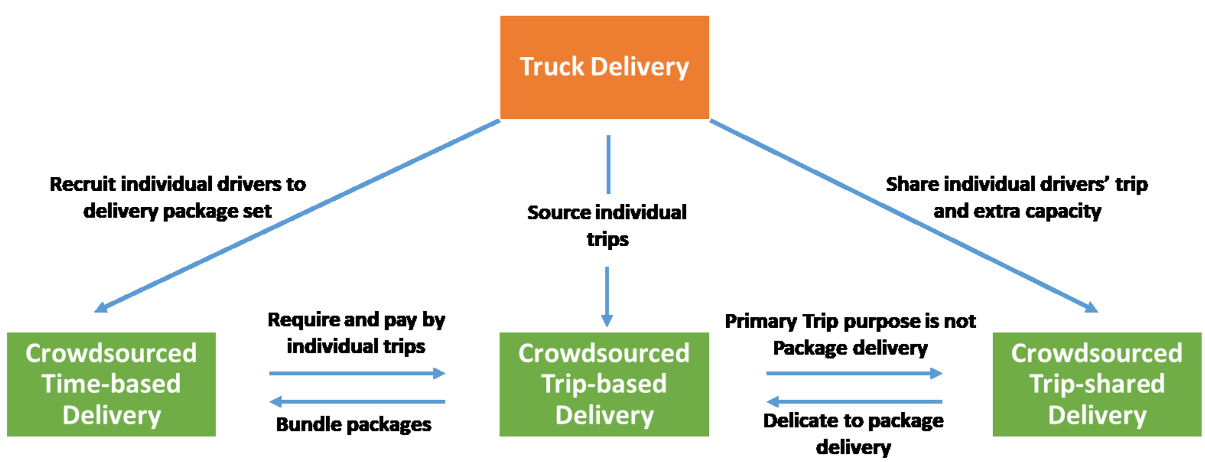

Figure 2.2 Relations between different crowdsourced delivery and traditional truck delivery

The first difference between crowdsourced delivery and traditional truck delivery is on the supply of mobility. For traditional truck delivery, usually the depot/logistics company owns a fleet, but for crowdsourced delivery, mobility supply is sourced from private vehicles by different ways. It is worth noticing that for crowdsourced shared-trip delivery, dedicated trucks are still necessary since when shared vehicles are limited, SPV mobility may not cover all package locations. The matching of demand and supply for all types of delivery are similar. Though, for crowdsourced time-based delivery, drivers decide their preferred packages set by selection, but the package set is bundled by logistics companies using optimization methods. For handling personnel, both crowdsourced time-based and trip-based delivery are using crowdsourced drivers as semi-pro drivers. This dissertation defines a semi-pro delivery person as a driver whose primary task is to deliver packages during flexible, self-decided working hours. The drivers for crowdsourced shared-trip delivery do not have the primary trip purpose of delivery packages, and they are amateur delivery persons. The compensation schemes are also various for different delivery type, which would be discussed in Chapter 5.

This dissertation narrows down to further study crowdsourced shared-trip delivery. The next section, literature review, will review related literature, which include studies related to crowdsourced delivery, vehicle routing problem, ridesharing problem and urban freight delivery integrated with other modes.

2.2 Terminology

Up to Section 2.1 of this dissertation, I have introduced a few new terms. This section summarizes and clarifies the terms that are used *in this dissertation* and related to crowdsourced delivery.

Crowdsourced delivery is the logistics mode that sources mobility supply from the general public. It is sometimes called ***crowd shipping***. By distance, crowdsourced delivery could be categorized to ***inter-city crowdsourced delivery*** and ***intra-city crowdsource delivery.*** Intra-city crowdsourced delivery, which is the focus of this dissertation, is also named ***urban last-mile crowdsourced delivery***.

This dissertation introduces a taxonomy for urban last-mile crowdsourced delivery and names three categorizes, ***crowdsourced trip-based delivery, crowdsourced time-based delivery and crowdsourced shared-trip delivery***. The definitions of the three categories could be found in Section 2.1 and are not repeated them in this section. The drivers who participate in crowdsourced delivery are called ***crowdsourced drivers***; their vehicles are called ***crowdsourced vehicles***. In the third category, crowdsourced shared-trip delivery, since the drivers share the space in their personal vehicles and part of their trips, they are specially called ***shared personal vehicle drivers***, and their vehicles are called ***shared personal vehicles (abbreviated as SPVs, or SVs)***. Any non-crowdsourced vehicles used in this dissertation are called ***dedicated vehicles (abbreviated as DVs)***, and since most of the time DVs are medium size ***trucks or vans***, trucks or vans in this dissertation also refer to dedicated vehicles.

2.3 Literature review

This section reviews related literature. The review of literature starts with ridesharing problems, followed by the people and freight integrated transportation problem, followed by the crowdsourced delivery problem and ends with literature on the Vehicle Routing Problem (VRP).

Ridesharing problems are a category of problems arising from shared economy. It focuses on the sharing of vehicles of routes between two separate parties, who have common segment of routes and accept reasonable detour. Ridesharing problem are also intensively study in recent years.

Reviewing literature on ridesharing between people helps enhance the understanding of ridesharing between people and parcels.

Ridesharing problems

Ridesharing problems are a group of problems where a set of vehicles serve a set of passengers such that route overlapped passengers may be serve by the same vehicle. The objectives include maximize people served, maximize profit, minimize total route distances or minimize total wait and in-vehicle time for passengers.

(Shaheen & Cohen, 2019) present a taxonomy of rideshare services. In their categorizations, rideshare services include three major classes, namely core pooled services, ridesharing and on-demand ride services. Core pooled services define a broad type of services that encompasses pooling services without smartphone apps (e.g., public transit). Ridesharing consists of general carpool/vanpool of families, coworkers, in which smartphone apps are also not necessarily involved. The last category, on-demand ride services, utilize smartphone apps for matching drivers and riders. On-demand ride services are further classified into four sub-classes, including ride sourcing (Uber, Lyft), ride splitting (Uber pool, Lyft Line), taxi share and micro transit. Ridesharing services are proved to have large social and economic benefit (Rayle et al., 2016). The ridesharing services reduce vehicle miles travelled, ride sourcing cost and road network congestion (Wang & Yang, 2019). (Levin et al., 2017) simulate the city traffic of Austin with and without dynamic ridesharing. They find that dynamic ridesharing could reduce the additional empty vehicle repositioning trips.

These findings all inspire us to think whether similar benefits could be achieved by enabling people and freight to share the same trip. The difference between freight and people is that freight requires less space than people, and the time window requirement of freight is usually not as urgent as people (unless it is some emergency packages). Therefore, one may replicate some approaches used in ridesharing between people to rideshare between people and parcels.

On the other hand, technically, providing on-demand ridesharing service is complicated and challenging. One of the reasons is that dynamic matching of vehicles and passengers usually involves choosing “the best” out of large numbers of feasible vehicle-passenger pairs and routing vehicles in a dynamic and smart manner.

There are different ways of matching vehicle and passengers. The closet one to the people and parcel trip integration problem is the matching of a “single driver, multiple rider arrangement”, i.e. a driver could simultaneously serve multiple riders, and riders will not switch to new drivers during the process (Agatz et al., 2012). Match-up of passenger drivers could be modelled multiple ways using different objective functions and constraint sets. Objective functions are set to achieve different goals. Multiple objective functions are considered in previous studies, such as minimizing total vehicle miles travelled (VMT) (Pelzer et al., 2015; Simonetto et al., 2019), minimizing total delays and waiting (Alonso-mora et al., 2018), and maximize the total VMT saved (Qian et al., 2017). Besides the difference in objective functions, the formulation of the problem either involves an integer programming (IP) of assignment problem (Alonso-mora et al., 2018; Hosni et al., 2014; Simonetto et al., 2019) or dial-a-ride problem (DARP) (Cordeau & Laporte, 2003, 2007; Quadrifoglio et al., 2008). This paper uses the objective of minimizing total vehicle miles travelled and adopts the MIP structure for the vehicle-passenger bi-partite matching problem. Bi-partite matching problem (also called assignment problem) is a widely used 0-1 integer programming type, the dissertation explains the problem of bi-partite matching problem in the content of a ride matching problem as follows.

Bi-partite matching formulation

There is a set of vehicles V, and a set of passengers P. The task is to match passengers to vehicles in order that every passenger is served by a vehicle. The decision variable of the problem is x_{ij}, a binary variable that indicates whether a passenger i is served by vehicle j.

$$Max\ Z = \sum_{i,j} c_{ij} x_{ij} \ (2.1)$$

subject to:

$$\sum_{j} x_{ij} = 1, \forall i \in P \ (2.2)$$

$$\sum_{i} x_{ij} \leq q_j, \forall j \in V \ (2.3)$$

$$x_{ij} \in [0, 1] (2.4)$$

Objective (2.1) maximize the total matching number of packages. Constraint (2.2) ensures that a package is served by a vehicle. Constraint (2.3) limits the number of packages served by a vehicle must be smaller than its capacity. A bi-partite matching problem is an integer problem, but could be solved efficiently with linear relaxation, since all corner points of the feasible region are all integers (Y. Lee & Orlin, 1994). The problem is a special case of transportation problem, and is widely used in crew scheduling, ride matching and pairing problems.

The solution algorithms of the matching/assignment problem were well developed in literature. Most algorithms could achieve a polynomial time solution (Burkard & Çela, 1999). Efficient algorithms include Hungarian method (Kuhn, 1955), which has a complexity of $O(n^3)$; Shortest augmented path algorithms (Jonker & Volgenant, 1987), which could achieve a complexity of

$O(n^2 logn)$ or $O(n^2)$ for some special cases; Simplex-based algorithms (Hung, 1983), which also has a complexity of $O(n^3)$. Overall, the assignment problem is handled with polynomial algorithms and usually used as an important relaxation for traveling salesman (TSP) type of routing problems.

Besides the assignment of vehicles to passengers, the choice of routes for vehicles also impacts the efficiency of ridesharing system significantly. Routing a vehicle includes deciding the sequence of pickup and delivery when ride splitting appears and choosing the paths from one pickup/drop-off location to another. The decision of pickup/drop-off sequence is often tackled by solving the problem as a vehicle routing problem (VRP) or more specifically, a DARP, which could be seen in (Ma et al., 2015) and (Simonetto et al., 2019). The path from one location to another is determined by travel time or distance. In most cases, the shortest path is used. However, proper detour may increase the opportunity for a vehicle to serve additional passengers. This concept is also applicable in freight sharing ride problem. When the service vehicle detours with its time constraints, it enlarges the possibility of matching to additional parcels.

In summary, the problem of ridesharing between people has considerable similarities with the problem of ridesharing between people and parcels. The idea of vehicle detour routing and matching vehicles with passengers could be applied to freight share-trip delivery.

People and freight integrated transportation problem (PFIT problem)

The second part of literature reviews the problem that integrates people movement and parcel delivery into the same vehicle. Broadly, integrating freight and passenger systems is also a part of People and Freight Integrated Transportation Problems (W. Chen et al., 2017). PFIT problems are a category of problems, which study the movement of freight and passengers as a combined objective. In these problems, freight is handled combined with passenger movements. In literature, researchers propose three ways of integration (W. Chen et al., 2017), namely, freight delivery integrated with public transit, taxi or Transportation Network Company services (TNCs), and private/personal vehicles. This dissertation will review literature in the three categories respectively.

Freight movement integrated with public transit

The first type of integration concerns with transporting goods for relatively long distances by using idle capacity of transit or adding carts to passenger trains. The benefit of this type of integration is mainly for cross-city transportation of freight. Public transit, in literature, which includes buses, trains and other rail systems, are largely involved in this type of integration. The feature of public transit is that it has relatively large idle capacity during non-peak hours, but its service routes are fixed. Therefore, additional freight transfer may be needed.

(Trentini & Malhene, 2010) conceptually describe the integration of passengers and goods transport services. They provide examples in this paper about how freight and passengers can share transportation resources. A new design of Freight-Bus is mentioned, which combines passenger and freight transport together. They summarize possible scenarios where buses, rails and light rails systems are available for transferring good.

Masson et al. (2017) formulated the Mixed Urban Transportation Problem (MUTP) as a mixed integer programming and solved the problem with neighborhood search technique. A case study in the city of La Rochelle has been conducted to get preliminary results. Studies using similar approaches include (Trentini et al., 2013) (A mixed integer vehicle routing problem) and (Woensel, 2013) (A mixed integer pickup and delivery problem).

Fatnassi et al., (2015) propose a combined passenger and freight rapid transit system for urban freight. In their design, a rapid train is equipped with both passenger carts and freight carts. Passengers and freights can share the same train and also have their own time of operation. At each station, dedicated personnel will use forklift to handle the collection of parcels.

Freight movement integrated with taxi or TNCs

Integration between freight and taxis or vehicles of ride sourcing companies belongs to the second category. In this type of integration, packages are considered the same as a passenger to share the vehicle. (B. Li et al., 2014) initiated a study on people and freight integrated transportation problems (PFIT problems). They construct a scenario where people and freight are sharing vehicles. In that scenario, they attempted to insert freight into an itinerary of a people, which is so called a freight insert problem. They formulated a mixed integer programming for the problem and solved it with neighboring search algorithm. They have concluded that at certain level, taxi companies had to compromise their profit with matching rate if freight and passengers. In addition, traditional distribution vehicles might still be needed as a supplementary service.

Qi et al., (2018) conducted another study on shared mobility for e-commerce package delivery problem. In their problem set-up, they considered a case that both distribution trucks and shared vehicles of ride sourcing companies (crowdsourcing mobility services) are both serving multiple terminals for distribution. In this study, they provided analytical solutions for decisions of service zones, wage paid to shared-vehicle drivers, and cost density of using shared-vehicles and trucks. They concluded that, shared-mobility cost was not scalable as truck cost of delivery when demand increase. By using shared vehicle for delivery, a company could effectively reduce its fleet size, and receive benefit from not purchasing extra trucks. From environmental perspective, they also conclude that using shared vehicles for delivery might not reduce emission due to extra trips of each vehicle.

Freight movement integrated with personal vehicles

The third type of integration is the most relevant way related to this dissertation. It considers packages delivered by personal vehicles. This kind of integration sometimes is named crowdsourced delivery with ad-hoc/occasional drivers, or same-day delivery problem. The dissertation provides detailed reviews of related literature in next subsection.

The earliest literature that one could found about this type of integration is (Archetti et al., 2016), which formulate a vehicle routing problem with occasional drivers (VRPOD). They provide basic formulation and regulate the number of packages served by shared vehicles to be one. Similar formulations are also proposed by (Macrina, Di Puglia Pugliese, et al., 2017). This dissertation further extends the formulation with time window and heterogeneous capacity among SPVs.

Both Archetti et al., (2016) and Macrina, Di Puglia Pugliese, et al., (2017) uses Solomon VRPTW instances (Solomon, 1987) for their numerical experiments. Both works show potential reduction in cost and Vehicle Miles Travelled (VMT) by applying crowdsourced delivery. Arslan et al., (2019) formulate the crowdsourced delivery problem as a matching problem and propose a rolling horizontal framework for dynamic cases. Their numerical study results indicate a potential 37% saving in VMT by using occasional drivers.

Environmental impact of crowdsourced delivery service is studied by (S. Lee et al., 2016b) and (Rai et al., 2017). It is possible to reduce carbon emission by using SPVs to deliver packages.

Crowdsourced delivery

This section reviews related research in the literature and delineates the unique contribution of the current study relative to the existing literature. Previous research related to crowdsourced logistics has been wide-ranging in terms of research methodology. This section reviews crowdsourced logistics research that employ (i) empirical methods to model crowdsourced delivery behavior and demand, (ii) optimization methods to model, design, and analyze crowdsourced logistics systems/services, and (iii) other methods including analytical models and simulations.

J.-F. Rougès & Montreuil, (2014) study 18 startups in the crowd shipping industry and claim that the business-to-consumer (B2C) crowdsourced delivery works best for intra-urban deliveries due to the need for partnerships with retailers and population density in urban areas. (Punel et al., 2018) analyze the determinants of using crowd shipping after collecting 800 responses from a web-based survey. Their results indicate that crowd shipping package users believe the major advantages of crowd shipping relate to environmental benefits and vehicle utilization instead of affordability of crowd shipping items. The dissertation applies optimization for the modeling, analysis, and design of crowdsourced delivery; hence, the related literature will be reviewed in detail. The study of crowdsource delivery problem as an optimization problem has been conducted from both static and dynamic perspectives.

Static problems usually treat the crowdsource delivery problem as a multi-vehicle routing problem (m-VRP) or multi-vehicle pickup and delivery problem (m-PDP). Archetti et al., (2016) model the crowdsource delivery as an extension of classic static VRP. The proposed model assumes a maximum of one task per shared vehicle driver. The study applies a multi-start heuristic by first assigning all packages to dedicated trucks and then solving a series of small scale bi-partite matching problem to assign packages to SPVs. Macrina, di Puglia Pugliese, et al., (2017) extended the problem to a VRP with time-windows (VPRTW) and allow SPVs to carry multiple packages. Dahle et al., (2019) formulate the problem with consideration of pickups and drop-offs and formulate the problem as a pickup and delivery problem with time windows (PDPTW). The focus of Dahle et al., (2019) is to compare different compensation schemes. The study argues that the compensation needs to be large enough to exceed the threshold of the driver's willingness to deliver. The paper concludes that all crowdsource delivery would reduce total costs for logistic companies and the savings would be around 10-15%.

The crowdsourced delivery problem is also formulated as a dynamic problem in literature. Arslan et al., (2019) provide a dynamic way of modelling the problem. Besides a single store as depot, the paper also considers in-store customers willing to travel to another depot for pickup and delivery. To solve the problem, the paper proposes a rolling horizon approach that employs a matching problem solution technique. Dayarian & Savelsbergh, (2020) also model the dynamic crowdsourcing problem with the assumption of maximum one task per SPV. The paper proposes and compares decision strategies, namely, myopic assignment and sample scenario planning. Gdowska et al., (2018) model the problem as a bi-level stochastic problem. They consider the possibility that in-store customers reject a matching of delivering a package. The paper proposes a cost-driven heuristic technique for solutions.

Studies applying approaches other than empirical analysis and optimization include P. Chen & Chankov, (2018) (an agent-based simulation) and Qi et al., (2018) (an analytical approach). In P. Chen & Chankov, (2018), the simulation results indicate that the maximum willingness of detour effects the service level and the number of packages served by SPVs the most. Qi et al., (2018) develop a continuous approximation model for the open vehicle routing problem of SPV drivers.

The study points out that the major economic benefit of crowdsourced delivery is reducing fleet size and offering operational flexibility.

While formulating the crowdsource delivery problem based on VRP or PDP, simultaneously routing both DVs and SPVs is unavoidable and challenging for medium- and large-scale problem instances. Due to the NP-hard nature of routing problems, despite the rich literature related to solving VRP (Cordeau & Laporte, 2003; Golden et al., 2008; Laporte, 1992; Laporte, Gendreau, Potvin, et al., 2000), optimal routes are tough to obtain with even a dozen dedicated trucks, not to mention hundreds of potential SPVs in this problem. Moreover, the decision to assign a package to a dedicated truck or a SPV is not simple to make. The decision is ultimately driven by costs; however, the marginal cost of assigning a package to an SPV or DV is hard to precisely and accurately estimate.

This dissertation differentiates itself from previous studies on the following aspects. The comparison of major optimization related papers is listed in Table 1. First, this paper provides an additional set cover formulation of the problem along with a traditional VRPTW with detour consideration. Second, by decomposing the set covering problem to a package-SPV matching and a DV routing problem, the method can successfully handle a large number of SPVs and restrict the VRP to limited number of dedicated trucks. Formulating the Crowdsourced Shared-trip ProblemChapter 3 presents the mathematical model and Chapter 4 explains the solution approach.

Table 2.3 Comparison with other crowdsourced shared-trip delivery studies

Literature	Problem Nature	Formulate	SPV Capacity	SPV compensation	Solution Technique	Test Scale
(Archetti et al., 2016)	Static	VRP-based	One	Package location	Muti-Start Heuristic	100 Task, 100 SPVs
(Macrina, di Puglia Pugliese, et al., 2017)	Static	VRPTW-based	Multiple	Detour-based	CPLEX directly	100 Task, 100 SPVs
(Dahle et al., 2019)	Static	PDPTW-based	Multiple	Threshold of compensation	MOSEL	70 Task, 100 SPVs
This dissertation	Static	Set Cover Based	Max. 4	Fixed + Detour-based	Decompose heuristic	200 Task, 1200 SPVs
(Arslan et al., 2019)	Dynamic		Max. 4	Detour-based	Heuristic	
(Dayarian & Savelsbergh, 2020)	Dynamic		One	Store credit	Matching SPV first	
(Gdowska et al., 2018)	Dynamic		One	Package location	Heuristic	

Vehicle routing problem (VRP)

This freight share-s-trip delivery problem could also be viewed from a VRP point of view. VRP has a long history and has been well studied by researchers.

VRP is described as the problem of obtaining optimal delivery/pickup routes from the depots to a spatial distribution of customer locations. It works as a base for combinatorial optimization (Laporte, 1992).

The basic settings of a VRP includes the following factors:

- Graph $G = (N, A)$.

- Node/vertex set *N*: $N = \{n_0, n_1, n_2, \dots n_i\}$, where n_0 represents the depot.
- Arc set *A*: $A = \{(n_i, n_j)\}, \forall n_i \in N, i \neq j$.
- Vehicle set *V*: $V = \{v_1, v_2, \dots, v_k\}$.
- Cost matrix: c_{ij} for the travel cost between $Node\ n_i\ and\ n_j$.

The objective of the problem is to visit all nodes with given fleet at a minimum cost. There are certain number of deviations of standard VRP, which can be summarized as follows (Laporte, 1992).

- *Capacity constraints*: each vehicle in fleet can be assigned a capacity of carrying goods. This type of problem is referred as Capacity-restricted Vehicle Routing Problems (CVRPs).
- *Total time/route constraints:* the length of route may be restricted by an upper bound *U* due to the limitation of working hours/fuel/time. This type is of question is referred as time constrained VRPs.
- *Time window*: each location to be visited may be associated with a time interval $[t_a, t_b]$. This type is referred as VRP with time window (VRPTW).
- *Precedence relation*: some $Node\ n_i$ must be visited before $Node\ n_j$.

In addition to the aforementioned variants,(Irnich et al., 2014) also suggests the following elements that could modify the original VRP:

- The road network structure (Arc routing problem)
- The type of transportation requests (Collection, simple visit, load split or repeat supply)
- The constraints that affect each route individually (Route length, vehicle scheduling)
- The fleet characteristics (Mixed fleet, multiple depot)

- The inter-route constraints (Task, operation and movement synchronization)
- The optimization objectives (Single, multiple, hierarchical objectives)

VRP belongs to the category of NP-hard problem in computational complexity literature. Due to the NP-hardness, problem size of VRP problem is vital in computation. This dissertation addresses problems where both the number of nodes and the number of vehicles are large. In this section, both exact and heuristic algorithms to solve VRP will be reviewed.

Exact Algorithms for VRP

According to Laporte & Nobert, (1987) and Laporte, (1992), three categories of exact algorithms are summarized for VRP, namely, direct tree search, dynamic programming, and integer programming.

Early exact algorithms related to direct tree search include (Laporte et al., 1992; Laporte & Nobert, 1986). In these studies, VRP has been transformed to one of its relaxations, multiple-vehicle Travelling Salesman Problem (m-TSP). Then m-TSP problem is solved by branch-and-bound (Christofides, 1981; Laporte & Nobert, 1986). Branch-and-Cut (BC) algorithm is also widely applied in CVRPTW (Capacitated Vehicle Routing Problems with Time Windows). (Augerat et al., 1995) first apply Branch-and-Cut (BC) algorithm for CVRP, which can solve up to 135 locations. Baldacci et al., (2004) formulate a two-commodity flow version of CVRP and apply BC with rounded capacity inequalities.

Dynamic programming methods include (Christofides & Eilon, 1969; Eilon, 1971). The problem is solved by some relaxation and finding lower bound of optimal solutions.

Integer programming approach includes (Balinski & Quandt, 1964), J. Desrosiers et al., (1984), and Desrochers et al., (1990). A set partitioning formulation is suggested for VRP. Column generation method is used to generated possible path which covers subsets of nodes. Following set partitioning formulation, Fukasawa et al., (2006) use a column-and-cut generation method to get lower bound and Branch-and-cut-and-price to solve CVRP. Baldacci, Battarra, et al., (2008); Baldacci et al., (2012b), and Baldacci et al., (2011) are also related to set partitioning formulation and solutions. Exact methods work well for small scale problems, the reported experiment nodes for above studies ranges from 20 to 200.

Heuristic Algorithms

Due to the NP-hard nature of VRP, researchers have been working on heuristics reduce the computational time.

Classical heuristics are mainly constructed by two main techniques: merging existing routes using a saving criterion, or gradually assigning nodes to routes using an insertion cost. Gendreau et al., (1994). Early studies related to saving criteria include (Clarke & Wright, 1964)(Saving algorithm), (Gaskell, 1964), (Nelson et al., 1985)(enhanced saving algorithms), and (Desrochers & Verhoog, 1991) (merging of routes). Literature related to insertion algorithm includes, (Wren & Hollidayt, 1972), (Gillett et al., 1974) (sweep algorithm), and (Bramel & Simchi-levi, 1995) (clustering and routing).

Metaheuristics including simulated annealing and tabu search are early heuristics used to solve VRP (Gendreau et al., 1994). Tabu search in generation starts from an initial solution and move to a best neighbor in each iteration (Laporte, Gendreau, & Potvin, 2000). Related literature includes (Taillard, 1993), (Xu & Kelly, 1996), (Diana & Dessouky, 2004), (Cordeau & Laporte,

2003) (Tabu search), (Ã et al., 2010) (Variable Neighborhood Search) and (Braekers et al., 2014).

In Tabu Search literature, Gendreau & Laporte, (1992) develop a Generalized Insertion Procedure (GENI), which involves moving vertex from its current route to another route. Also, tabu-route procedure untightens the feasibility of solutions, which means, it produces a mixture of feasible and infeasible solutions to avoid trapping in local optimal.

Chapter 3 Formulating the Crowdsourced Shared-trip Problem

This chapter presents the mathematical formulation of the crowdsourced shared-trip delivery problem. This chapter will present both the static and dynamic version of the problem and alternative formulations. In the static version, the study assumes that logistics companies are fully aware of the planned itinerary of drivers, and all packages that need to be delivered are well prepared before the decision process starts. The dissertation assumes all package and shared vehicle information (e.g., location of delivery, time window) are not fully known before the beginning of the first stage and may appear over time.

A detailed list of notations used in this dissertation can be found in Appendix A.

3.1 Problem Description

A set of package delivery orders that require delivery is defined as P. Each package order (p_i) may have multiple packages. The study assumes that all packages are small- to medium- sized and easily fit in a normal sedan. Each package order p_i has a designated drop-off location, an earliest pickup time $T_d^{p_i}$, and latest delivery time, $T_a^{p_i}$.

Two types of vehicles are used for delivery in the crowdsourced shared-trip delivery system, shared-personal vehicles (SPVs, usually family size sedans or wagons) and dedicated vehicles (DVs, usually vans or trucks) for delivery. Let V be the set of all vehicles, S be the set of SPVs and D be the set of DVs; hence, $V = \{S \cup D\}$. An individual SPV is represented as s_k ($s_k \in S$). The driver of an SPV may indicate the maximum number of package orders they are capable or willing to carry/serve, and the parameter is denoted q_{s_k}. Each SPV has its own origin and destination pair. If any package orders are assigned to an SPV, the SPV must travel from its origin to the depot first, pick up the packages, deliver all package orders, and lastly travel to its own destination. Let $T_d^{s_k}$ denote the earliest time an SPV s_k can pick up packages at the depot and let $T_a^{s_k}$ denote the latest arrival time that an SPV s_k should arrive at its own destination. A DV is represented as d_k ($d_k \in D$). The study assumes that all DVs are identical and have a maximum number of stops they can make, denote q_d. The maximum number of stops is determined jointly by the size and range of the vehicle, the maximum consecutive working hours for a driver, and the maximum driving distances of the vehicle and driver. DVs are required to return to the depot/hub after completing delivery tasks.

The service network is defined on a graph $G = (N, A)$. N is the set of nodes, including the hub, all package drop-off locations, and all origins and destinations of SPVs. A is the arcs/links connecting nodes, represented by tuple (i, j), where i and j are nodes. The departure and arrival hubs of DVs are represented as 0 and h (physically they are both the depot). The drop-off location of each package order p is represented as N_p. The designated destination of each SPV s_k is represented as N_{s_k}. The monetized travel cost of a link (i, j) is represented as c_{ij}^s and c_{ij}^d for SPVs and DVs respectively. The travel time of a link (i, j) is represented as τ_{ij}. An SPV driver is compensated by both the number of delivery orders completed and the total detour distances from delivery. The per delivery order compensation is represented by e. The detour distance calculation is demonstrated in Figure 3.1. The monetized cost for each SPV to travel from its origin to the depot is represented as c_{0s_k}. The monetized cost for each SPV to travel directly from its origin to destination is represented as c_{s_k}. Therefore, the total detour cost(compensation) for an SPV is calculated as $c_{0s_k} + delivery\ distance\ cost - c_{s_k}$. Every time a DV is used, a fixed cost, F_d, is incurred, which includes the labor, administrative, and miscellaneous overhead costs associated with an additional DV.

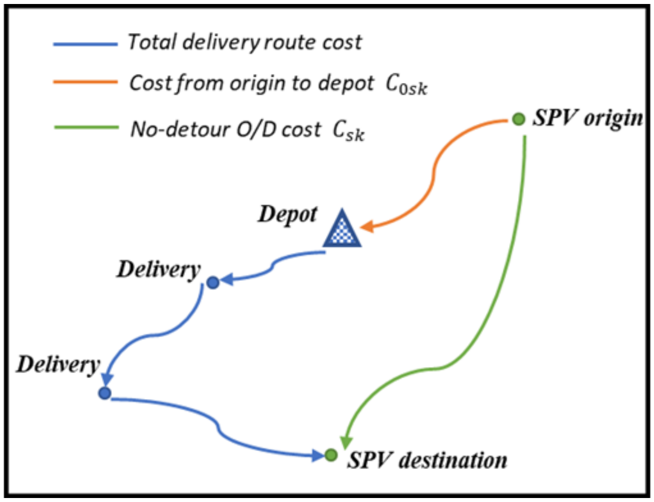

Figure 3.1 Detour Distance of SPVs

3.2 m-VRPTW based formulation

The natural way to formulate a delivery problem is using vehicle routing problem (VRP) formulation, since the delivery problem has the requirements of "unrepeated route" and "returning to depot". The crowdsource delivery problem is a special variant of the original VRP. First, this problem needs to route two general types of vehicles, SPVs (sedans) and DVs (trucks). Therefore, the problem is considered as a heterogeneous fleet or mixed fleet VRP (Baldacci, Battarra, et al., 2008; Irnich et al., 2014). In addition, in this crowdsourced shared-trip problem, SPVs will not return to the depot similar to another variant of VRP, the so-called *Open VRP* (F. Li et al., 2007). The combined variant could be named as a *Mixed Fleet Open Capacitated Vehicle Routing Problem with Time Windows (MFOCVRPTW)*.

In order to adapt to the combined nature of the *MFOCVRPTW*, one may consider "dummy packages for SPVs" in this problem in formulation. A dummy package will not occupy space or capacity in the SPV. The vehicles need to "deliver" the dummy packages to a specific location.

The study assigns a dummy package to each SPV in the system. The dummy package must be picked up at the depot/distribution center by assigned vehicle. Each dummy package has a unique drop-off location, which is the same location as the assigned vehicle, and it also has a latest drop-off time, which represents the latest arrival time of the personal vehicle at its destination.

Assigning a dummy package to SPV guarantees that each vehicle completes its trip at the designated destination and before the required time. Under the regulation of dummy packages, SPV drivers are required to come to the depot first. Therefore, they have two possible routes after leaving the depot, one is going to dummy package destination directly, and the other is delivering real packages first then going to the dummy package destination. Figure 3.2shows possible routes.

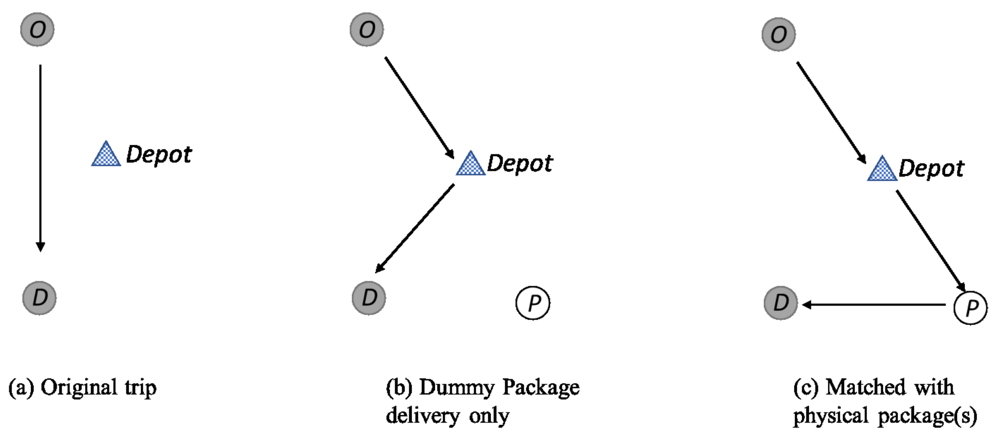

Figure 3.2 Three Types of Shared Vehicle Routes

Figure 3.2(a) denotes the original trip of SPVs. When only a dummy package is assigned, and no physical package is matched, the driver is actually driving to the depot then to his destination (Figure 3.2(a). In the formulation, I assume that it is the necessary trip to be completed by a driver. However, in practice, if no physical package is assigned, SPV drivers will not be required to travel to the depot, and in calculating objective function, this unnecessary detour is eliminated. Figure 3.2c is the case when physical packages are assigned to SPVs. The SPVs will perform delivery before they travel to their destinations. The total detour is calculated by the distance travelled in Figure 3.2(c) minus the distances travelled in Figure 3.2(a).

The detour distance calculated

Decision variables are defined as follows:

- $z_k \in \{0,1\}, \forall\, k \in S.$ $\quad z_k = 1$, if SPV k is used.
- $u_k \in \{0,1\}, \forall k \in D.$ $\quad u_k = 1$, if vehicle k is used.
- $x_{ij}^k \in \{0,1\}, \forall (i,j) \in A, \forall\, k \in V.$ $\quad x_{ij}^k = 1$, if arc (i,j) is visited by vehicle k.
- $t_i^k \in \mathbb{R}^+, \forall i \in N, \forall k \in V.$ $\quad$ Arrival time of vehicle k at node i.
- ***Formulation 1***

$$\min_{\boldsymbol{x,z,t,u}} \boldsymbol{\Theta_1} = \sum_{k \in S} \left(z_k \left(\left(c_{0s_k} + \sum_{(i,j) \in A} c_{ij}^s x_{ij}^k - c_{s_k} \right) + e\Big(\sum_{(i,j) \in A} x_{ij}^k - 1 \Big) \right) \right) + \sum_{k \in D} \sum_{(i,j) \in A} c_{ij}^d x_{ij}^k + \sum_{k \in D} F_d u_k \tag{3.1}$$

subject to

$$\sum_{j \in \{N_p, N_{s_v}\}, j \neq i} \sum_{k \in V} x_{ij}^k = 1 \qquad \forall i \in \{N_p\} \tag{3.2}$$

$$\sum_{i \in \{0, N_p\}, i \neq j} \sum_{k \in V} x_{ij}^k = 1 \qquad \forall j \in N \backslash \{h\} \tag{3.3}$$

$$\sum_{i\in\{0,N_p\},i\neq j}\sum_{k\in V}x_{ij}^k-\sum_{l\in\{N_p,N_{s_v}\},l\neq j}\sum_{k\in V}x_{jl}^k=0 \qquad \forall j\in\{N_p\} \tag{3.4}$$

$$\sum_{i\in N\setminus\{N_{s_k}\}}x_{i,N_{s_k}}^k=1 \qquad \forall k\in S \tag{3.5}$$

$$\sum_{j\in\{N_{p_i}\}}x_{0j}^k-\sum_{i\in\{N_{p_i}\}}x_{i,h}^k=0 \qquad \forall k\in D \tag{3.6}$$

$$z_k\leq 1-x_{0,N_{s_k}}^k \qquad \forall k\in S \tag{3.7}$$

$$u_k\geq\sum_{j\in\{N_{p_i}\}}x_{0j}^k \qquad \forall k\in D \tag{3.8}$$

$$\sum_{(i,j)\in A\setminus\{(0,N_{s_k})\}}x_{ij}^k\ \leq z_k\times(q_{s_k}+1) \qquad \forall k\in S \tag{3.9}$$

$$\sum_{(i,j)\in A}x_{ij}^k\leq u_k\times(q_d+1) \qquad \forall k\in D \tag{3.10}$$

$$T_d^k\leq t_0^k \qquad \forall k\in S \tag{3.11}$$

$$t_{N_{s_k}}^k\ \leq\ T_a^k \qquad \forall k\in S \tag{3.12}$$

$$T_d^p\leq t_0^k+\left(1-\sum_{i\in N\{N_{s_k}\}}x_{i,N_p}^k\right)\times M \qquad \forall p\in P,\ \forall k\in S \tag{3.13}$$

$$t_{N_p}^k\leq T_a^p+\left(1-\sum_{i\in N\{N_{s_k}\}}x_{i,N_p}^k\right)\times M \qquad \forall p\in P,\ \forall k\in S \tag{3.14}$$

$$t_i^k+\tau_{ij}\leq t_j^k+\left(1-x_{ij}^k\right)\times M \qquad \forall i,j\ \in N,\ i\neq j,\ \forall k\in V \tag{3.15}$$

$$x_{ij}^k \in \{0,1\} \qquad \forall i,j \in N \quad \forall k \in V \tag{3.16}$$

$$z_k \in \{0,1\} \qquad \forall k \in S \tag{3.17}$$

$$u_k \in \{0,1\} \qquad \forall k \in D \tag{3.18}$$

$$t_i^k \geq 0 \qquad \forall i \in N \quad \forall k \in S \tag{3.19}$$

In this formulation, objective function aims to minimize the total cost of delivery using SPVs and DVs. The term, $\left(c_{0s_k} + \sum_{(i,j)\in A} c_{ij}^s x_{ij}^k - c_{s_k}\right)$, represents the total detour cost of an SPV k. The term $e(\sum_{(i,j)\in A} x_{ij}^k - 1)$ is the compensation for delivery orders completed by SPV k. The first term is multiplied by z_k, the indicator variable for whether an SPV k is used. If z_k is 0, the SPV travels directly from its origin to its destination. The second term, $\sum_{k\in D}\sum_{(i,j)\in A} c_{ij}^d x_{ij}^k$, is the total delivery routing cost of all DVs. The last term, $\sum_{k\in D} F_d u_k$, calculates the total fixed cost for using dedicated vehicles.

Constraints (3.2) to (3.6) are routing constraints. Constraints (3.2) and (3.3) indicate that every node must be visited once and only once by each vehicle (either type). The constraints in Eqn. (3.4) are the flow balance constraints at each node. The constraints in Eqn. (3.5) indicate that an SPV must arrive at its designated destination. The constraints in Eqn. (3.6) are for DVs, and they guarantee that if a DV leaves the depot, it must return to the depot. The constraints in Eqn. (3.7) state that if an SPV k delivers no packages on its way to destination, then $z_k = 0$. The constraints in Eqn. (3.8) represent the DV usage constraint, meaning that only DVs that are activated can serve requests. The constraints in Eqn. (3.9), together with the constraints in Eqn.

(3.7), ensure that when $z_k = 0$, the only activity for the SPV k is to travel to its own destination (N_{s_k}), i.e., this vehicle is not used for package delivery. Additionally, the constraints in Eqn. (3.9) regulate the number of delivery locations an SPV k can visit and is the so-called "capacity" constraint. Similarly, the constraints in Eqn. (3.10) regulate the number of delivery locations a DV can visit.

Constraints (3.11) to (3.15) are time window constraints. The constraints in Eqn. (3.11) guarantee that the trip for an SPV starts after its earliest departure time. The constraints in Eqn. (3.12) guarantee that and SPV's arrival time at its destination must be no later than its latest arrival time. The constraints in Eqn. (3.13) ensure that a package is only picked up after its earliest pickup time. The constraints in Eqn. (3.14) ensure that a package must be delivered no later than its latest delivery time. The constraints in Eqn. (3.15) indicate that if $x_{ij}^k = 1$ (i.e., $Node\ j$ is visited right after $Node\ i$ by $vehicle\ k$), the arrival time of $vehicle\ k$ at $Node\ j$ must be later than the arrival time of $vehicle\ k$ at $Node\ i$ plus the necessary travel time of arc (i, j). The constraints in Eqn. (3.15) also serve as sub-tour elimination constraints. The constraints in Eqn. (3.16) to (3.19) are the binary and no-negativity constraints for decision variables.

Formulating the crowdsourced shared-trip delivery problem from a vehicle routing perspective enables us to solve the problem by leveraging rich literature in VRP. Exact methods include Branch-and-bound (Christofides & Eilon, 1969; Little et al., 1963) and branch-and-cut or generating cuts (Baldacci, Christofides, et al., 2008; Baldacci et al., 2012a; Laporte et al., 1985). Heuristics include the earliest and famous Clarke and Wright saving heuristic (Clarke & Wright, 1964) to multiple meta heuristics (Gendreau & Potvin, 2005; Hansen et al., 2001; Nikolaev & Jacobson, 2010; Prins, 2004). The MFOCVRPTW is an NP-hard problem and using exact methods for large-scale problem is computationally infeasible. Therefore, for large-scale problems, heuristics are preferable. Inspired by the literature, this study constructs a decomposition heuristic algorithm to solve the problem. The basis for the decomposition is a set partitioning formulation, presented in the following subsection.

3.3 Set-partitioning formulation

The m-VRP can be reformulated from a set partitioning perspective (Baldacci, Christofides, et al., 2008; M. Desrosiers et al., 1992; Laporte, 1992; Y. H. Lee et al., 2008; Ropke & Cordeau, 2009). The paper treats the collection all package locations (N_p) as a set of nodes to be covered/contained by a collection of route sets (the collections of vehicle routes). Then the objective is to assign the origin and destination locations of each package to one feasible vehicle route while minimizing the total cost of the collection of vehicle routes.

Similar to the approaches of (Baldacci, Christofides, et al., 2008) and (Ropke & Cordeau, 2009), the study alternatively formulates the crowdsourced shared-trip delivery problem as a set partitioning problem by using new variables. Let $y_{i,k}^{s}$ be a binary decision variable and represent whether the i^{th} feasible route of $SPV\ k$ is used. Correspondingly, let binary variable $y_{i,k}^{d}$ represent whether the i^{th} feasible route of $DV\ k$ has been used. Let $c_{i,k}^{s}$ and $c_{i,k}^{d}$ be the cost of travelling on $route\ i$ of shared/dedicated $vehicle\ k$ respectively. Let $a_{i,j,k}^{s}$ and $a_{i,j,k}^{d}$ be two binary parameters. When $a_{i,j,k}^{s}$ or $a_{i,j,k}^{d}$ is 1, it is feasible for $route\ i$ of shared/dedicated $vehicle\ k$ to service $package\ delivery\ order\ j$. The set partitioning formulation of the problem is written as:

Formulation 2

$$\min_{y} \ \boldsymbol{\Theta_2} = \sum_{k}\sum_{i}\left(c_{i,k}^{s} - c_{s_k}\right)y_{i,k}^{s} + e\sum_{k}\sum_{i}\sum_{j} a_{i,j,k}^{s} y_{i,k}^{s} + \sum_{k}\sum_{i} c_{i,k}^{d} y_{i,k}^{d} + F_d \sum_{k}\sum_{i} y_{i,k}^{d} \tag{3.20}$$

subject to

$$\sum_{k}\sum_{j} a_{i,j,k}^{s} \times y_{i,k}^{s} + \sum_{k}\sum_{j} a_{i,j,k}^{d} \times y_{i,k}^{d} = 1 \qquad \forall i \in \{N_p\} \tag{3.21}$$

$$\sum_{i} y_{i,k}^{s} = 1 \qquad \forall k \in S \tag{3.22}$$

$$\sum_{i} y_{i,k}^{d} \leq 1, \qquad \forall k \in D \tag{3.23}$$

$$y_{i,k}^{s}, y_{i,k}^{d} \in \{0,1\} \qquad \forall (r,k) \in R \tag{3.24}$$

The objective function (3.20) is similar to the MFOCVRPTW formulation and minimizes the total cost. The first term, $\sum_k \sum_i \left(c_{i,k}^{s} - c_{s_k}\right)y_{i,k}^{s}$, is the total detour cost of SPVs, and the second term ($e\sum_k \sum_i \sum_j a_{i,j,k}^{s} y_{i,k}^{s}$) is the total "per package order completed compensation" for SPVs. The third term is the total routing cost of DVs, and the last term is the total fixed cost of using DVs.

The constraints in Eqn. (3.21) ensure that each package order must appear once and only once on all vehicle routes. The constraints in Eqn. (3.22) guarantee that one and only one feasible route for each SPV is selected in the optimal solution. The constraints in Eqn. (3.23) state that no more than one feasible route for a DV should be used. The constraints in Eqn. (3.24) are binary constraints for the decision variables.

A set partitioning problem can easily be converted to a bi-partite matching problem. Therefore, Formulation 2 provides a new approach for solving the crowdsourced shared-trip delivery problem as a matching/assignment problem between the delivery orders and vehicle routes. Since the bi-partite matching problem has the feature of total unimodularity (Yannakakis, 1985), it allows linear relaxation of an integer problem.

However, a major challenge still remains. To ensure optimality, it is necessary to enumerate all possible routes for each SPV and DV. Doing so for SPVs is challenging, whereas, doing so for DVs is computationally infeasible. Hence, most research relies on generating a ‘sufficient’ number of promising, yet distinct, routes for each vehicle (Ryan et al., 1993). The first obstacle is to enumerate a large number of routes for SPVs. The second challenge is that even with a huge number of SPV routes, some package locations may still be unvisited, therefore the DV routes generated must cover all unvisited locations to guarantee the feasibility of problem. To cope with these challenges, the paper introduces a novel decomposition heuristic, which handles the routing and assignment of SPVs and DVs separately and considers potential package switching for solution improvement. Therefore, the study introduces a decomposition heuristic algorithm in Solution Algorithm to solve the problem.

Chapter 4 Solution Algorithm

This chapter explains the solution algorithm proposed by this dissertation. The solution algorithm aims at solving large scale the crowdsourced shared-trip delivery problem. I believe that the algorithm would be a solid supplementary to the current solution methods in literature. This chapter also reviews and compares some existing algorithms in detail.

In general, solving a crowdsourced shared-trip delivery problem is challenging, because the combinatorial feature of the problem. Depending on the mixed integer programming with VRP type of formulation limits the scale of the problem. To the best of our knowledge, the current solution scale of the problem is at the level of 50 packages and 100 vehicles. This dissertation proposes a decomposition heuristic which is capable of handling cases of hundreds of packages and thousands of vehicles. The decomposition heuristic includes four major steps that would be explained in detail in this chapter. At the end of the chapter, the dissertation compares the computational efficiency and accuracy of this algorithm with traditional VRP solvers.

4.1 Solution algorithm overview

Chapter 3 presents the *MFOCVRPTW* formulation of the problem. However, solutions based on this formulation would be extremely time costly. Therefore, an alternative formulation, set partitioning formulation, is presented. One may rely on the unimodularity of a matching problem to reduce the complexity of computing. However, generating vehicle routes for a set partitioning problem may be another challenge.

For an ordinary VRP, with a set partitioning/covering formulation, the first step is usually generating the possible vehicle routes. Since in ordinary VRP, no mixed fleet exists. The generation of possible vehicle routes is straight forward. Heuristics such petal methods are recommended. It is worth noticing that since it is nearly impossible to exhaustively generate all possible routes of vehicles. Therefore, constructing a set of promising routes and solving the problem heuristically are suggested (Ryan et al., 1993).

On contrast, the crowdsourced shared-trip delivery problem has two types of vehicles, SPVs and DVs. To solve the problem, I first assume that on average, the per package cost of SPV delivery is lower than the cost of DV/truck delivery. Similar assumption could also be found in (Arslan et al., 2019). To substantiate the reasonability of assumption, I also did some estimation with different cost structures. First, for mile-based variable cost, a heavy truck doubles the fuel cost of a normal sedan. Once the driver's hourly rate, insurance cost and depreciation of vehicles are added, the per mile cost of a truck is more than 70 cents. While for the SPV, it is reasonable to use the business reimbursement rate suggested by IRS (Internal Revenue Service, 2021), the per mile cost is $56 cents. This calculation has not added the fixed cost of purchasing a truck and maintenance cost. With this assumption, the design of the algorithm could attempt to matching all possible packages to SPVs while using trucks to serve the rest. Considering this assumption and following the approach of solving an ordinary VRP, the study needs to generate routes for both SPVs and DVs. At this point, one may have two possible approaches, first simultaneously generating routes for both SPVs and DVs then matching packages to both sets of routes at once. Second, generating SPV routes and match packages to SPVs first then handle the rest using DVs. Why the first option is less preferable than the second one? Because the "promising" set of DV routes is depending on the package delivery ability of the set of SPVs. Without incorporating the

SPV route information into the generation of DV routes may lead to answers that are far from the optimal solution. Therefore, the main structure of the algorithm is to match packages and SPVs first, then considers DV routes.

The next question pops up naturally. How to generate SPV routes? As described in Chapter 3, an SPV has a travel budget that is regulated by earliest starting time and latest arrival time. Therefore, the feasible route set for an SPV is the route from the origin or depot to the destination that is smaller than or equal to the travel budget. The dissertation calls this problem *the budgeted k shortest path problem* and explain the details in Section 4.2. Once the routes of SPV are generated, the next step is to match packages to SPVs. The matching problem is close to a bi-partite matching and is explained in Section 4.3. After matching packages to SPVs, one could route DVs for the rest of packages. Due to the cost structure of a VRP, it may be more cost efficient to switch some packages that are originally matched to SPVs to DVs. Therefore, the dissertation conducts a cost analysis and package switching process. After that, a final route of DVs could be decided and the delivery decisions could be finalized.

The solution procedure is described as follows:

- ***Initialization***:
  - *Initialize counter* $i = 1$
  - *Initialize current set of SPVs,* $S_0 = \{\ \}$
  - *Slice the total SPV set S into* m *subsets* $\{s_1, s_2, \dots s_m\}$ *with roughly the same number of SPVs in each set. Add subset* s_1 *to the current set of SPVs,* S_0*.*
- ***Step 1****: SPV Route Generation.*
  - *Generate a set of feasible routes for each SPV in* S_0*.*
- ***Step 2:*** *Delivery order-shared vehicle assignment problem.*
  - *Assign SPVs to delivery orders. This problem is close to a bi-partite matching problem and can be efficiently solved.*
- ***Step 3:*** *DV routing problem.*

   - *Route one DV to serve the delivery orders not served in Step 2 without any capacity constraints. The step is to obtain an estimated route and delivery cost for each order by using DV.*
- ***Step 4:*** *Swapping packages from SPVs to DVs problem.*
   - *For delivery orders served by SPVs, calculate the insertion cost of the SPV-served order if it would have been instead served by a DV, based on DV route estimated from Step 3.*
      - *If all DVs are at capacity, the insertion cost should include the cost of adding a new DV to the fleet.*
   - *Rank order the packages by their insertion cost.*
   - *Starting with the SPV package with the lowest insertion cost, insert SPV packages into the DV route.*
   - *Terminate when the insertion cost of an order exceeds the SPV service cost.*
- ***Step 5:*** *DV and SPV re-routing problem.*
   - *Route DVs based on an m-VRP solution algorithm to serve all package orders assigned to DVs in Step 3 and Step 4.*
   - *Rematch the delivery orders still assigned to SPVs to SPV routes.*
- ***Step 6:*** *Optimality check.*
   - *Calculate the cost of the assignment and routes determined in Step 5*
   - *Compare this new cost with the current best solution and store the new solution if it is smaller than the current best solution.*
- ***Step 7: Terminate or Increment i and S_0.***
   - *If S_0 includes all SPVs in S, terminate.*
   - *Otherwise, increment i by one and add a new random subset of SPVs, s_i to the current set of SPVs S_0 and go to Step 1.*

Pseudocode for the decomposition algorithm is presented as follows.

Algorithm 1: Decomposition heuristic for crowdsourced shared-trip delivery

A set of SPVs $S = \{s_N\}$; A set of DVs $\{D_k\}$; A set of package delivery orders $\{P\}$;
Travel budget for SPV $s_k = B_k$;
Initialization: $current\ best = +\infty$
  Slice $S = \{s_1, \dots, s_n\} + \{s_{n+1}, \dots s_{2n}\} + \cdots +$
    $\{s_{mn+1}, \dots, s_N\}, S_0 = \{s_1, \dots, s_n\}$
While $S_0 \subsetneq S$:
  For $s_k\ in\ \{s_1, \dots, s_n\}$:
    Find $R_{s_k} = \{r | r_{OD} = (depot\ and\ D_{s_k})\ and\ c_r \le B_k\}$
  End For
  Do Matching $\{P\}\ and\ \{R_s = R_{s_1} + R_{s_2} + \cdots + R_{s_N}\}, \{P\} = \{P_{sv}\} + \{P_{dv}\}$
    Calculating $c_{i,j,k}$ *cost of serving* p_i *by route j of SPV k*
    Route single VRP for $\{P_{dv}\}$, get $DV\ Route = R_{dv}$
    Find the smallest insertion cost λ for $p\ in\ \{P_{sv}\}$ to route R_{dv}
  While $\lambda \le c_{i,j,k}$:
    Move *package order* i *to* $\{P_{dv}\}$

For $p\ in\ \{P_{sv}\}$:
Find the smallest insertion cost λ for $p\ in\ \{P_{sv}\}$ to route R_{dv}
End For
End While
Do m-VRP for $\{D_k\}$ and $\{P_{dv}\}$, calculate total cost $c_T = c_{SPV} + c_{DV}$
If $c_T \leq current\ best\ solution$:
$current\ best = c_T$
End If
Increment S_0 by $\{s_{kn+1}, \dots s_{(k+1)n}\}$
End While
Return: $current\ best$

The reason for implementing an incremental approach, whereby, the algorithm adds a subset of SPVs at every iteration, is to avoid being trapped in a local minimum. The cause for local minimum is the non-convexity of truck routing, which can lead to inefficient package switching between SPVs and DVs in the case where all SPVs are included in S_0 from the beginning. Without the incremental approach, the researchers found that there were instances where cost increased as the number of SPVs increased, which indicated the solution algorithm does not necessarily improve as the solution space expands. For the incremental approach, the fewer SPVs in every batch, the more batches, the higher the solution quality but the longer the computational time.

The advantages of applying the decomposition heuristic are as follows. First, generating routes for an SPV is a relatively straight-forward task since the route length is bounded by the detour willingness the SPV. Hence, SPV route generation can be conducted off-line. With day-to-day operations, the “promising” SPV routes identified or ‘learned’ from prior days can be stored and retrieved as needed.

Additionally, under the assumption that on average delivery packages via SPV is cheaper than delivering packages via DV, it makes sense to initially assign as many vehicles to SPVs as possible, which the algorithm does. Separating the SPVs and DVs in the routing process enables a straight-forward assignment problem between delivery orders and the set of SPV routes.

On the other hand, serving the remaining, non-SPV, delivery orders with DVs are also straightforward given the wide range of exact, approximate, and heuristic solution algorithms for the VRPTW available in the literature.

Moreover, using an insertion heuristic algorithm for DV routing also simplifies the procedure of obtaining marginal cost of serving an additional delivery order. Based on extensive empirical analysis, the SPV to DV package switching step significantly improves the overall solution quality. The main reason for the improvement is that even though the average delivery cost for SPVs is much lower than the average delivery cost for DVs, once a DV is put into operation, the marginal cost of serving a package via DV is smaller than the marginal cost of serving most packages via an SPV, until the DVs near their capacity limits.

The following sections are arranged as follows. Section 4.2 describes the algorithms to obtain *budgeted k-shortest paths*, which corresponds to the aforementioned Step 1 of Algorithm 1. Section 4.3 explains the large-scale package vehicle route assignment problem, which extends Step 2 of Algorithm 1 in further details. Section 4.4 presents the insertion algorithm used for vehicle routing problems in Step 3 and Step 5. Section 4.5 is the procedure for deciding package switch between SPVs and DVs, which is the Step 4 that is described before. Section 4.6 is a comparison of computational time and results between the decomposition heuristic and the exact method. Section 4.7 summarizes this chapter.

4.2 Budgeted k-shortest path

This section presents the algorithm for generating k-shortest paths with budget constraints, which is the first step of the decomposition heuristic. This step is an off-line procedure, in which we generate the possible routes from the depot to different SPV destination locations with various levels of detour willingness. The quality of k-shortest paths is the main determinant of the solution quality of the package/vehicle route assignment problem, and we attempt to exhaustively generate all possible routes for an SPV under a travel budget constraint.

The k-shortest paths with budget constraints problem could be described as the follows. Given a graph $G = (V, E)$, find all possible paths between a start node s and a target node t that are within the travel time/cost/budget of B.

It is worth noting that the budget B in this section is not equivalent to maximum willingness to detour for SPV drivers in the crowdsourced delivery problem. The maximum willingness to detour for an SPV driver is defined as the maximum time the driver is willing to delay arriving at their destination given the departure time from their origin. In this step, we generate routes for SPVs from the depot to their destinations (the bule paths in Figure 3.1), therefore the budget B for an SPV k equals the maximum willingness to detour of the SPV (B_k^M) minus the shortest path travel time from the SPV origin to the depot (c_{0s_k}) and possible order pickup and drop-off time (τ_{pd}), represented as $B_k = B_k^M - c_{0s_k} - \tau_{pd}$.

To solve the k-shortest path with budget constraints problem we first present an intuitive recursion algorithm. The pseudo code is as follows.

Algorithm 2: Recursion algorithm for budgeted k-shortest paths

$G = (V, E), source = s, sink = t, budget = B$
Function $\boldsymbol{BgtKPath}(G, s, t, B, CurrentPath, PathCost)$:
Initialization:
    $NodeVisited = [s]$, $CurrentPath = [s]$
    $kPathHeap = [\ ]$, $PathCost = 0$
If $s == t$:
    $kPathHeap.push(CurrentPath, Pathcost)$
Else:
    For $u \in Neighbor[s]$:
        If $u \notin NodeVisited$:
            If $edgeCost[s][u] + PathCost < B$:
                $PathCost{+}{=}\ edgeCost[s][u]$
                $B{-}{=}\ edgeCost[s][u]$
                Recursion
                $kPathHeap = \boldsymbol{BgtKPath}(G, u, t, B, CurretCost, PathCost)$
            End If
        End If
        Reset vertex $CurrentPath.pop(u)$
        $NodeVisited.pop(u)$
        Reset cost $PathCost{-}{=}\ edgeCost[s][u]$
        $B{+}{=}\ edgeCost[s][u]$
    End For
End If
Return: $kPathHeap$

The recursion algorithm has the advantage of being intuitive and easy to code. The complexity of the algorithm depends on the number of nodes and the connectivity of the network. It has reasonable computational time for sparse networks. In the numerical example, presented in Section 6, we test the recursion algorithm on the City of Irvine network. For the 2000 SPV case, the computational time ranges from 12 mins (10 mins detour willingness) to 6 hours (30 mins detour willingness). For a network with a high level of connectivity, Algorithm 2 runs slowly since it has a complexity of $O(n!)$.

In order to cope with the complexity issue, we attempt to apply Yen's algorithm (Yen, 1971) by adding budgeted constraints to it. The pseudo code of the budgeted Yen's algorithm is as follows:

Algorithm 3 Yen's algorithm with budgeted constraints

$G = (V, E), source = s, target = t, budget = B$
Function $\boldsymbol{BgtYenKsp}(G, s, t, B)$:
Initialization:
    $Find\ the\ shortest\ path\ between\ s\ and\ t, store\ in\ list\ A$
    $A = [Dijkstra(G, s, t)], A_{cost} = [SHP\ cost]$
    $Initialize\ a\ heap/priority\ queue\ for\ cost\ comparison$
    $C = heap.queue()$
    $BudgetFlag =$ **TRUE**
While $BudgetFlag\ is$ **TRUE**:
        For $path\ in\ A$:
            For $node\ in\ path, find\ spur\ nodes$:
                $spurNode = node$
                $rootPath = path[:node]$
                $edgeRemove = [\ ]$
                For $path\ in\ A$:
                    If $rootPath = path[:spurNode]$
                        $Remove\ links\ shared\ by\ rootPath$
                        $edgeRemove.append(links)$
                $Find\ SHP\ between\ spurNode\ to\ t$
                $spurPath = Dijkstra(G, spurNode, t)$
                If $no\ loop\ in\ (spurPath, rootPath)$:
                    $totalPath = rootPath + spurPath$
                    $totalPathCost = rootPathCost + spurPathCost$
                    $C.heap.push((totalPathCost, totalPath))$
                For $edge\ in\ edgeRemoved$:
                    $Add\ back\ to\ removed\ edges\ to\ G$
            If $len(C) > 0$:
                $Get\ the\ first\ time\ in\ B;\ pathAddCost, pathAdd = C.get()$
            If $pathAdd\ no\ in\ A, and\ pathAddCost \leq B$:
                $A.append(pathAdd), A_{cost}.append(pathAddCost)$
            elif $pathAddCost > B$:
                $BudgetFlag =$ **FALSE**
Return A, A_{cost}

The complexity of original Yen's algorithm is $O(kV(E + V\log V)$, with k number of paths to be generated. Algorithm 3 keeps the heap structure of storing paths and applying Dijkstra's algorithm for shortest path finding, which is $O(E + VlogV)$. In the worst-case scenario, when the network is fully connected and the budget is huge, along the spur path, all nodes will be visited. The Dijkstra's algorithm would be called $B \times |V|$ times if we treat budget B as a large constant. The overall complexity for Algorithm 3 is $O(BV(E + V\log V)$.

4.3 Large-scale matching problem of packages and vehicles

The previous subsection presents the algorithms of generating k $shortest$ $paths$ for SPVs. The next step is to match the packages with SPVs. To be more specific, instead of matching packages to individual SPVs, the decomposition algorithm attempts to match packages to SPV routes that were generated from the previous step. The following formulation is therefore presented.

Formulation 3:

$$\max_{x_{prk}} \Theta_3 = \sum_k \sum_r \sum_p \omega A_{prk} x_{prk} - \sum_k \sum_r c_{rk} z_{rk} \qquad (4.1)$$

subject to:

$$\sum_k \sum_r x_{prk} \leq 1 \qquad \forall p \in P \qquad (4.2)$$

$$\sum_r z_{rk} \leq 1 \qquad \forall k \in S \qquad (4.3)$$

$$\sum_p x_{prk} \leq z_{rk} q_{s_k} \qquad \forall (r,k) \in R \qquad (4.4)$$

$$x_{prk}, z_{rk} \in \{0,1\} \qquad \begin{array}{c} \forall p \in P \\ \forall (r,k) \in R \end{array} \qquad (4.5)$$

- $x_{prk} \in \{0,1\}$, Equal to one if package p assigned to route r of SPV k
- $z_{rk} \in \{0,1\}$, Equal to one if SPV k uses route r

The objective function (4.1) maximizes the total benefit of matching delivery orders to SPV routes. To encourage successful matchings, ω (a larger number) is introduced as a reward term. This operationalizes the strategy to match as many packages to SPVs as possible initially. In the objective function, A_{prk} is a binary parameter that indicates whether the r^{th} route/path of SPV k can feasibly serve package delivery order p. The constraints in Eqn. (4.2) guarantee that a delivery order is served by at most one route and at most one vehicle. The constraints in Eqn. (4.3) ensure each SPV only travels on at most one path through the depot. The constraints in Eqn. (4.4), ensure a package p is only assigned to route r if a SPV k is assigned to route r. If z_{rk}

equals zero, SPV k does not use route r, and therefore, no package order should be served by vehicle k on route r. Moreover, if z_{rk} equals one, then vehicle k does use route r and the total orders carried by the vehicle should not exceed the maximum number of orders SPV s_k is willing to serve. Constraint (4.4) also acts as a linking constraint between decision variables x_{prk} and z_{rk}.

It is worth noting that without Constraints (4.3) and the decision variable z_{rk}, Formulation 3 becomes a bi-partite matching problem, which is solvable in polynomial time and has complexity of $O(n^3)$. Commercial solvers, such as Gurobi (Gurobi, 2021), could solve the large-scale assignment problem in reasonable time. For a mixed integer programming, such as Formulation 3, solving a linear relaxation of the problem also provides a good approximation of the optimal. For normal size problem, the study applies Gurobi to solve Formulation 3 directly. When the number of vehicles, the travel budget (detour buffer) and the number of packages increase, the problem of matching packages to vehicle routes would expand to a large-scale problem. The computation time still raises sharply since the total number of routes increases exponentially as the travel budget expands. To cope with potentially large-scale scenarios and to take advantage of the complexity of the bi-partite matching problem, the dissertation implements a Bender's decomposition to Formulation 3 for large-scale cases when the detour willingness of SPVs is high. The formulation and procedure for performing Benders decomposition are presented as follows.

Formulation 4

Master Problem (MP)

$$\max_{z_{rk}} \boldsymbol{\Theta}_{\mathbf{MP}} = \quad Z \tag{4.6}$$

subject to

$$\sum_{r} z_{rk} \leq 1 \qquad \forall k \in S \qquad (4.7)$$

$$Z \leq Cuts \qquad (4.8)$$

$$z_{rk} \in \{0,1\} \qquad \forall (r,k) \in R \qquad (4.9)$$

Subproblem (SP)

$$\max_{x_{prk}} \boldsymbol{\Theta}_{\mathbf{SP}}(\bar{\boldsymbol{z}}_{\boldsymbol{rk}}) = \sum_{(r,k)} \sum_{p} \omega A_{prk} x_{prk} - \sum_{(r,k)} c_{rk} \bar{z}_{rk} \qquad (4.10)$$

subject to

$$\sum_{(r,k)} x_{prk} \leq 1 \qquad \forall p \in P \qquad (4.11)$$

$$\sum_{p} x_{prk} \leq \bar{z}_{rk} q_{s_k} \qquad \forall (r,k) \in R \qquad (4.12)$$

$$x_{prk} \geq 0 \qquad \forall (r,k) \in R \qquad (4.13)$$

The dual subproblem (DSP) could be written as:

$$\min_{\lambda} \boldsymbol{\Theta}_{\mathbf{DSP}}(\bar{\boldsymbol{z}}_{\boldsymbol{rk}}) = \sum_{p} \lambda_p + \sum_{(r,k)} \bar{z}_{rk} q_{sk} \lambda_{(r,k)} + \sum_{(r,k)} \lambda_{(r,k)} \bar{z}_{rk} - \sum_{(r,k)} c_{rk} \bar{z}_{rk} \qquad (4.14)$$

subject to

$$\lambda_p + \lambda_{(r,k)} \geq A_{prk} \qquad \forall p \in P, \forall (r,k) \in R \qquad (4.15)$$

$$\boldsymbol{\lambda} \geq 0 \qquad (4.16)$$

Formulation 4 presents the master problem (MP) and subproblem (SP) of the delivery order-vehicle route assignment problem. The SP is a linear assignment problem with a time complexity of $O(n^3)$. To obtain optimal cuts (corner solutions), the study solves the dual problem (DSP). When initialized with a feasible solution from the MP, the SP is always feasible because the linear assignment problem always has a solution given its parameter settings are valid. Hence, the DSP is never unbounded, and one does not need to generate feasibility cuts (extreme rays) for the MP. The solution procedure is a standard Benders decomposition procedure, which is presented as follows.

Algorithm 4: Benders decomposition for delivery order - vehicle assignment problem

---

Initialization:

A feasible solution $\tilde{\boldsymbol{z}}^{\boldsymbol{0}}_{\boldsymbol{r},\boldsymbol{k}}$; $LB = -\infty, UB = +\infty$.

Iteration counter t = 1

While $UB - LB > \varepsilon$:

Do Solve the dual subproblem (DSP)

Get extreme points $\bar{\boldsymbol{\lambda}}^{\boldsymbol{t}}$

Add cut $Z \leq \sum_i \bar{\lambda}^t_p + \sum_{(r,k)} \bar{z}^{t-1}_{r,k} q_{sk} \bar{\lambda}^t_{(r,k)} + \sum_{(r,k)} (\bar{\lambda}^t_{(r,k)} - c_{rk}) z_{rk}$

$LB = \max\{LB, \sum_i \bar{\lambda}^t_p + \sum_{(r,k)} \bar{z}^{t-1}_{r,k} [(q_{sk} + 1)\bar{\lambda}^t_{(r,k)} - c_{rk}]\}$

Do Solve the master problem (MP)

Get solution $UB = \Theta^*_{\text{MP}}$

End While

Return: $LB, \tilde{\boldsymbol{x}}_{\boldsymbol{prk}}, \tilde{\boldsymbol{z}}_{\boldsymbol{rk}}$

In Formulation 4, the SP is convex and is a restricted problem of the original problem (Formulation 3). Therefore, the SP is an underestimation of the optimal value for the original problem and solving SP gives a lower bound (LB) for the original problem. On the other hand, the MP is non-convex with all integer constraints, and it is a relaxation of the original problem. Solving MP gives an upper bound (UB). Comparing with the original problem, the problem size (number of constraints and variables) is significantly reduced. Adding cuts that are generated from the SP gradually restricts the MP and alleviates the gap between LB and UB.

4.4 Solving the vehicle routing problem

In Algorithm 1, the decomposition heuristic, the routing of trucks is generated by solving a VRP. The truck routing problem that is used in Algorithm 1 is a single depot, single/multiple-vehicle routing. The algorithm that is chosen for solving the problem is an insertion algorithm that was described in (Campbell & Savelsbergh, 2004). Other related algorithms are reviewed in Chapter 2.

The reasons for choosing an insertion algorithm for solving the vehicle routing are as follows. First, Algorithm 1 includes a step of comparing marginal price of serving a package if the package is switched from an SPV to a DV. An insertion algorithm provides a directly ways of obtaining the marginal cost. The following section, Section 4.5 elaborates the usage of marginal cost in detail. Moreover, as suggested by (Campbell & Savelsbergh, 2004), an efficient insertion heuristic achieves a time complexity of $O(n^3)$, and therefore could handle problem in large-scale.

The insertion algorithm has been used as for a single vehicle routing (Step 3) and a multi-vehicle routing (Step 5) in Algorithm 1. Step 3 performs a single vehicle routing in order to obtain an estimation of overall delivery cost for the packages that are to be served by DVs. More importantly, a single route that is formed by the all the DV-served packages provides a direct estimation of the insertion cost if any packages are to be moved from SPV-served set. The marginal cost (for the DVs) of switching a package from the SPV-served set is calculated as follows. Assume the original link in a DV route is link (i, j), and the node to insert is node u.

$$Marginal\ Cost = c_{i,u} + c_{u,j} - c_{i,j} \quad (4.17)$$

Algorithm 4: Insertion Algorithm for m-VRP

Initialization:

  $P_{dv} = the\ set\ of\ unrouted\ package\ locations$
  $R = the\ set\ of\ routes$; Include an empty route

While $P_{dv} \neq \emptyset$:

  Minimum insertion cost $c^* = +\infty$
  Node to insert $u^* = None$
  Link to insert $(i, j)^* = None$
  Route to insert $r^* = None$
  For $u \in P_{dv}$:
    For $r \in R$:
      If $quantity[p] + load[r] \leq q_k$:
        For $(i, j) \in r$:
          Do $c_{ins} = c_{i,u} + c_{u,j} - c_{i,j}$
          If $c_{ins} < c^*$ and time window fits:

$u^* = u;\ c^* = c;$
$(i,j)^* = (i,j); r^* = r;$
End If
End For
End If
End For
End For
Insert $u^*\ to\ (i,j)^*\ in\ r^*$, update r^*.
$P_{dv} \setminus u^*$
End While
Return r$oute\ set\ R$

Step 5 applies the insertion algorithm for a multi-vehicle routing, the pseudo code for the insertion algorithm is presented in Algorithm 4.

In computational experiments, the insertion algorithm achieves high computational efficiency and relative optimal results. The overall performance of Algorithms 1, which includes the application if Algorithm 4, is discussed in Section 4.6.

4.5 Decision of package switching – cost driven approach

Step 4 of the Algorithm 1 involves a procedure of switching packages from the set to be served by SPVs to the set to be served by DVs. The questions that one may be interested in about this step include why switching packages reduces cost for the system and which packages should be switched. This section explains the reasons for the two questions in detail.

To answer the first question, one may need to understand the cost elements of operating a DV or truck for logistics company. Usually, operating a truck for delivery involves two major categories of cost, the fixed cost items and variable cost items. Fixed cost items are the cost items that are not depending on the volume of the business. In other words, some cost that has to be paid regardless of making profit or not. For a package delivery business, fixed cost items, for example, usually includes vehicle purchasing cost, rentals, administrative cost and wages/benefits for necessary employees. On the other hand, variable cost is the cost that depending on the volume of the business, i.e., the more packages that are delivered, the higher the cost (also more profit at the same time). The variable cost component usually contains fuel cost, depreciation of truck based on mileage, and labor cost based on mileage or working hours. One may assume that in package delivery business, the variable cost unit is measured by the vehicle miles travelled (VMT). Therefore, total variable cost equals variable cost per package times the number of packages. In this dissertation, I apply the “fixed plus variable” cost structure for estimating truck cost.

The basic rule for package switching is that if the delivery cost for a package is cheaper on a truck than on SPVs, the package would be served by a truck. Therefore, it is necessary to estimate the "per package cost" on truck. As describe above, the truck total cost includes a fixed and a variable component, therefore, the average cost of delivering a package by truck is calculated by $(Fixed\ cost + Varible\ cost\ per\ mile \times VMT\ Total)/Num\ of\ Packs$. When the number of packages delivered increases, the fixed cost is averaged. Therefore, the average package cost decreases with the number of packages.

The total variable cost estimation for truck delivery is complex. The main reason is that for a given set of packages, it is hard to obtain an optimal route of delivery and therefore hard to estimate the total variable cost. Dedicated vehicle delivery cost has been studied in literature with different estimation schemes (Daganzo, 1984; Figliozzi, 2008, 2009). The literature uses the number of packages, average distance from package to depot, truck capacity and service region area to estimate average delivery cost for dedicated vehicles. The study demonstrates the delivery cost estimation by applying (Daganzo, 1984) estimation as:

$$\bar{c}_t(n) = \frac{2rn}{Q} + 0.57\sqrt{nA} \quad (4.18)$$

In the above equation, $\overline{c_t}$ is the average travel cost for making n stops, r is the average distance from stops to depot. Q is the truck loading capacity. A is the area.

Let us denote the fixed cost of using a truck as c_f. The average total vehicle delivery cost is estimated as $\bar{c}_d = (m \times c_f + n \times \bar{c}_t)/n$, where m is the total number of trucks used. Each truck is also associated with a capacity, which is usually determined by truck size and truck driver working hours. Every time a new truck is used, the total cost increase by c_f. Figure 4.1 presents a sample estimation for average VMT and average total cost. It indicates that when the number of packages is the low, the average cost per package is high. As the number of packages increases, the average cost drops. However, when the number of packages is beyond a truck load (or delivery time longer than the truck driver working hours, additional trucks are required. The finding indicates that if a truck is used, the most cost-efficient way is to exhaust its capacity. The figure also demonstrates the non-convexity of truck delivery cost function, which is one of the difficulties for precise VRP solutions. An exact way of calculating average cost is to run a DV routing problem, this procedure is executable if the unassigned SPV package number is small. However, when the unassigned SPV package number is large, an approximation heuristic or an estimation scheme would be more time efficient.

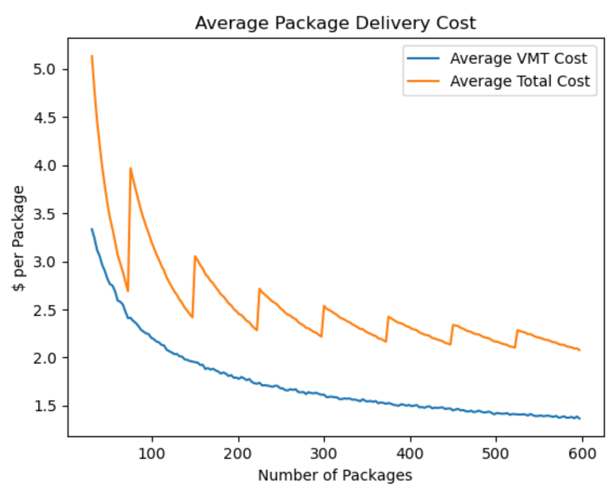

Figure 4.1 Average Package Delivery Cost by Truck

After knowing the cost structure of dedicated vehicle delivery, the dissertation designs the switching scheme as Step 4 in Algorithm 1 and describe it in further details as follows.

Step 4.0 Obtain truck route from single VRP as in Step 3.

Step 4.1 For every package location to be served by SPVs (i), find the nearest location that is to be served by DVs (j). The DV round-trip cost ($2 \times c_{(i,j)}^{dv}$) between i *and* j provides an upper bound cost of using DV to serve this location. If $2 \times c_{(i,j)}^{dv} < c_i^{SPV}$(the cost of serving location i by SPV), it is more cost efficient to using DVs to serve the package location i. Location j is the nearest neighbor for location i. The study uses the insertion cost of either inserting i into link (j^{-}, j) or link (j, j^{+}) to be the estimation of insertion cost of SPV location i to the truck route.

Step 4.2 Calculate the potential cost saving for location i that if packages in location i are switched from SPVs to DVs.

$$Cost\ Saving = SPV\ serve\ cost - DV\ route\ insertion\ Cost \\ - Addtion\ DV\ Fixed\ Cost(if\ any)$$

Compare over all locations to be served by SPVs. Find the minimum one. If the load of DV after switching goes beyond a truck load, an additional truck is required, and therefore additional truck fixed cost needs to be added to the cost saving estimation.

Step 4.3 Switch packages, update truck routes and SPV location list. Repeat Step 4.1 and 4.2. Terminate when all cost savings are negative.

The pseudo code for package switching is as follows.

Algorithm 5: Package Switching from SPVs to DVs

Initialization:
    $loc^{spv} = \{1,2, \ldots i\} = the\ set\ of\ SPV\ locations;$
    $loc^{dv} = \{1,2, \ldots j\} = the\ set\ of\ DV\ locations;$
    $c_{sav}^{min} = +\infty$
While $c_{sav}^{min} > 0$:
    Find nearest DV locations to SPV locations $= \{i: j\}$
    Location to switch $l^{*} = None$
    $c_{sav}^{min} = +\infty$
    For $i \in loc^{spv}$:
        If $2 \times c_{(i,j)}^{dv} < c_i^{SPV}$:
            $c_i^{ins} = \min\{c_{(j^{-},i,j)}, c_{(j,i,j^{+})}\}$
            If $int\left(\frac{load_{dv}+q_i}{q_{dv}}\right) > int\left(\frac{load_{dv}}{q_{dv}}\right)$:
                $c_i^{sav} = c_i^{SPV} - c_i^{ins} - c_f$
            Else: $c_i^{sav} = c_i^{SPV} - c_i^{ins}$
            End If
            If $c_i^{sav} < c_{sav}^{\min}$:
                $c_{sav}^{min} = c_i^{sav}; l^{*} = i$
            End If
        End If
    End For

Insert i to DV route, update

End While

4.6 Computational results comparison

This section compares the computational results between the decomposition heuristic and exact method of multi-vehicle routing problem to assess the effectiveness and efficiency of the decomposition heuristic(D-H). The numerical experiment is conducted by using the network of the City of Irvine, CA, USA. In total, the number of packages ranges from 10 to 100, and packages are distributed uniformly in the study area. In addition, a number of SPV samples are generated randomly. The numerical experiment regulates that every SPV has a fixed detour time of 20 minutes. The depot is located on the boundary of the network. The detailed settings are listed below.

Table 4.1 Experiment setting

Parameter	Value
Depot location	Node 152688 (boundary)
Number of packages	10 ~ 100
Number of SPVs	10 ~ 1,000
SPV capacity	1 ~ 4
SPV max detour willingness	20 $mins$
SPV detour compensation rate	\$ 0.56 $/mile$
SPV package deliver compensation	\$ 0.5 $/package$
Truck capacity	60
Truck per mile cost	\$ 0.56 $/mile$
Truck fixed cost	\$ 120 $/vehicle$

The comparison is conducted between the decomposition heuristic (Algorithm 1) and the exact method of solving the crowdsourced shared-trip delivery problem (Formulation 1, Chapter 3). Algorithm 1 is implemented in Python 3.7 language, and the exact solution is obtained by using commercial solver Gurobi 9.1. The computation is executed on a 2.20 GHz Intel Xeon Server with 128 GB RAM. The comparison focuses on the computational time and the optimality gaps.

The computational time limit that is set for Gurobi is 1,200 seconds. If Gurobi does not finish the optimization process after 1,200 seconds, the optimality gap between the primal and dual problems is reported along with the optimality gap between D-H and exact solutions. The experiment contains both small-scale problems (e.g., 10 packages, 10 SPVs) to larger-scale problems (e.g., 100 packages, 100 SPVs). The results are summarized in Table 4.2.

The study first compares the computational results for relatively small-scale problems. In the four cases of 10 packages, the optimality gap between the D-H and exact method is from 0% to 0.36%. Optimality gap is calculated by $(Obj_{DH} - Obj_{VRP})/Obj_{VRP}$. The computational time for D-H ranges from 10% to 25% of the computational time of exact method. In the four cases of 20 packages, D-H even solves the problem with smaller fraction of time comparing to an exact method. While at the same time, D-H maintains the optimality gap to be smaller than 1.5%. To summarize, in small-scale problems, D-H achieves solutions fairly close to the true optimal solution (less than 1.5%), while uses much less time for computation than the exact method.

For problems with large scales, the exact method starts to slow down with commercial solvers. For the four cases with 50 packages, Gurobi could not find the optimal within time limit. For the case of 50 packages and 50 SPVs, the exact method provides a solution better than D-H, but the optimality gap is smaller than 1% (0.94%). For the cases of 100 SPVs and 250 SPVs, the exact method could not turn out a solution better than D-H within time limit. Moreover, for cases larger than 50 packages and 500 SPVs, with the exact method, Gurobi could not finish pre-computation within the time limit and provides no optimality gaps between the primal and dual problems. Therefore, for large-scales problems, the D-H dominants the exact method.

Comparing the computational time of different cases with D-H, one could find that the scalability of D-H is decent under current settings. The computational time increase is relatively linear (10 packages 100 SPVs takes 11.3 secs, while 10 times the problem uses 96 secs).

In summary, by comparing computational results for different cases, the study finds that the D-H algorithms is scalable. For small-scales problems, the algorithm achieves close solutions to the optimal objective value with a fractional of time that is used by the exact method. For large-scale problems, the algorithm outperforms the exact method within the time limit.

Table 4.2 Computational results comparison between decomposition heuristic and exact m-VRP

Package No.			m-VRP	D-H	m-VRP	D-H	m-VRP	D-H	m-VRP	D-H
	10	SPV No.	10		20		50		100	
		Time (s)	3.11	0.33	2.06	0.38	4.14	0.83	38.87	11.33
		Cost	134.91	134.91	134.43	134.91	134.43	134.91	134.22	134.22
		OPT Gap	0.00%		0.36%		0.36%		0.00%	
	20	SPV No.	20		40		100		200	
		Time (s)	36.93	0.67	88	1.06	221.74	18.55	1105.26	25.54
		Cost	138.69	140.72	138.69	140.44	138.69	140.44	138	138.2
		OPT Gap	1.46%		1.26%		1.26%		0.14%	
Package No.	**50**	SPV No.	50		100		250		500	
		Time (s)	1200	3.83	1200	11.21	1200	29.89	1200	60.48
		Cost	147.93	149.33	157.81	149.33	272.4	149.33	-	149.33
		OPT Gap	0.94%, (13.3%) *		-5.37%, (51.5%) *		-45.18%, (93.99%) *		-	
	100	SPV No.	100		200		500		1000	
		Time (s)	1200	11.62	1200	13.72	1200	41.15	1200	96.94
		Cost	-	283.61	-	283.47	-	192.71	-	186.9
		OPT Gap	-		-		-		-	

OPT Gap shows	1. The optimality gap between D-H method and real optimal (m-VRP) if Gurobi turns out a solution in 1200 secs.
	2. The optimality gap and the dual gap (in parenthesis) between the solutions of primal and dual problem in if Gurobi doesn't finish in 1200 secs. (Mark with *)
	3. Nothing if Gurobi doesn't start cutting planes in 1200 secs (Mark with -).

4.7 Summary of the solution algorithms

The previous chapters describe the decomposition heuristic and details of the algorithms that is used in every step. The procedure of decomposition heuristic involves solving different subproblems, and for every subproblems, different solution techniques are required.

For the first subproblem, *budgeted k-shortest paths problem,* an algorithm (Algorithm 3) with a time complexity of $O(BV(E + V\log V)$ is presented. In addition, the study has discussed the necessity of generating the entire set of *k-paths.* For the large-scale package/vehicle route assignment problem, the dissertation presents the formulation and explain the procedure for a Benders decomposition when the problem size is huge. For the vehicle routing problems, the study applies an insertion heuristic, which handles the problem efficiently and provides important cost reference for the comparison in the next solution step. For the package switch procedure, the dissertation utilizes the cost-driven approach that effectively decides the packages to be shifted from the SPV set to the DV set. The dissertation also compares the decomposition heuristic with the exact method of solving the *MFOCVRPTW*, the decomposition heuristic is comparable with the exact method in solution quality for small scale problems and outperforms the exact algorithms for large-scale problems.

For further improvements of the decomposition heuristics, one may consider further improving the solution efficiency of the *budgeted k-shortest paths problem*, and the large-scale matching problem.

The next chapter applies the decomposition heuristic in real-world case study and presents the results for different metrics.

-End of Chapter 4-

Chapter 5 Application of Models and Algorithms: A Case Study in the City of Irvine

5.1 Introduction

This chapter presents an application of the models and algorithms described in previous chapters. The chapter attempts to model the crowdsourced shared-trip delivery behavior in the City of Irvine. The problem instance description is detailed below.

This application is a large-scale deterministic case of the crowdsourced shared-trip problem. Information about driver trips and packages are available to the analysts or operator significantly in advance of the first driver trip and first package delivery. The study assumes that a depot is in the city and responsible for distributing packages in the service area. The depot could be a distribution center, a store (e.g., Walmart, Whole Foods, etc.), or a warehouse. The parcel sizes are small to medium size and can fit in a normal sedan or wagon (shared personal vehicles, or SPVs). Vehicle drivers, who are willing to deliver packages, may register their trip plans to the depot. Each SPV trip plan must include origin (*O*), destination (*D*), earliest starting time (*EST*), and latest arrival time (*LAT*). The driver also indicates the maximum number of packages they are willing to transport. The operator then determines the combination of packages and drivers. There are also a group of trucks on duty for delivery. The following flow chart explains the service process.

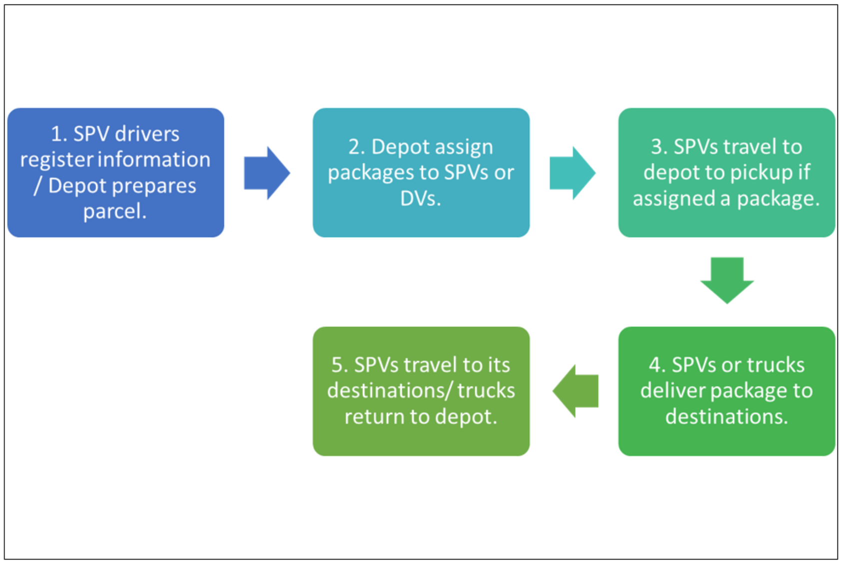

Figure 5.1 Process of crowdsourced shared-trip delivery

The problem is a crowdsourced shared delivery problem, which fits the formulation explained in Formulating the Crowdsourced Shared-trip Problem. To solve the problem, the study applies the algorithms described in Solution Algorithm The application chapter mainly analyze the results and findings and discuss about the factors that may affect the operation of crowdsourced shared-trip delivery.

5.2 Literature review and background information

The methodological aspects of literature have been reviewed in Chapter 2

And Chapter 3. This literature review focuses on parameters that would affect the operation of crowdsourced shared-trip delivery.

The operation usually contains only one depot. A slightly more complex operation is to use multiple depots and allow vehicles to transfer between depots (Arslan et al., 2019). The complex design of operations provides more options for package flow, but does not change the key factor for operation, which is the availability of vehicles. Therefore, this study only considers a single depot design. However, I believe that the location, or using network study jargon, the connectivity of the depot has a large impact on the availability of shared vehicles. The following sections presents the sensitivity analysis results of different depot locations.

Compensation schemes, or incentives of the service, is another factor that impacts the quality. In several studies(Archetti et al., 2016; Arslan et al., 2019; Dahle et al., 2019), researchers all mention the importance of a payment scheme to the reduction of total cost. Different payment schemes, such as fixed amount, variable amount, or grocery store coupon award are suggested and tested for payment. Fixed compensation means the driver would receive a fixed amount (e.g., $1) for each package that they deliver. Variable compensation usually depends on the travel distance or detour distance of drivers. (Archetti et al., 2016) suggest that a variable compensation scheme would benefit the driver more. This dissertation follows this conclusion and treats the payment amount as variable, but it also includes a fixed amount for each successful delivery. Hence, this study applies a mixed payment schedule, which is different from studies in the literature. Besides the payment schemes, another cost factor, cost of dedicated vehicles is sometimes overlooked in the literature. This study includes the fixed cost of using a truck, which is the summation of facility, administrative, and miscellaneous cost, into the decision process of package assignment.

In addition to payment scheme, the effectiveness of crowdsourced shared delivery depends significantly on the detour willingness/buffer time of a shared vehicle drivers. Studies, such as (Archetti et al., 2016), show that higher willingness to detour results in lower costs. This conclusion is also tested in this dissertation chapter. To sum up, Figure 5.2 complies the connection between the potential factors.

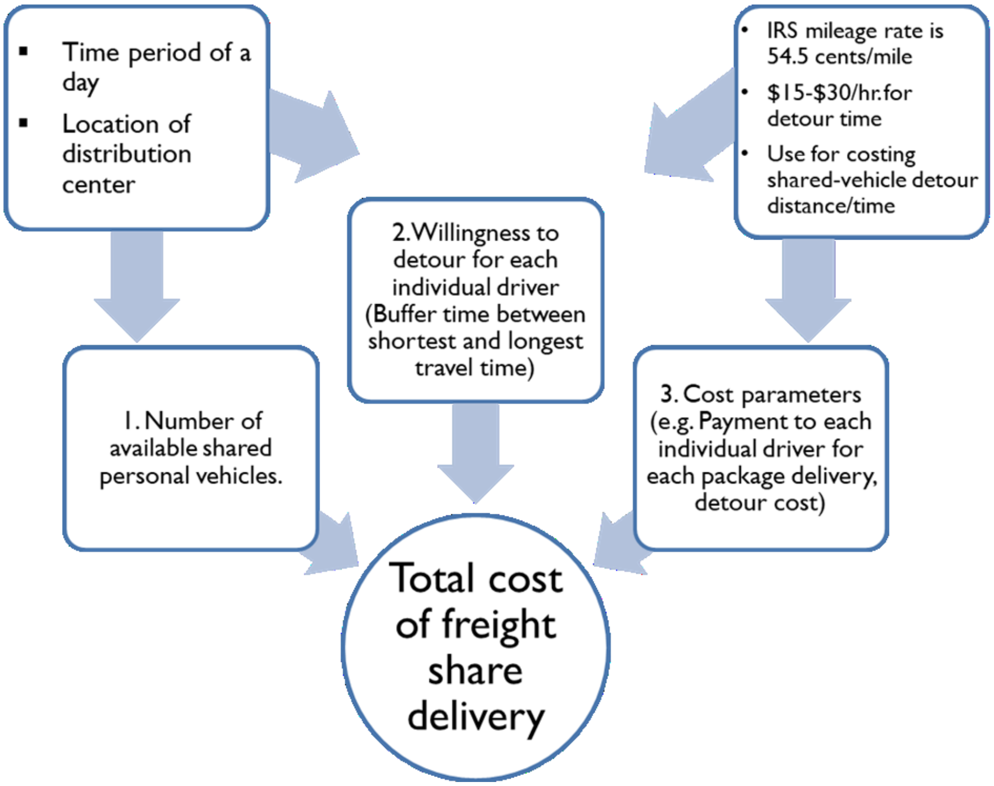

Figure 5.2 Factors that affects crowdsourced shared-trip delivery

This chapter utilizes the results of optimizing a crowdsourced shared-trip delivery problem to answer the following questions:

1. What is the cost and VMT savings for crowdsourced shared-trip delivery service? What are the differences between drivers starting from origins and starting from the depot?
2. What is the impact of driver willingness to detour on the operations and operational costs? What percentage of packages can be served by SPVs?
3. What is the impact of the location of the depot? To what extent does it impact the available vehicles?

The following sections present the detailed numerical study parameters and demonstrate the results.

5.3 Case study settings

This study conducts a numerical case study using the road network of the City of Irvine, CA, USA. The network contains 442 nodes and 648 links. Two nodes are selected as potential depots for package delivery. Depot 1 is on the boundary and in a plaza that could potentially be a grocery store location. Depot 2 is close to the center of city; the node is also in a plaza of a wholesale club. In the benchmark case, the study uses Depot 1 as the depot. In the sensitivity analysis, the study compares the results of using Depot 2 to Depot 1.

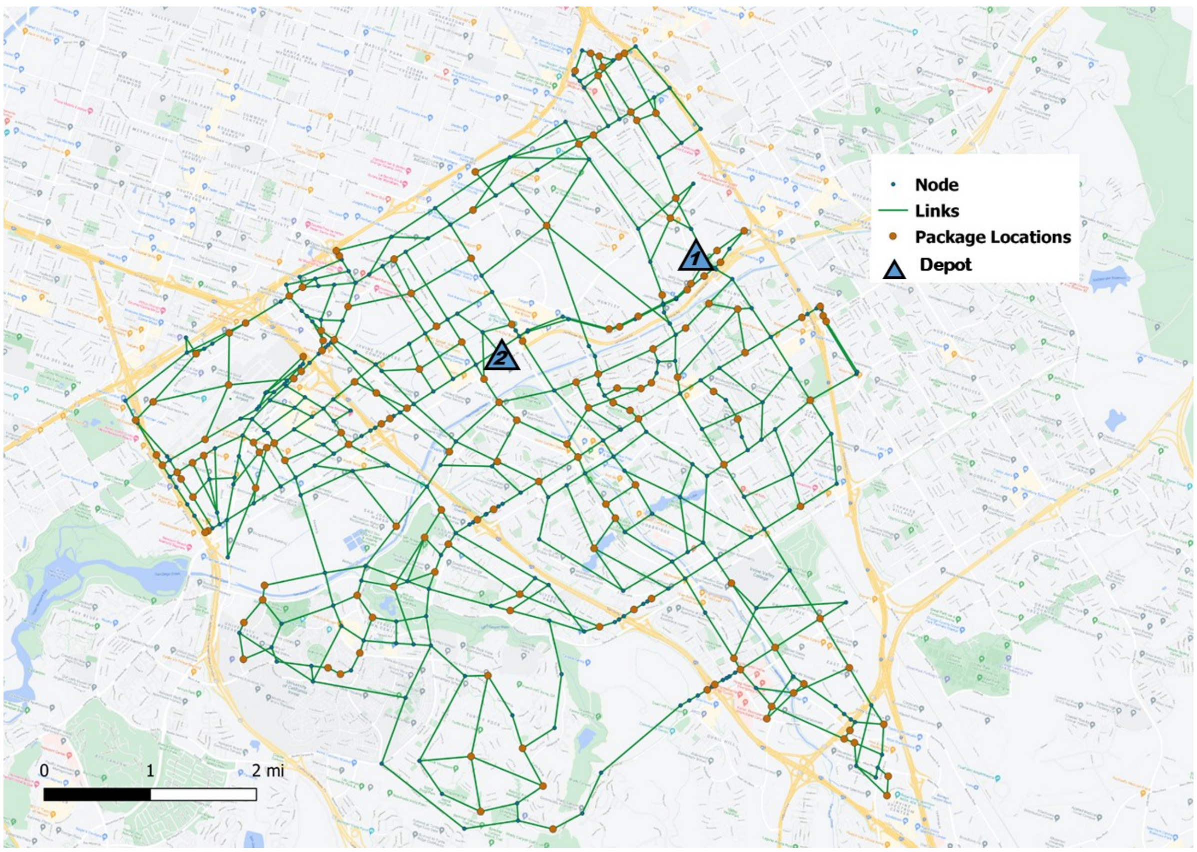

Figure 5.3 Irvine Network. Depot and Package Locations

The numerical study includes 200 packages and a maximum of 1200 SPVs. The package locations are randomly chosen, uniformly distributed in the entire area. Multiple packages may share the same location. Each package has a latest time that it needs to be delivered (time window). The latest delivery time for packages is randomly chosen from 12:00 at noon, 4:00 p.m. or 8 p.m. The SPV trips are generated from the California State Travel Demand Model (CSTDM). An individual SPV has a capacity limit for carrying packages. The driver has an EST and an LAT. The maximum detour willingness of a vehicle is calculated by $LAT - EDT - Shortest\ Path\ Time$. The study will test detour willingness of 10, 15, 20 and 25 minutes. The vehicles are assumed to travel at an average speed of 45 miles per hour. The study also assumes that once a package is matched to an SPV driver, the driver will not reject the assignment. The SPV drivers are compensated based on the number of packages assigned to them (a fixed amount of $0.5 per package) and the detour distance that is incurred from the delivery (the study uses the IRS reimbursement rate for business travel, $0.56/mile).

There are also a number of trucks available at the depot. The trucks are responsible for delivering the packages that are not served by the SPV. Trucks are identical and have a capacity of 60 packages per trip. A truck has a fixed cost per trip of $120, which includes facility cost, administrative cost, and miscellaneous cost. The truck also has a variable cost depending on the mileage it travels. Variable cost includes fuel cost, insurance, and depreciation and has the value of $0.56 as well.

The following is the summary table of the parameters used in the numerical study.

Table 5.1 Summary of parameter values for numerical study

Parameter	Value
Area	32 $mile^2$
Depot location	152688 (Depot 1) *, 131052 (Depot 2)
Number of nodes	442
Number of delivery orders	200
Number of SPVs	0 ~ 1,200
SPV average speed	40 mph
SPV max stop willingness	1 ~ 4
SPV max detour willingness	20, 25, 30*, 35 mins
SPV detour compensation rate	$ 0.56 per mile
SPV package deliver compensation	$ 1.5 per delivery order
DV average speed	30 mph
DV max stop numbers	60
DV per mile cost	$ 1.5 per mile
DV fixed cost	$ 120 per use

* The benchmark case

The numerical study results are presents in the following section.

5.4 Results

The case study uses the numerical case where the depot is at Node 156288 (Depot 1 in Figure 5.3) and the SPV detour willingness equals 20 minutes as the benchmark situation. The numerical study obtains results for cases where the number of SPVs range from 100 to 1,200.

Number of delivery orders served by SPVs

The first metric to compare in this study is the number of delivery orders served by SPVs. According to Chapter 4, the delivery orders are matched to the SPV first (an initial matching), and then possible delivery orders are switched from SPVs to DVs (final matching). This section presents the matching number for both initial matching and final matching. Figure 5.4 Package served by SPVs and trucks draws the initial and final matching numbers against the number of SPVs.

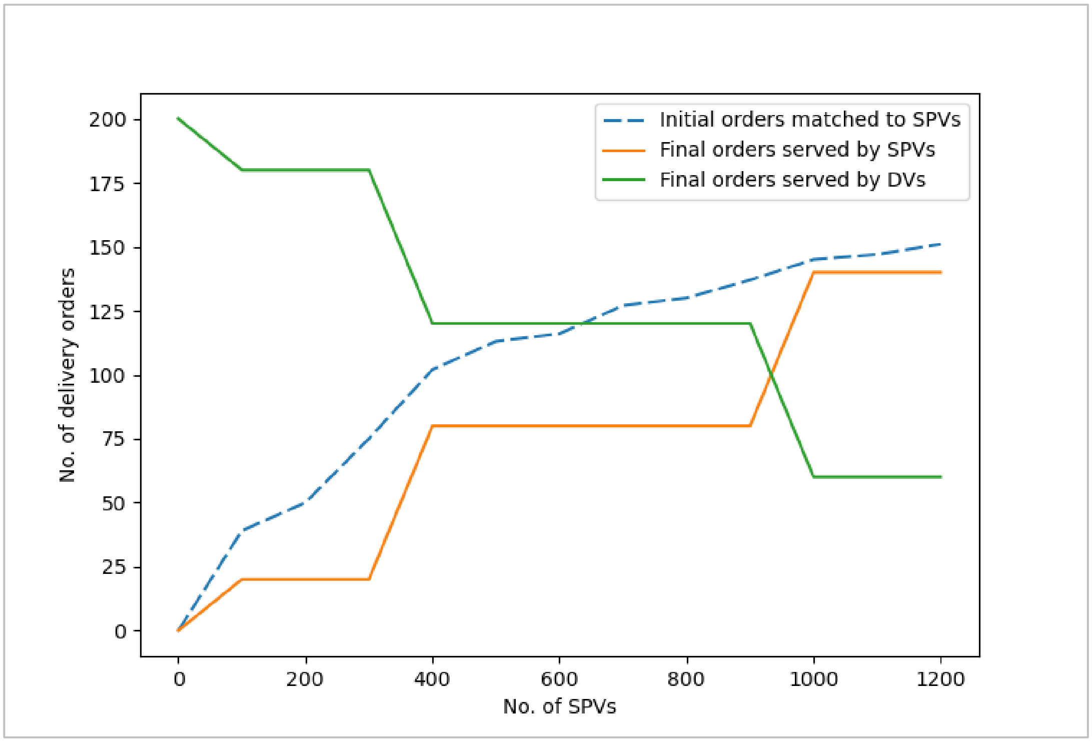

Figure 5.4 Package served by SPVs and trucks

The blue line in Figure 5.4 represents the initial number of orders matched with SPVs. Section 4.3 has shown that the initial matching number is also the largest possible number of delivery orders that could be served by SPVs. Unsurprisingly, the blue dash line shows a trend that the maximum number of package orders that could be served by SPVs increases as the number of SPVs increases. The final SPV served order number (orange line) is bounded by the blue dash line. The trend is a curve bending toward the x-axis, which indicates that the marginal increase in the number of SPV served packages decreases as the number of SPVs increases.

It is also worth noting that even when the number of SPVs reaches a relatively high level, the entire deployed SPVs could not serve the entire group of delivery orders (matching rate at around 75%). If the company keeps increasing the number of SPVs, it is possible that all the orders will be served by SPVs, and no truck will be needed. However, the total number of SPVs required may be an enormous number, which leads to a question whether the logistics company could obtain sufficient supply of SPVs (the matching of packages and SPVs are also highly dependent on the detour willingness of SPVs). Hence, in nearly all cases, at least one dedicated vehicle is needed to serve packages.

The orange line in Figure 5.4 represents number of SPV served orders at the optimal or near-optimal cost; the orange line forms a step function. The green line represents the number of delivery orders served by DVs, which are usually at its maximum stop times. These findings indicate that the optimal solution involves utilizing DV to full capability in operation but also minimizing the number of DVs in operation. When the number of SPVs increases to a level that the feasible SPVs could serve a truckload of delivery orders, the number of DVs required to deliver orders drops by one, and the total cost drops to a lower level. The results about total cost are shown in the following subsections.

Delivery cost

Total delivery cost is the most critical metric as it is the objective function for the original problem formulation. Figure 5.5 displays SPV cost, truck cost, and total delivery cost as a function of the number of SPVs in the system. Figure 5.5 displays the per package costs for SPVs, DVs, and all vehicles combined.

Figure 5.5 and Figure 5.6 illustrate that the total delivery cost (and by definition average cost) decreases as the number of available SPVs increases. This finding is expected given more SPVs represent a larger feasible region or more options for package orders to be delivered by SPVs. Consistent with the results in Figure 5.4, the total delivery cost in Figure 5.5 reduces in steps as the number of SPVs increases. This is, again, the result of reducing the number of DVs necessary to deliver the packages.

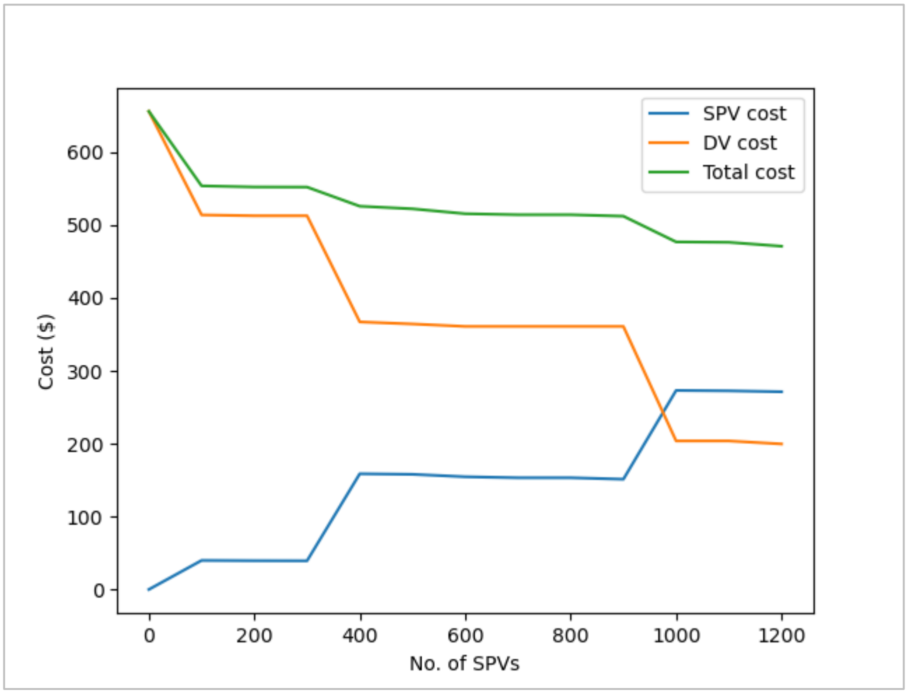

Figure 5.5 Cost Comparison

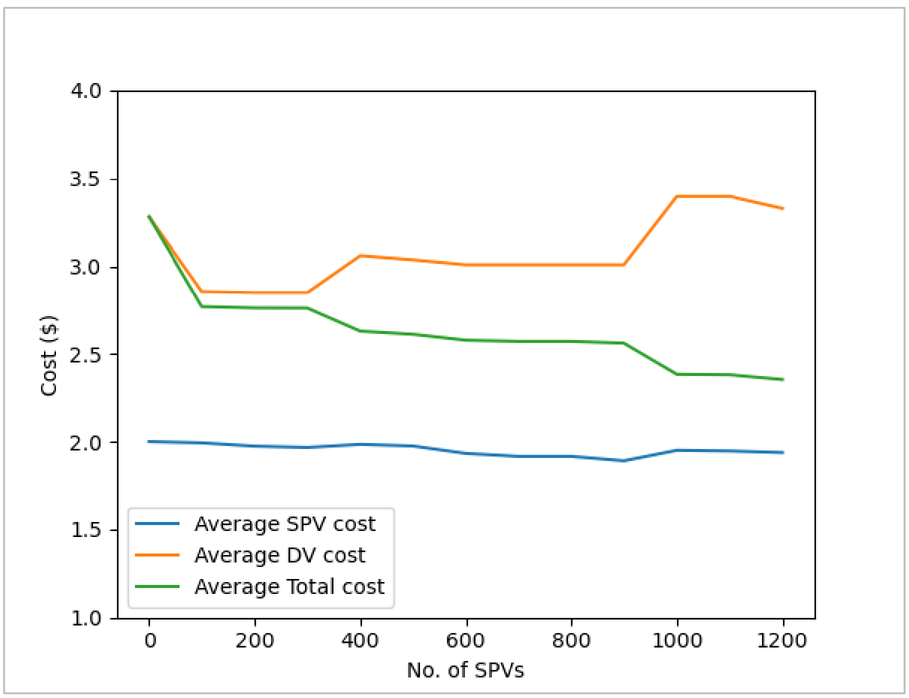

Figure 5.6 Average Cost of Package Delivery

The SPV cost line in Figure 5.5 shows an increasing trend while both truck (orange line) and total cost (green line) are decreasing. SPV costs are increasing with the number of SPVs due to the increase in packages served by SPV, while truck costs are decreasing due to the decrease in packages served by trucks or more accurately the decrease in number of trucks needed to serve packages. This result also indicates that the savings from crowdsourced share delivery are mainly the result of truck fixed cost reduction. The fact that total costs are decreasing with the number of SPVs indicates that the marginal increases in total costs across SPVs are smaller in magnitude than the marginal decrease in total costs across trucks.

The results show that if properly managed, logistics companies that employ crowdsourced delivery service can reduce the number of dedicated trucks they purchase as well as the associated storage, maintenance, insurance, and other overhead costs associated with each truck.

Figure 5.6 shows that total cost is not entirely flat between vehicle size 400 to 900. It is decreasing relatively slowly. The decrease is caused by SPV cost saving due to additional SPVs, but the cost reduction before reaching a truck load is not significant. Table 5.2 also substantiates that freight share delivery saves cost and the cost saving percentage ranges from 20% to 50%. Since it is hard to serve all packages using SPVs, it is hard to eliminate truck service, and the cost saving may stall at 50% for even higher number of SPVs available.

Besides the total cost, the average cost of delivering a package is also decreasing as the number of SPV increases. Both Figure 5.6 and Table 5.2 **Delivery cost by SPVs and trucks** demonstrate the trend. It is worth noticing that while the overall average delivery cost is decreasing, the average truck delivery cost increases slightly as the number of SPVs increase. This reason is that once packages are assigned to SPVs, the benefits from scale economy start to diminish for trucks. The average SPV delivery cost is relatively stable around $2, which indicates that the crowdsourced shared-trip delivery program has a decent utilization of the SPV trips.

Table 5.2 Delivery cost by SPVs and trucks

	Orders served by		Cost (S)				AVG cost per order ($)		
SPV Number	SPV	DV	SPV	DV	Total	% Saving w.r.t SPV=0	by SV	by DV	overall
0	0	200	0.00	656.06	656.06	-	-	3.28	3.28
100	20	180	39.86	513.94	553.79	15.59%	1.99	2.86	2.77
200	20	180	39.48	512.81	552.30	15.82%	1.97	2.85	2.76
300	20	180	39.33	512.81	552.15	15.84%	1.97	2.85	2.76
400	80	120	158.75	367.13	525.88	19.84%	1.98	3.06	2.63
500	80	120	158.01	364.31	522.32	20.39%	1.98	3.04	2.61
600	80	120	154.65	360.94	515.58	21.41%	1.93	3.01	2.58
700	80	120	153.30	360.94	514.24	21.62%	1.92	3.01	2.57
800	80	120	153.30	360.94	514.24	21.62%	1.92	3.01	2.57
900	80	120	151.29	360.94	512.22	21.92%	1.89	3.01	2.56
1000	140	60	273.09	203.81	476.91	27.31%	1.95	3.40	2.38
1100	140	60	272.57	203.81	476.38	27.39%	1.95	3.40	2.38
1200	140	60	271.30	199.69	470.99	28.21%	1.94	3.33	2.35

Vehicle Miles Travelled (VMT)

Besides the total cost, the total vehicle mile travelled (VMT), a proxy for both congestion and environmental impact, is another important performance metric. Figure 5.7 and Table 5.3 present the total VMT to deliver packages for SPVs, trucks and overall vehicles.

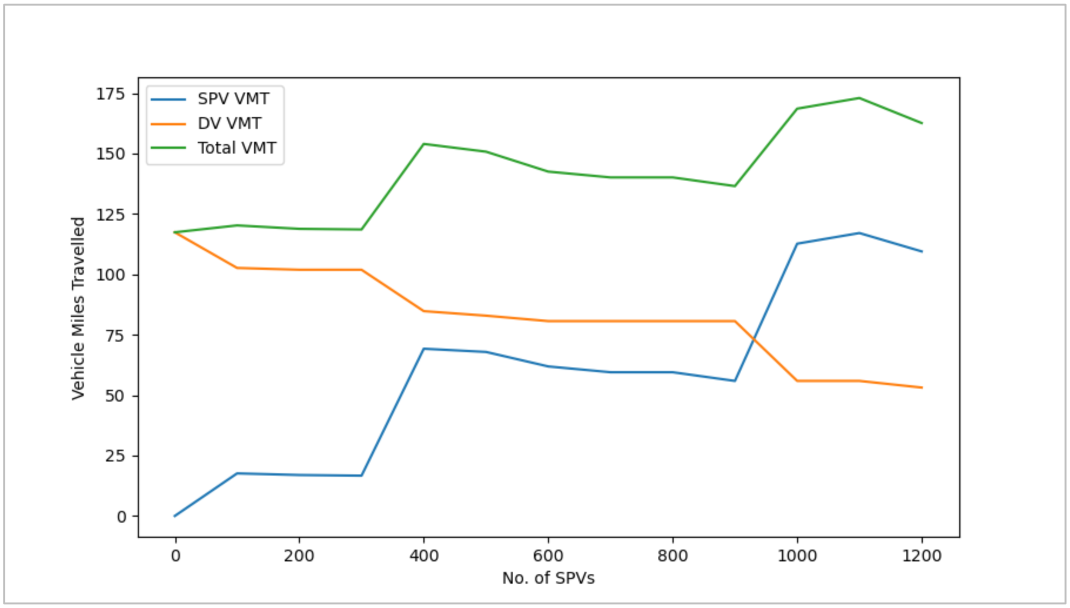

Figure 5.7 VMT from Package Delivery

The results indicate that total VMT tends to increase with the number of SPVs, although non-monotonically. This finding is unsurprising given that as the number of SPVs increases the number of delivery orders assigned to SPVs increases thereby causing more SPVs to detour to pickup and drop-off packages, as opposed to relatively VMT-efficient DV delivering orders.

Given that VMT increases with the number of SPVs, the question then becomes, does the crowdsourced delivery system increase congestion and worsen environmental impacts of package delivery? The answer for congestion is most likely 'yes' unless the DVs were blocking traffic lanes in dense urban areas. The answer for environmental impact is more nuanced. Since SPVs are family size sedans, the SPVs themselves are significantly more energy efficient and environmentally friendly than trucks or vans that dedicated delivery may use on a per mile bases. Therefore, crowdsourced shared-trip delivery does likely reduce environmental emissions relative to exclusive dedicated truck delivery, unless the trucks are fully electric.

Also, worth mentioning is that the VMT estimation for each individual SPV in Figure 5.7 includes SPV travel from the SPV driver's origin to the depot. If the logistics company only considers people who shop at the stores near the depot, which is the case in almost all previous studies (Archetti et al., 2016; Arslan et al., 2019; Dayarian & Savelsbergh, 2020), then total VMT decreases with the number of SPVs, as shown in Figure 5.8. In Figure 5.8, all the solid lines that are marked with "1" (indicating Case 1, SPVs have origins as their origins) are the same as Figure 5.7, while the dash lines represent the case (Case 2) where the case study assumes all SPV drivers are at the depot/store. While the VMT gap between DVs is relatively small in both cases, the gap between the VMT from SPVs is quite drastic in the two cases. Hence, the VMT savings from 'assuming' all the SPVs are already at the store/depot drives the overall savings in VMT between the two green lines. The dash green line is consistent with much of the literature and indicates that share-ride delivery saves VMT.

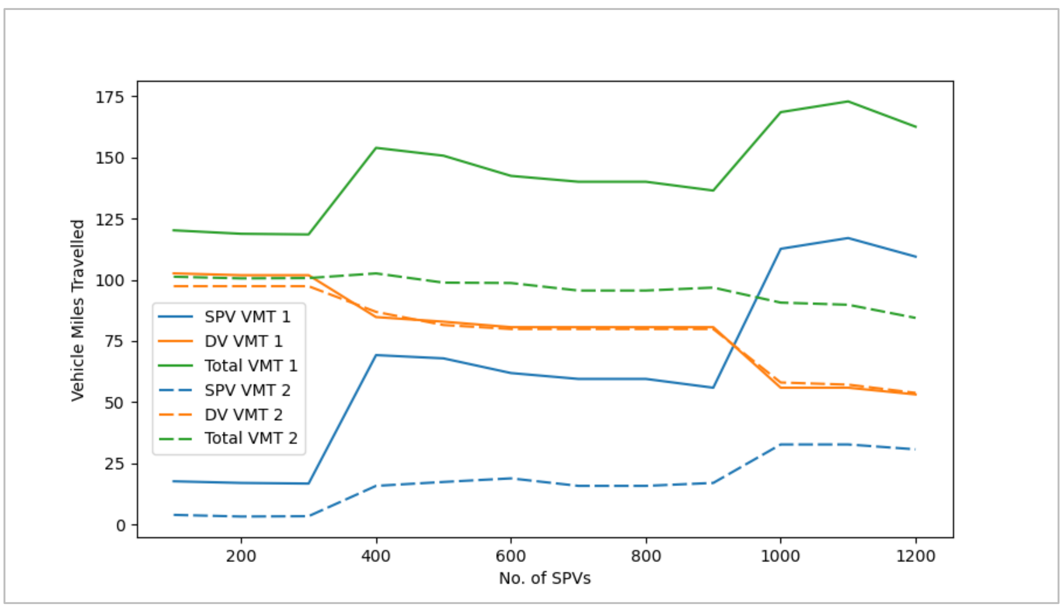

Figure 5.8 VMT under different origins of SPVs

Table 5.3 VMT for package delivery

	Orders served by		VMT (miles)				AVG VMT per order (miles)		
SPV Number	SPV	DV	SPV	DV	Total	% Saving w.r.t SPV=0	by SV	by DV	overall
0	0	200	0.00	117.38	117.38	-	-	0.59	0.59
100	20	180	17.60	102.63	120.23	-2.43%	0.88	0.57	0.60
200	20	180	16.93	101.88	118.81	-1.22%	0.85	0.57	0.59
300	20	180	16.67	101.88	118.54	-0.99%	0.83	0.57	0.59
400	80	120	69.20	84.75	153.95	-31.16%	0.87	0.71	0.77
500	80	120	67.87	82.88	150.74	-28.43%	0.85	0.69	0.75
600	80	120	61.87	80.63	142.49	-21.40%	0.77	0.67	0.71
700	80	120	59.47	80.63	140.09	-19.35%	0.74	0.67	0.70
800	80	120	59.47	80.63	140.09	-19.35%	0.74	0.67	0.70
900	80	120	55.87	80.63	136.49	-16.29%	0.70	0.67	0.68
1000	140	60	112.67	55.88	168.54	-43.59%	0.80	0.93	0.84
1100	140	60	117.07	55.88	172.94	-47.34%	0.84	0.93	0.86
1200	140	60	109.47	53.13	162.59	-38.52%	0.78	0.89	0.81

Vehicle route usage

This section analyzes the SPV (route) usage from the supply perspective to understand how the SPVs are used. The study defines an SPV with minimum possible detour as an SPV such that it travels to depot on the shortest path and then travels to its destination on the shortest path while delivering at least one package order. The route that the SPV traverses is called a minimum detour route. The numerical study finds that the average total detour distance for SPVs with delivery tasks is 1.06 miles. In addition, 14% SPVs uses its minimum detour route. These results are not abnormal since the algorithm also attempts to route SPVs on shorter routes to save cost. Another more important metric that is presented in this section is the number orders served by SPVs with different detour levels. This metric helps understanding how SPVs are used in crowdsourced share-trip delivery. Figure 5.9 shows the total number of package orders (sum up the orders from 100 SPV case to 1200 SPV case) delivered by SPVs with different detour levels. In Figure 5.9, on the x-axis, “0” indicates the route is a minimum detour route, and the other integer numbers are the travel time differences between the used route and the minimum detour route.

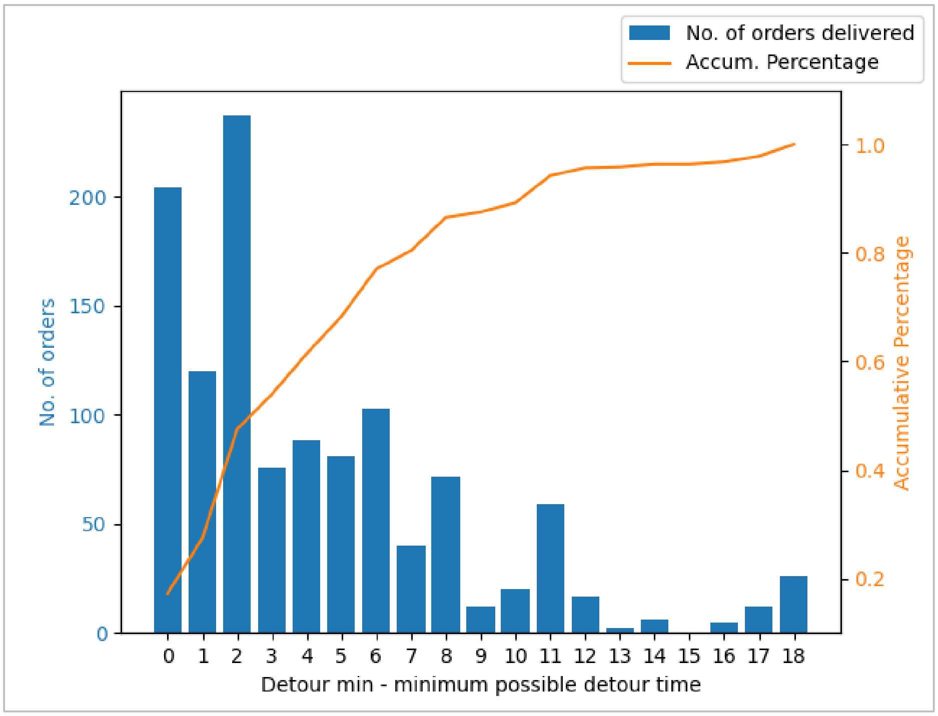

Figure 5.9 The number of packages served by routes with different detours

Figure 5.9 demonstrates that half of the SPV delivered orders are carried by SPVs that travers a route which has a travel time that within 2 mins more than the minimum detour route. Especially 17% of the orders are delivered by SPVs on a minimum detour route. Almost 90% of the SPV served orders are carried by an SPV on a route that is within 10 mins more than the minimum detour route. Therefore, the routes which have a longer detour distance are less preferable for SPVs to use.

The reason may be that long-detour routes, when used, incurs higher detour compensations, which are less preferable for logistics companies. In addition, the compensations that paid to long-detour routes may be ultimately higher than the cost of using a DV to deliver the order, and therefore, are rarely used. This finding suggested that though longer detour of vehicle would potentially serve more package orders, most orders are delivered by SPVs that travels with a short detour distance. If we reduce the maximum detour willingness requirements for SPVs, when the total number of SPV is large, the result would not be impacted significantly. This finding also motivates us to conduct a sensitivity analysis on the impact of the maximum willingness to detour for SPVs. The finding also indicates that for route generation in Section 4.2, one could limit the budget of k-shortest paths so as to reduce computational time for both Step 1 and 2 in Algorithm 1.

Impact of detour willingness

A sensitivity analysis to assess the impact of detour willingness of SPVs is conducted in this sub-section. As explained in both Section 4.2 and Section 5.3, the maximum willingness to detour for an SPV includes the travel time from its origin to depot, the necessary order pickup/drop-off time and the travel time from the depot to the destination. The benchmark case has a maximum detour willingness of 30 minutes for every SPV. The study uses 20, 25 and 35 minutes of maximum detour willingness to generate results and compare them with the benchmark.

The first metrics to compare is the total delivery cost. Figure 5.10 shows the cost comparison over different detour willingness.

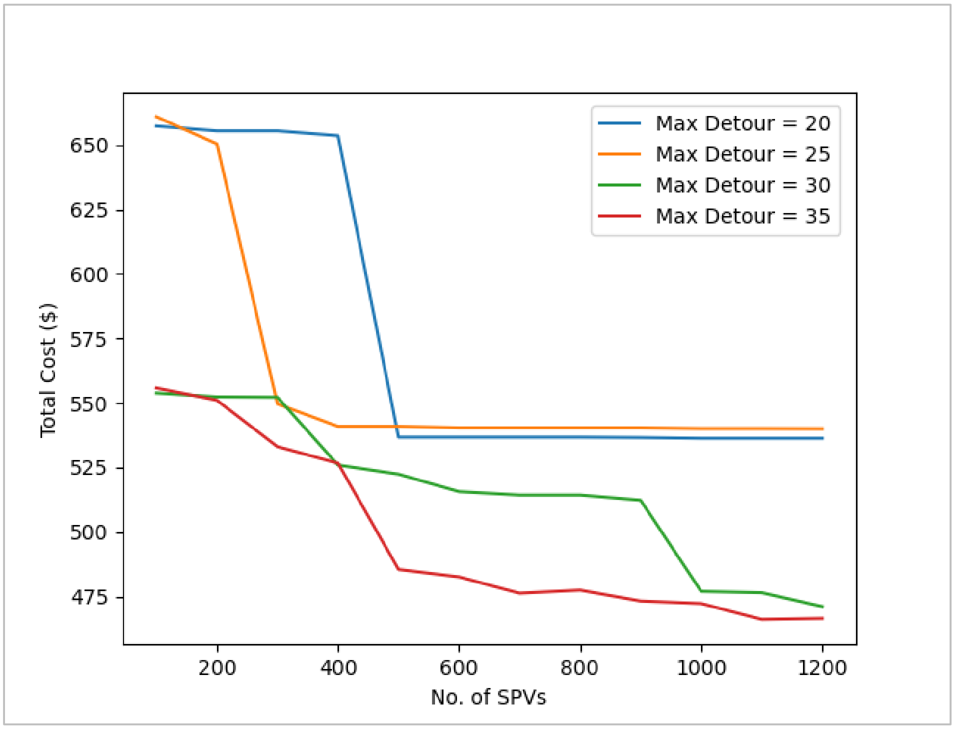

Figure 5.10 Cost Comparison over Detour Willingness

First, for the cases where the maximum detour willingness is 20 and 25 minutes, after the number of SPVs reaches 500, the total cost for delivery stabilizes at around $550. Further reduction in the total cost would require a relatively high number of SPVs to participate in the program. In addition, for the case of 500 SPVs and onwards, the total cost of for 20- and 25-minute cases is almost the same. These two findings indicate that if SPV drivers are all with a low level of detour willingness, achieving high level of cost reduction would require a considerably large number of SPVs (more precisely, a large SPV/order ratio).

Higher detour willingness cases achieve the same cost reduction with fewer SPVs than low detour willingness cases. For example, the 35-minute (red line) and 30-min (green line) cases require only 100 SPVs to achieve a total cost around $550, while the 15-min case needs at least 300 SPVs and the 10-min case requires around 500 SPVs. Naturally, a longer detour time allows

SPVs to travel to more 'unpopular' nodes that may have a delivery demand. Therefore, it would be interesting to examine the maximum number of delivery orders that all SPVs could serve under different scenarios. Figure 5.11 demonstrates the results of this analysis.

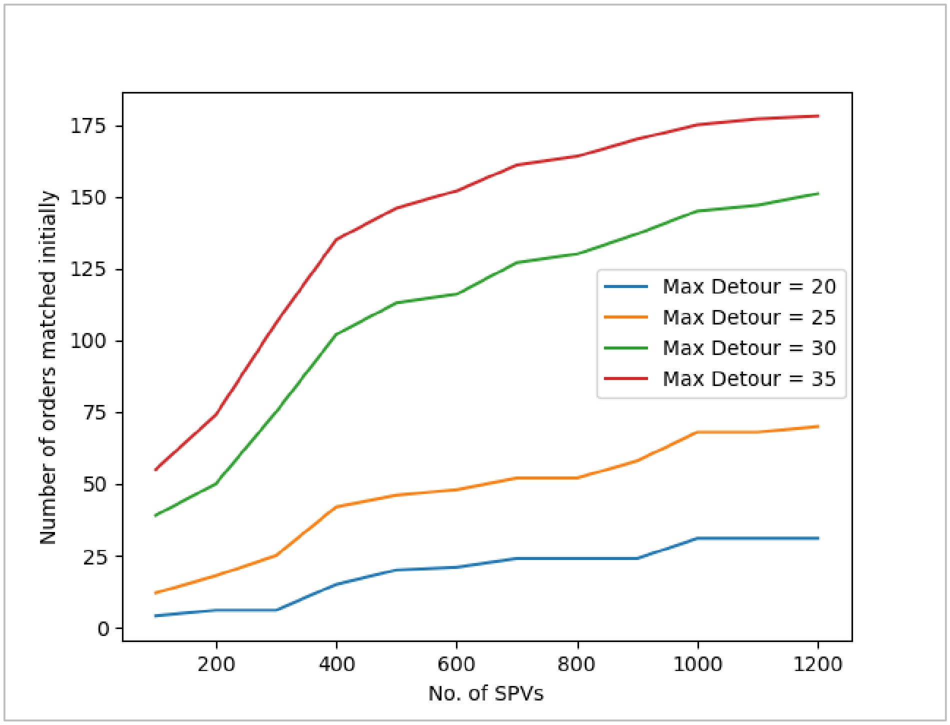

Figure 5.11 Maximum delivery orders that SPVs could serve

Figure 5.11 shows that with the same number of SPVs, the 35-min detour can serve 4 to 6 times more the orders than the 20 or 25-min cases. Moreover, compared to the 35-min case, the 30-min case can significantly increase the number of potential orders served. On the contrary, the additional orders that SPVs can serve between 35-min and 30-min cases are fewer than the improvement between 30- and 25-min cases. While these results are a function of the network structure, for the Irvine network the results indicate that the logistics company should incentivize drivers to accept a 30-min detour time.

A related parameter of interest is the percentage of SPVs that can deliver at least one order without violating any time-window constraints -- $(no. feasible\ SPVs/total\ SPV\ number)\%$ -- as a function of the maximum willingness to detour. According to computational results for the Irvine case study with a 20-min detour willingness, only 2% ~3% of SPVs are feasible SPVs. While with a 25-min detour, the percentage increases to 18%. For the 30-min and 35-min cases, the feasible SPV percentages are 40% and 52%. This result substantiates the finding that a decent detour willingness significantly increases the potential of the crowdsourced shared-trip delivery service. Without a decent detour willingness for SPVs, a considerable large number of SPVs will be needed to mitigate the differences.

This study also compares the SPV usage across different scenarios in Figure 5.12. Figure 5.12 ascertains the aforementioned finding that a decent detour willingness (30-, 35-min cases) significantly improves the potential of delivering orders by SPVs. In addition, it shows that in different maximum detour willingness scenarios, the majority of orders delivered by SPVs are carried by SPVs that uses a low-detour route, meaning that most SPVs do not need to detour a lot from the depot to their own destinations to deliver package orders.

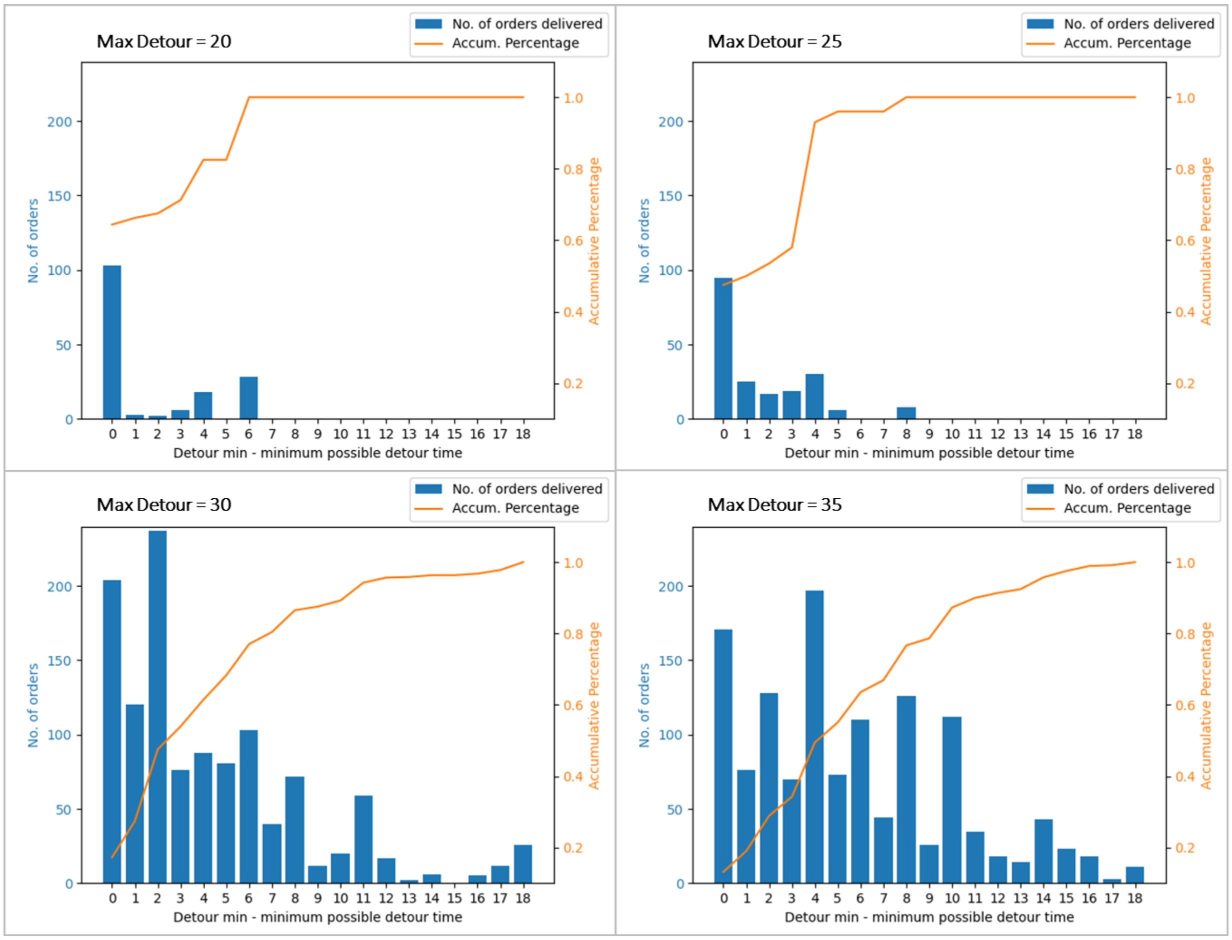

Figure 5.12 SPV usage under different maximum detour willingness

Combining the findings of Figure 5.11 and Figure 5.12, we may conclude that a higher maximum detour willingness would allow more SPVs (especially for those SPVs whose origins and destinations are far away from the depot) to participate in the crowdsourced delivery system. However, individual SPVs who are matched with delivery orders, highly likely do not need to detour a lot when travels from the depot to their destinations. Therefore, in Step 2 of Algorithm 1, we may not necessarily need a large budget to enumerate SPV routes from the depot to their destinations and therefore could potentially save computational time.

Impact of the depot location

This subsection presents another sensitivity analysis to examine the impact of depot locations on several performance measures. As described at the beginning of this section, the study compares two cases: Case 1, where the depot is at the service region boundary; Case 2 where the depot is at the center of the city.

The study first compares the total cost and total VMT based on the depot location. Figure 5.13 shows that, in general, the center depot delivery costs are lower than the boundary depot costs. The reason is that a center located depot attracts more SPVs than boundary located ones. However, when the number of SPVs is high, the difference (about 5.4%) in the total cost between a center depot and a boundary depot is not significant. This finding indicates that a higher number of SPVs can mitigate the total cost deficit caused by depot location selection. Hence, from a managerial perspective, the location of depots may be less important, in terms of system cost, within a crowdsourced shared-trip delivery system than a conventional dedicated vehicles only system.

For the total VMT metric Figure 5.14), in most cases, a center located depot results in less VMT than a boundary located vehicle, since a certain number of SPVs does not need to detour a lot to reach the depot for order pickup. Though the VMT different is significant (25%) for the SPV = 1200 case, the boundary located case does show a trend of reducing. Therefore, it is reasonable to believe that when the number of SPVs reaches a larger level (may be huge), the differences between VMT could also be alleviated.

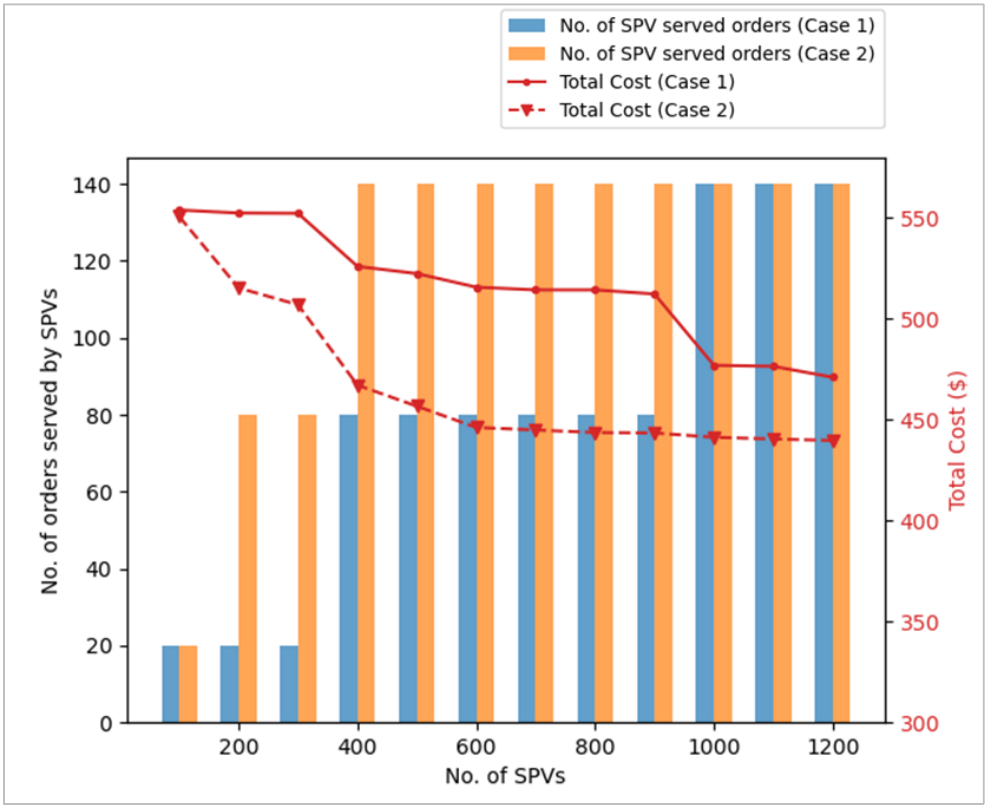

Figure 5.13 Cost Comparison over Different Depots

On the other hand, the depot location affects the percentage of SPVs that can feasibly serve at least one package order. The center located depot tends to produce more feasible drivers than a boundary depot. The percentage of feasible drivers under a 30-min detour willingness assumption for the center located depot case is 60%, while the same metric for a boundary located depot is only 40% (as shown in the previous section). The higher percentage of feasible drivers also leads to a higher matching rate of package orders by the SPV. In Figure 5.15, for the maximum number of packages that could be served by SPV, the center located depot case (orange bars) still outperforms the boundary located depot case (blue bars). Both cases are moving toward 200, the total delivery order number, but still several hundreds of SPVs are needed to deliver all package orders by SPV.

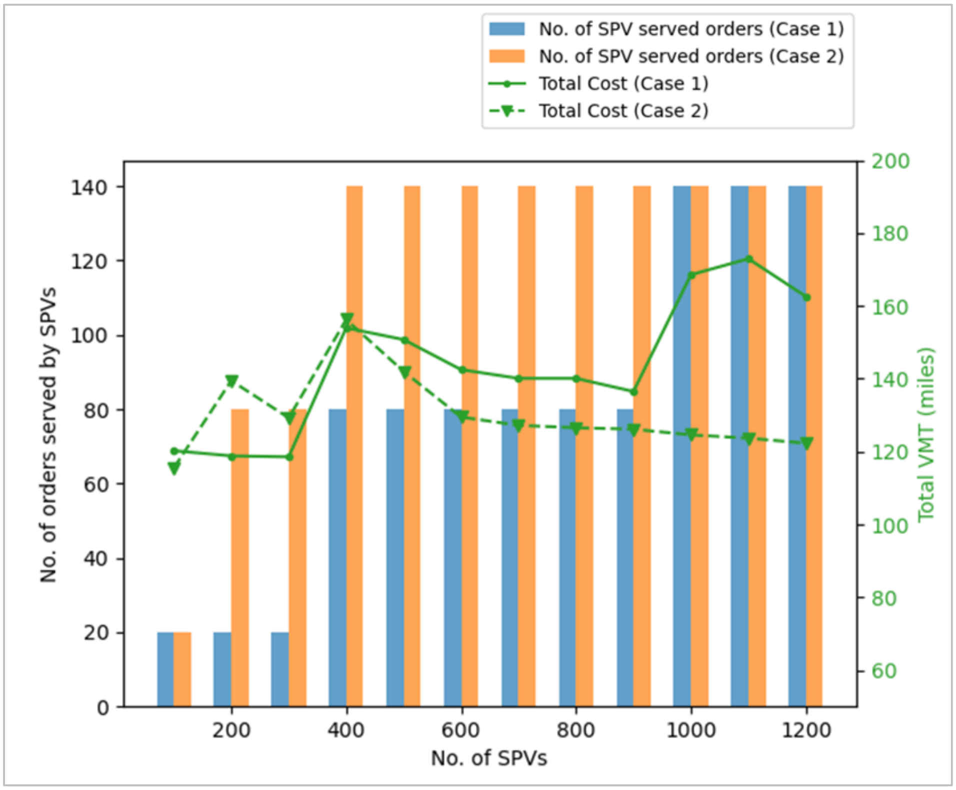

Figure 5.14 VMT Comparison over Different Depots

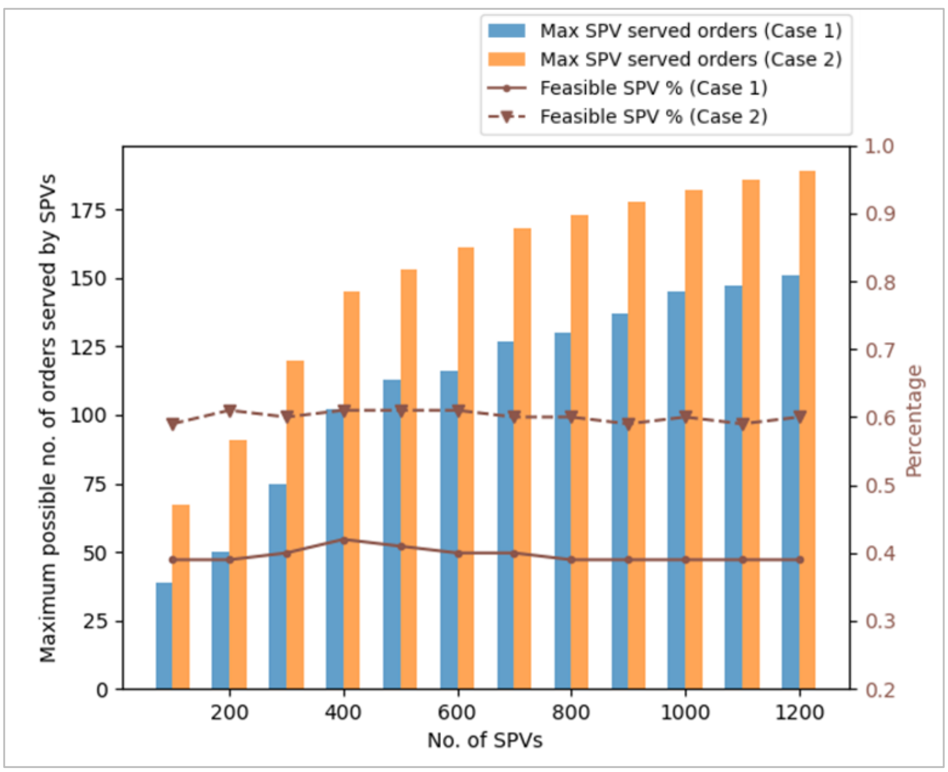

Figure 5.15 Feasible SPV percentages and SPV served packages

5.5 Minimum compensation required

The above discussion and results assume that crowdsourced drivers accept any order assigned to them. Conversely this section assumes, realistically, that crowdsourced drivers can and will reject orders if the expected payment is smaller than their minimum acceptable compensation. The study tests the system performance of crowdsourced shared-trip delivery with different levels of minimum acceptable compensation for drivers and compares the results to the crowdsource time-based delivery. The comparison should help answer the question of when a company should deploy crowdsourced shared-trip delivery, or crowdsourced time-shared delivery, or simply use conventional truck delivery.

Assume a shared personal driver (or simply using SPV k) has a minimum compensation that he would accept to deliver, and the minimum compensation, represented by w_k, is measured by monetarized cost. Assume there are two ways that drivers determine their minimum acceptable compensation. The first one is to compute the minimum compensation by value of time (or detour distance travelled). The analysis introduces the parameter w_k^d to represent the required compensation per mile detour for vehicle k. In the first case, $w_k = w_k^d \times detour\ distance$. The second one is to compute the minimum compensation by the number of packages, represented by w_k^p, and $w_k = w_k^p \times number\ of\ pacakges\ assigned$. The shared vehicle drivers could have different values of $w_k^d\ or\ w_k^p$.

Therefore, another constraint based on the minimum acceptable compensation could be added to Formulation 1 of Chapter 3.

$$\left(c_{0s_k} + \sum_{(i,j)\in A} c_{ij}^s x_{ij}^k - c_{s_k}\right) + e \times \left(\sum_{(i,j)\in A} x_{ij}^k - 1\right) \geq z_{sk} \times w_k, \forall k \in S \quad (5.1)$$

In Case 1, where drivers' minimum acceptable compensation is based on detour time/distance, the constraint is written as:

$$\left(c_{0s_k} + \sum_{(i,j)\in A} c_{ij}^{s} x_{ij}^{k} - c_{s_k}\right) + e\left(\sum_{(i,j)\in A} x_{ij}^{k} - 1\right) \geq z_{sk} \times \left(c_{0s_k} + \sum_{(i,j)\in A} c_{ij}^{s} x_{ij}^{k} - c_{s_k}\right) \times w_k^d, \forall k \in S \quad (5.2)$$

which could be rewritten as:

$$\left(z_{sk} \times w_k^d - 1\right) \times \left(c_{0s_k} + \sum_{(i,j)\in A} c_{ij}^{s} x_{ij}^{k} - c_{s_k}\right) \leq e \times \left(\sum_{(i,j)\in A} x_{ij}^{k} - 1\right), \forall k \in S \quad (5.3)$$

In Case 2, where drivers' minimum acceptable compensation is based on number of packages delivered, the constraint is:

$$\left(c_{0s_k} + \sum_{(i,j)\in A} c_{ij}^{s} x_{ij}^{k} - c_{s_k}\right) + e\left(\sum_{(i,j)\in A} x_{ij}^{k} - 1\right) \geq z_{sk} \times w_k^p \times e\left(\sum_{(i,j)\in A} x_{ij}^{k} - 1\right), \forall k \in S \quad (5.4)$$

which is equivalent to:

$$\left(z_{sk} \times w_k^p - 1\right) \times e\left(\sum_{(i,j)\in A} x_{ij}^{k} - 1\right) \leq \left(c_{0s_k} + \sum_{(i,j)\in A} c_{ij}^{s} x_{ij}^{k} - c_{s_k}\right), \forall k \in S \quad (5.5)$$

Similarly, in Formulation 3 (Chapter 4), when matching packages to vehicle routes, the new constraint for driver acceptable compensation is introduced as follows.

$$e \times \sum_{p} x_{prk} + \omega_d c_{rk} z_{rk} \geq w_k \times z_{rk}, \forall (r,k) \in R \ (5.6)$$

In Case 1, where drivers' minimum acceptable compensation is based on detour time/distances, the constraint is:

$$e \times \sum_{p} x_{prk} + \omega_d c_{rk} z_{rk} \geq w_k^d c_{rk} \times z_{rk}, \forall (r,k) \in R \ (5.7)$$

which can be rearranged to get:

$$e \times \sum_{p} x_{prk} \geq \left(w_k^d - \omega_d\right) c_{rk} z_{rk}, \forall (r,k) \in R \ (5.8)$$

Inequality 5.8 indicates that, if the crowdsourced driver expects more compensation for per-mile detour than the actual amount that the company pays ($if\ w_k^d > \omega_d$), then the compensation per package delivered should be sufficiently large to fill the deficit in order to attract the crowdsourced drivers to work. For a driver k, the minimum required compensation for a single package is calculated as:

$$e_k = max\left[\frac{\left(w_k^d - \omega_d\right) \times detour\ distance}{No. of\ Pacakges\ delivered}, 0\right] (5.9)$$

In Case 2, where the drivers' minimum acceptable compensation is based on the number of packages that are assigned to them, the constraint is:

$$e \times \sum_{p} x_{prk} + \omega_d c_{rk} z_{rk} \geq w_k^p \times \sum_{p} x_{prk} \times z_{rk} (5.10)$$

When $\sum_p x_{prk} \neq 0$ (the driver serves at least one packages), one could eliminate $\sum_p x_{prk}$ from both sides and rearrange the terms:

$$e \geq w_k^p z_{rk} - \frac{z_{rk}}{\sum_p x_{prk}}(\omega_d c_{rk}) \ (5.11)$$

When a route is active and $z_{rk} = 1$, the minimum requirement for per package compensation is:

$$e \geq w_k^p - \frac{\omega_d \times detour\ distance}{No.of\ pacakges\ delivered} \ (5.12)$$

This study assumes that all crowdsourced drivers are homogenous (they choose the same evaluation criteria for the minimum acceptable compensation). This study tests the impact of incentive pricing schemes on the total cost and the alternative operating policies of logistics companies. The following scenarios will be considered:

1. Under crowdsourced shared-trip delivery, crowdsourced drivers are compensated by both the number of packages delivered and the detour distance. The drivers value the minimum acceptable compensation to deliver by a constant.
2. Similar compensation scheme as scenario 1, but the drivers measure their minimum acceptable compensation by their maximum detour time.
3. Similar compensation scheme as scenario 1, but drivers measure their minimum acceptable compensation by the number of packages that are assigned to them.
4. Under crowdsourced time-based delivery, crowdsourced drivers are paid based on the time they work. To determine their willingness to participate, they compare the hourly rate to the income they would otherwise receive from crowdsourced trip-based delivery.

The numerical results of different scenarios are discussed as follows.

Scenario 1: Crowdsourced shared-trip delivery and driver minimum acceptable compensation is based on a constant.

In this scenario, each driver has a lower bound for acceptable compensation, i.e., the payment to a driver must be larger than a constant to attract the driver. Note that compensation is still based on the detour distance and number of packages delivered; hence, in this derived scenario, drivers effectively want to detour more and want to deliver more packages, while not exceeding a maximum detour or time window constraint.

This section varies the minimum payment constant from 0 to 3 and obtains the total cost under different cases. The findings are shown in Figure 5.16.

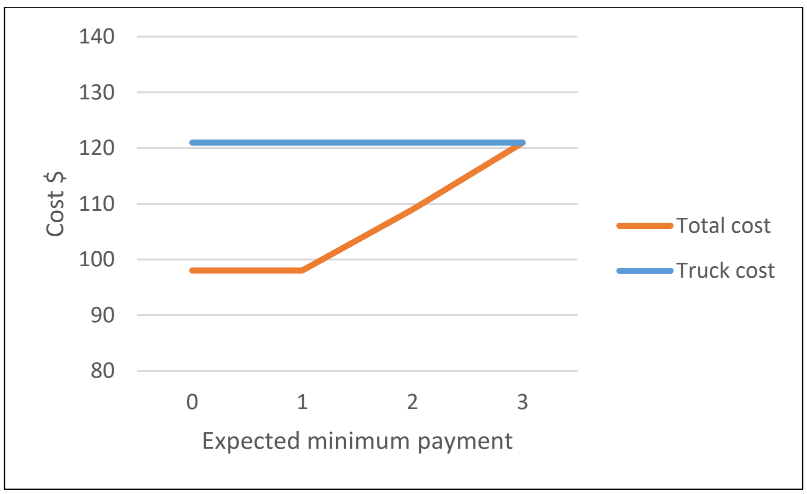

Figure 5.16 Minimum acceptable compensation valued by a constant

Figure 5.16 indicates that when the minimum willingness is one dollar, the total cost is the same as when the minimum willingness of deliver is zero. The finding indicates that most drivers would receive a minimum payment of $1 or more. When the minimum willingness to deliver increases to $2, the total cost increases but is still lower than the conventional dedicated truck delivery. When crowdsourced drivers have a higher minimum compensation requirement, the logistics company tends to assign packages to routes that have longer detour distances to fulfill the driver expectation of higher payment. However, if the requirement becomes too high for logistics companies, it becomes less attractive to contract with crowdsourced drivers and companies would switch to dedicated delivery. This appears to be the case when the minimum compensation is $3, as the total cost becomes equivalent to the truck-only cost.

Scenario 2: Crowdsourced shared-trip delivery and driver minimum acceptable compensation is based on detour time.

In this scenario, drivers measure their minimum acceptable compensation based on detour time. The detour time for a driver is calculated by the total time used for a trip minus the shortest path travel time. The study obtains the total cost based on four seperate value of time (VOT) levels. The comparison is shown in Figure 5.17.

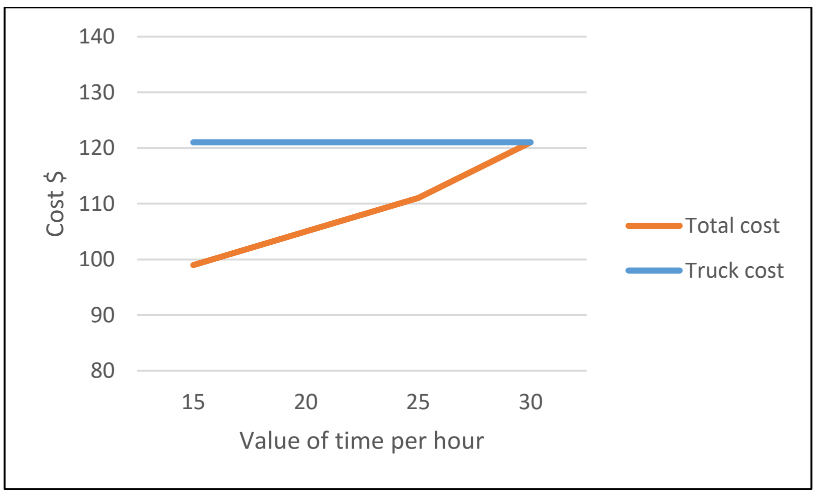

Figure 5.17 Minimum acceptable compensation to deliver valued by VOT

The figure indicates that when crowdsourced drivers have a VOT of less than $30 per hour, crowdsourced shared-trip delivery is a better option than dedicated delivery. The result also indicates that with higher VOT, longer distance routes that can serve multiple packages become favorable, because these routes provide higher return for drivers, but also have a reasonable per package delivery cost.

Scenario 3: Crowdsourced shared-trip delivery where driver minimum acceptable compensation is based on the number of packages delivered.

In this scenario, we found that based on the current compensation schemes, if drivers require a minimum per package compensation larger than 62 cents, dedicated delivery only becomes more preferable for companies. The reason is that in the current setting, about 20% of the packages are delivered by drivers that do not detour. For these drivers, their compensation from the detour time is limited, and therefore, they would require higher compensation per package delivered. The suggestion from the result is that the company may consider separate compensation schemes for drivers that complete delivery tasks with very small detour distances.

Scenario 4. Crowdsourced time-based delivery

In this scenario, the company attempts to contract with drivers, who work for a time period to deliver packages. Arranging package sets for drivers in this scenario is solved by a m-VRP. The drivers require hourly payment that is comparable to alternative crowdsourced trip-based delivery. We vary the alternative payment from $18/hour to $24/ hour and test the total cost of contracting with time-based crowdsourced drivers.

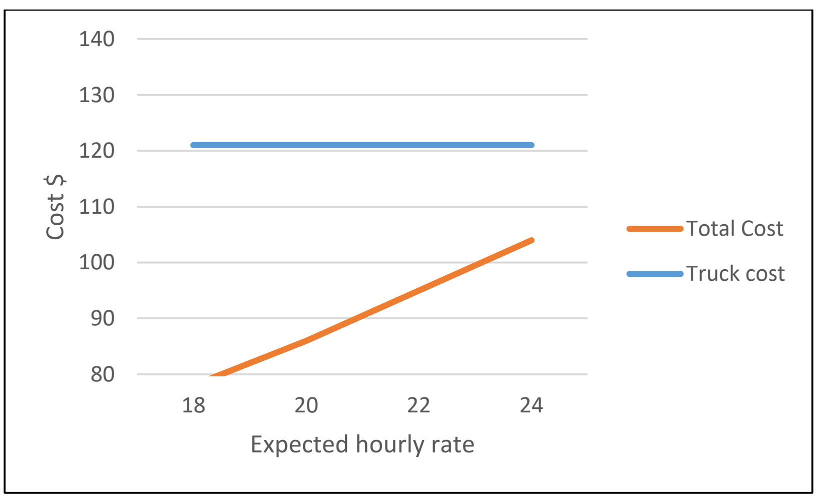

Figure 5.18 Crowdsourced time-shared delivery

The result indicates that in all tested-cases, crowdsourced time-based delivery returns a lower cost than the dedicated delivery. The major savings are from avoiding using a truck and therefore saves fixed cost. When the expected payment for crowdsourced drivers is lower than \$24, the service is also more preferrable than crowdsourced shared-trip delivery.

5.6 Summary of findings and implementations

Chapter 5 has presented a real-world city-scale numerical case study of crowdsourced shared-trip delivery. The results of computational time of using the decomposition heuristic, total cost, total VMT of crowdsourced shared-trip delivery system and two sensitivity analyses are presented. Though, all the results are generated based on the geometrical features and typology of the City of Irvine, they do bring lead to interesting findings and significant implications.

The computational experiments shows that the decomposition heuristic outperforms the exact solver solution on computational time in all cases. In the cases where SPVs and package orders are small, the decomposition heuristic solution quality is comparable to the exact solution. In large cases, where the solver could not turn out solutions through exact method, the decomposition heuristic could still solve the problem in reasonable time. The ability of solving large-scale problems in crowdsourced delivery problem enables us to include more SPVs into the system and provides flexibility of modeling crowdsourced delivery problem.

The case study also shows that the ability of delivering package orders with SPVs is linearly related to the number of participating SPVs. However, the total cost of delivering package orders does not have a linear relation with the number of participating SPVs. Using SPVs in a crowdsourced shared-trip delivery system saves cost compared to conventional dedicated delivery (from 15% to 40%), but the major cost savings are from the reduction of using DVs. Totally getting rid of DVs for delivery is possible, but a significantly large number of participating SPVs are required to achieve such goal. The total VMT of crowdsourced delivery highly depends on the origins of SPVs. If SPVs are located near the depot, the VMT savings from crowdsourced delivery compared to dedicated delivery is about 10% to 20%. When the SPVs are far from the depot and needs to drive to the depot for package order pick, then the crowdsourced shared-trip delivery produces more VMT, and the amount of VMT increases as more SPVs are used.

To further understand the impact of the maximum willingness to detour on crowdsourced delivery, we conduct a sensitivity analysis. The results indicates that, on one hand, when a higher maximum detour willingness is imposed, the system has more participating SPVs, and higher detour willingness leads to lower total cost. On the other hand, the SPVs which are matched with delivery orders are likely not required to detour a lot for the trip from the depot to their destinations. This finding could further inspire the improvement on the decomposition heuristic by only generating short detour routes from the depot to the destinations.

Another sensitivity study on the depot location shows that a center located depot would attracts more feasible SPVs, have less total cost and VMT compared to a boundary located depot. When choosing the depot location, the decision makers may need a cost-benefit analysis since a center located depot may incur higher land use cost.

As for the minimum compensation required for crowdsourced drivers, the findings indicate that in crowdsourced shared-trip delivery service, when the drivers have a minimum accepted compensation (no matter whether it is constant or by detour time), the current compensation scheme (incentive = compensation per package carried plus compensation per mile detour) would prefer to choose drivers with longer detour, and the system results in a higher VMT. To avoid the unnecessary VMT, the logistics companies should consider differentiates compensation schemes for drivers that could deliver packages with low detour miles and drivers that needs long detour distances to deliver packages. For the former type of crowdsourced drivers, compensation schemes that based on the number of packages that could be carried may both achieve reduced cost for the company and avoid unnecessary mileages travelled by the drivers. On the other hand, when crowdsourced drivers have a relatively high expectation on their compensation, the company may consider switch to alternative services such as crowdsourced time-based delivery or dedicated delivery.

Crowdsourced time-based delivery, in this case, may be preferrable from the total cost perspective. The findings show that under different expectations of incentives, crowdsourced time-based delivery is always preferable than dedicated delivery. If the hourly minimum acceptable compensation is low than $24, crowdsourced time-based delivery is preferrable than share-trip delivery from the cost perspective. However, more discussions are need from the social impact perspective to compare the two types of services.

Chapter 6 Incentive and Operational Policies in Crowdsourced Freight Delivery

6.1 Introduction

Chapter 2 presents the taxonomy of crowdsourced delivery and compares the three types of crowdsourced freight delivery types, namely crowdsourced trip-based delivery, crowdsourced time-based delivery, and crowdsourced shared-trip delivery. The three types of crowdsourced delivery mainly characterized by the source of mobility and freight types. The crowdsourced trip-based delivery is often used for high urgency freight, such as meals, and handled by drivers/vehicles that have the primary purpose of delivering goods. The crowdsourced time-based delivery usually handles freight with low urgency. The same as crowdsourced trip-based delivery, the drivers for time-based delivery are semi-professional (semi-pro) drivers with trips that have a primary purpose of delivering goods. The term “semi-pro” indicates that the driver is not hired by the logistics company but works as a “contractor” for certain period of time or delivering certain tasks. The third type, crowdsourced shared-trip delivery handles freight with low urgency and utilizes trips of non-professional drivers.

Figure 6.1 positions the three types of crowdsourced delivery and dedicated truck delivery on a coordinate system characterized by the urgency of freight and the professionalism of drivers. Crowdsourced time-based delivery, trip-shared delivery, and dedicated truck delivery all serve packages that have low urgency. Therefore, the three types of services “compete” on the demand side. From another perspective, they are supplementary services. From the supply perspective, crowdsourced trip-based delivery services and crowdsourced time-based delivery services both contract with semi-pro drivers, who work for a dedicated period of time or who conduct specific delivery tasks. Hence, crowdsourced trip-based delivery and time-based delivery compete for drivers. The overlap of demand and supply of the four types of services raises an interesting pair of questions for logistics service providers: when delivering a set of low urgency parcels, which crowdsourcing type should be chosen and what compensation should be paid to the drivers?

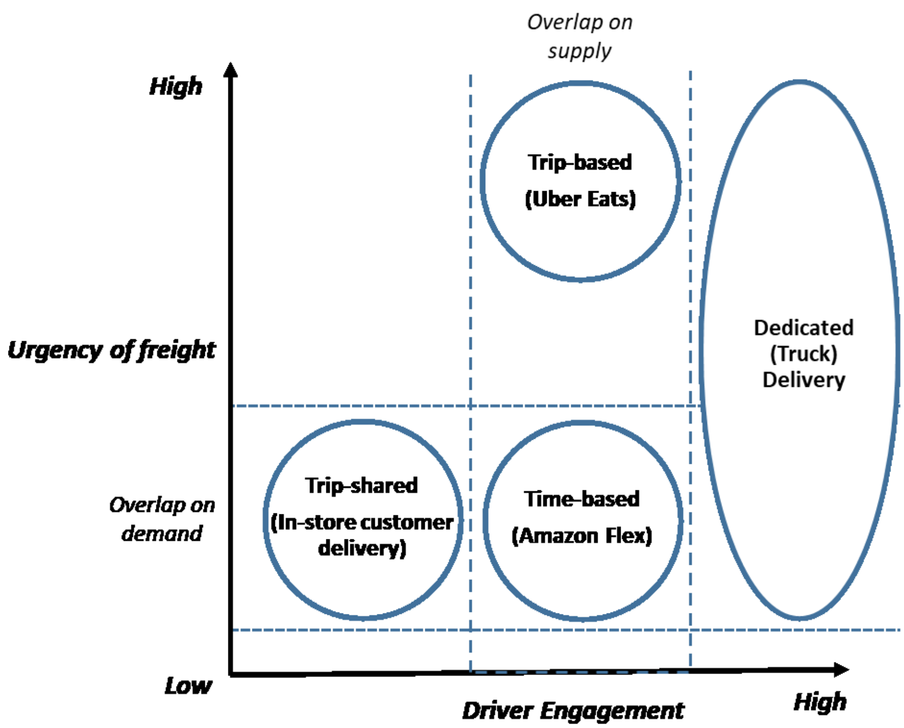

Figure 6.1 Three crowdsourced delivery types and dedicated delivery

This chapter aims to compare the three types of delivery services (crowdsource time-based delivery, crowdsourced shared-trip delivery, and dedicated delivery) for low urgency package delivery. Moreover, the chapter aims to provide insights about the incentives to drivers provided by the services and increase understanding of the social impact of deploying each service type.

The incentive schemes (compensation to crowdsourced drivers) and operational policies have been discussed from different perspectives in literature. Several studies include different compensation schemes for crowdsourced shared-trip delivery drivers (Archetti et al., 2016; Arslan et al., 2019; Dahle et al., 2019; Dayarian & Savelsbergh, 2020); auction-based pricing for long-haul crowdsourced drivers (Rechavi & Toch, 2020); domain-of-influence pricing based on individual package delivery radius (Zhou et al., 2021); user-centric incentive model to motivate driver participation (Hong et al., 2019); and the willingness of drivers to deliver goods (Qi et al., 2018).

Previous studies on incentives and operations of urban last-mile crowdsourced delivery all focus on a single crowdsourcing type, the crowdsourced shared-trip delivery, without considering the interchangeability of different types of services. In other words, an over-compensated scheme in crowdsourced shared-trip delivery may drive the logistics company to switch to crowdsourced time-based delivery or traditional dedicated truck delivery services. However, an under-compensated scheme would fail to attract sufficient crowdsourced drivers as supply of mobility for the delivery service, and as a result, impair the service quality. This research attempts to build the linkage between incentive schemes and choice of crowdsourced delivery modes, and to address the gap in incentive strategies in urban last-mile crowdsourced delivery. It is worth noting that the decision of crowdsourced delivery types and compensation schemes are highly related to the type of parcels that is to deliver. For example, high urgency goods, such as food delivery, usually requires quick response and high reliability. Therefore, crowdsourced trip-based delivery or time-based delivery are preferable for these services since the crowdsourced drivers are more committed than the crowdsourced trip-shared type.

This chapter is organized as follows. Section 6.2 reviews related literature and explains background information. Section 6.3 presents the research design, research methods, and the numerical study parameters. Section 6.4 discusses the results and findings. Section 6.5 summarizes the chapter.

6.2 Background information and literature review

For delivering a set of low-urgency packages, a logistics service provider can deploy conventional dedicated truck delivery, contract with crowdsourced drivers for either time-based delivery (committed drivers) or trip-shared delivery (ad-hoc drivers). The results from the numerical case study in Chapter 5 indicate that crowdsourced shared-trip delivery reduces delivery costs by 50% compared to conventional dedicated truck service. The savings are mainly from the reduction in facility costs, such as vehicle procurement and maintenance. Therefore, not keeping a fleet of dedicated vehicle brings the benefit of reducing vehicle purchasing and maintenance cost. However, not keeping a dedicated vehicle fleet also means that the company needs to manage the uncertainty of crowdsourced driver supply. The planning level question that the logistics company would consider is whether they need to keep a fleet of dedicated vehicles, or more specifically, whether the company could deploy a mixture of vehicles for the service.

On the other hand, the supply of crowdsourced drivers is related to the incentive or compensation schemes provided by the logistics company. In the literature, Archetti et al. (2016) compare different pricing options for crowdsourced shared-trip delivery (or in their terms, package delivery with occasional drivers). Their options of pricing include compensating the drivers based on the number of packages delivered or the distance travelled. Their conclusion claims that paying drivers based on distance benefits the drivers the most. Qi et al. (2018) conclude that in order to satisfy the constraints surrounding a driver's willingness to deliver packages, the major cost savings in crowdsourced delivery service stems from facility cost savings instead of mileage savings. Dahle et al. (2019) also test different pricing options for crowdsourced delivery services with the requirement of meeting drivers' willingness of deliver. Their results indicate that using compensation schemes that are based on detour would encourage vehicles to travel on non-shortest paths.

6.3 Methodology

A logistics company needs to deliver a set of package delivery orders (abbreviated as PDOs, denoted by $P = \{p_1, p_2, \dots, p_n\}$) in a given time period. The company has a fixed number of k ($k \geq 0$) dedicated vehicles that are available for PDO delivery. Since the number of PODs varies from time to time, during some time periods, the company needs to "hire" or contract with crowdsourced drivers for delivery. There are two types of crowdsourced drivers, committed crowdsourced drivers (called as committed drivers) and ad-hoc crowdsourced drivers (called ad-hoc drivers). Committed drivers are crowdsourced drivers who signed up in smart phone apps in advance to become a dedicated driver for a certain time slot. Ad-hoc drivers are crowdsourced drivers who arrive at the depot randomly and only serve a limited number of PDOs en-route to their destinations. We here assume that ad-hoc drivers would start the trip at the depot and not come back, but committed drivers, during their committed time slots, would return to the depot multiple times for PDO pickup.

The cost of dedicated vehicles is calculated by the total vehicle miles travelled multiplied by the cost factor c_{ij}^d. For committed drivers, there are three ways of compensation, namely, compensation by time, compensation by orders delivered, and compensation by total distance travelled. If committed drivers are compensated by time, they will be paid a fixed hourly rate f_t^c. This indicates that under this payments scheme, the company will attempt to minimize the number of committed drivers. When committed drivers are compensated by the orders delivered, total compensation is calculated by the number of orders delivered times a fixed amount per package order f_p^c. If the committed drivers are compensated by the distance travelled for delivery, the cost factor per mile is represented by c_{ij}^c. Ad-hoc drivers will be compensated in two ways. The first one is compensation by the travel distance, in which the cost factor per mile is represented by c_{ij}^{a}. The second one is compensation by the number of packages carried, in which the compensation amount per package order is represented by f_p^a.

The logistics company considers one of the following five operating strategies for the delivery service.

1. The logistics company keeps a fleet of dedicated vehicles and only deploys dedicated vehicles for delivery.
2. The logistics company keeps a fleet of dedicated vehicles and contracts with some committed drivers for delivery when it is necessary.
3. The logistics company keeps a fleet of dedicated vehicles and contracts with both committed and ad-hoc drivers for delivery when it is necessary.
4. The logistics company only contracts with committed drivers for delivery.
5. The logistics company contracts with both committed and ad-hoc drivers for delivery.

In order to compare the cost of the aforementioned strategies, this chapter formulates the crowdsourced delivery problem as a Pickup and Delivery Problem (PDP).

The decision variables are listed as follows:

- $x_{ij}^{k} \in \{0,1\}, \forall (i,j) \in A, \forall\ k \in V.$ $\quad x_{ij}^{k} = 1$, if arc (i,j) is visited by vehicle k.
- $t_i^k \in \mathbb{R}^{+}, \forall i \in N, \forall k \in V.$ $\quad$ Arrival time of vehicle k at node i.
- $w_i^k \in \mathbb{N}^{0}, \forall i \in N, \forall k \in V.$ $\quad$ Load of vehicle k at node i.
- $z_k^d \in \{0,1\}, \forall k \in V.$ $\quad$ Whether a dedicated vehicle is used.
- $z_k^c \in \{0,1\}, \forall k \in V.$ $\quad$ Whether a committed driver is used.
- $z_k^a \in \{0,1\}, \forall k \in V.$ $\quad$ Whether an ad-hoc driver is used.

Strategy 1 works as the base case, the formulation for it is as follows:

Formulation 5:

$$Min\ \Theta \quad (6.1)$$

subject to:

$$\sum_{j\in N}\sum_{k\in D} x_{ij}^{k} = 1 \qquad \forall i \in N\backslash\{0\} \qquad (6.2)$$

$$\sum_{i\in N}\sum_{k\in D} x_{ij}^{k} = 1 \qquad \forall j \in N\backslash\{h\} \qquad (6.3)$$

$$\sum_{i\in N, i\neq j}\sum_{k\in D} x_{ij}^{k} - \sum_{l\in N, l\neq j}\sum_{k\in D} x_{jl}^{k} = 0 \qquad \forall j \in N\backslash\{0,h\} \qquad (6.4)$$

$$\sum_{j\in\{N_p\}} x_{0j}^{k} - \sum_{i\in\{N_d\}} x_{i,h}^{k} = 0 \qquad \forall k \in D \qquad (6.5)$$

$$\sum_{i\in N} x_{ij}^{k} - \sum_{i\in N} x_{i,j+n}^{k} = 0 \qquad \forall j \in N_p \;\; \forall k \in D \qquad (6.6)$$

$$z_k^d \geq \sum_{j\in N, j\neq 0} x_{0j}^{k} \qquad \forall k \in D \qquad (6.7)$$

$$t_j^k + \left(1 - \sum_{i\in N} x_{ij}^{k}\right) \times M \geq T_j \qquad \forall j \in N_p \;\; \forall k \in D \qquad (6.8)$$

$$t_j^k \leq \left(1 - \sum_{i\in N} x_{ij}^{k}\right) \times M + T_j \qquad \forall j \in N_d \;\; \forall k \in D \qquad (6.9)$$

$$t_i^k + \tau_{ij} \leq t_j^k + \left(1 - x_{ij}^{k}\right) \times M \qquad \forall i,j \in N,\ i \neq j \;\; \forall k \in D \qquad (6.10)$$

$$t_i^k \leq t_{i+n}^k \qquad \forall i \in N_p \;\; \forall k \in D \qquad (6.11)$$

$$w_j^k + q_j \leq Q_k + \left(1 - \sum_{i\in N} x_{ij}^{k}\right) \times M \qquad \forall j \in N_p, \;\; \forall k \in D \qquad (6.12)$$

$$w_j^k = w_i^k + q_j \times \sum_{i\in N} x_{ij}^{k} \qquad \forall i,j \in N,\ i \neq j \;\; \forall k \in D \qquad (6.13)$$

$$x_{ij}^k \in \{0,1\} \qquad \forall i,j \in N \;\; \forall k \in V \qquad (6.14)$$

$$z_k^d \in \{0,1\} \qquad \forall k \in D \qquad (6.15)$$

$$w_k \in \{0,1\} \qquad \forall k \in D \qquad (6.16)$$

$$t_i^k \geq 0 \qquad \forall i \in N \;\; \forall k \in D \qquad (6.17)$$

The objective function is to minimize the total cost. The specific function depends on the compensation schemes that is used for committed drivers and ad-hoc drivers. Eqn. 6.2 to 6.4 are standard routing constraints that require that each node is visited once and only once. Eqn. 6.5 requires every dedicated vehicle that leaves the depot to return to the depot. Eqn. 6.6 ensures that every order is served. Eqn. 6.7

indicates whether a vehicle is used or not. Eqn. 6.8 to 6.11 are time window constraints. Eqn.6.8 and 6.9 ensure that a pickup happens after the package order is ready and delivery happens before the time window is closed. Eqn. 6.10 is the sequencing constraint and also eliminates sub-tours. Eqn. 6.11 ensures delivery happens after pickup. Eqn. 6.12 and 6.13 are capacity constraints. Eqn.6.14 to 6.17 are binary and non-negativity constraints.

For Strategy 2, when committed drivers are included, the following constraints are added to Formulation 5.

$$z_k^c \geq \sum_{j \in N, j \neq 0} x_{0j}^k \qquad \begin{array}{l} \forall i \in N \\ \forall k \in C_m \end{array} \tag{6.18}$$

As mentioned previously, the objective function depends on the incentive scheme for committed drivers.

When crowdsourced drivers are compensated by hourly rate, the objective is to minimize the total number of vehicles required.

$$Min\ \Theta_4 = \sum_{k \in D} \sum_{(i,j) \in A} c_{ij} x_{ij}^k + \sum_{k \in D} f^d z_k^d + \sum_{k \in C_m} f_t^c z_k^c \quad (6.19)$$

When crowdsourced drivers are compensated by the number of orders delivered the objective function is:

$$Min\ \Theta_5 = \sum_{k \in D} \sum_{(i,j) \in A} c_{ij} x_{ij}^k + \sum_{k \in D} f^d z_k^d + f_p^c \sum_{k \in C_m} \sum_{j \in N_d} \sum_{i \in N} x_{ij}^k \quad (6.20)$$

When crowdsourced drivers are compensated by the distance travelled, the objective function is:

$$Min\ \Theta_6 = \sum_{k \in D} \sum_{(i,j) \in A} c_{ij} x_{ij}^k + \sum_{k \in D} f^d z_k^d + \sum_{k \in C_m} \sum_{(i,j) \in A} c_{ij} x_{ij}^k \quad (6.21)$$

For Strategy 3, when ad-hoc drivers are included. The formulation needs to include the destination constraints and capacity constraints for ad-hoc drivers. The following constraints are added to Formulation 5.

$$z_k^a \le 1 - x_{0,N_{s_k}}^k \qquad \forall k \in A_{hoc} \qquad (6.22)$$

$$\sum_{i \in N \setminus \{N_{s_k}\}} x_{i,N_{s_k}}^k = 1 \qquad \forall k \in A_{hoc} \qquad (6.23)$$

$$\sum_{i \in N} \sum_{j \in N_d} x_{ij}^k \le z_k^a \times (q_{s_k} + 1) \qquad \forall k \in A_{hoc} \qquad (6.24)$$

$$\sum_{i \in N} \sum_{j \in N_p} x_{ij}^k \le q_{s_k} \qquad \forall k \in A_{hoc} \qquad (6.25)$$

Eqn. 6.22 indicates whether an ad-hoc driver is used or not. Eqn.6.23 ensures every ad-hoc driver arrives at their own destination after delivery. Eqn.6.24 is the capacity constraint for ad-hoc drivers. Constraint 6.25 ensures ad-hoc drivers only leave the depot once and do not return.

For the objective function, this study assumes that both committed drivers and ad-hoc drivers are compensated by the homogenous schemes (not necessarily the same parameters) when all crowdsourced drivers are compensated by the number of delivery orders fulfilled, the objective function is:

$$Min\ \Theta_7 = \sum_{k \in D} \sum_{(i,j) \in A} c_{ij} x_{ij}^k + \sum_{k \in D} f^d z_k^d + f_p^c \sum_{k \in C_m} \sum_{j \in N_d} \sum_{i \in N} x_{ij}^k + f_p^a \sum_{k \in A_{hoc}} \sum_{j \in N_d} \sum_{i \in N} x_{ij}^k \ (6.26)$$

When all crowdsourced drivers are compensated by the distance traveled, the objective function is:

$$Min\ \Theta_8 = \sum_{k \in D} \sum_{(i,j) \in A} c_{ij} x_{ij}^k + \sum_{k \in D} f^d z_k^d + \sum_{k \in C_m} \sum_{(i,j) \in A} c_{ij} x_{ij}^k + \sum_{k \in A_{hoc}} \sum_{(i,j) \in A} c_{ij} x_{ij}^k \ (6.27)$$

The formulation of Strategy 4, deploying only committed drivers, is similar to the base case (Strategy 1) and will not be rewritten in this section. The formulation of Strategy 5 is also similar to Strategy 2.

6.4 Summary and discussion of future research

This chapter discuss a novel problem of deciding and planning the mode of urban last mile delivery for logistics companies. The logistics company needs to decide, in relatively long-run, the mode of delivery, whether to use dedicated vehicles or crowdsourced vehicles.

The chapter presents a PDP-based formulation with different objective functions to choose between different operational policies and pricing options. The formulation is able to capture the decision for five different operation strategies and three different incentive/compensation policies for crowdsourced drivers.

In the next step of research, a dynamic and stochastic supply function could be considered for crowdsourced drivers. The compensation/incentive rate would impact the availability of crowdsourced drivers. With real-world data, this model is capable of capture the dynamics of crowdsourced driver supply. The optimization model is applicable for deciding the optimal long-term operational policies for logistics company.

Chapter 7 Conclusions

7.1 Conclusions

This dissertation provides a comprehensive study on urban last-mile crowdsourced delivery. Beginning with different types of crowdsourced delivery and their features, the dissertation creates a taxonomy for urban crowdsourced delivery. Based on the sourcing type and driver working type, the dissertation categorizes urban crowdsourced delivery into crowdsourced time-based delivery, trip-based delivery, and shared-trip delivery. The features and applications of the three types of crowdsourced services are discussed in the dissertation.

Crowdsourced shared-trip delivery, which is the focus of the dissertation, has great potential in terms of cost and VMT savings in the shared economy. Small-scale crowdsourced shared-trip delivery problems are well studied by the literature, while large-scale problems are rarely attempted. The dissertation addresses the research gap on large-scale crowdsourced delivery problems by developing new mathematical models and algorithms and applying them to large real-world problems.

Following the literature on crowdsourced delivery, the dissertation first models the crowdsourced shared-trip delivery as an *MFOCVRPTW* (mixed fleet open capacitated vehicle routing problem with time window). This formulation captures the generalized features of crowdsourced shared-trip delivery. However, finding an optimal solution based on this formulation is time-consuming. To solve the problem in an efficient manner, the dissertation reformulates the problem as a set partitioning problem. The alternative set partitioning formulation also inspires a new solution approach, which is a novel decomposition heuristic.

The novel solution algorithm decomposes the problem into a set of shared personal vehicles and a set of dedicated vehicles and matches the packages to the two vehicle sets separately. The decomposition heuristic also solves 4 subproblems, namely, the budgeted k-shortest paths problem, the large-scale matching problem, the package switching problem, and the multiple vehicle routing problems. Solution algorithms for each subproblem are also discussed. The decomposition heuristic is compared with an exact method to solve the crowdsourced shared-trip delivery problem. The novel heuristic approach can obtain solutions with a 1.5% optimality gap and the heuristic is much faster than the exact method.

The models and algorithms are applied to city-scale problems. In the case study of the City of Irvine, the dissertation analyzes major factors that would impact the efficiency of a crowdsourced shared-trip delivery. The findings indicate that when the number of participating drivers is small, the system requires crowdsourced drivers to have a longer willingness to detour in order to achieve significant cost reductions. In addition, when the number of drivers is small, a depot located in the center of the service region can achieve significantly lower total costs than a depot located along the boundary of the service region. However, the impact from both detour willingness and depot location is mitigated when the number of participating drivers increases. The dissertation also finds that the major cost saving components of crowdsourced delivery stem from reductions in facility costs, mainly the purchasing of dedicated trucks.

Comparing the vehicle miles travelled for crowdsourced delivery and dedicated delivery, the dissertation finds that VMT savings depend on the distance that the drivers travel to the depot for package pick-up. The dissertation also discusses the choice of crowdsourcing service for a logistics company under different driver willingness to deliver assumptions and different compensation schemes. The findings indicate that to achieve reductions in both costs and VMT, the company should differentiate between compensation schemes for drivers with short detour distances and long detour distances.

Overall, the dissertation significantly advances science and the state-of-the-art through novel, computationally efficient and operationally effective, mathematical models and associated heuristic solution algorithms. Moreover, the dissertation contributes to industrial practice by addressing real-world problem instances (through numerical case studies) and summarizing important managerial implementations. Potential research areas that are inspired by this dissertation are discussed in the following section.

7.2 Future research directions

While writing this dissertation, it became relatively clear that the existing empirical evidence for crowdsourced delivery is not sufficient. On one hand, logistics companies may not share data for academic studies. On the other hand, there are still gaps in literature for finding empirical evidence for multiple aspects of crowdsourced delivery, such as driver detour willingness and expectation for compensation. Therefore, the first future research direction is to collect data through surveys/questionnaires or experiments that would help estimate critical parameter values for the design and operation of specific crowdsourced delivery services.

The second research direction relates to algorithm advancements. This dissertation leveraged Yen's algorithm with a budget constraint for computation. The results show that not all vehicle routes are need for matching. Therefore, setting up an intelligent algorithmic scheme for finding potentially valuable routes is likely to be computationally beneficial without negatively impacting the algorithms search for optimal solutions. The computational time reduction would improve the applicability of the algorithm in real-world scenarios.

The third research direction is to integrate crowdsourced delivery with other modes of delivery, such as delivery with transit or drone delivery. The "internet of things" enables items to be connected and also enables faster information exchange. Therefore, the integration of crowdsourced delivery and other modes would benefit the society by providing more reliable delivery options and lowing the carbon footprint.

Bibliography

Ã, S. N. P., Doerner, K. F., & Hartl, R. F. (2010). Computers Operations Research Variable neighborhood search for the dial-a-ride problem. *Computers and Operation Research*, *37*(6), 1129–1138. https://doi.org/10.1016/j.cor.2009.10.003

Agatz, N., Erera, A., Savelsbergh, M., & Wang, X. (2012). Optimization for dynamic ride-sharing: A review. *European Journal of Operational Research*, *223*(2), 295–303. https://doi.org/10.1016/j.ejor.2012.05.028

Alonso-mora, J., Wallar, A., Frazzoli, E., Rus, D., Alonso-mora, J., Samaranayake, S., Wallar, A., Frazzoli, E., & Rus, D. (2018). On-demand high-capacity ride-sharing via dynamic trip-vehicle assignment. *Proceedings of the National Academy of Sciences of the United States of America*, *115*(3), E555. https://doi.org/10.1073/pnas.1721622115

Archetti, C., Savelsbergh, M., & Speranza, M. G. (2016). The Vehicle Routing Problem with Occasional Drivers. *European Journal of Operational Research*, *254*(2), 472–480.

Arslan, A. M., Agatz, N., Kroon, L., & Zuidwijk, R. (2019). Crowdsourced delivery—a dynamic pickup and delivery problem with ad hoc drivers. *Transportation Science*, *53*(1), 222–235. https://doi.org/10.1287/trsc.2017.0803

Augerat, P., Naddef, D., Belenguer, J., Benavent, E., Corberan, A., & Rinaldi, G. (1995). *Computational results with a branch and cut code for the capacitated vehicle routing problem*.

Baldacci, R., Battarra, M., & Vigo, D. (2008). Routing a Heterogeneous Fleet of Vehicles. In B. Golden, S. Raghavan, & E. Wasil (Eds.), *The Vehicle Routing Problem: Latest Advances and New Challenges* (pp. 3–27). Springer US. https://doi.org/10.1007/978-0-387-77778-8_1

Baldacci, R., Christofides, N., & Mingozzi, A. (2008). An exact algorithm for the vehicle routing problem based on the set partitioning formulation with additional cuts. *Mathematical Programming*, *115*(2), 351–385. https://doi.org/10.1007/s10107-007-0178-5

Baldacci, R., Hadjiconstantinou, E., & Mingozzi, A. (2004). *An Exact Algorithm for the Capacitated Vehicle Routing Problem Based on a Two-Commodity Network Flow Formulation*. *August 2021*. https://doi.org/10.1287/opre.1040.0111

Baldacci, R., Mingozzi, A., & Roberti, R. (2011). New route relaxation and pricing strategies for the vehicle routing problem. *Operations Research*, *59*(5), 1269–1283. https://doi.org/10.1287/opre.1110.0975

Baldacci, R., Mingozzi, A., & Roberti, R. (2012a). Recent exact algorithms for solving the vehicle routing problem under capacity and time window constraints. *European Journal of Operational Research*, *218*(1), 1–6. https://doi.org/10.1016/j.ejor.2011.07.037

Balinski, M. L., & Quandt, R. E. (1964). On an Integer Program for a Delivery Problem. *Operations Research*, *12*(2), 300–304. https://doi.org/10.1287/opre.12.2.300

Bhatti, A., Akram, H., & Khan, A. U. (2020). *E-commerce trends during COVID-19 Pandemic E-commerce trends during COVID-19 Pandemic*. *June*.

Braekers, K., Caris, A., & Janssens, G. K. (2014). Bi-objective optimization of drayage operations in the service area of intermodal terminals. *Transportation Research Part E*, *65*, 50–69. https://doi.org/10.1016/j.tre.2013.12.012

Bramel, J., & Simchi-levi, D. (1995). A Location based Heuristic for General Routing Problems. *Operations Research*, *43*(4), 649–660.

Burkard, R. E., & Çela, E. (1999). Linear Assignment Problems and Extensions. In D.-Z. Du & P. M. Pardalos (Eds.), *Handbook of Combinatorial Optimization: Supplement Volume A* (pp. 75–149). Springer US. https://doi.org/10.1007/978-1-4757-3023-4_2

Campbell, A. M., & Savelsbergh, M. (2004). Efficient Insertion Heuristics for Vehicle Routing and Scheduling Problems. *Transportation Science*, *38*(August 2004), 369–378. https://doi.org/10.1287/trsc.1030.0046

Chen, P., & Chankov, S. M. (2018). Crowdsourced delivery for last-mile distribution: An agent-based modelling and simulation approach. *IEEE International Conference on Industrial Engineering and Engineering Management*, *2017-Decem*(2), 1271–1275. https://doi.org/10.1109/IEEM.2017.8290097

Chen, W., Mes, M., & Schutten, M. (2017). Multi-hop driver-parcel matching problem with time windows. *Flexible Services and Manufacturing Journal*, *30*(3), 517–553. https://doi.org/10.1007/s10696-016-9273-3

Christofides, N. (1981). *State-Space Relaxation Procedures for the Computation of Bounds to Routing Problems*. *1*, 145–164.

Christofides, N., & Eilon, S. (1969). An Algorithm for the Vehicle-dispatching Problem. *Journal of the Operational Research Society*, *20*(3), 309–318. https://doi.org/10.1057/jors.1969.75

Clarke, G., & Wright, J. W. (1964). Scheduling of Vehicles from a Central Depot to a Number of Delivery Points. *Operations Research*, *12*(4), 568–581. https://doi.org/10.1287/opre.12.4.568

Cordeau, J. F., & Laporte, G. (2003). A tabu search heuristic for the static multi-vehicle dial-a-ride problem. *Transportation Research Part B: Methodological*, *37*(6), 579–594. https://doi.org/10.1016/S0191-2615(02)00045-0

Cordeau, J. F., & Laporte, G. (2007). The dial-a-ride problem: Models and algorithms. *Annals of Operations Research*, *153*(1), 29–46. https://doi.org/10.1007/s10479-007-0170-8

Daganzo, C. F. (1984). The Distance Traveled To Visit N Points With a Maximum of C Stops Per Vehicle: an Analytical Model and an Application. *Transportation Science*, *18*(4), 331–350. https://doi.org/10.1287/trsc.18.4.331

Dahle, L., Andersson, H., Christiansen, M., & Speranza, M. G. (2019). The pickup and delivery problem with time windows and occasional drivers. *Computers and Operations Research*, *109*, 122–133. https://doi.org/10.1016/j.cor.2019.04.023

Dayarian, I., & Savelsbergh, M. (2020). Crowdshipping and Same-day Delivery : Employing In-store Customers to Deliver Online Orders. *Production and Operations Management*, *29*(9), 2153–2174. https://doi.org/10.1111/poms.13219

Desrochers, M., Lenstra, J. K., & Savelsbergh, M. W. P. (1990). A classification scheme for vehicle routing and scheduling problems. *European Journal of Operational Research*, *46*(3), 322–332. https://doi.org/10.1016/0377-2217(90)90007-X

Desrochers, M., & Verhoog, T. W. (1991). A NEW HEURISTIC FOR THE FLEET SIZE AND MIX VEHICLE ROUTING PROBLEM. *Computers Operations Research*, *18*(3), 263–274.

Desrosiers, J., Sournis, F., & Desrochers, M. (1984). Routing with Time Windows by Column Generation. *Networks*, *14*(4), 545–565. https://doi.org/10.1002/net.3230140406

Desrosiers, M., Desrosiers, J., & Marius, S. (1992). A New Optimization Algorithm for the Vehicle Routing Problem with Time Windows. *Operations Research*, *40*(2), 342–354. http://www.jstor.com/stable/171457

Diana, M., & Dessouky, M. M. (2004). *A new regret insertion heuristic for solving large-scale dial-a-ride problems with time windows*. *38*, 539–557. https://doi.org/10.1016/j.trb.2003.07.001

Eilon, S. (1971). *Management control*. Macmillan International Higher Education.

Fatnassi, E., Chaouachi, J., & Klibi, W. (2015). Planning and operating a shared goods and passengers on-demand rapid transit system for sustainable city-logistics. *Transportation Research Part B: Methodological*, *81*, 440–460. https://doi.org/10.1016/j.trb.2015.07.016

Figliozzi, M. A. (2008). Planning approximations to the average length of vehicle routing problems with varying customer demands and routing constraints. *Transportation Research Record*, *765*(2089), 1–8. https://doi.org/10.3141/2089-01

Figliozzi, M. A. (2009). Planning approximations to the average length of vehicle routing problems with time window constraints. *Transportation Research Part B: Methodological*, *43*(4), 438–447. https://doi.org/10.1016/j.trb.2008.08.004

Fukasawa, R., Longo, H., Lysgaard, J., & Poggi, M. (2006). *Robust Branch-and-Cut-and-Price for the Capacitated Vehicle*. *511*, 491–511.

Gaskell, T. J. (1964). *Bases for Vehicle Fleet Scheduling*. 281–295.

Gdowska, K., Viana, A., & Pedroso, J. P. (2018). Stochastic last-mile delivery with crowdshipping. *Transportation Research Procedia*, *30*, 90–100. https://doi.org/10.1016/j.trpro.2018.09.011

Gendreau, M., Hertz, A., Laporte, G., Gendreau, M., Hertz, A., & Laporte, G. (1994). *Linked references are available on JSTOR for this article : A Tabu Search Heuristic for the Vehicle Routing Problem*. *40*(10), 1276–1290.

Gendreau, M., & Laporte, G. (1992). *New Insertion and Postoptimization Procedures for the Traveling Salesman Problem Author (s): Michel Gendreau , Alain Hertz and Gilbert Laporte Published by : INFORMS Stable URL : https://www.jstor.org/stable/171722 REFERENCES Linked references are avail*. *40*(6), 1086–1094.

Gendreau, M., & Potvin, J.-Y. (2005). Tabu Search. In E. K. Burke & G. Kendall (Eds.), *Search Methodologies: Introductory Tutorials in Optimization and Decision Support Techniques* (pp. 165–186). Springer US. https://doi.org/10.1007/0-387-28356-0_6

Gillett, B. E., Miller, L. R., Apr, M., Apr, N. M., & Miller, L. R. (1974). *A Heuristic Algorithm for the Vehicle-Dispatch Problem Published by : INFORMS Stable URL : https://www.jstor.org/stable/169591l*. *22*(2), 340–349.

Golden, B., Raghavan, S., & Wasil, E. (2008). The vehicle routing problem: Latest advances and new challenges. In *Operations Research/ Computer Science Interfaces Series* (Vol. 43). https://doi.org/10.1007/978-0-387-77778-8

Gurobi. (2021). *GUROBI Optimization*. https://www.gurobi.com/

Hansen, P., Mladenović, N., & Perez-Britos, D. (2001). Variable neighborhood decomposition search. *Journal of Heuristics*, *7*(4), 335–350. https://doi.org/10.1023/A:1011336210885

Hong, H., Li, X., He, D., Zhang, Y., & Wang, M. (2019). Crowdsourcing Incentives for Multi-Hop Urban Parcel Delivery Network. *IEEE Access*, *7*, 26268–26277. https://doi.org/10.1109/ACCESS.2019.2896912

Hosni, H., Naoum-Sawaya, J., & Artail, H. (2014). The shared-taxi problem: Formulation and solution methods. *Transportation Research Part B: Methodological*, *70*, 303–318. https://doi.org/10.1016/j.trb.2014.09.011

Huang, Y., Savelsbergh, M., & Zhao, L. (2018). Designing logistics systems for home delivery in densely populated urban areas. *Transportation Research Part B*, *115*, 95–125. https://doi.org/10.1016/j.trb.2018.07.006

Hung, M. S. (1983). A Polynomial Simplex Method for the Assignment Problem. *Operations Research*, *31*(3), 595–600. https://doi.org/10.1287/opre.31.3.595

Internal Revenue Service. (2021). *Standard Mileage Rates*. https://www.irs.gov/tax-professionals/standard-mileage-rates

Irnich, S., Toth, P., & Vigo, D. (2014). Chapter 1: The Family of Vehicle Routing Problems. In *Vehicle Routing* (pp. 1–33). https://doi.org/10.1137/1.9781611973594.ch1

Ivanov, B. (2018). *Growth of E-Commerce and Ride-Hailing Services is Reshaping Cities Innovative Goods Delivery Solutions for Cities of the Future*.

Jonker, R., & Volgenant, A. (1987). A shortest augmenting path algorithm for dense and sparse linear assignment problems. *Computing*, *38*(4), 325–340. https://doi.org/10.1007/BF02278710

Kuhn, H. W. (1955). The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, *2*(1–2), 83–97.

Laporte, G. (1992). The traveling salesman problem: An overview of exact and approximate algorithms. *European Journal of Operational Research*, *59*(2), 231–247. https://doi.org/10.1016/0377-2217(92)90138-Y

Laporte, G., Gendreau, M., & Potvin, J. (2000). *Classical and modern heuristics for the vehicle routing problem*. *7*.

Laporte, G., Gendreau, M., Potvin, J. Y., & Semet, F. (2000). Classical and modern heuristics for the vehicle routing problem. *International Transactions in Operational Research*, *7*(4–5), 285–300. https://doi.org/10.1111/j.1475-3995.2000.tb00200.x

Laporte, G., Mercure, H., & Nobert, Y. (1992). A Branch and Bound Algorithm for a Class of Asymmetrical Vehicle Routeing Problems. *Journal of the Operational Research Society* , *43*(5), 469–481. https://doi.org/https://doi.org/10.1057/jors.1992.73

Laporte, G., & Nobert, Y. (1986). An Exact Algorithm for the Asymmetrical Capacitated Vehicle Routing Problem. *Networks*, *16*(1), 33–46. https://doi.org/https://doi.org/10.1002/net.3230160104

Laporte, G., & Nobert, Y. (1987). Exact algorithms for the vehicle routing problem. *North-Holland Mathematics Studies*, *132*, 147–184. https://doi.org/https://doi.org/10.1016/S0304-0208(08)73235-3

Laporte, G., Nobert, Y., & Desrochers, M. (1985). Optimal Routing under Capacity and Distance Restrictions. *Operations Research*, *33*(5), 1050–1073.

Lee, S., Kang, Y., & Prabhu, V. V. (2016b). Smart logistics : distributed control of green crowdsourced parcel services. *International Journal of Production Research*, *7543*, 0. https://doi.org/10.1080/00207543.2015.1132856

Lee, Y. H., Kim, J. I., Kang, K. H., & Kim, K. H. (2008). A heuristic for vehicle fleet mix problem using tabu search and set partitioning. *Journal of the Operational Research Society*, *59*(6), 833–841. https://doi.org/10.1057/palgrave.jors.2602421

Lee, Y., & Orlin, J. B. (1994). On very large scale assignment problems. In *Large Scale Optimization* (pp. 206–244). Springer.

Levin, M. W., Kockelman, K. M., Boyles, S. D., & Li, T. (2017). A general framework for modeling shared autonomous vehicles with dynamic network-loading and dynamic ride-sharing application. *Computers, Environment and Urban Systems*, *64*, 373–383. https://doi.org/10.1016/j.compenvurbsys.2017.04.006

Li, B., Krushinsky, D., Reijers, H. A., & van Woensel, T. (2014). The Share-A-Ride Problem: People and parcels sharing taxis. *European Journal of Operational Research*, *238*(1), 31–40. https://doi.org/10.1016/j.ejor.2014.03.003

Li, F., Golden, B., & Wasil, E. (2007). The open vehicle routing problem: Algorithms, large-scale test problems, and computational results. *Computers and Operations Research*, *34*(10), 2918–2930. https://doi.org/10.1016/j.cor.2005.11.018

Little, J. D. C., Murty, K. G., Sweeney, D. W., & Karel, C. (1963). An Algorithm for the Traveling Salesman Problem. *Operations Research*, *11*(6), 972–989. https://www.jstor.org/stable/167836

Ma, S., Zheng, Y., & Wolfson, O. (2015). Real-Time City-Scale Taxi Ridesharing. *IEEE Transactions on Knowledge and Data Engineering*, *27*(7), 1782–1795. https://doi.org/10.1109/TKDE.2014.2334313

Macrina, G., Di Puglia Pugliese, L., Guerriero, F., & Laganà, D. (2017). The Vehicle Routing Problem with Occasional Drivers and Time Windows. *Springer Proceedings in Mathematics and Statistics*, *217*, 577–587. https://doi.org/10.1007/978-3-319-67308-0_58

Macrina, G., di Puglia Pugliese, L., Guerriero, F., & Laganà, D. (2017). The Vehicle Routing Problem with Occasional Drivers and Time Windows. *Springer Proceedings in Mathematics and Statistics*, *217*, 577–587. https://doi.org/10.1007/978-3-319-67308-0_58

Masson, R., Trentini, A., Lehuédé, F., Malhéné, N., Masson, R., Trentini, A., Lehuédé, F., Malhéné, N., Péton, O., & Olivier, P. (2017). *Optimization of a city logistics transportation system with mixed passengers and goods To cite this version : HAL Id : hal-01068305 Optimization of a city logistics transportation system with mixed passengers and goods*.

Nelson, M. D., Nygard, K. E., Griffin, J. H., & Shreve, W. E. (1985). Implementation techniques for the vehicle routing problem. *Computers and Operations Research*, *12*(3), 273–283. https://doi.org/10.1016/0305-0548(85)90026-7

Nikolaev, A. G., & Jacobson, S. H. (2010). Simulated Annealing. In M. Gendreau & J.-Y. Potvin (Eds.), *Handbook of Metaheuristics* (pp. 1–39). Springer US. https://doi.org/10.1007/978-1-4419-1665-5_1

Pelzer, D., Xiao, J., Zehe, D., Lees, M. H., Knoll, A. C., & Aydt, H. (2015). A Partition-Based Match Making Algorithm for Dynamic Ridesharing. *IEEE Transactions on Intelligent Transportation Systems*, *16*(5), 2587–2598. https://doi.org/10.1109/TITS.2015.2413453

Prins, C. (2004). A simple and effective evolutionary algorithm for the vehicle routing problem. *Computers and Operations Research*, *31*(12), 1985–2002. https://doi.org/10.1016/S0305-0548(03)00158-8

Punel, A., Ermagun, A., & Stathopoulos, A. (2018). Studying determinants of crowd-shipping use. *Travel Behaviour and Society*, *12*(April), 30–40. https://doi.org/10.1016/j.tbs.2018.03.005

Punel, A., & Stathopoulos, A. (2017). Modeling the acceptability of crowdsourced goods deliveries : Role of context and experience effects. *Transportation Research Part E*, *105*, 18–38. https://doi.org/10.1016/j.tre.2017.06.007

Qi, W., Li, L., Liu, S., & Shen, Z. J. M. (2018). Shared mobility for last-mile delivery: Design, operational prescriptions, and environmental impact. *Manufacturing and Service Operations Management*, *20*(4), 737–751. https://doi.org/10.1287/msom.2017.0683

Qian, X., Zhang, W., Ukkusuri, S. V., & Yang, C. (2017). Optimal assignment and incentive design in the taxi group ride problem. *Transportation Research Part B: Methodological*, *103*, 208–226. https://doi.org/10.1016/j.trb.2017.03.001

Quadrifoglio, L., Dessouky, M. M., & Ordóñez, F. (2008). Mobility allowance shuttle transit (MAST) services: MIP formulation and strengthening with logic constraints. *European Journal of Operational Research*, *185*(2), 481–494. https://doi.org/10.1016/j.ejor.2006.12.030

Rai, H. B., Verlinde, S., Merckx, J., & Macharis, C. (2017). Crowd logistics: an opportunity for more sustainable urban freight transport? *European Transport Research Review*, *9*(3), 1–13. https://doi.org/10.1007/s12544-017-0256-6

Rayle, L., Dai, D., Chan, N., Cervero, R., & Shaheen, S. (2016). Just a better taxi? A survey-based comparison of taxis, transit, and ridesourcing services in San Francisco. *Transport Policy*, *45*, 168–178. https://doi.org/10.1016/j.tranpol.2015.10.004

Rechavi, A., & Toch, E. (2020). Crowd logistics: Understanding auction-based pricing and couriers' strategies in crowdsourcing package delivery. *Journal of Intelligent Transportation Systems: Technology, Planning, and Operations*, *0*(0), 1–16. https://doi.org/10.1080/15472450.2020.1797503

Ropke, S., & Cordeau, J. F. (2009). Branch and cut and price for the pickup and delivery problem with time windows. *Transportation Science*, *43*(3), 267–286. https://doi.org/10.1287/trsc.1090.0272

Rougès, J., & Montreuil, B. (2014). *Crowdsourcing delivery : New interconnected business models to reinvent delivery*. 1–19.

Rougès, J.-F., & Montreuil, B. (2014). Crowdsourcing Delivery : New Interconnected Business Models to Reinvent Delivery. *1st International Physical Internet Conference*, *1*, 1–19.

Ryan, D. M., Hjorring, C., & Glover, F. (1993). Extensions of the petal method for vehicle routeing. *Journal of the Operational Research Society*, *44*(3), 289–296. https://doi.org/10.1057/jors.1993.54

Shaheen, S., & Cohen, A. (2019). Shared ride services in North America: definitions, impacts, and the future of pooling. *Transport Reviews*, *39*(4), 427–442. https://doi.org/10.1080/01441647.2018.1497728

Simonetto, A., Monteil, J., & Gambella, C. (2019). Real-time city-scale ridesharing via linear assignment problems. *Transportation Research Part C: Emerging Technologies*, *101*(January), 208–232. https://doi.org/10.1016/j.trc.2019.01.019

Solomon, M. M. (1987). *Algorithms for the Vehicle Routing and Scheduling Problems with Time Window Constraints Author (s): Marius M . Solomon Published by : INFORMS Stable URL : https://www.jstor.org/stable/170697 REFERENCES Linked references are available on JSTOR for this a*. *35*(2), 254–265.

Statista. (2021). *Retail e-commerce revenue in the United States from 2017 to 2024*. https://www.statista.com/statistics/272391/us-retail-e-commerce-sales-forecast/

Taillard, E. (1993). *Parallel Iterative Search Methods for Vehicle Routing Problems*. 661–673.

Trentini, A., & Malhene, N. (2010). Toward a Shared Urban Transport System Ensuring Passengers & Goods Cohabitation. *TeMA - Trimestrale Del Laboratorio Territorio Mobility Ambiente*, *3*. https://doi.org/10.6092/1970-9870/165

Trentini, A., Masson, R., Lehuédé, F., Malhéné, N.,(2013). *A shared " passengers goods " city logistics system To cite this version : HAL Id : hal-00861728*.

Ulmer, M. W., Thomas, B. W., Campbell, A. M., & Woyak, N. (2021). The restaurant meal delivery problem: Dynamic pickup and delivery with deadlines and random ready times. *Transportation Science*, *55*(1), 75–100. https://doi.org/10.1287/TRSC.2020.1000

Voccia, S. A., Campbell, A. M., & Thomas, B. W. (2019). The same-day delivery problem for online purchases. *Transportation Science*, *53*(1), 167–184. https://doi.org/10.1287/trsc.2016.0732

Wang, H., & Yang, H. (2019). Ridesourcing systems : A framework and review. *Transportation Research Part B: Methodological*, *129*, 122–155.

Woensel, V. (2013). *Integrating passenger and freight transportation : model formulation and insights*.

Wren, A., & Hollidayt, A. (1972). *Computer Scheduling of Vehicles from One or More Depots to a Number of Delivery Points*. 333–344.

Xu, J., & Kelly, J. P. (1996). A network flow-based Tabu Search heuristic for the vehicle routing problem. *Transportation Science*, *30*(4), 379–393. https://doi.org/10.1287/trsc.30.4.379

Yannakakis, M. (1985). On a Class of Totally Unimodular Matrices. *Mathematics of Operations Research*, *10*(2), 280–304.

Yen, J. Y. (1971). Finding the K Shortest Loopless Paths. *Management Science*, *17*(11), 712–716. http://www.jstor.org/stable/2629312

Yildiz, B., & Savelsbergh, M. (2019). Service and capacity planning in crowd-sourced delivery. *Transportation Research Part C*, *100*(January), 177–199. https://doi.org/10.1016/j.trc.2019.01.021

Zhou, Z., Chen, R., & Guo, S. (2021). A domain-of-influence based pricing strategy for task assignment in crowdsourcing package delivery. *IET Intelligent Transport Systems*, *15*(6), 808–823. https://doi.org/10.1049/itr2.12062

Appendix A Mathematical notations used in this dissertation

Notation	Description
0	*Distribution center depot*
A_{prk}	*Binary, whether a package* p *could be visited by* r^{th}*route of SPV k*
$a^{s}_{i,j,k}$	*Binary, whether a node j could be visited by* i^{th}*route of SPV k*
$a^{d}_{i,j,k}$	*Binary, whether a node j could be visited by* i^{th}*route of dedicated vehicle k*
a_t	*The action to take at the stage t*
b_k	*Travel budget of vehicle k*
γ	*Adjustment factor in dynamic programming*
C_t	*The total delivery cost incurred during stage t*
$c^{s}_{i,j}$	*Monetized travel cost to use link* $(i,\ j)$ *for shared vehicle*
$c^{d}_{i,j}$	*Monetized travel cost to use link* $(i,\ j)$ *for dedicated vehicle*
c_{0s_k}	*Monetized cost from origin to hub 0 for shared vehicle k*
c_{sk}	*Monetized cost from origin to destination for shared vehicle k*
c^{s}_{ik}	*Cost of using route* i *for shared vehicle k*
c^{s}_{ik}	*Cost of using route* i *for dedicated vehicle k*
c_{ir}	*Cost of using route r of shared vehicle k to serve package* i
c_{rk}	*Cost of* r^{th} *route of shared vehicle k*
c_f	*Fixed cost when a truck/DV is used*
$\bar{c}_t$	*Average variable cost of package delivery using truck*
D	*Dedicated delivery vehicle set*
d_k	*Individual dedicated vehicle k,*d_k $\in D$
E_t	*The state of a stage*
e	*Compensation to shared vehicle for each drop-off, fixed*
$G\ =\ (N, A)$	*Network G consists of Vertexes/Nodes and Arcs/Links*
(i, j)	*A tuple to describe a link between node* i *and node j*
h	*Distribution center arriving hub*
K	*Number of all available vehicles*
K_s	*Number of shared-personal vehicles*
K_d	*Number of dedicated vehicles*
M	*A large Number*
N_{sk}	*Destination node for shared vehicle k*
N_p	*Drop-off node for package p*
P	*Set of packages to be delivered*
p	*Individual package p,* $p \in P$

q_{sk}	*Capacity for shared vehicle k*
R	*The set of feasible routes of all shared personal vehicles*
S	*Shared-personal vehicle set*
s_k	*Individual shared vehicle k,* $s_k \in S$
τ_{ij}	*Time cost to use link* $(i,\ j)$
T_d^{sk}	*Departing time of vehicle k from hub 0*
T_a^{sk}	*Arriving time of vehicle k to destination*
T_d^p	*Earliest pickup time for package p*
T_a^p	*Latest arrival time for package p*
t_i^k	*The time for Vehicle k to arrive at Node* i
Θ	*Objective function values*
u_i	*Binary, indicates whether a truck* i *is used*
U_t	*Total cost of stage* t *and all stage afterwards*
V	*Total vehicle set*
ω	*Service reward adjustment factor, a relatively large number*
x_{ij}^k	*Binary, indicates whether a link* $(i,\ j)$ *is visited by vehicle k*
x_{ir}^k	*Binary, indicates whether a package* i *is served by* r^{th}*route of vehicle k*
x_{prk}	*Binary, indicates whether a package* p *is carried by* r^{th}*route of vehicle k*
$y_{i,k}^s$	*Binary, whether the* i^{th} *feasible route of shared vehicle k is used*
$y_{i,k}^s$	*Binary, whether the* i^{th} *feasible route of dedicated vehicle k is used*
z_{rk}	*Binary, whether the* r^{th}*route of shared vehicle k is used*
z_{sk}	*Binary, indicates whether a shared vehicle k is carrying any packages*