



THE HONG KONG  
POLYTECHNIC UNIVERSITY

香港理工大學

Pao Yue-kong Library

包玉剛圖書館

---

## Copyright Undertaking

This thesis is protected by copyright, with all rights reserved.

**By reading and using the thesis, the reader understands and agrees to the following terms:**

1. The reader will abide by the rules and legal ordinances governing copyright regarding the use of the thesis.
2. The reader will use the thesis for the purpose of research or private study only and not for distribution or further reproduction or any other purpose.
3. The reader agrees to indemnify and hold the University harmless from and against any loss, damage, cost, liability or expenses arising from copyright infringement or unauthorized usage.

### IMPORTANT

If you have reasons to believe that any materials in this thesis are deemed not suitable to be distributed in this form, or a copyright owner having difficulty with the material being included in our database, please contact [lbsys@polyu.edu.hk](mailto:lbsys@polyu.edu.hk) providing details. The Library will look into your claim and consider taking remedial action upon receipt of the written requests.

**DESIGN AND OPTIMIZATION OF SHARED  
MOBILITY SYSTEMS**

**HUANG JIANGYAN**

**PhD**

**The Hong Kong Polytechnic University**

**2025**



**The Hong Kong Polytechnic University**  
**Department of Industrial and Systems Engineering**

**Design and Optimization of Shared Mobility Systems**

**HUANG Jiangyan**

**A thesis submitted in partial fulfillment of the  
requirements for the degree of Doctor of Philosophy**

**August 2024**



## Certificate of Originality

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degree or diploma, except where due acknowledgement has been made in the text.

\_\_\_\_\_ (Signed)

\_\_\_\_\_ HUANG Jiangyan \_\_\_\_\_ (Name of student)



## Abstract

Rapid urbanization poses significant challenges to urban mobility. Shared mobility services, defined as the collective utilization of transportation resources, have emerged as a promising solution to these challenges. However, their implementation faces complex decision-making issues including vehicle routing, order assignment, pricing scheme design, etc., particularly under dynamic or uncertain conditions. Addressing these issues is crucial for fostering an efficient and sustainable mobility ecosystem.

This thesis addresses three key decision-making problems within shared mobility services: dynamic vehicle dispatching for shared-and-autonomous-mobility (SAM) services incorporating ride-pooling, compensation scheme design for integrative shared mobility (ISM) services under stochastic demand, and public transit line planning (PTLP) with bike-sharing integration.

The first research problem investigates dynamic vehicle dispatching for SAM services with ride-pooling options. An algorithmic framework based on a rolling horizon approach is proposed, continually updating vehicle dispatch plans based on real-time demand information by solving a series of static subproblems. Each static subproblem is formulated as a mixed-integer programming (MIP) model and solved by a customized hybrid algorithm, named ARA-LNS, which integrates an adaptive request assignment (ARA) into a large neighborhood search (LNS) heuristic framework to efficiently optimize the request assignment and vehicle routing plans.

The second research problem explores the optimal compensation scheme design for ISM services that simultaneously provide both passenger ride and parcel delivery services using an on-demand shared vehicle fleet. To address the extra ride duration (ERD) caused by additional stops, the service operator compensates passengers, whose tolerance for ERD depends on the compensation amount. The problem is formulated as a two-stage stochastic programming model considering passengers' nonlinear acceptable ERD (AERD) profile and stochastic demands and solved by a sample

average approximation method. A customized ALNS-CSA algorithm that combines an adaptive large neighborhood search (ALNS) heuristic and an efficient compensation scheme adjustment (CSA) method is developed to iteratively determine the optimal demand serving, passenger compensation, and vehicle routing (DPV) solution and improve the compensation scheme accordingly while respecting the AERD constraints.

The third research problem focuses on the optimal design of the public transit line with integrated bike-sharing services to determine the optimal bus stop location and service frequency by minimizing total system costs, including both user and operator expenses. A simulation-based optimization modeling framework powered by a multi-agent-based simulation (MABS) system is developed to capture disaggregate behaviors and interactions of various entities in the bus operation system, especially incorporating the bike-sharing complementary feeder mode services. A surrogate-based optimization (SBO) solution method is introduced to solve the black-box simulation-based PTLP problem by efficiently approximating the mapping relationship between bus transit planning decision inputs and expected system cost output. This method allows us to identify high-quality stop location and service frequency solutions within a few objective function simulation evaluations.

The efficacy of all the proposed models and solution methods for the three research problems is evaluated through extensive numerical experiments. Impact analyses of potentially influential factors are also conducted to derive managerial insights to guide the practical management and operations of shared mobility services.

## Publications Arising from the Thesis

### Publications

1. **Huang, J.**, Xu, M., 2024. Travel time estimation with improved neural network. *Proceedings of the 103rd Annual Meeting of Transportation Research Board*. Washington D.C., U.S. January 2024.
2. Wong, W. F., **Huang, J.**, Xu, M., 2022. Exploration of the public's willingness to participate in crowd-sourcing services: A case study of Hong Kong. *Proceedings of the 22nd COTA International Conference of Transportation Professionals*. Changsha, Hunan, China. July 2022.

### Working Papers

1. Xu, M., **Huang, J.**, 2024. Optimal compensation scheme design for integrative shared mobility services considering stochastic demands. Submitted to *Transportation Research Part B: Methodological*, under review.
2. **Huang, J.**, Xu, M., 2024. Dynamic vehicle dispatching for shared-and-autonomous-mobility services considering ride-pooling.
3. **Huang, J.**, Xu, M., 2024. Simulation-based optimization of public transit line design considering bike-sharing integration.
4. **Huang, J.**, Xu, M., 2024. An overview of emerging logistic solutions of integrative urban delivery systems.
5. **Huang, J.**, Xu, M., 2024. Improved travel time estimation with neural network considering residual prediction.

### Conference Presentations

1. **Huang, J.**, Xu, M., 2023. Enhancing bus line performance based on simulation and optimization framework. *Presentation at the 27th Hong Kong Society for Transportation Studies*. Hong Kong, China. December 2023.

2. **Huang, J.**, Xu, M., 2023. Optimal compensation scheme design for integrative shared mobility services. *Presentation at the 9th International Symposium on Transport Network Resilience.* Hong Kong, China. December 2023.

## Acknowledgements

I am profoundly grateful to my supervisor, Prof. Xu Min, for her invaluable guidance and support throughout my three-year PhD studies at the Hong Kong Polytechnic University. Prof. Xu's expertise has profoundly shaped my research skills and critical thinking abilities. Her dedication and passion for research have inspired me and will continue to motivate my future career endeavors.

I would like to express my deep gratitude to the Hong Kong Polytechnic University for providing me with financial support that has been essential to my academic development. I also wish to be thankful to my research team members—Peng Shouguo, Yang Lu, Ji Rumei, Su Hang, Ren Qiaoqiao, Wang Yilun, Chen Xiaohua, Ran Tanghong, Tu Lin, Xiong Ya, Wang Li, Wu Ting, Yan Xiaoyuan, Lei Da, and Zhang Lin. Heartfelt thanks are extended to Zhang Zhuoye from The University of Hong Kong, Cheng Qixiu from the University of Bristol, Liu Yang from the Tsinghua University, An Qinhe from Southeast University, and Zhu Manman, He Yonghuan, Li Xin, CHU Yin Ting, Lin Yuqian, Zhu Siyu from The Hong Kong Polytechnic University for their generous helps and support. Their companionship ensured that I never felt alone, transforming my challenging academic journey at PolyU into a series of cherished memories filled with camaraderie and surprise.

Last but not least, my heartfelt thanks go to my family, whose unconditional love and support have been the backbone of my perseverance throughout these three years. Thanks to my parents for not only giving me the wings to soar but also ensuring a safe landing during times of struggle. Thanks to them for enlightening me profound values of lifelong learning and the importance of pursuing my passions with heart and conviction. Their unswerving belief in me has been my most reliable safety net, giving me the courage to overcome any obstacle. Their love is, without a doubt, the most precious gift I could ever receive.



## Table of Contents

<b>Abstract.....</b>	<b>I</b>
<b>Publications Arising from the Thesis .....</b>	<b>III</b>
<b>Acknowledgements .....</b>	<b>V</b>
<b>Table of Contents .....</b>	<b>VII</b>
<b>List of Figures.....</b>	<b>XI</b>
<b>List of Tables.....</b>	<b>XIV</b>
<b>Glossary of Abbreviations .....</b>	<b>XV</b>
<b>Chapter 1 Introduction.....</b>	<b>1</b>
1.1 Background.....	1
1.1.1 Practice state of shared mobility services .....	2
1.1.2 Decision-making challenges under shared mobility context .....	9
1.2 Research Scope and Objectives .....	12
1.3 Thesis Organization .....	14
<b>Chapter 2 Literature Review .....</b>	<b>17</b>
2.1 Passenger-Sharing Transportation Services.....	17
2.2 Passenger-and-Goods Transportation Services.....	21
2.3 Public Transit Planning and Potential Shared Mobility Integration .....	24
2.4 Limitations of Existing Studies .....	28
<b>Chapter 3 Dynamic Vehicle Dispatching for Shared-and-Autonomous-Mobility Services Considering Ride-Pooling .....</b>	<b>33</b>

3.1 Assumptions, Notations and Problem Statement.....	34
3.1.1 Demand characterization .....	35
3.1.2 Passengers' satisfaction.....	36
3.1.3 SAV states .....	37
3.2 Dynamic Vehicle Dispatching Framework.....	38
3.3 Static Problem Formulation.....	42
3.4 Solution Methodology .....	46
3.4.1 ARA-LNS structure for S-SAVD.....	47
3.4.2 SAV dispatching plan generation.....	52
3.4.3 Adaptive request assignment scheme.....	55
3.5 Numerical Experiments .....	60
3.5.1 Instance generation and parameter settings .....	60
3.5.2 Evaluation against the benchmark approach.....	63
3.5.3 Impact analysis.....	65
3.6 Concluding Remarks .....	75
3.7 Appendix. Notation.....	76

**Chapter 4 Optimal Compensation Scheme Design for Integrative Shared Mobility Services..... 79**

4.1 Assumptions, Notations and Problem Statement.....	80
4.1.1 Compensation scheme .....	82
4.1.2 Passenger's AERD profile .....	83
4.2 Two-Stage Stochastic Programming Model .....	84

4.3 Solution Algorithm Design .....	88
4.3.1 ALNS-CSA heuristic algorithm framework.....	89
4.3.2 New DPV solution generation .....	92
4.3.3 Compensation scheme determination .....	99
4.4 Numerical Experiments .....	104
4.4.1 Test instances and parameter setting.....	105
4.4.2 Algorithm performance.....	107
4.4.3 Impact analysis.....	114
4.5 Concluding Remarks .....	119
4.6 Appendix. Notation.....	121
<b>Chapter 5 Simulation-Based Optimization of Public Transit Line Design Considering Bike-Sharing Integration.....</b>	<b>123</b>
5.1 Problem Statement and Optimization Framework .....	124
5.1.1 Problem statement.....	124
5.1.2 Optimization framework.....	126
5.2 MABS System for Public Transit Line.....	129
5.2.1 Passenger travel behavior, bus motion and cost function formulation.....	130
5.2.2 Simulation system development .....	140
5.3 Solution Method .....	147
5.3.1 Gaussian process surrogate .....	149
5.3.2 Acquisition function.....	151

5.4 Numerical Experiments .....	154
5.4.1 Computational performance.....	154
5.4.2 Real-world case study .....	165
5.5 Concluding Remarks .....	174
5.6 Appendix. Notation.....	175
<b>Chapter 6 Conclusions and Future Research Recommendations .....</b>	<b>179</b>
6.1 Overview and Research Contributions .....	179
6.2 Recommendations for Future Research.....	180
<b>References.....</b>	<b>183</b>

## List of Figures

Figure 1.1. Shared mobility market size in USD billion from 2023 to 2034 (Precedence Research, 2024).....	6
Figure 1.2. Major global passenger-sharing service platforms (Burger, 2019) .....	7
Figure 1.3. The integration of micromobility with public transit systems in Jakarta (ITDP, 2021) .....	8
Figure 1.4. Research topics investigated in this thesis.....	14
Figure 3.1. SAV dispatching illustration.....	38
Figure 3.2. Dynamic vehicle dispatching framework.....	40
Figure 3.3. Impact of the flexibility time on the total profit.....	71
Figure 3.4. Impact of the flexibility time on the average waiting time .....	72
Figure 3.5. Impact of the flexibility time on the number of served requests.....	72
Figure 3.6. Impact of the flexibility time on the served request ratio.....	72
Figure 3.7. Impact of the flexibility time on the number of pooled requests .....	73
Figure 3.8. Impact of the flexibility time on the pooled request ratio .....	73
Figure 4.1. Compensation scheme illustration.....	83
Figure 4.2. Stepwise compensation scheme illustration .....	84
Figure 4.3. Increase adjustment illustration.....	103
Figure 4.4. Decrease adjustment illustration.....	103
Figure 4.5. Flowchart of our proposed CSA method.....	104
Figure 4.6. Variations in computation time of the ALNS-CAS algorithm with request numbers.....	112

Figure 4.7. Average call percentage of removal and insertion operators.....	113
Figure 4.8. Variations of compensation level and profit with parcel number increase .....	117
Figure 4.9. Variations in the compensation level and profit under different AERD parameters.....	119
Figure 5.1. Different bus stop-to-stop movement patterns .....	137
Figure 5.2. Relationship between different agents.....	143
Figure 5.3. State transitions of bus agent.....	143
Figure 5.4. Workflow of the agent-based public transit line simulation system.....	147
Figure 5.5. Flowchart of the iterative SBO solution method.....	154
Figure 5.6. Computational efficacy comparison between proposed solution method and benchmark approach .....	164
Figure 5.7. The KMB 3D bus route .....	167
Figure 5.8. Cumulative passenger counts of the KMB 3D bus route during the morning peak-hour period .....	167
Figure 5.9. Variations of the passenger and operator costs with an increase in the demand level.....	169
Figure 5.10. Variations of the stop density with an increase in the demand level.....	170
Figure 5.11. Variations of the average waiting time, accessing and egressing time and in-vehicle time of passengers with an increase in the demand level .....	171
Figure 5.12. Variations of the passenger and operator costs with an increase in the bus capacity .....	172
Figure 5.13. Variations of the average waiting time, accessing and egressing time and	

in-vehicle time of passengers with an increase in the bus capacity ..... 172

## List of Tables

Table 3.1. Comparison of the proposed solution method with the benchmark method .....	65
Table 3.2. Comparison of the SAMw/oP and SAM service .....	67
Table 3.3. Effect of flexibility time on system performance with different SAV and request number combinations .....	73
Table 4.1. Small-scale instances evaluation of ALNS-CSA algorithm versus Gurobi .....	110
Table 4.2. Large-scale instances evaluation of ALNS-CSA algorithm.....	111
Table 4.3. Contribution of operators in terms of solution gap .....	113
Table 4.4. Performance of operator selection mechanism in the ALNS-CSA algorithm .....	114
Table 4.5. Result comparison of ISM services with two benchmark services.....	116
Table 4.6. Impacts of the parcel request number .....	117
Table 4.7. Impacts of AERD parameter .....	118
Table 5.1. Comparison of the proposed solution method and the benchmark approach for different scenarios on test instances .....	164
Table 5.2. Bus stop locations of No. 3D bus route in both directions .....	166
Table 5.3. Comparison of the proposed solution method and the benchmark approach on KMB 3D bus route.....	173
Table 5.4. Effect of demand levels on system performance .....	173
Table 5.5. Effect of bus capacity on system performance .....	173

## Glossary of Abbreviations

SAM	Shared-and-autonomous-mobility
SAMw/oP	Shared-and-autonomous-mobility without pooling
ISM	Integrative shared mobility
PTLP	Public transit line planning
SAV	Shared autonomous vehicle
MIP	Mixed-integer programming
ARA	Adaptive request assignment
LNS	Large neighborhood search
ERD	Excess ride duration
AERD	Acceptable excess ride duration
ALNS	Adaptive large neighborhood search
CSA	Compensation scheme adjustment
DPV	Demand servicing, passenger compensation, and vehicle routing
SBO	Surrogate-based optimization
AV	Autonomous vehicle
TNC	Transportation network companies
RT-SAVD	Real-time shared autonomous vehicle dispatching
S-SAVD	Static shared autonomous vehicle dispatching
DARP	Dial-a-ride problem
EV	Electric vehicle
OMS	On-demand mobility services
VRP	Vehicle routing problem
SARP	Share-a-ride problem
PDP	Pickup and delivery problem

SAA	Sample average approximation
VOT	Value of time
CSD	Compensation scheme design
Pw/P	Passenger transportation services with pooling
Pw/oP	Passenger transportation services without pooling
MABS	Multi-agent-based simulation
GP	Gaussian process
EI	Expected improvement
ABC	Artificial bee colony

# Chapter 1 Introduction

## 1.1 Background

The global urban population is projected to approach nearly 5 billion by 2030 (Seto et al., 2012), and the escalating urbanization rates directly connect to the surging travel demand within urban cities, putting immense pressure on existing transportation infrastructures. According to the Organization for Economic Co-operation and Development, post-pandemic travel data in cities like Tokyo and London show an obvious increase in commuter traffic, highlighting the quick rebound in urban travel demands (OECD, 2023). A recent report by the American Public Transportation Association predicts continued growth in public transit usage and indicates the potential overstress across major US cities (APTA, 2024). In the meantime, the population growth affects not only passenger transport but also urban delivery systems, presenting both opportunities and challenges. This growth in urban deliveries has been propelled by growing e-commerce sales, a trend that has been further accelerated during the pandemic (Bhatti et al., 2020; United Nations, 2021). By 2024, it is anticipated that global retail e-commerce sales will exceed \$6.3 trillion, with projections of even greater growth in the years ahead (van Gelder, 2024). These surges in both passenger and goods transportation demand intensify urban challenges such as prolonged traffic congestion, increased pollution levels, and strains on existing transport systems.

In response to these pressing issues, cities worldwide have begun to embrace more flexible and sustainable transportation services to ensure efficient urban mobility. The concept of the sharing economy has driven shared mobility into a transformative force that could reshape future mobility patterns, affecting not just people, but for goods transportation as well. Within the academic field of transportation, the shared mobility service signifies a significant evolution in investigating how we access and utilize transportation resources. The shift from personal ownership to broader access to shared mobility resources represents a fundamental change in transportation dynamics and

introduces complexities in transportation planning and management. The background of this thesis will explore the current state of shared mobility in practice and identify critical decision-making challenges associated with shared mobility. It will highlight the significance and necessity of designing and optimizing these integrative systems to maximize efficiency, cost-effectiveness, and sustainability in urban transportation.

### **1.1.1 Practice state of shared mobility services**

This subsection will provide an overview of the fundamental aspects of shared mobility, followed by a discussion of its global implementation and practical applications.

#### **(1) Basic facts of shared mobility**

Shared mobility involves the collective utilization of transport modes such as vehicles and bicycles either concurrently or one after another (Shaheen and Cohen, 2013). Enabled by technological advancements, shared mobility service systems offer mobility as a service, allowing users to utilize transport options on-demand and on a short-term basis. The recent fascination with shared mobility has opened a new epoch of how we conceptualize the transportation of people and goods, significantly reducing reliance on personal vehicles and fostering sustainable urban mobility. Specifically, this trend has led to the development of various forms of passenger-sharing transportation alternatives, which decrease the need for private car ownership and support environmentally friendly travel habits. Additionally, the innovative idea of simultaneously transporting passengers and goods has gained growing interest in recent years (Cheng et al., 2023). Such a combination of passenger and goods transportation streams in urban cities can further facilitate efficient use of the existing passenger transportation vehicles and meet the rising demand for intracity parcel deliveries (e.g., food and grocery delivery) (Bouton et al., 2017). Besides, the shared mobility concept extends beyond simply sharing vehicles to address urban transport issues. It represents

a broader shift towards interconnected transportation networks by seamlessly integrating with existing public transport infrastructures to enhance connectivity and accessibility of urban mobility (Teixeira et al., 2021). In what follows, we will specify these shared mobility services in three broad categories, focusing on passenger-sharing transportation, combined passenger-and-goods transportation, and the integration of shared mobility with current urban transport planning.

### Passenger-sharing transportation

The passenger-sharing concept has transformed urban transport landscapes, offering adaptable solutions to evolving mobility demands. For a generalized definition, the systems with travelers sharing their rides to improve vehicle utilization and decrease travel costs can be termed “ridesharing” (Furuhata et al., 2013; Mourad et al., 2019). According to different service features, they could be further divided into more specialized business models, like carsharing, carpooling, ride-sourcing, vanpooling, dial-a-ride and shared-taxi service. Specifically, carsharing provides temporary private use of vehicles for a short duration as required. Effective vehicle relocations are essential to address spatial and temporal imbalances in vehicle distribution caused during operations. Carpooling is another form of shared mobility variant typically popular for daily commutes in one-off journeys, where drivers can find peer travelers with comparable travel itineraries to share the costs of the journey, such as fuel and tolls (Agatz et al., 2011, 2012; Sun et al., 2020). Ride-sourcing, also known as ride-hailing, leverages digital platforms to link passengers with nearby drivers who employ their personal, non-commercial vehicles to offer transportation services on demand (Rayle et al., 2016; Shaheen and Cohen, 2019). Vanpooling uses larger vehicles for group travel, significantly cutting individual commuting expenses and reducing environmental impact (Kaan and Olinick, 2013). Additionally, programs like dial-a-ride provide flexible and adaptable door-to-door services that typically cater to the specific transportation needs of specific client groups, such as seniors or individuals with

disabilities. (Cordeau and Laporte, 2006; Ho et al., 2018). Similarly, shared-taxi services enable multiple passengers, often strangers, to share rides in taxis, optimizing passenger assignments and vehicle routes to improve efficiency and reduce travel costs. (Hosni et al., 2014; Hua et al., 2022). In both dial-a-ride and shared-taxi systems, users typically specify their pick-up and drop-off locations, preferred timing, and maximum ride time in advance.

### *Combined passenger-and-goods transportation*

The innovative business model combines the transportation of both passengers and goods within the same vehicle by leveraging the spare capacities of existing passenger transportation systems in urban mobility, e.g., rail-hailing vehicles, taxis, buses, urban rail, etc. for goods delivery. Based on the utilization of different passenger transportation means, it can be categorized into individual vehicle-based, public transit-based and on-demand mobility vehicle-based service types. First, individual-based passenger-and-goods transportation leverages the spare capacity of private vehicles or bicycles (Buldeo Rai et al., 2017). Individual participants act as couriers by registering on digital platforms or mobile apps that enable them to deliver packages while traveling to their intended destinations (Le et al., 2019). Second, public transit-based combined transportation incorporates goods transportation into existing public transit systems, such as buses and trams. Goods are transported alongside passengers in vehicles that operate on fixed routes at pre-determined service frequencies (Elbert and Rentschler, 2022). Last, the on-demand mobility vehicle-based combined transportation offers door-to-door services by employing on-demand mobility vehicles like taxis or shared autonomous vehicles (SAVs). These vehicles are particularly effective for transporting urban dwellers and simultaneously accommodating some small parcels, documents, or takeaway meals (Mulley et al., 2018).

### Integration with existing urban transport planning

Apart from providing transportation alternatives, shared mobility services can be strategically incorporated into the broader framework of the urban transport system to develop a unified, efficient, and sustainable transportation network (Shaheen et al., 2016). Shared mobility options like bike-sharing, scooter-sharing, car-sharing, and ride-hailing play a crucial role in this integration. For instance, bike-sharing can effectively solve the first- and last-mile connectivity issue by providing a convenient mode of transportation to and from public transit stations, especially beneficial in densely populated urban areas with short travel distances (Yin et al., 2024). Similarly, scooter-sharing offers an ideal option for short-distance travel in congested urban areas, facilitating easy and quick access to and from public transit hubs. Carsharing presents a flexible alternative to car ownership, effectively decreasing the number of private vehicles on urban streets while facilitating smoother transitions for commuters on their journeys at major transit points. Ride-hailing services can also enhance the public transit system by providing flexible travel options during off-peak times and in areas with limited transit access.

## **(2) Global outlook of shared mobility**

Shared mobility services have experienced remarkable development over the past decades, holding great potential to continue to thrive in the following years. According to the Precedence Research in 2024, the global shared mobility market was valued at around \$350 billion in 2023, with a projected market size reaching nearly \$885 billion by 2034 (see Figure 1.1).

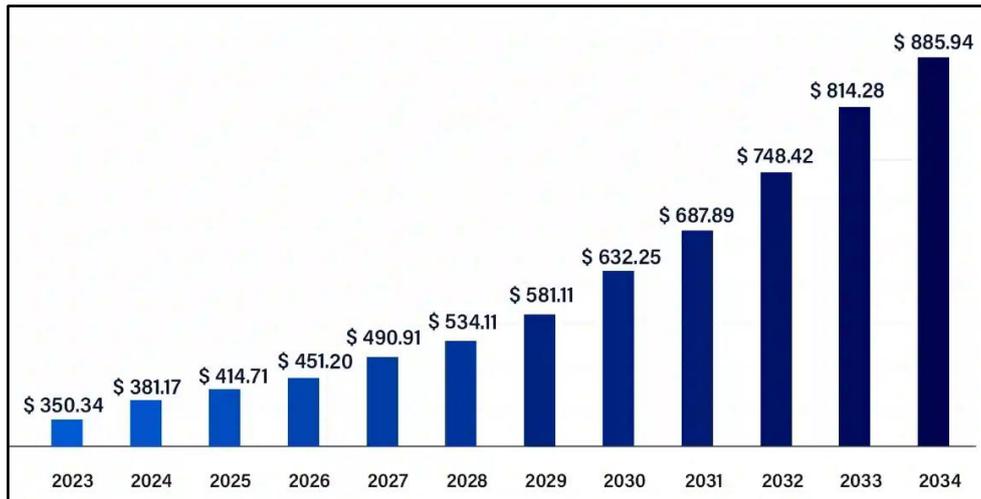


Figure 1.1. Shared mobility market size in USD billion from 2023 to 2034  
(Precedence Research, 2024)

### Expansion of passenger-sharing services

As for passenger-sharing transportation, for instance, carsharing services exemplified by Zipcar and Car2Go have seen rapid adoption, drawing millions of users across regions including China, the United States, and Europe (Shaheen et al., 2018). Concurrently, transportation network companies (TNCs), like Uber, Lyft, and Didi, have experienced rapid global expansion in their ride-sourcing services (Hartmans and Leskin, 2019; Shaheen and Cohen, 2019). These TNC platforms not only cater to immediate transportation needs but also include ride-pooling options, like UberPool and Lyft Shared. These options allow multiple passengers traveling in the same direction towards their destination(s) to share both the driver and vehicle, thereby optimizing route efficiencies and reducing individual travel costs. As depicted in Figure 1.2, the map displays the global distribution of major passenger-centric shared mobility platforms such as Uber, Lyft, and Ola in recent years. These platforms have not only transformed mobility in their home countries but also indicate a broader shift towards shared daily transportation solutions globally.





Figure 1.3. The integration of micromobility with public transit systems in Jakarta  
(ITDP, 2021)

*Potential of integrating passenger and goods transportation*

In addition to passenger transportation, some ride-hailing service providers have also expanded their services in terms of collaboration in passenger and goods movements. Pushed by a sharp revenue decline in its primary ride-hailing business, China's leading ride-hailing company, DiDi, launched an on-demand delivery service in order to offer its drivers a new avenue for income generation, after the ride-hailing demand got hammered by COVID-19 in 2020 (Sun and Goh, 2020). Grab Express, particularly valuable in Southeast Asia, is another similar successful on-demand vehicle-based initiative that facilitates the delivery of goods via a network of drivers who efficiently utilize their vehicles to transport packages to the doorsteps of recipients (Grab, 2021). Sidecar, a US-based vehicle-for-hire company, merges its original ride-hailing services with the delivery of packages containing hot food, flowers, groceries, and other items (Lien, 2015). Roadie is an example of individual vehicle-based passenger-and-goods combined transportation services that enable ordinary people, referred to as "Roadies", to utilize their vehicles' spare capacity for transporting packages while en route to their destinations, thus promoting a community-driven approach to logistics. Bussgods is a Swedish logistics and transportation service that

utilizes existing bus networks to transport packages along with passengers (Van Duin et al., 2019). This system utilizes the underutilized luggage compartment or a dedicated goods compartment in public buses to ensure efficient and economic goods delivery.

### *Operation, management and environmental benefits*

Distinctive as they are from one another today, it has been found that these shared mobility business models in practical terms have demonstrated substantial operational and environmental advantages. Passenger-sharing transportation systems are increasingly recognized for mitigating rising private vehicle use, traffic congestion, and air pollution (Sun et al., 2018). The business model of integrating passenger and goods transportation not only contributes to the reduction in freight vehicle numbers and kilometers traveled but also maximizes the efficiency of existing transport services and infrastructure by allowing people and goods to share the same road space and vehicles for portions of their journey, particularly within urban areas (Bruzzone et al., 2021; Van Duin et al., 2019). Additionally, the shared micromobility can further enhance the public transportation systems by improving the accessibility and interconnectivity among transportation modes. This integration exemplifies how traditional transport systems can be promoted by integrating with more flexible, adaptive shared mobility options (Bozzi and Aguilera, 2021). Overall, these fruitful benefits of shared mobility services indicate their indispensable role in addressing urban transport challenges by enhancing the urban mobility ecosystem with increased flexibility, efficiency, and sustainability.

### **1.1.2 Decision-making challenges under shared mobility context**

Although there are sufficient reasons to be optimistic about incorporating various shared mobility business models in the transportation mobility market, some challenges still hold back their widespread adoption. These challenges that remain to be resolved are summarized below:

## **(1) Challenges in shared-and-autonomous-mobility services**

First, the advancement in self-driving technology and autonomous vehicles (AVs) holds great potential to bring fundamental changes to urban mobility landscape (Katrakazas et al., 2015). Recent studies have indicated that sharing AVs could significantly reduce the required vehicle number and the associated fuel consumption (Levin et al., 2017). As a result, the successfully developed passenger-sharing transportation services are expected to be consolidated into one major SAM (shared-and-autonomous-mobility) service through the introduction of SAVs (Lokhandwala and Cai, 2018). The SAM service promises more efficient and seamless vehicle dispatch, allows both solo rides and pooling arrangements, and maximizes sharing benefits while reducing labor costs through manageable SAV fleet operations. Despite these favorable merits, the ever-evolving SAM system introduces many operation and decision-making challenges for service providers. Effective integration of autonomous vehicles into urban mobility requires sophisticated management of request match/assignments, vehicle dispatching (i.e., repositioning and relocation) and route optimization. Beyond operations, providers face the challenge of balancing efficiency with passenger satisfaction, particularly when coordinating multiple rides into single-vehicle trips. These pooling arrangements necessitate careful consideration of passenger comfort and convenience concerns associated with sharing space with strangers. To this end, it is imperative to build advanced models and design effective algorithms that can resolve these issues and guide the operation of SAM services.

## **(2) Challenges in combined people-and-goods mobility services**

In addition to the challenges in passenger-sharing transportation, the emerging field of combined passenger-and-goods transportation, referred to as ISM (integrative shared mobility) for short thereafter, also presents a set of challenges. Although the idea of integration of freight shipments into passenger trips is not new to some TNCs, this business model is still in the early stages of exploration due to planning and operation

complexities such as vehicle routing, trip pricing and service design, etc. (Cleophas et al., 2019; Elbert and Rentschler, 2022). At the operational level, integrating goods movement within different passenger transportation services requires customized service designs. Service providers must efficiently manage simultaneous transportation of passengers and freight within a unified transportation network. This involves decision-making in order allocation, route planning, and fleet management considering factors like vehicle capacity and customer-related constraints. Moreover, traditional pricing models designed for passenger or freight transport alone may not be suitable for such integrative services. Another significant challenge for service providers is to develop flexible and adaptable trip pricing or compensation schemes that can not only ensure profitability but also enhance user acceptance and encourage participation in these services while balancing the diverse needs of both passenger and goods clients. In order to enhance the market competitiveness of ISM service systems, the advancement of optimization models and algorithms that support practical operations and management is highly desired.

### **(3) Challenges in shared micromobility integration into urban transport planning**

Besides, incorporating shared micromobility into current urban transportation networks presents both opportunities and challenges. Shared micromobility like bike-sharing has proven to enhance public transport by solving the first- and last-mile connectivity issues, thereby promoting overall accessibility to transit networks (van Mil et al., 2021). However, incorporating these new transportation options into existing public transit operation frameworks complicates the optimal public transit design (Wu et al., 2020). Bike-sharing serves as a complementary feeder mode for public transit, replacing walking for some users while introducing new challenges in travel choice behavior that need careful management. Considering detailed disaggregate travel behaviors of passengers, coupled with bus fleet operation, is crucial for optimizing public transit, as these elements are integral for accurate system performance

measurement. However, this incorporation often leads to optimization problems characterized by nonlinearity, stochasticity, realistic constraints, and high computational costs. Consequently, to ensure a robust and efficient urban transport system that benefits both passengers and service providers, there is a genuine need to develop advanced models and effective solution methods that tackle these complexities brought by shared micromobility and aid in decision-making in public transit planning.

#### **(4) Challenges in managing dynamics and uncertainties in shared mobility**

Finally, the dynamic and unpredictable nature of urban transportation also presents significant challenges. Variations in weather conditions, demand fluctuations during different times, and special events can all impact the effectiveness of these shared mobility systems. To navigate these complexities, it is essential to integrate dynamic real-time frameworks and stochastic considerations into shared mobility research to obtain more practical and reliable solutions tailored to the evolving needs of the urban transportation environment. Thus, advanced methodologies and techniques are imperative to effectively manage these dynamic and uncertain factors, enhancing both the reliability and effectiveness of shared mobility services.

### **1.2 Research Scope and Objectives**

This thesis aims to tackle the major challenges highlighted in Subsection 1.1.2 by providing decision-making and technical support for optimizing operations and management of the future shared mobility system. As illustrated in Figure 1.4, this thesis will explore three critical areas of shared mobility covering passenger transportation, passenger-and-goods transportation and the integration with public transit to offer practical guidance to service operators and policymakers. The specific research topics and their objectives are detailed as follows:

### **Topic I. Dynamic vehicle dispatching for SAM services considering ride-pooling.**

This research topic deals with SAM services that offer passenger transportation services with the ride-pooling option. The main research issue involves developing optimization models and customized effective algorithmic to determine SAV routing plans and timely respond to new passenger requests in real-time while incorporating passengers' ride-pooling stranger number limit and satisfaction constraints to ensure the required level of service quality. **Topic I** will be investigated in Chapter 3.

### **Topic II. Compensation scheme design for ISM services under stochastic demand.**

This research topic focuses on ISM services that use an on-demand shared vehicle fleet to provide transportation services for passengers and parcels simultaneously. To encourage the acceptance of the ERD caused by parcel and passenger pickups or drop-offs, the ISM service operator will offer compensation to onboard passengers, whose tolerance for ERD is elastic in relation to the received compensation amount. The main research issue is to build the optimization model and efficiently tailored algorithm while incorporating passengers' nonlinear AERD (acceptance ERD) profile under stochastic passenger and parcel transportation demands. **Topic II** will be addressed in Chapter 4.

### **Topic III. Public transit line planning with integration of shared micromobility.**

This research topic explores the optimal design of the public transit line integrated with shared micromobility feeder services to guide the government and public transit operators. The main research issue is to construct effective and holistic models and algorithms for the simultaneous determination of the reliable bus stop location and service frequency for public transit line while considering disaggregate behaviors and interactions of involved entities in the bus operation system coupled with nonlinearity, stochasticity and realistic constraints. **Topic III** will be explored in Chapter 5.

These three research topics collectively and comprehensively address the decision-making problems relevant to shared mobility services, resolving logistical, economic,

and systemic issues to improve urban transportation efficiency and foster a sustainable shared mobility ecosystem in the future.

<p><b>Topic I</b> <b>(Chapter 3)</b></p>	<p><b>SAM service provider</b></p> <ul style="list-style-type: none"> <li>• Request assignment</li> <li>• Dynamic SAV routing</li> <li>• Ride-pooling strangers and satisfaction</li> </ul>	<p>Shared mobility for passenger transportation</p>
<p><b>Topic II</b> <b>(Chapter 4)</b></p>	<p><b>ISM service provider</b></p> <ul style="list-style-type: none"> <li>• Compensation scheme design</li> <li>• Passengers' elastic tolerance</li> <li>• Passenger and goods demand uncertainty</li> </ul>	<p>Shared mobility for passenger and goods transportation</p>
<p><b>Topic III</b> <b>(Chapter 5)</b></p>	<p><b>Government and public transit operator</b></p> <ul style="list-style-type: none"> <li>• Joint optimization of bus stop location and service frequency</li> <li>• Disaggregate behavior and practical factors</li> </ul>	<p>Shared mobility integration with public transit</p>

Figure 1.4. Research topics investigated in this thesis

### 1.3 Thesis Organization

The thesis is structured into six distinct chapters as follows:

Chapter 1 offers a background overview and motivations behind this research, outlining the research scope and objectives of the thesis.

Chapter 2 provides a comprehensive and critical review of the existing literature on decision-making problems arising from the passenger-sharing transportation services, combined passenger-and-goods transportation services and public transit planning with the potential integration of shared mobility modalities. Limitations of existing studies are identified, and research gaps are highlighted.

Chapter 3 investigates the problem of real-time vehicle dispatch for SAM services while considering ride-pooling strangers and passengers' satisfaction. A dynamic vehicle dispatching algorithmic framework based on the rolling horizon approach is

established to periodically handle a series of static subproblems. For each static subproblem, an MIP model is built, considering the maximum limit on the number of pooling strangers and passengers' satisfaction constraints. A customized algorithm is subsequently put up to address the proposed model. Numerical experiments are performed to evaluate the solution method efficacy and to analyze effects of some potentially influential factors.

Chapter 4 investigates the compensation scheme design problem for the ISM services while considering passengers' elastic tolerance for detours under stochastic demand. A two-stage stochastic programming model is built by maximizing profit. The SAA approach is utilized, and a tailored hybrid algorithm that integrates an adaptive large neighborhood search heuristic with an efficient compensation scheme adjustment is proposed to obtain the optimal compensation scheme. Numerical experiments are carried out to assess the proposed solution method, and to conduct impact analysis.

Chapter 5 seeks to optimize the design of the public transit line supported by shared bikes in pursuit of total system cost minimization. A simulation-based optimization framework is proposed to determine the stop location and service frequency solution, featuring an agent-based public transit line simulation system that is developed to mimic real-world operations and evaluate system performance. A surrogate-based optimization solution approach is developed to address the expensive-to-evaluate simulation-based optimization problem to obtain high-quality solutions through a few objective function evaluations. Numerical experiments involving both randomly generated scenarios and a real-world case study are conducted to assess the solution method efficiency and perform the impact analysis of several influential factors.

Chapter 6 summarizes the findings of this thesis and offers recommendations for future research.



## Chapter 2 Literature Review

In this chapter, we will review the related studies of the three main research topics of interest in this thesis: (i) passenger-sharing transportation services, (ii) combined passenger-and-goods transportation services, and (iii) public transit planning with the potential shared mobility integration.

### 2.1 Passenger-Sharing Transportation Services

The combination of shared mobility services with SAVs (shared autonomous vehicles), characterized by high flexibility and reliability, is foreseeable to emerge as one of the promising directions in future mobility scenarios (Zhou and Roncoli, 2022). Previous studies have shown that autonomous vehicles can significantly reduce traditional private car ownership and alleviate road network congestion (Farhan and Chen, 2018; Levin, 2017; Narayanan et al., 2020). Relevant research on the SAM (shared-and-autonomous-mobility) service operations using SAV fleets already exists, and many of them have demonstrated significant benefits of ride-pooling opportunities (Farhan and Chen, 2018; Martinez and Viegas, 2017; Fagnant and Kockelman, 2018; Loeb et al., 2018; Ge et al., 2021; Lokhandwala and Cai, 2018; W. Zhang et al., 2015). For example, Fagnant and Kockelman (2018) investigated dynamic SAV operations with the ride-pooling option, employing a discrete-time agent-based simulation framework. They found that incorporating ride-pooling greatly contributed to the reduction in customers' waiting time and overall vehicle travel mileage, while avoiding new congestion problems. Similarly, Lokhandwala and Cai (2018) developed an agent-based model to explore ride-pooling services using both traditional vehicles (with drivers' shifts and breaks) and SAVs (available all day), considering individual heterogeneous preferences in sharing rides. A real-world investigation in New York City revealed that integrating SAVs and ride-pooling services could potentially reduce fleet size and carbon emissions. Other studies have examined the use of the electric SAV fleet in SAM services. Farhan and Chen (2018) examined the operation of electric

SAVs using an agent-based simulation model, and the results indicated that ride-pooling services decrease fleet size and the need for charging stations. Loeb et al. (2018) investigated the electric SAV operation performance on an agent-based simulator MATSim, focusing on charging strategies and infrastructure decisions. Nevertheless, it is evident that most studies on dynamic SAV operations have concentrated on impact analysis using simulation frameworks with rule-based methods, rarely exploring the fields of operational-level decision-making optimization problems and algorithm design for SAM services. In our study, we consider a real-time SAV dispatching for the SAM service considering ride-pooling in pursuit of profit maximization. The optimization problem addressed in this chapter is abbreviated as RT-SAVD thereafter.

The deployment of SAVs in SAM services enhances operational flexibility, potentially leading to improved system performance. This is achieved through continuous optimization of vehicle dispatching plans with seamless diversion and self-relocation to accommodate all user requests for pickups and drop-offs (Ma et al., 2017). In fact, the proposed RT-SAVD problem can be considered as a specific variant of dynamic dial-a-ride problem (DARP) or shared-taxi problem focusing on route planning. Under these circumstances, the SAV fleet is operated in an online manner to fulfill passenger requests with diverse origins and destinations. The dynamic DARP and shared-taxi problem are dynamic extensions of the static DARP. The static DARP, a variant of the pickup and delivery problem (PDP) with time windows, focuses on optimizing vehicle routing and scheduling to accommodate the transportation needs for individuals, typically patients or the handicapped with specified pickup and drop-off locations, time windows, and maximum trip duration (Cordeau and Laporte, 2007). Unlike the static DARP, where all detailed information regarding requests and vehicles are predetermined, the dynamic version handles requests that arrive unexpectedly and decision-makers can adjust previously established routes in response to newly acquired information. Notably, the dynamic DARP is considered deterministic if decision-makers have complete information about all ongoing and upcoming operations, except

for the unexpected new arrivals (Ho et al., 2018).

Among the studies on dynamic DARP, the decision-making problems typically focus on exploring the optimal accommodation of incoming requests on short notice and various metaheuristics and hybrid algorithms have been proposed (Attanasio et al., 2004; Berbeglia et al., 2010, 2012; Coslovich et al., 2006; Gendreau et al., 2006; Häll et al., 2015; Häme, 2011; Häme and Hakula, 2015; Maalouf et al., 2014; Marković et al., 2015; Souza et al., 2022). For example, Attanasio et al. (2004) introduced a parallel tabu search to efficiently insert newly arrived requests into existing routes and optimize routing solutions continuously. Coslovich et al. (2006) proposed a two-stage insertion algorithm utilizing route perturbances. The first stage, executed offline, involved the route neighborhood updating step to generate a collection of alternative feasible routes within the vicinity of the incumbent route, while the second stage, executed in real-time, provided a quick response to a newly arrived request using a simple insertion method to evaluate all possible insertions into the obtained feasible routes. Häme (2011) proposed an adaptive insertion algorithm with a priori clustering method to find the optimal solution. All possible insertion positions were evaluated for a specific request, and all resulting partial routes were preserved, particularly suitable for the highly restricted problem. Berbeglia et al. (2012) presented a hybrid approach combining the exact constraint programming with a tabu search algorithm to detect feasible insertions of new requests efficiently. Souza et al. (2022) investigated a dynamic DARP under a partially dynamic environment with both static requests known a priori and dynamically arrived ones over time. They developed a two-phase hybrid algorithm with the first phase dedicated to solving a static problem using the variable neighborhood search heuristic and the second phase responsible for dealing with the dynamic requests through a simple insertion heuristic. This method is designed in pursuit of operational cost minimization and customers' convenience maximization.

Further extensions in dynamic DARP have also been explored by considering individual behaviors, stochastic factors, etc. (Sayarshad and Chow, 2015; Sayarshad

and Oliver Gao, 2018; Schilde et al., 2014; Azadeh et al., 2022; Tafreshian et al., 2021; Xiang et al., 2008). For example, Azadeh et al. (2022) proposed a choice-driven dynamic DARP integrating customer behaviors by incorporating assortment optimization with routing decisions. They proposed a pricing strategy while incorporating a choice model to offer personalized options of services (i.e., service type and pickup time) along with their corresponding price levels. This approach also generates possible assortments with associated probabilities for each new request. Schilde et al. (2014) investigated a dynamic and stochastic DARP considering the uncertainty of time-dependent travel speeds. Tafreshian et al. (2021) investigated an on-demand shuttle dispatching problem and developed an efficient data-driven two-phase algorithmic framework considering stochastic demand information.

The shared-taxi problem represents another novel variant of the DARP, which seeks to optimally allocate passengers to taxis and identify corresponding optimal routes in an online taxi-dispatch system (Mourad et al., 2019). Hosni et al. (2014) pioneered to formulate an MIP model for a generalized DARP that considered vehicles at various initial locations with onboard passengers, facilitating application within a dynamic system. The near-optimal solution was obtained using the Lagrangian decomposition approach combined with two heuristic approaches. To address dynamic scenarios, this algorithm was periodically invoked and evaluated against a commercial solver. Given the large-scale nature and inherent dynamics of the shared-taxi problem, relevant studies primarily focused on developing efficient heuristic algorithms to produce high-quality vehicle routing solutions in real-time with good computation speed (Hua et al., 2022; Wang and Yang, 2019). To further explore extensions, Santos and Xavier (2015) investigated a dynamic shared-taxi problem by incorporating the monetary incentive and proposed an improved heuristic with a path-relinking technique to solve the problem. Jung et al. (2016) tackled the dynamic shared-taxi problem by developing an advanced hybrid simulated annealing algorithm, which aimed to maximize occupancy rates and minimize travel time for passengers. Zhan et al. (2021)

examined the overall advantages of dynamic shared-taxi services focusing on the maximization of the served customer number and the minimization of travel costs. A rolling horizon approach was adopted and each static subproblem was solved by an efficient adapted artificial bee colony (ABC) algorithm integrated with a vantage-point tree technique for algorithm acceleration. Zhan et al. (2022) further incorporated electric vehicles (EVs) within shared-taxi systems and addressed two subproblems involving dynamic matching and EV charging respectively. These two subproblems are addressed concurrently, with the algorithm introduced by Zhan et al. (2021) and CPLEX solving them in parallel, utilizing a rolling horizon approach.

Additionally, other studies have tackled even larger situations and proposed efficient approximate methods. Alonso-Mora et al. (2017) proposed a dynamic vehicle dispatching approach to address a relaxed on-demand shared-taxi problem leveraging the concept of shareability networks to pair trips with vehicles. Their approach was evaluated with taxi data from Manhattan, demonstrating its effectiveness for large-scale applications. Later, Simonetto et al. (2019) developed a less computationally demanding algorithm, restricting the matching of only one passenger to a vehicle per optimization epoch and converting the initial problem into a linear assignment problem. Their findings revealed that the myopic optima preserved service quality comparable to the study by Alonso-Mora et al. (2017). Furthermore, Zhou and Roncoli (2022) extended the algorithmic framework from Simonetto et al. (2019) by incorporating congestion prediction.

## **2.2 Passenger-and-Goods Transportation Services**

Over the past decade, the surge in shared mobility has sparked a significant rise in research focused on the development of optimization models and algorithms to support and facilitate the implementation of diverse mobility solutions, including car-sharing, ridesharing, and ride-sourcing/ride-hailing and crowd-shipping services (Cleophas et al., 2019; Furuhata et al., 2013; Mourad et al., 2019). Although fruitful development

has been achieved in shared mobility for passengers, the investigation into ISM (integrative shared mobility) services that combine passenger and parcel transportation is not sufficient (Bruzzone et al., 2021; Mourad et al., 2019; Sampaio et al., 2019). In fact, people and freight transportation have long been studied separately within dedicated networks in an urban environment in the literature, whereas research efforts exploring ISM services have been relatively limited. Research in this field can be generally classified into three groups according to different passenger transportation modes: (i) private cars, (ii) public transport such as bus and urban rail systems, and (iii) on-demand mobility services (OMS) like ride-hailing services and taxis (Mulley et al., 2018).

For private cars, Archetti et al. (2016) pioneered the study of in-store customers using their own vehicles to execute individual deliveries in return for a specified compensation amount. A vehicle routing problem (VRP) was proposed involving regular and occasional drivers in pursuit of total operating cost minimization and was addressed using a multi-start heuristic approach. Later, many variants based on this problem setting have been studied. For example, Macrina et al. (2017) explored the constraints imposed by customers' and occasional drivers' time windows and explored the possibility of multiple deliveries by occasional drivers. Al Hla et al. (2019) integrated regular and occasional drivers' risk-taking behaviors. Triki (2021) investigated the bidding behavior of occasional drivers for assignments of delivery. Other studies considered the stochasticity and dynamics aspects of private car-based transportation for both people and freight (Arslan et al., 2019; Dayarian and Savelsbergh, 2020; Mousavi et al., 2022).

Regarding public transit-based integrated ISM services, many studies investigated PDP variants by using supportive vehicles for connecting with the fixed public transportation routes for the initial and last journey of delivery (Cheng et al., 2023). For instance, Masson et al. (2017) proposed using available space in buses to transport goods from the depot to different bus stations located in the downtown area. The goods

would then be transferred to supportive vehicles and delivered to customers. They introduced a PDP model with transfers and formulated an MIP model with transshipment synchronization constraints between buses and supportive vehicles at bus stops aiming to minimize both the overall traveled distance and the utilized city freighter number. Ghilas et al. (2016a) studied another transit-based integrated ISM setting where parcel requests were either directly served by supportive vehicles or partially transported using scheduled line services. They proposed an MIP formulation for a variant of PDP incorporating time windows and scheduled lines and only optimally solved small instances. In follow-up research, Ghilas et al. (2016b) developed an ALNS heuristic specifically designed to solve large-size instances. Furthermore, Ghilas et al. (2016c) expanded upon the previously proposed deterministic problem into a stochastic version that considers uncertainty in parcel demand. They introduced an SAA method integrated with the ALNS heuristic to address this issue.

For studies on OMS-based ISM services, Li et al. (2014) were pioneers in introducing the share-a-ride problem (SARP), where the taxi fleet was used to provide integrated transport services for passenger and parcel requests simultaneously. Vehicles were allowed to detour under the constraints of the maximum number of stopovers for the pickup and drop-off of parcels during the riding periods of on-board passengers. A mixed-integer linear programming model was formulated to determine the optimal vehicle routing solutions, which were then solved by a commercial solver. Li et al. (2016a) later developed an ALNS algorithm to address large-scale SARP instances. Li et al. (2016b) further determined vehicle routing plans under uncertain delivery locations and travel times. The stochastic variant of the SARP problem was solved by the ALNS coupled with different sampling strategies. Later on, more extensions have been conducted on SARP. For instance, Yu et al. (2018) introduced a variant of SARP that took passenger pooling into consideration. Ren et al. (2021) investigated a multi-depot dynamic SARP for an online ride-hailing system to determine the optimal routing plans for coupled passenger-parcel transportation. Lu et al. (2022) focused on another

SARP variant by employing a diverse fleet consisting of both gasoline-powered vehicles and EVs within the combined people-and-goods transportation systems.

### **2.3 Public Transit Planning and Potential Shared Mobility Integration**

Over the past decades, extensive research in public transit planning has primarily concentrated on two fundamental areas: the optimal bus stop locations and service frequencies. The primary objectives are to improve the service quality and reduce social costs associated with public transit operations (Ibarra-Rojas et al., 2015; Perumal et al., 2022). The strategic placement of bus stops is particularly crucial when establishing and planning new bus routes in urban cities. Numerous previous studies have developed mathematical models to optimize bus stop placements or the continuous spacing and density of stops along specific bus routes or networks. These models incorporate discrete or continuous demand functions along with various objective functions that focus on enhancing service quality for passengers, optimizing operations for service providers, or achieving a balance between these elements (Ceder et al., 2015; Chien and Qin, 2004; Gleason, 1975; Ibeas et al., 2010; Medina et al., 2013; Murray, 2003; Otto and Boysen, 2014; Schittekat et al., 2013; Wirasinghe and Ghoneim, 1981). For example, Wirasinghe and Ghoneim (1981) formulated an analytical model to optimize the bus stop spacing within a grid-based street network, and the model was solved by a trial-and-error approach. Chien and Qin (2004) developed a mathematical model aimed at optimizing both the quantity and placement of bus stops by minimizing the generalized cost for users and the operator. Similarly, Ibeas et al. (2010) explored the optimization of bus stop location by employing a bi-level modeling approach in pursuit of the generalized cost minimization, incorporating operational constraints at the upper level and adjusting demand assignment based on a mode choice assignment approach at the lower level. More recent research has continued to advance this field in diverse public transit systems. For instance, Zhang et al. (2020) investigated the stop spacing optimization problem within an on-demand public transit system. Considering the

specific operational and line configuration constraints of the system, an optimization model was proposed in pursuit of total passenger travel time minimization and was addressed by a dynamic programming approach. Cheng et al. (2019) formulated a bi-level programming model for the stop spacing optimization in a rapid transit system with the objective of minimizing user and operator costs. Chen et al. (2021) formulated an integrated mathematical model for a customized bus service system to determine the optimal service plan in terms of stop deployment, operating routes, and timetable by minimizing the system cost while considering the passengers' inconvenience.

As for another key component, i.e., service frequency, for transit line design, Ceder and Wilson (1986) pioneered to investigate the service frequency optimization of transit lines, focusing on solving a deterministic model with static demand. Subsequent developments have introduced diverse approaches to tackling the service frequency issues. For example, Daganzo (2009) proposed a control strategy aimed at effectively mitigating bus bunching and ensuring consistent headways through strategic frequency adjustments. Sun et al. (2017) proposed a stochastic approach combining queuing theory and Markov Decision Processes to model the unpredictable nature of passenger flows. Ceylan and Ozcan (2018) introduced a model integrating bi-level simulation and optimization techniques with the upper level minimizing the weighted sum of user and operator costs and lower level handling the transit assignment by VISUM transportation planning software. Further studies have enriched the field by incorporating various significant influential factors into service frequency optimization. Hörcher and Tirachini (2021) emphasized the impact of discomfort from in-vehicle crowding on decisions related to service frequency, whereas Cats and Glück (2019) addressed how variations in headways and crowding affect frequency through a simulation-based optimization approach. Gkiotsalitis et al. (2022) tackled the frequency setting problem by formulating a stochastic optimization programming model considering uncertain passenger demands while minimizing both costs associated with bus operations and passenger wait times. Additionally, research dedicated to optimizing the service

frequency, particularly within automated bus systems, has seen significant growth in recent years. For instance, Hatzenbühler et al. (2020) investigated the strategic incorporation of autonomous buses within existing public transportation networks. They developed a comprehensive modeling framework that combines analytical models and agent-based simulations to optimize the service frequencies in pursuit of minimizing the weighted sum of operator and user costs. Sadrani et al. (2022) designed a mathematical modeling framework to determine the optimal frequency and dimensions of buses in an automated transit system by minimizing overall costs while considering travel time uncertainty and in-vehicle crowding externalities.

Although there has been considerable research focused on optimizing bus service frequency and stop locations for public transit design, most of them deal with these two issues separately. The joint optimization of both elements holds more significant potential to enhance service quality and optimize resource use within public transit systems. However, comprehensive strategies for simultaneous optimization remain scarce in the literature. Among these limited studies, Hurdle (1973) pioneered to consider the joint optimization problem of physical setting and vehicle departure headway for feeder transit operation system by minimizing overall system costs with the demand characterized by a continuous function across both time and space. Chang and Schonfeld (1991) extended this study by developing a multiple-period analytical optimization model considering the demand elasticity to determine closed-form analytic solutions for route spacing and headway that adapt to time-varying demand and vehicle operation dynamics. However, these considered problems specifically investigated the optimal placement (spacing) of parallel feeder transit lines perpendicular to a rapid transit line and their service frequencies, rather than focusing on the exact bus stop locations along the line itself. For more relevant studies, dell'Olio et al. (2006) developed a bi-level programming model aimed at jointly optimizing stop locations and service frequencies over specific time intervals with known passenger travel demand. The primary objective at the upper level was total system cost

minimization considering operational restrictions, while the lower-level optimization task focused on modeling user behavior. Following this, Medina et al. (2013) developed a mathematical model aimed at simultaneously optimizing stop density and headways (time intervals between successive bus departures) based on a continuous and multi-period demand approximation by minimizing the total system cost. A two-phase solution method was proposed to first address the headway optimization, followed by the determination of the optimal stop density.

Additionally, the emergence and prevalence of shared mobility facilitates the combination of shared micromobility like bike-sharing into the existing public transit systems. Current studies primarily have explored dimensions of the interaction between bike-sharing and public transit: modal substitute and modal integration (Kim, 2023; Kong et al., 2020). Modal substitution denotes the replacement of trips previously taken by public transit with those made by bike-sharing, while modal integration represents situations where riders can use shared bicycles as an auxiliary mode, complementing pedestrian travel for either the initial or final mile of their journey in conjunction with public transit. (Kong et al., 2020). However, the majority of research investigated the impact analysis of the integrated bike-sharing services into public transit (Campbell and Brakewood, 2017; Montes et al., 2023; Radzimski and Dzięcielski, 2021; Shelat et al., 2018; Yang et al., 2018; Yin et al., 2024). The development of decision-making models and algorithm design for public transit lines that integrate shared micromobility is limited. Among the limited studies, for instance, Liu et al. (2019) investigated the problems of fleet sizing and scheduling for feeder buses that integrate with bike-sharing systems for metro and residential areas connection. A multi-objective model was formulated by minimizing passenger waiting times and maximizing operator profits, incorporating variable demands from bike-sharing usage, and was addressed by a customized hybrid heuristic algorithm. Additionally, Wu et al. (2020) explored a design optimization problem for the transit network supported by shared bikes. Continuous optimization models were proposed to strategically optimize both the placement and

number of the bike docking stations and the general configuration and frequency of transit routes in pursuit of the overall system cost minimization.

## **2.4 Limitations of Existing Studies**

The literature review above highlights three major limitations in current studies:

(1) *Existing studies on SAM services ignore critical issues about passengers' ride-pooling preference and satisfaction.*

The above literature review shows that numerous studies have been conducted to investigate dynamic DARP or shared-taxi problems for passenger-sharing mobility services with different objectives, addressing challenges in efficient request assignment and vehicle dispatch by developing various algorithms. However, the ride-pooling services considered in these studies often involve simultaneously serving two or more potential requests only respecting vehicle capacity constraints. The dynamic decision-making process focuses on finding optimal solutions to accommodate incoming requests without specific consideration of passengers' shared 'strangers' during their shared trips. In other words, the number of strangers sharing with each passenger is assumed to be unlimited in these studies. In reality, however, passengers might have the maximum ride-pooling stranger number preferences. For example, consider a passenger who is willing to share his/her trip with only one stranger. Suppose the concerned passenger has already been (or is being) transported with another passenger in the same vehicle. In that case, the service operator is not expected to continue pooling this passenger with additional passengers in its remaining journey, even if the vehicle has sufficient capacity. Therefore, for the sake of service quality, it is recommended that SAM services impose restrictions on the number of strangers sharing rides with each request. Notably, the maximum number of ride-pooling strangers constraint distinguishes the SAM services from traditional dial-a-ride services.

In fact, the SAM services maintain the pooling feature found in well-known ride-

sourcing services and extend the traditional ride-pooling service design to assemble multiple passengers in a more generalized manner. Specifically, the SAM services consider the limited number of passengers traveling together in a shared trip and allow shared rides to be formed in the middle of the delivery. In addition, given the inherent personal attributes and the characteristics of the ride-pooling trips, passengers with potentially feasible ride-pooling options may not satisfy the arranged shared trip. Their approval for sharing rides with strangers will inevitably affect the vehicle dispatching operations that provide pooled rides service, which is largely ignored in existing studies. To our knowledge, previous research has not ever explored how to determine vehicle dispatching plans in real-time for the SAM services while incorporating ride-pooling stranger number limit and considering passengers' acceptance of sharing trips with others.

(2) *Current literature for ISM services fails to consider the potential influence of economic incentives on passengers' tolerance for excess ride durations.*

The above literature review shows that most studies for passenger-and-goods transportation services have focused on vehicle route optimization at an operational level. At its core, the route optimization problem in OMS-based ISM services represents a PDP variant, which has been thoroughly investigated in the current body of literature (Berbeglia et al., 2007, 2010). For the sake of service quality, those studies often assumed that passengers had a fixed maximum ride duration, to avoid lengthy detours made for delivering other passengers or parcels. In practice, however, passengers' approval of prolonged travel times, caused by 'ride-pooling' among passengers and parcels, may largely depend on whether they are offered compensations (e.g., trip discounts), and their tolerances of ERD (excess ride duration) could be elastic in relation to the amount of compensation, somehow, in a nonlinear manner. For instance, a passenger may be happy with a detour of 10 min if he/she receives compensation of \$5 but may not tolerate another 10 min detour even if the compensation is doubled. In

addition, from the viewpoint of the service provider, the revenue generated from parcel delivery could be partially shared with passengers in the form of compensation to gain operational flexibility and achieve higher profitability while ensuring customer satisfaction. To our knowledge, previous research has not explored how to use economic instruments, such as trip compensation, to proactively stimulate passengers' acceptance of detours in OMS-based ISM services.

*(3) A holistic approach in public transit design integrated with shared micromobility services is needed.*

The existing studies in the realm of public transit planning have predominantly focused on bus operations and passenger demand in an approximate and aggregate way by developing analytical models that often simplify decision-making processes for individuals and overlook the inherent dynamic and uncertain nature of urban public transit environments. Additionally, it can be seen that the potential for bike-sharing to act as a feeder to bus networks simultaneously with walking mode could significantly alter public transit design by increasing access speeds and reducing the need for closely spaced stops. This, in turn, could lead to reduced operational costs and shorter overall trip times, benefiting both transit agencies and users. However, existing literature rarely tackles the joint optimization of bus stop locations and service frequencies with an integrated bike-sharing mobility service. All these considerations suggest a gap in shifting towards sophisticated modeling building and algorithm design that more accurately capture the dynamic and uncertain elements as well as disaggregate behaviors of transit systems, reflecting the real-world complexities of integrating diverse mobility services. By adopting such advanced methodologies, we can derive more robust, adaptable, and effective transit planning solutions in the context of integrated bike-sharing services.

To close these research gaps, three research problems will be investigated in this thesis in detail. Chapter 3 will explore the dynamic vehicle dispatching for SAM

services while considering the maximum ride-pooling stranger number limit and ensuring passengers' satisfaction. Chapter 4 will further deal with the optimal compensation scheme design for ISM services while considering passengers' elastic tolerance for detours under stochastic passenger and parcel transportation demands. Chapter 5 will investigate the public transit line planning with the bike-sharing feeder mode complement considering complexities of practical factors to simultaneously determine the optimal bus stop location and service frequency.



### **Chapter 3 Dynamic Vehicle Dispatching for Shared-and-Autonomous-Mobility Services Considering Ride-Pooling**

This chapter investigates the real-time shared autonomous vehicle dispatching (RT-SAVD) problem for SAM (shared-and-autonomous-mobility) in an online environment, where passenger requests are released dynamically. In particular, we impose constraints on the ride-pooling services by allowing each request only to be transported, not necessarily simultaneously, with a limited number of strangers during its trip. Individual acceptance of the shared trip is also considered based on the satisfaction measurement. The RT-SAVD aims to determine the optimal SAV dispatching plans in real-time in pursuit of profit maximization. A dynamic algorithmic framework based on the rolling horizon approach is proposed, which includes periodical optimization with a fixed time step over the entire planning horizon. In each optimization run, a static SAV dispatching (S-SAVD) subproblem will be solved with respect to a batch of active passenger requests that can be (re)scheduled at the current iteration of the rolling horizon. Each S-SAVD subproblem is formulated as an extension of DARP, taking into account additional constraints on the ride-pooling stranger number and passengers' satisfaction. A customized hybrid algorithm named ARA-LNS is developed to solve the S-SAVD subproblem. It integrates an ARA (adaptive request assignment) method into the traditional LNS (large neighborhood search) heuristic framework, enabling the decomposition of the S-SAVD problem into several single-vehicle problems to obtain the optimal request assignment and corresponding vehicle routing plans efficiently. Extensive numerical experiments are conducted to demonstrate the efficacy of our solution method compared to benchmark approaches. We also investigate the effects of incorporating the ride-pooling option in SAM services and analyze the impact of passenger requests' flexible time windows to gain managerial insights.

This chapter is structured as follows. Section 3.1 offers an in-depth explanation of the assumptions, notation and the RT-SAVD problem. Section 3.2 establishes an algorithmic framework for dynamic vehicle dispatching that utilizes a rolling horizon

approach. The S-SAVD problem is formulated in Section 3.3. Our customized hybrid ARA-LNS solution method is presented in Section 3.4 to address the S-SAVD problems defined in the dynamic algorithmic framework. The efficacy of our proposed solution method and impact analysis are demonstrated in Section 3.5 through extensive numerical experiments. Section 3.6 provides the conclusion for this chapter. Section 3.7 lists notations used throughout this chapter.

### 3.1 Assumptions, Notations and Problem Statement

We investigate SAM services operating a fleet of homogeneous, centrally managed SAVs to offer door-to-door passenger transportation services within an urban area over the operational period  $[0, T]$ . Let  $\mathcal{R}$  and  $\mathcal{V}$  denote sets of passenger requests and SAVs respectively. Passenger requests will dynamically arrive over time and space, and their information can only be known upon announcements through the SAM platform. Given the passenger request information, SAVs can be dispatched for picking up and dropping off requests. We mainly consider ad-hoc orders that are expected to be acknowledged by the SAM service platform within a specified timeframe. This acknowledgment indicates whether the requests will be confirmed to be accommodated by an SAV, although they are not required to be picked up by this confirmation deadline. In particular, we consider the SAM services that allow both solo rides and pooled rides among different passenger requests. To guarantee service quality, the satisfaction levels of pooled passenger requests are explicitly considered, reflecting their willingness to be arranged in specific pooled trips, which will be illustrated in detail in Subsection 3.1.2. To be applicable in an online context, the SAV dispatching plans will be generated and updated dynamically according to the latest known request information and provide services with pooled trip arrangements. Considering the constraints of a finite vehicle fleet, we aim to dispatch SAVs in a real-time fashion to serve the dynamically arriving passenger requests during the planning horizon in pursuit of total profit maximization. In order to thoroughly introduce the RT-SAVD problem, the subsequent three

subsections will provide detailed explanations of the demand characterization, passengers' satisfaction and SAV states during the dispatch procedure, respectively.

### 3.1.1 Demand characterization

Each passenger request  $r \in \mathcal{R}$  is characterized by a tuple  $\{v_r^o, v_r^d, t_r^r, \delta_r, t_r^{lp}, t_r^{ld}, w_r\}$ , specifying the location for pick-up  $v_r^o$ , location for drop-off  $v_r^d$ , request announcement time (earliest pick-up time)  $t_r^r$ , maximum waiting time  $\delta_r$  to receive the response (i.e., maximum confirmation duration of service), latest pick-up time  $t_r^{lp}$  from origin, latest drop-off time  $t_r^{ld}$  at destination, and the passenger number  $w_r$ . It is worth noting that every passenger is assumed to be willing to tolerate a maximum waiting duration  $\delta_r$  after their announcements before receiving confirmation from the SAM service platform about the vehicle arrangement for service. If the request  $r$  has not been arranged to any SAV by the time  $(t_r^r + \delta_r)$ , it will be considered canceled. All passenger requests are open to sharing a trip with others while respecting the constraint of maximum ride-pooling stranger number, denoted by  $Q$ . Note that even if the number of previously pooled strangers for a request does not exceed the pre-defined limit, the request may still decline a pooled trip arrangement due to satisfaction concerns. Let  $G_r$  and  $\hat{G}_r = (1 - \nu) \cdot G_r$  denote the service charge of the request  $r$  in a solo trip and a pooled trip respectively, where  $\nu$  denotes the discount rate applied to the basic service charge to compensate for the requests sharing their trips with others. To maximize the profit, potential passenger requests are allowed to be rejected and are deemed unprofitable.

### 3.1.2 Passengers' satisfaction

As we have mentioned earlier, to incorporate passengers' acceptance of ride-pooling and guarantee the level of service, each (potentially pooled) request is associated with a satisfaction function. This function, which could be nonlinear, characterizes the passengers' satisfaction measured based on their own attributes and experience of the pooled trips. We assume that each request  $r$  is characterized by a value of time (VOT)  $p_r$  and a privacy-sensitivity value  $g_r$  which describes the degree of reluctance of the passenger to share with strangers. Higher privacy sensitivity values suggest a greater discomfort with the presence of strangers in ride-pooling scenarios. According to the empirical research by Lavieri and Bhat (2019), the passenger's acceptance of a shared ride with others will be affected by the increased travel time due to additional pickups and drop-offs, as well as their approval/comfort level with strangers sharing the same travel space. In our study, it can be reasonably assumed that passengers' satisfaction largely depends on the impedances or benefits of ride-pooling in terms of the request's intrinsic attributes such as the VOT and privacy sensitivity and the pooling trip attributes such as the number of ride-pooling strangers, the shared journey duration and the additional travel time resulting from ride-pooling. In addition to the attributes of the concerned request, we introduce a variable  $\bar{q}_r$  to represent the number of strangers encountered by the passenger request  $r$  during the shared ride. Therefore, the satisfaction function for a particular request  $r$ , denoted by  $F_r(\bullet)$ , can be defined as a multivariate function of these factors and expressed as  $F_r(\nu, p_r, g_r, \hat{q}_r, \hat{\zeta}_r, \hat{\xi}_r)$ , where  $p_r$  and  $g_r$  are the attributes of the concerned request, i.e., VOT and privacy-sensitivity,  $\hat{q}_r$  represents the ride-pooling stranger number,  $\hat{\zeta}_r$  corresponds to the ride-pooling duration (with other requests onboard), while  $\hat{\xi}_r$  denotes the extra (additional) travel time. To ensure service quality, a minimum passengers' satisfaction threshold  $\underline{F}$  is introduced. This ensures that the ride-pooling

arrangement considers not only operational constraints like time windows and vehicle capacity but also places a critical emphasis on passenger satisfaction. Incorporating passenger satisfaction into vehicle dispatch can effectively refine ride-pooling configurations, balancing the efficiency and comfort in SAM services.

### 3.1.3 SAV states

Each SAV  $h \in \mathcal{H}$  will be initialized at a specific location  $v_h^0$  in the service area with no onboard passengers at the beginning of the period  $[0, T]$ . During the daily operation of an SAV  $h \in \mathcal{H}$ , it can be dispatched to handle multiple passenger requests simultaneously, depending on its carrying capacity  $W_h$ . Departing from the initial location, an SAV can transport many passenger requests between any designated pick-up and drop-off points of these requests with ride-pooling services. Let  $\tau_{v_r^o, v_r^d}$  and  $\kappa_{v_r^o, v_r^d}$  denote travel time and incurred cost traveling from the location for pick-up to the location for drop-off of request  $r \in \mathcal{R}$ , respectively. Following this notation, the travel duration and travel cost from the initial location of an SAV  $v$ , i.e.,  $v_h^0$ , to the pick-up location of request  $r \in \mathcal{R}$  is denoted as  $\tau_{v_h^0, v_r^o}$  and  $\kappa_{v_h^0, v_r^o}$ , respectively.

In a dynamic environment, an SAV may happen to be in any of the following states: (i) transporting passengers to corresponding drop-off locations while having additional scheduled requests; (ii) transporting passengers to corresponding drop-off locations without further scheduled requests; (iii) relocating (without onboard requests) to serve the next scheduled request en route; (iv) being idle with no onboard requests. Therefore, the current states and positions of both SAVs and passenger requests must be taken into account when making vehicle dispatch decisions in an online context. The SAV information, including their various states and positions, can be updated based on the continuously optimized SAV routing plans throughout the operational period. Figure

3.1 depicts an instance of a specific SAV dispatching plan at a particular time, highlighting the various vehicle states involved. The locations in red represent drop-off points that vehicles are required to serve along the routes shown by solid lines, while the locations in blue denote scheduled pickup or drop-off points the vehicles will serve following the routes indicated in dashed lines.

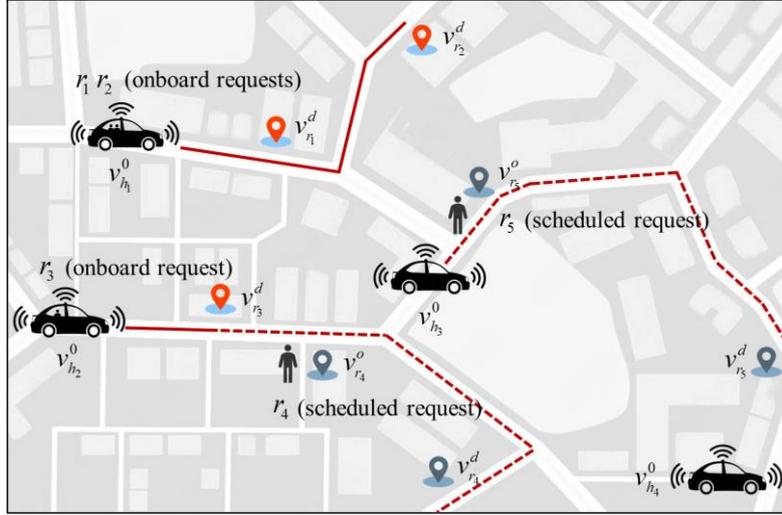


Figure 3.1. SAV dispatching illustration

Given the dynamic passenger request demand, the RT-SAVD problem is to dynamically identify the optimal SAV routing plans by maximizing the profit of the service operator such that: (i) each request is served by one SAV at most; (ii) constraints related to requests' time windows and vehicle capacity are respected; and (iii) ride-pooling stranger number limit and passengers' satisfaction threshold are not violated. The dispatching solutions of SAVs will be constantly adjusted and updated depending on the latest passenger request arrival information.

### 3.2 Dynamic Vehicle Dispatching Framework

To effectively deal with the passenger demand dynamics, we develop a dynamic vehicle dispatching framework based on a closed-loop rolling horizon method to solve the RT-SAVD problem sequentially over the entire planning horizon  $[0, T]$ . Figure 3.2

illustrates the rolling horizon-based dynamic vehicle dispatching algorithmic framework. Instead of solving the RT-SAVD problem across the entire planning horizon, this framework is designed to successively execute the planning at a given sequence of  $K$  time points, denoted by set  $\mathcal{T} = \{t_0, t_1, t_2, \dots, t_{k-1}, t_k, \dots, t_K\}$ , where  $t_0 = 0$  and  $t_K = T$ . These time points are evenly spaced with the increment of a same time step  $\Delta t$ , i.e.,  $t_k = k\Delta t$ , where  $\Delta t = T / K$ . For any iteration  $k$  at the *decision time instant*  $t_k$ , we will use a planning horizon that extends from the present moment, including all available passenger request information. This framework aims to repeatedly slide the horizon forward by the time step  $\Delta t$  and address a series of S-SAVD sub-problems to update the SAV dispatching solutions periodically. In particular, we impose the computation time limit  $\lambda$  for each S-SAVD subproblem solved at each decision time instant  $t_k$  to ensure the timely response to the passenger demand dynamics. In other words, by the time  $t_k + \lambda$ , a subproblem solution will be obtained for the implementation over the period  $[t_k + \lambda, t_{k+1} + \lambda]$ .

For each rolling iteration  $k$ , let  $\mathcal{R}_k \in \mathcal{R}$  represent the *active* request set eligible for consideration in the corresponding S-SAVD subproblem at decision time instant  $t_k$ . Intuitively, these requests involve the ones considered in the previous static subproblem but have not been finalized (and can be reconsidered/rescheduled) by the time point  $t_k + \lambda$  and the newly arrived passenger requests during the period  $[t_{k-1}, t_k)$ . To be specific, we have  $\mathcal{R}_k = (\mathcal{R}_{k-1} \cup \tilde{\mathcal{R}}_{k-1}) \setminus (\bar{\mathcal{R}}_{k-1} \cup \hat{\mathcal{R}}_{k-1})$ , where  $\tilde{\mathcal{R}}_{k-1}$  represents the set of dynamically *arrived* passenger requests during the period  $[t_{k-1}, t_k)$ ;  $\hat{\mathcal{R}}_{k-1}$  indicates the *finished* requests that have already been served by the SAVs during the period  $[t_{k-1} + \lambda, t_k + \lambda)$ ; and  $\bar{\mathcal{R}}_{k-1}$  denotes the *expired* passenger requests including those

with the latest pick-up time the falling in the period  $[t_{k-1} + \lambda, t_k + \lambda)$  and those with waiting time exceeding the maximum limit but without being arranged to any vehicle. In particular, the active requests involved in the first optimization run, i.e.,  $\mathcal{R}_0 = \emptyset$ . In this way, the decisions made in each iteration based on the currently active requests  $\mathcal{R}_k$  may be reconsidered at later decision periods as long as the associated requests have not been finished or expired, allowing the service provider to effectively handle the demand dynamics and achieve high-quality solutions.

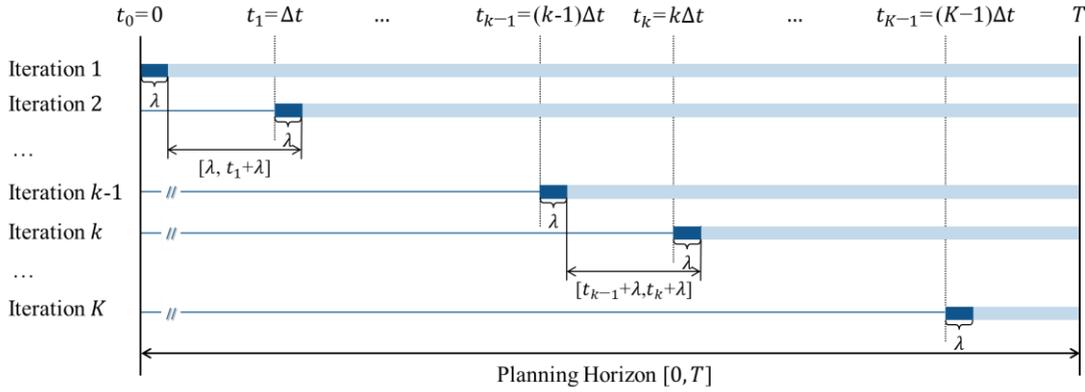


Figure 3.2. Dynamic vehicle dispatching framework

For ease of presentation, we further divide the requests  $\mathcal{R}_k$  in the incumbent S-SAVD problem into three sets: (1) *onboard* request set  $\mathcal{O}_k$ , (2) *confirmed* request set  $\mathcal{C}_k$ , and (3) *unscheduled* request set  $\mathcal{N}_k$ , i.e.,  $\mathcal{R}_k = \mathcal{O}_k \cup \mathcal{C}_k \cup \mathcal{N}_k$ . In particular, the set  $\mathcal{O}_k$  includes the onboard passenger requests that have already been picked up and transported by specific pre-arranged SAVs. The set  $\mathcal{C}_k$  includes the passenger requests that have been designated to be served by specific pre-assigned SAVs in the last rolling horizon computation but have not been boarded yet. They have been confirmed service and cannot change their corresponding arranged SAVs but can still be rescheduled. The set  $\mathcal{N}_k$  includes the passenger requests that are received by the system either during the current or the preceding time interval, whose waiting times have not yet reached

their maximum confirmation duration limit, thus allowing them to be (re)arranged.

In addition to the request information, the information for each SAV  $h \in \mathcal{H}$ , including the current vehicle location  $v_h$ , arrival time at the current location  $\hat{t}_h$ , set of pre-assigned onboard and confirmed requests  $\mathcal{S}_{k,h}$  and corresponding scheduled partial route information  $\Theta_{k,h}$  consisting of sequence of locations for pick-up and drop-off for requests in  $\mathcal{S}_{k,h}$ , is another important input to the incumbent S-SAVD sub-problem. Specifically, according to the SAV dispatching solution in iteration  $(k-1)$ , each vehicle might be in the course of being relocated to serve another passenger request, under the service of a trip with the solo request or pooled requests, or in the idle state upon completion of the optimization run period for the incumbent S-SAVD sub-problem, i.e., time instant  $t_k + \lambda$ . Considering the central-controlled characteristic of SAVs in the SAM services, we assume that a vehicle can be flexibly diverted to serve another passenger request if the vehicle either during the relocation operation to another passenger request or under the service for a request/pooled requests, i.e., with onboard passengers, by the time  $t_k + \lambda$  as long as the number of accumulated pooled strangers for each concerned request does not exceed the pre-defined limit while the vehicle capacity constraint is respected. Nevertheless, the future dispatching plans of each SAV should consider the previous schedules related to the fixed arrangement of its onboard requests and confirmed requests. Therefore, once a solution to the S-SAVD subproblem in iteration  $(k-1)$  is obtained and implemented at the time instant  $t_k + \lambda$ , the current vehicle location  $v_h$ , arrival time at the current location  $\hat{t}_h$ , previously arranged request  $\mathcal{S}_{k,h}$  as well as the corresponding scheduled partial route information  $\Theta_{k,h}$  for the S-SAVD subproblem in iteration  $k$  will be known. In particular, for the first optimization run, the information of each SAV  $h \in \mathcal{H}$  will be initialized as  $v_h = v_h^0$ ,

$$\hat{t}_h = t_0 + \lambda, \mathcal{S}_{0,h} = \emptyset \text{ and } \Theta_{0,h} = \emptyset.$$

Given the information on the involved passenger requests  $\mathcal{R}_k$  and the current state of each SAV  $h \in \mathcal{H}$ , the S-SAVD subproblem will be solved to determine the routing plans of SAVs in  $\mathcal{H}$  to serve the request/request pairs in  $\mathcal{R}_k$  in pursuit of total profit maximization over the current planning period. Each S-SAVD subproblem will be solved to guide the vehicle dispatching solutions update in response to incoming passenger requests during the period  $(t_{k-1}, t_k]$ . More and more passenger requests will be finalized to be served or expire without service as the operational horizon rolls forward. The real-time response to the newly arrived passenger requests can be realized by the short rolling time step  $\Delta t$  in the proposed dynamic vehicle dispatching rolling horizon framework.

### 3.3 Static Problem Formulation

According to the dynamic vehicle dispatching framework, the key is to efficiently address the S-SAVD subproblem. We formulated an MIP model for the S-SAVD subproblem by maximizing the overall system profit while ensuring compliance with the time window, SAV capacity, maximum ride-pooling stranger number and passenger satisfaction constraints. Consider the incumbent S-SAVD problem in iteration  $k$  with input of active passenger request  $\mathcal{R}_k = \mathcal{O}_k \cup \mathcal{C}_k \cup \mathcal{N}_k$  and current information of SAVs  $\mathcal{H}$ . For model building, we define a directed network  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ , in which  $\mathcal{V}$  denotes the node set and  $\mathcal{E}$  denotes the edge set. Considering different types of request sets, set  $\mathcal{V}$  is partitioned as  $\mathcal{V} = \mathcal{V}_1 \cup \mathcal{V}_2 \cup \mathcal{V}_3 \cup \mathcal{V}_4$ .  $\mathcal{V}_1$  is the subset of nodes that correspond to current vehicle locations, where  $v_h \in \mathcal{V}_1$  indicates the node associated with the location of SAV  $h$ . The subset of nodes  $\mathcal{V}_2$  represents drop-off locations for requests in set  $\mathcal{O}_k$ .  $\mathcal{V}_3$  is the subset of nodes denoting pick-up locations

for requests in set  $\mathcal{C}_k$  and  $\mathcal{N}_k$ , while  $\mathcal{V}_4$  is subset of nodes denoting drop-off locations for requests in set  $\mathcal{C}_k$  and  $\mathcal{N}_k$ . Set  $\mathcal{V}$  for all nodes will vary over time and have updated before the incumbent optimization invoking time  $t_k$ . Note that  $v_r^o$  and  $v_r^d$  denote indices corresponding to the pick-up and drop-off locations of passenger request  $r$ . Each edge  $(i, j) \in \mathcal{E}$  is linked to both a specific travel time  $\tau_{ij}$  and the corresponding cost  $\kappa_{ij}$ . For request  $r \in \mathcal{R}_k$ , in addition to the request announcement time  $t_r^a$ , latest time for pick-up  $t_r^{lp}$ , latest time for drop-off  $t_r^{ld}$ , the number of passengers  $w_r$ , revenue of pooling  $\hat{G}_r$  and without pooling  $G_r$ , let  $q_r^*$  denote the recorded number of strangers that have previously shared with request  $r$  known by the time  $t_k + \lambda$ . Specifically,  $q_r^* \geq 0$ ,  $\forall r \in \mathcal{O}_k$  and  $q_r^* = 0$ ,  $\forall r \in \mathcal{C}_k \cup \mathcal{N}_k$ . Additionally, for each request  $r \in \mathcal{O}_k$ , let  $h_r$  denote the SAV that is pre-arranged to the request  $r$ ,  $\zeta_r^*$  denote the recorded (ride-pooling) duration (with other requests onboard) of the implemented (shared) trip up to the time  $t_k + \lambda$  and  $t_r^*$  denote the past pick-up time instant in the (shared) trip. Note that confirmed and unscheduled requests have not been picked up by any vehicle and have no previously implemented pooling arrangement information.

We define binary decision variable  $x_{ij}^{rh}$  to denote whether request  $r$  traverses on edge  $(i, j)$  onboard of vehicle  $h$ ; binary decision variable  $y_{ij}^h$  to denote whether the vehicle  $h$  traverses on edge  $(i, j)$ ; binary decision variable  $d_{rh}$  to denote whether request  $r$  is picked up by vehicle  $h$ ; binary decision variable  $z_r$  to denote whether request  $r$  is pooled with any other request during its trip; non-negative variable  $q_r$  to denote the number of strangers in the shared trip of request  $r$  (obtained in the

incumbent subproblem); non-negative variable  $u_{hi}$  to indicate the time at which vehicle  $h$  arrives at node  $i$ . Thus, the S-SAVD subproblem can be formulated as the following MIP model:

**[S-SAVD]**

$$\max_{\{\mathbf{x}, \mathbf{y}, \mathbf{d}, \mathbf{z}, \mathbf{q}, \mathbf{u}\}} \sum_{h \in \mathcal{H}} \sum_{r \in \mathcal{R}_k} \left( \hat{G}_r z_r + G_r d_{rh} (1 - z_r) \right) - \sum_{h \in \mathcal{H}} \sum_{i \in \mathcal{V}} \sum_{j \in \mathcal{V}} \kappa_{ij} \cdot y_{ij}^h \quad (3.1)$$

subject to

$$\sum_{j \in \mathcal{V}} x_{v_h, j}^{rh} = 1, \quad \forall r \in \mathcal{O}_k \quad (3.2)$$

$$\sum_{j \in \mathcal{V}} x_{v_r^o, j}^{rh} = 1, \quad \forall r \in \mathcal{C}_k \quad (3.3)$$

$$\sum_{j \in \mathcal{V}} x_{j, v_r^d}^{rh} = 1, \quad \forall r \in \mathcal{O}_k \cup \mathcal{C}_k \quad (3.4)$$

$$\sum_{j \in \mathcal{V}} x_{j, v_r^o}^{rh} = \sum_{j \in \mathcal{V}} x_{v_r^o, j}^{rh} - d_{rh}, \quad \forall r \in \mathcal{N}_k, h \in \mathcal{H} \quad (3.5)$$

$$\sum_{j \in \mathcal{V}} x_{j, v_r^d}^{rh} = \sum_{j \in \mathcal{V}} x_{v_r^d, j}^{rh} + d_{rh}, \quad \forall r \in \mathcal{N}_k, h \in \mathcal{H} \quad (3.6)$$

$$\sum_{j \in \mathcal{V}} x_{ji}^{rh} = \sum_{j \in \mathcal{V}} x_{ij}^{rh}, \quad \forall i \in \mathcal{V} \setminus (\mathcal{V}_1 \cup \{v_r^o, v_r^d\}), r \in \mathcal{R}_k, h \in \mathcal{H} \quad (3.7)$$

$$q_r = \sum_{h \in \mathcal{H}} \sum_{r' \in \mathcal{R} \setminus r} \left( \sum_{j \in \mathcal{V}} \sum_{i \in \mathcal{V}} x_{ij}^{rh} x_{ij}^{r'h} - \max \left\{ \sum_{j \in \mathcal{V}} \sum_{i \in \mathcal{V}} x_{ij}^{rh} x_{ij}^{r'h} - 1, 0 \right\} \right) w_{r'} \leq Q - \hat{q}_r, \quad \forall r \in \mathcal{R}_k \quad (3.8)$$

$$F_r \left( v, p_r, g_r, q_r, \sum_{h \in \mathcal{H}} \sum_{r' \in \mathcal{R} \setminus r} \sum_{j \in \mathcal{V}} \sum_{i \in \mathcal{V}} x_{ij}^{rh} x_{ij}^{r'h} \tau_{ij}, \sum_{h \in \mathcal{H}} d_{rh} (u_{hv_r^d} - u_{hv_r^o} - \tau_{v_r^o, v_r^d}) \right) \geq \underline{F} z_r, \quad \forall r \in \mathcal{C}_k \cup \mathcal{N}_k \quad (3.9)$$

$$F_r \left( v, p_r, g_r, q_r + q_r^*, \zeta_r^* + \sum_{h \in \mathcal{H}} \sum_{r' \in \mathcal{R} \setminus r} \sum_{j \in \mathcal{V}} \sum_{i \in \mathcal{V}} x_{ij}^{rh} x_{ij}^{r'h} \tau_{ij}, \sum_{h \in \mathcal{H}} d_{rh} (u_{hv_r^d} - t_r^* - \tau_{v_r^o, v_r^d}) \right) \geq \underline{F} z_r, \quad \forall r \in \mathcal{O}_k \quad (3.10)$$

$$\sum_{r \in \mathcal{R}} x_{ij}^{rh} w_r \leq W_h y_{ij}^h, \quad \forall i, j \in \mathcal{V}, h \in \mathcal{H} \quad (3.11)$$

$$z_r = \min \{1, q_r + q_r^*\}, \quad \forall r \in \mathcal{R}_k \quad (3.12)$$

$$\sum_{h \in \mathcal{H}} d_{rh} \leq 1, \quad \forall r \in \mathcal{N}_k \quad (3.13)$$

$$\sum_{j \in \mathcal{V}} y_{ji}^h \leq 1, \quad \forall i \in \mathcal{V}, h \in \mathcal{H} \quad (3.14)$$

$$\sum_{j \in \mathcal{V}} y_{ij}^h \leq 1, \quad \forall i \in \mathcal{V}, h \in \mathcal{H} \quad (3.15)$$

$$\sum_{j \in \mathcal{V}} y_{ij}^h \leq \sum_{j \in \mathcal{V}} y_{ji}^h, \quad \forall i \in \mathcal{V} \setminus \mathcal{V}_1, h \in \mathcal{H} \quad (3.16)$$

$$\sum_{j \in \mathcal{V}} y_{ji}^h \leq 0, \quad \forall i \in \mathcal{V}_1, h \in \mathcal{H} \quad (3.17)$$

$$\sum_{h \in \mathcal{H}} \sum_{j \in \mathcal{V}} y_{ij}^h \leq 1, \quad \forall i \in \mathcal{V}_1 \quad (3.18)$$

$$u_{hj} - u_{hi} \geq \tau_{ij} - M(1 - y_{ij}^h), \quad \forall i, j \in \mathcal{V}, h \in \mathcal{H} \quad (3.19)$$

$$t_r^r d_{rh} \leq u_{hv_r^o} \leq t_r^{lp} d_{rh}, \quad \forall r \in \mathcal{C}_k \cup \mathcal{N}_k, h \in \mathcal{H} \quad (3.20)$$

$$u_{hv_r^d} \leq t_r^{ld} d_{rh}, \quad \forall r \in \mathcal{R}_k, h \in \mathcal{H} \quad (3.21)$$

$$x_{ij}^{rh} \in \{0, 1\}, y_{ij}^h \in \{0, 1\}, d_{rh} \in \{0, 1\}, z_r \in \{0, 1\}, q_r \geq 0, u_{hi} \geq 0, \quad \forall i, j \in \mathcal{V}, r \in \mathcal{R}_k, h \in \mathcal{H} \quad (3.22)$$

The objective function (3.1) seeks to maximize the overall profit of the SAM services. This profit is determined by deducting the operational expenses from the total revenue earned from serviced requests. Constraint (3.2) ensures that onboard requests in set  $\mathcal{O}_k$  carried by the corresponding pre-arranged SAVs leave their current locations. Constraint (3.3) guarantees that all confirmed requests in  $\mathcal{C}_k$  are picked up from their pick-up locations by their pre-arranged SAVs. Constraint (3.4) ensures that both onboard and confirmed requests in  $\mathcal{O}_k \cup \mathcal{C}_k$  are delivered to their designated drop-off locations by their pre-arranged SAVs. Constraints (3.5) and (3.6) guarantee that the unscheduled requests in  $\mathcal{N}_k$  are collected from origins and finally transported to designated destinations after being assigned to a specific SAV. Constraint (3.7) expresses the flow balance indicating that each request entering the node (other than its pick-up/currently boarded vehicle location and drop-off location) will always leave from the same node. Constraint (3.8) imposes the limit of the maximum ride-pooling

stranger number for each request. Constraints (3.9) and (3.10) ensure the satisfaction threshold of pooled requests for both on-board and unscheduled ones in their shared trips are not exceeded. Constraint (3.11) guarantees that the number of passengers carried by each SAV remains within the vehicle capacity limit. Constraint (3.12) ensures that each request is accommodated in a pooled trip sharing with at least one stranger. Constraint (3.13) means that no more than one SAV can serve any unscheduled request. Constraints (3.14) and (3.15) guarantee that an SAV cannot simultaneously arrive at a node from multiple sources or depart from a single source to multiple nodes. Constraint (3.16) guarantees that each SAV will either traverse the node (i.e.,  $\sum_{j \in \mathcal{V}} y_{ij}^h = \sum_{j \in \mathcal{V}} y_{ji}^h$ ) or end its trip at the drop-off node (i.e.,  $\sum_{j \in \mathcal{V}} y_{ij}^h < \sum_{j \in \mathcal{V}} y_{ji}^h$ ) after starting from the current location. Constraint (3.17) guarantees that each SAV can only leave from its current location without revisiting any of SAVs' current locations. Constraint (3.18) guarantees that only one SAV is permitted to depart from any existing vehicle location within set  $\mathcal{V}_1$ . Constraint (3.19) defines the time instant of each SAV visiting each node, where  $M$  denotes a sufficiently large value. Time window conditions for each request are specified in Constraints (3.20) and (3.21). Constraint (3.22) defines the feasible domain of each decision variable.

### 3.4 Solution Methodology

To efficiently obtain good-quality solutions to the model [S-SAVD] for each static subproblem within a limited amount of time, we develop a customized iterative hybrid algorithm that integrates an ARA scheme into the LNS heuristic framework. The algorithm is referred to as ARA-LNS algorithm. The LNS algorithm, widely utilized in VRP including variants like DARP, iteratively improves a solution by destroying and repairing the current one by exploring a large solution space through removal and insertion operators. Despite its efficacy for vehicle routing-related problems, it is still not very efficient to directly use the LNS for solving the multiple-vehicle dispatching

that requires high computational speed considering the real-time context and large problem scale. To this end, we extend the traditional LNS algorithm by incorporating an ARA scheme to decompose the optimization of the multiple-SAV dispatching problem into several single-SAV dispatching problems. The ARA scheme involves the selection and application of various assignment operators to assign passenger requests to suitable SAVs, thus filtering the requests to be scheduled for each SAV. The LNS aims to effectively solve these single-SAV dispatching problems to identify the optimal routing solution for each SAV. However, if the overall profit for routing solutions fails to improve after a specified number of iterations, the ARA scheme will be invoked to reassign the unscheduled requests to different vehicles. Additionally, selection probabilities of assignment operators are updated adaptively according to past performance of routing solutions. This adaptive assignment mechanism not only guides the algorithm in searching for the optimal solution more efficiently but also enables parallel optimization for single-SAV dispatching problems, thereby accelerating the overall optimization process.

#### **3.4.1 ARA-LNS structure for S-SAVD**

Given the request and SAV information available for the incumbent S-SAVD subproblem, the iterative ARA-LNS algorithm begins by constructing initial routing solutions to accommodate unscheduled requests based on the previously scheduled partial routes (for onboard and confirmed requests) of each SAV. In each iteration, the LNS heuristic is employed to improve the incumbent routing solution of each SAV individually by using removal and insertion operators. If the search process fails to yield any improvement in the overall objective of single SAV dispatching solutions after a given number of iterations, the ARA scheme will be invoked to execute the reassignment process for all removed requests by selecting and applying different assignment operators. Subsequently, each SAV routing solution will be improved based on the newly assigned request set until the assignment scheme is called again. The

newly generated SAV routing solutions will be evaluated to be potentially accepted as the incumbent solution and optimal solution. Furthermore, the performance of these SAV routing solutions will also provide feedback to adaptively adjust the selection probability of assignment operators in the ARA scheme. This iterative procedure will continue until a specific stop criterion is reached.

The pseudocode for the proposed ARA-LNS algorithm is presented in **Algorithm 3.1**. Before initialization, potentially available SAVs for each unscheduled request  $\mathcal{N}_k$  are filtered based on the current vehicle information in set  $\mathcal{H}$ . Supposing that all SAVs can divert to directly serve unscheduled requests. For each unscheduled request, pick-up time window requirements should be verified for all SAVs, disregarding other constraints, to roughly filter out the impossible vehicle arrangements. This process generates the possible request-SAV mapping  $\Omega$  to define the assignment of each request to possible SAVs (see Line 1). Essentially, this mapping context reflects that each unscheduled request may have multiple vehicle choices to be arranged, which will set the foundation for the subsequent request assignment procedures. Then, the algorithm generates the initial solution  $s_0$  based on our proposed assignment operators and greedy insertion heuristics, which is described in detail in Subsection 3.4.2. For ease of presentation, the solution  $s_0$  is defined to be composed of  $s_0^h$ ,  $\forall h \in \mathcal{H}$ , where the superscript  $h$  is added in the notation to denote the single-SAV solution. The incumbent solution  $s^h$  and optimal solution  $s_b^h$  for each SAV are initialized to be the initial solution  $s_0^h$ , and the corresponding solutions for all SAVs, i.e.,  $s$  and  $s_b$ , can be obtained accordingly (see Lines 2–3). We will also initialize the selection probability of the assignment operators denoted by  $\mathcal{A}$ . Meanwhile, both the iteration number counter, denoted by  $N$ , and the counter of consecutive iterations during which the assignment operator fails to yield an improved solution, indicated as  $\mu$ , are both set to 0 (see Line 4).

After the initialization, the algorithm initiates a loop designed to iteratively improve the incumbent solution for each SAV by implementing the ARA-embedded LNS (see Lines 5–31). In the loop, the removal operator will be used to the incumbent solution for each SAV  $h$  to generate a destroyed solution  $s_d^h$  and the removed requests are stored in set  $\mathcal{M}_h$  (see Lines 6–8). When the non-improvement counter  $\mu$  is larger than the maximum number of consecutive attempts  $\mu_{\max}$ , the request reassignment process of the ARA scheme will be triggered (see Lines 9–15), which is detailed in Subsection 3.4.3. This assignment process operates on the mixed removed requests in set  $\mathcal{M}$  using five assignment operators  $\mathcal{A}$  to update the request bank  $\mathcal{M}_h$  for each  $h \in \mathcal{H}$  and the counter  $\mu$  will be reset to 0. (see Lines 10–11). Additionally,  $PrbUpd(\bullet)$  is the subfunction used to update the selection probability of all assignment operators according to their historical performance after the ARA scheme has been called for  $\varphi$  times (see Line 13). Then the insertion operator will be used to select the requests in  $\mathcal{M}_h$  for each SAV  $h$  and insert them into its previously destroyed solution, thus obtaining the new solution  $s'$  composed of  $s^{h'}$ ,  $\forall h \in \mathcal{H}$  (see Lines 16–19). If the current solution dose not result in an improvement during the incumbent iteration, the non-improvement counter  $\mu$  will be incremented by 1 (see Lines 20–22).

---

**Algorithm 3.1. Pseudocode of the ARA-LNS algorithm.**

---

**Input:** Active request set  $\mathcal{R}_k = \mathcal{O}_k \cup \mathcal{C}_k \cup \mathcal{N}_k$ , vehicle set  $\mathcal{H}(v_h, t_h, \mathcal{S}_{k,h}, \Theta_{k,h})$

**Output:** Optimal SAV dispatching solution  $s_b$

---

```
1  Filter possible request-SAV mapping  $\Omega$  ;
2  Initialize  $s_0$  composed of  $s_0^h$ ,  $s^h \leftarrow s_0^h$ ,  $s_b^h \leftarrow s_0^h$ ;
3   $s \leftarrow \{s^h\}_{h \in \mathcal{H}}$ ;  $s_b \leftarrow \{s_b^h\}_{h \in \mathcal{H}}$ ; // initialize the incumbent and best solution
4   $N \leftarrow 0$ ,  $\mu \leftarrow 0$ , initialize selection probability of assignment operators  $\mathcal{A}$ ;
5  While  $\neg (N = N_{\max} \vee \text{elapsed CPU time exceeds } U_{\max})$  do
6      For each  $h \in \mathcal{H}$  do
7           $s_d^h \leftarrow \text{Removal}(s^h)$  and put removed requests into  $\mathcal{M}_h$ ;
8      EndFor
9      If  $\mu > \mu_{\max}$ , then // whether re-assign the request
10         Mix the removed requests  $\mathcal{M} \leftarrow \{\mathcal{M}_h\}_{h \in \mathcal{H}}$ ,  $\mu \leftarrow 0$ ;
11          $\{\mathcal{M}_h\}_{h \in \mathcal{H}} \leftarrow \text{ARA}(\mathcal{A}, \mathcal{M}, \{s_d^h\}_{h \in \mathcal{H}})$ ;
12         If the end of every  $\varphi$  times of assignment, then // assignment segment
13              $\mathcal{A} \leftarrow \text{PrbUpd}(\mathcal{A})$  and initialize scores;
14         EndIf
15     EndIf
16     For each  $h \in \mathcal{H}$  do
17          $s^{h'} \leftarrow \text{Insertion}(s_d^h, \mathcal{M}_h)$ ; // obtain (repaired) new solution;
18     EndFor
19      $s' \leftarrow \{s^{h'}\}_{h \in \mathcal{H}}$ ;
20     If  $s'$  improves upon  $s$ , then
21          $\mu \leftarrow \mu + 1$ ;
22     EndIf
23     If  $\text{AcpIcm}(s', s)$  is true, then
24          $s \leftarrow s'$ ;
25     EndIf
26     If  $\text{AcpOpt}(s', s_b)$  is true, then
27          $s_b \leftarrow s'$ ;
28     EndIf
29      $\mathcal{A} \leftarrow \text{ScrUpd}(\mathcal{A})$ ; // Update scores of the selected assignment operator;
30      $N \leftarrow N + 1$ ;
31 EndWhile
32 Return  $s_b$  that is composed of  $s_b^h$ ,  $\forall h \in \mathcal{H}$ .
```

---

Two evaluation subfunctions  $AcpIcm(\bullet)$  and  $AcpOpt(\bullet)$  are employed to assess whether the new solution should be accepted as the incumbent solution and optimal solution respectively (see Lines 23–28). The subfunctions will return ‘true’ if specific criteria are satisfied. The subfunction  $AcpIcm(\bullet)$  is defined to be the simulated annealing accept criterion, which accepts the newly generated solution  $s^{h'}$  in place of the incumbent one  $s^h$  with a probability  $\exp[-(f_{profit}(s^h) - f_{profit}(s^{h'})) / \ell]$ , where  $f_{profit}(\bullet)$  denotes the objective value obtained from the objective function expressed by Eq. (3.3), and  $\ell > 0$  denotes temperature. The initial temperature is determined following the method introduced by Ropke and Pisinger (2006) and the temperature will be iteratively decreased within the simulated annealing process based on a cooling rate  $0 < \nu < 1$  such that  $\ell \leftarrow \nu\ell$ . The subfunction  $AcpOpt(\bullet)$  is defined to accept a new solution as optimal if it demonstrates superiority over the current best one, i.e.,  $f_{profit}(s^{h'}) > f_{profit}(s_b^h)$ . As for the assignment operator score,  $ScrUpd(\bullet)$  is the subfunction that updates the score associated with the currently chosen assignment operator based on the newly generated solution (see Line 29). The iteration number  $N$  will be increased by one at the end of each loop (see Line 30). The loop terminates once it either reaches maximum iteration count,  $N_{max}$ , or exceeds CPU time threshold,  $U_{max}$  (see Line 5).

In what follows, we will elaborate on the SAV dispatching routing plan generation using removal and insertion operators and ARA scheme design in Subsections 3.4.2 and 3.4.3. respectively.

### 3.4.2 SAV dispatching plan generation

#### Initial solution

We will use our proposed adaptive assignment operators and a simple greedy heuristic to construct an initial solution based on the partial routes of all SAVs. To efficiently generate the initial solution, each assignment operator (which will be elaborated in detail in Subsection 3.4.3) will first be used to allocate unscheduled requests among SAVs. Then, according to the outcomes of these assignments, we will construct the initial solution for each vehicle individually based on its partial route. Since LNS is not highly sensitive to the initial solution, a basic greedy insertion heuristic is employed for the construction of each SAV's initial routing plan. Specifically, assigned unscheduled requests will be organized in ascending order based on latest pick-up time and are inserted in the current constructed route of the corresponding vehicle according to this order. Each request will be inserted into the position that brings the maximum profit improvement under all the constraints stated in Section 3.3. For example, the scheduled partial route solution of an SAV  $h \in \mathcal{H}$  is represented as  $\Theta_{k,h} := (v_{r_1}^d, v_{r_2}^d, v_{r_3}^o, v_{r_3}^d)$ , indicating the order of pick-up and drop-off nodes of pre-assigned onboard requests  $r_1, r_2 \in \mathcal{S}_{k,h}$  and confirmed request  $r_3 \in \mathcal{S}_{k,h}$ . For the assigned unscheduled  $r_4$ , we try to insert the  $v_{r_4}^o$  and  $v_{r_4}^d$  in every possible position that maintains the existing routing order of operations, i.e.,  $(v_{r_4}^o, v_{r_4}^d, v_{r_1}^d, v_{r_2}^d, v_{r_3}^o, v_{r_3}^d)$ ,  $(v_{r_4}^o, v_{r_1}^d, v_{r_4}^d, v_{r_2}^d, v_{r_3}^o, v_{r_3}^d)$ ,  $\dots$ . The profit improvement of each feasible insertion can be computed and the insertion that yields the highest profit improvement will be selected.

#### Removal and insertion procedure

In every search iteration, both removal and insertion operators are employed in the LNS heuristic to generate new routing solutions based on the incumbent solution for

each SAV  $h \in \mathcal{H}$ . The removal operator randomly selects and removes a percentage  $m$  of requests from the current routing solution to enhance search diversity. To guarantee the fulfillment of pre-arranged onboard and confirmed requests, we have  $\mathcal{S}_{k,h} \not\subseteq \mathcal{M}_h$ . In other words, these removed requests, which are grouped into set  $\mathcal{M}_h$ , only comprising unscheduled requests in  $\mathcal{N}_k$  and become candidates to be selected and reinserted into the destroyed solution. We employ a greedy insertion operator to construct a new routing plan for SAV  $h$  based on the destroyed solution. This operator performs iteratively to insert one candidate request from the removed request set  $\mathcal{M}_h$  at a time to repair the destroyed solution. Each request is associated with a *best insertion position* in the current repaired routing solution, in which this insertion in the route brings the maximum profit improvement while ensuring the feasibility of the solution. In each iteration, the request offering the greatest profit improvement is chosen and placed at the best insertion position in the route. This process repeats until all requests are considered. If a request's best insertion position leads to a profit decrease, this request will be removed completely, indicating that the service of this request will be rejected to make the service profitable.

Our proposed insertion heuristic will be implemented by additionally checking the feasibility of the ride-pooling arrangement. For example, we proceed to insert request  $r \in \mathcal{M}_h$  into an already existing partial routing solution  $\Upsilon$ . Intuitively, we will check every potential insertion position for the concerned request's the pick-up and drop-off nodes, progressing sequentially from first to last insertion position. For a specific pick-up node insertion position, we will first examine whether inserting this drop-off node insertion would exceed the ride-pooling stranger number constraint for all potentially influenced requests or vehicle capacity constraints. If either constraint is violated, the checking process for the following drop-off node insertion positions can be stopped, as all the subsequent drop-off node insertion positions will lead to violations. If no violations occur, we will proceed to assess the feasibility of time windows and check

the passengers' satisfaction with the proposed insertion.

Following the above idea, the procedure of inserting the candidate request  $r \in \mathcal{M}_h$  into the partial routing solution  $\Upsilon$  with ride-pooling feasibility checking is outlined in **Algorithm 3.2**. For ease of presentation, let  $I$  denote the last possible insertion position for evaluating possible pick-up node insertions for request  $r$ . Define  $Q_r$  as the set of requests previously shared with request  $r$ . For each combination of the potential insertion positions, i.e.,  $i$  and  $j$ , of the pick-up node and drop-off node, the new solution  $\Upsilon_r^{ij}$  will be generated with corresponding profit improvement  $\Delta f_r^{ij}$  initialized to  $-\infty$  (see Lines 1–6). The potentially affected sequence partial route  $\bar{\Upsilon}_r^{ij}$  will be identified to check the feasibility of capacity and ride-pooling stranger number constraints (see Lines 7–17). Provided that constraints on the time window and passengers' satisfaction are respected, the profit improvement  $\Delta f_r^{ij}$  resulting from this insertion will be updated. The insertion position  $(i^*, j^*)$  that yields the highest profit improvement  $\Delta f_r^{i^* j^*}$  is selected to insert request  $r$  into  $\Upsilon$ , thus generating new partial solution  $\Upsilon_r^{i^* j^*}$  (see Line 24). Let  $TT^{(v_1, \dots, v_n)}$  denote the total travel time when an SAV sequentially arrives nodes  $(v_1, \dots, v_n)$ . Mathematically, request  $r$  will be accommodated in an updated routing solution of vehicle  $h$  only if the time window constraints  $t_h + TT^{(v_h, \dots, v_r^o)} \leq t_r^{lp}$  and  $t_h + TT^{(v_h, \dots, v_r^d)} \leq t_r^{ld}$  are respected. For satisfaction feasibility constraints, it is essential to consider previous pooling information regarding the accumulated pooled duration  $\zeta_r^*$  and pick-up time  $t_r^*$  of the previously unfinished pooled trip in the satisfaction calculation for onboard requests.

---

**Algorithm 3.2. Insertion algorithm with ride-pooling feasibility check.**

---

```
1  For  $i \in \{1, 2, \dots, I\}$  do
2    Insert  $v_r^o$  to  $\Upsilon$  in positions  $i$  and generate  $\Upsilon_r^i$ ;
3    If time window constraint is violated for  $v_r^o$ , then
4      | Break;
5    EndIf
6    For  $j \in \{i+1, i+2, \dots, I+1\}$  do
7      Insert  $v_r^d$  to  $\Upsilon_r^i$  in positions  $j$  and generate  $\Upsilon_r^{ij}$ ;  $\Delta f_r^{ij} \leftarrow -\infty$ ;
8       $\bar{\Upsilon}_r^{ij} \leftarrow (v_r^o, \dots, v_r^d)$ ; // potentially affected node sequence
9      For  $v$  in  $\bar{\Upsilon}_r^{ij}$  do
10     | If capacity constraint at node  $v$  is not respected, then
11     | | Break;
12     | EndIf
13     | Set  $r_v$  as the associated request of node  $v$ ;  $Q_{r_v}^{ij} \leftarrow Q_{r_v}$ ;
14     | Add  $r_v$  in  $Q_{r_v}^{ij}$  if  $r_v$  is not in  $Q_{r_v}^{ij}$ ;
15     | If  $|Q_{r_v}^{ij}| > Q$ , then
16     | | Break;
17     | EndIf
18     | EndFor
19     | If  $\Upsilon_r^{ij}$  is feasible for time window and satisfaction constraints, then
20     | |  $\Delta f_r^{ij} \leftarrow f_{profit}(\Upsilon_r^{ij}) - f_{profit}(\Upsilon)$ 
21     | | EndIf
22     | EndFor
23  EndFor
24   $(i^*, j^*) \leftarrow \arg \max_{i \in \{1, 2, \dots, I\}, j \in \{i+1, i+2, \dots, I+1\}} \{\Delta f_r^{ij}\}$ ; // best insertion position
25  Return  $\Upsilon_r^{i^* j^*}$ ,  $\Delta f_r^{i^* j^*}$ 
```

---

### 3.4.3 Adaptive request assignment scheme

As we have mentioned earlier, the adaptive request assignment scheme will be invoked when no further improvement of the total profit of all SAVs' routing solutions can be achieved for a certain number of iterations. Different assignment operators will be selected according to their respective selection probabilities and then utilized for allocating each unscheduled request to an SAV. Note that the selected SAV for a request must be among the filtered possible SAVs assigned to this request, as indicated in the

mapping context.

### Assignment operators

Five assignment operators will be proposed to determine the suitable assignment of each unscheduled request in the set of mixed removed requests  $\mathcal{M}$  to a specific vehicle. These operators will define different measures for assignment and the request bank of each vehicle, i.e.,  $\mathcal{M}_h$ ,  $\forall h \in \mathcal{H}$ , will be updated after checking all unscheduled requests using the assignment operator.

**A1 (random assignment):** This operator randomly assigns each request to an SAV to improve the diversification of the search.

**A2 (distance assignment):** This operator aims to assign each unscheduled request to the nearest SAV. A distance level is defined to measure the spatial closeness between a request and an SAV by calculating the distance from the current position of SAV to the origin of concerned request. This distance criterion will be used to obtain the distance level of all SAVs for each concerned request. The SAV with the lowest distance level will be chosen to determine the assignment of this request.

**A3 (violation assignment):** This operator aims to assign each unscheduled request based on the overlapping length of pick-up time windows among all the pre-assigned requests to the SAV. A violation level is proposed to measure the average overlap between the concerned request and all other assigned requests to an SAV. A smaller overlapping length of the time windows between requests indicates that there will be more potential to insert this request into the route (possibly with ride-pooling with other requests) without causing time infeasibility and generate better routing solutions for vehicles. We first define the violation level of a request  $r$  to another request  $r'$  regarding overlap of the pick-up time window, which can be computed by

$$\theta_r^v(r') = \frac{\max\{0, \min\{t_r^{lp}, t_{r'}^{lp}\} - \max\{t_r^r, t_{r'}^r\}\}}{t_r^{lp} - t_r^r} \quad (3.23)$$

The violation level of assigning a request  $r$  to the SAV  $h$  can be calculated as

$$\theta_{rh}^v = \frac{\sum_{r' \in \tilde{\mathcal{S}}_h} \theta_r^v(r')}{|\tilde{\mathcal{S}}_h|} \quad (3.24)$$

where  $\tilde{\mathcal{S}}_h$  denotes the currently assigned requests to the SAV  $h$  excluding the onboard requests. Request assignment should respect to the lowest violation level criterion. Note that if there are currently no assigned requests to the vehicle, the violation level is set to 0.

**A4 (matching assignment):** Inspired by the study of Hou et al. (2018), this operator aims to assign an unscheduled request to an SAV by evaluating the matching level among its pre-assigned requests. We first define the two-request matching level measurement as

$$\theta_{rr'}^m = \frac{\min\{l_{v_r^o v_r^d}, l_{v_{r'}^o v_{r'}^d}\}}{T_{\min}(rr')} \quad (3.25)$$

where  $T_{\min}(rr')$  denotes the feasible trip minimum travel distance among all the feasible trips that connecting the nodes for picking up and dropping off associated with two requests under constraints on passengers' time windows and satisfaction (if any). It can be seen that  $\theta_{rr'}^m$  can measure the rerouting change that should be made when inserting the request  $r'$  into the existing route with request  $r$  already arranged. If no feasible trip can be found, the value of the matching level will be set to 0.

The two-request matching measurement can be extended to evaluate the matching level of assigning an unscheduled request  $r$  to an SAV  $h$ , denoted by  $\theta_{rh}^m$ . It is determined by the average matching level with all assigned requests to the concerned vehicle and expressed as follows:

$$\theta_{rh}^m = \frac{\sum_{r' \in \tilde{\mathcal{S}}_h} \theta_{r'r'}^m}{|\tilde{\mathcal{S}}_h|} \quad (3.26)$$

where  $\tilde{\mathcal{S}}_h$  denotes the currently assigned requests for SAV  $h$ . Request assignment should respect to the highest matching level criterion. Note that although this operator only measures the two-request pooling trips for matching level evaluation, it is expected that this matching-level-oriented assignment still finds out requests for each SAV that have the potential to accept more shared requests during their trips and sets the foundation for generating better routing solutions.

**A5 (observation-based assignment):** This operator aims to assign each unscheduled request to an SAV based on historical experience gained from the previous performance of the algorithm. Specifically, for every request assigned to an SAV, the operator keeps track of a limited collection of objective values over the latest several iterations, i.e.,  $\mathcal{F} = \{f_1^h, \dots, f_{|\mathcal{K}|}^h\}$ . The observation level of assigning request  $r$  to an SAV  $h$ , denoted by  $\theta_{rh}^o$ , is determined by the average observed value of the profit set across all  $|\mathcal{K}|$  iterations and is calculated as:

$$\theta_{rh}^o = \frac{\sum_{i=1}^{|\mathcal{K}|} f_i^h}{|\mathcal{K}|} \quad (3.27)$$

where the recorded  $|\mathcal{K}|$  is capped at a maximum number of recorded iterations  $|\mathcal{K}|_{\max}$ . Request assignment should respect the lowest observation level criterion.

### Adaptive strategy design

Drawing inspiration from Pisinger and Ropke (2007) where different operators are employed with use frequencies depending on their historical performance according to roulette wheel selection principle, we introduce a similar adaptive mechanism for

selecting the assignment operator in our proposed request assignment scheme. Specifically, we define a segment for the request assignment process that consists of  $\varphi$  times of assignments and is independent of the overall algorithm iteration number. Each assignment operator is associated with an iteration-specific score and a segment-specific weight. The score is iteratively updated to measure the recent performance of the selected assignment operator. A higher score indicates a successful assignment achieved by the corresponding operator. Upon completion of an assignment segment, the operators' weights will be revised based on the cumulative score within this segment, thus updating the selection probability of a particular assignment operator. Let  $\pi_a$  denote the recorded score of assignment operator  $a \in \mathcal{A}$  at last assignment in a specific segment (at the iteration when the corresponding non-improvement counter  $\mu$  reaches  $\mu_{\max}$ ) and  $\varpi_a$  denote the operator's weight for this segment. The weight of the assignment operator  $a$  for the next segment can be updated according to the following expression:

$$\varpi_a = (1 - \psi)\varpi_a + \psi \frac{\pi_a}{\sum_{a \in \mathcal{A}} \pi_a} \quad (3.28)$$

where  $\psi$  reflects the reaction parameter determining the rate at which the weight adjustment respond to the historical changes in algorithm effectiveness. The assignment operator selection probability is computed by  $\frac{\varpi_a}{\sum_{a \in \mathcal{A}} \varpi_a}$ , derived from the roulette wheel selection method proposed in Prins (2004). Note that all the assignment operators are equally weighted in the first assignment segment.

The chosen assignment operator's score is incremented according to the newly generated solutions in each iteration. At the start of every assignment segment, scores of all operators are initialized to zero. For the selected operator  $a \in \mathcal{A}$ , score  $\pi_a$  is updated varying in each iteration, depending on different increment levels, i.e.,  $\sigma_1$ ,

$\sigma_2$ ,  $\sigma_3$  and  $\sigma_4$ , associated with the newly generated solutions as follows:

- $\pi_a \leftarrow \pi_a + \sigma_1$ : New solution is accepted as new global optimum.
- $\pi_a \leftarrow \pi_a + \sigma_2$ : New solution is accepted and superior to the incumbent one but worse than the optimal one.
- $\pi_a \leftarrow \pi_a + \sigma_3$ : New solution is accepted and inferior to the incumbent one.
- $\pi_a \leftarrow \pi_a + \sigma_4$ : New solution is unacceptable.

### 3.5 Numerical Experiments

This section details the results of a range of numerical experiments for randomly generated instances. First, we will introduce the test instance generation and experimental settings. Second, we will evaluate the efficacy of the proposed solution method through a comparative analysis with benchmark approach. Finally, we will explore potential benefits of the SAM service and analyze effects of passenger request flexibility time on system performance. The solution algorithm is coded with Python on a personal computer with Intel (R) Core (TM) i7, 2.80GHz CPU, 16.0 GB RAM.

#### 3.5.1 Instance generation and parameter settings

Due to the absence of benchmark instances for SAM services in the previous research, instances used for our numerical experiments will be randomly generated. Considering that our proposed problem is categorized as a variant of dynamic DARP, the generation of random instances in this study is inspired by the benchmark data of DARP proposed by Cordeau (2006). Assuming that all the instances are created within a  $10 \text{ km} \times 10 \text{ km}$  square service area. The locations for picking up and dropping off  $|\mathcal{R}|$  passenger requests, i.e.,  $v_r^o$  and  $v_r^d$ ,  $\forall r \in \mathcal{R}$ , and the initial locations of  $|\mathcal{H}|$  SAVs, i.e.,  $v_h^0$ ,  $\forall h \in \mathcal{H}$ , are randomly generated in this square region following the

uniform distribution. Euclidean distance is utilized to determine distance from the start to end location points. It is assumed that all SAVs travel at an average speed  $V = 40$  km/hr, and each standard SAV is designed to comfortably accommodate up to four passengers, i.e.,  $W_h = 4$ . Let  $dis(i, j)$  denote the distance from location  $i$  to  $j$ . Travel time from the pick-up to drop-off location of request  $r$  can be computed by  $\tau_{ij} = dis(i, j) / V$ . Considering per-unit traveling distance operating cost of SAVs at 0.2 \$/km, travel cost from  $i$  to  $j$  can be determined by  $\kappa_{ij} = 0.2 \cdot dis(i, j)$ .

The planning horizon is set at 3 h to mimic typical peak hour period during the operational day, typically from 7:00 a.m. to 10:00 a.m. for morning rush and from 4:00 p.m. to 7:00 p.m. for afternoon rush, and the corresponding generated data can mirror the commuter travel patterns. If time duration is measured in minutes, then the planning horizon will be represented as  $[0, 180]$ . For each passenger request  $r \in \mathcal{R}$ , the earliest pick-up time, i.e.,  $t_r^r$ , is generated randomly, with a 25% chance of falling within the interval  $[0, 60)$ , a 50% probability of being within  $[60, 120]$ , and a 25% likelihood from the interval  $(120, 180]$ . The maximum confirmation duration  $\delta_r$  is set to 10 min. The corresponding latest pick-up time of the request  $r$  is  $t_r^{lp} = t_r^r + \chi(r)$ , where  $\chi(r)$  denotes the *flexibility time* that is drawn from a uniform distribution  $u(5, 30)$ . The latest arrival time is determined by considering latest pick-up time, i.e.,  $t_r^{ld} = t_r^{lp} + \tau_{v_r^o, v_r^d}$ . We consider single-passenger and two-passenger requests in our experiments, indicating that  $w_r$  is randomly set to 1 or 2. The maximum number of ride-pooling strangers is defined to align with the typical scenario of pooling two orders, as commonly observed in current ride-sourcing services, i.e.,  $Q = 2$ . We assume that the SAM service operator will charge each request based on travel distance. Given the unit service charge for the solo-ride trip of 1 \$/km, the service charge of the request in a solo-ride trip is calculated by  $G_r = 1 \cdot dis(v_r^o, v_r^d)$ . The request pooled with others enjoys

a 10% discount rate, i.e.,  $\nu = 0.1$ , and therefore, the service charge of the request in a pooled-ride trip is  $\hat{G}_r = 0.9 \cdot G_r$ . Additionally, the VOT of each request, i.e.,  $p_r$ , will be chosen at random from set  $\{0.1, 0.2, 0.3, \dots, 1\}$ . Similarly, the privacy-sensitivity of each request, e.g.,  $g_r$ , is determined by randomly selecting an integer from set  $\{1, 2, 3, 4, 5\}$ , with each value of sensitivity level having an equal probability of being chosen. Without loss of generality, the multivariate satisfaction function is represented as

$$F_r(\nu, p_r, g_r, \hat{q}_r, \hat{\zeta}_r, \hat{\xi}_r) = 1 - \frac{\nu^2}{5} - \frac{p_r^2}{5} - \frac{g_r / 5 \times \hat{q}_r / Q}{5} - \frac{(\hat{\zeta}_r / \hat{\zeta}_{\max})^2}{5} - \frac{(\hat{\xi}_r / \hat{\xi}_{\max})^2}{5} \quad (3.29)$$

where  $\hat{\zeta}_{\max}$  and  $\hat{\xi}_{\max}$  denote the upper limits for ride-pooling duration and extra travel time of request  $r$  respectively, and both are set to 30 min. The minimum passengers' satisfaction value  $\underline{F}$  is set to 0.6.

Regarding rolling horizon framework, predefined time interval between two consecutive decision time instants and the computational time for each static subproblem, i.e.,  $\Delta t$  and  $\lambda$ , are set to 120s and 15s, respectively. As for the algorithm-related parameters, the percentage of requests to be removed in the neighborhood search is defined as 25%. Parameters in terms of the score increment, denoted as,  $\sigma_1$ ,  $\sigma_2$ ,  $\sigma_3$  and  $\sigma_4$ , are set to 6, 4, 1 and 0.2. The simulated annealing accept criterion parameter  $\nu$  is set to 0.9954. The segment size and assignment frequency parameters of the ARA method, denoted as,  $\varphi$  and  $\mu_{\max}$ , are assigned values of 5 and 5, respectively. The parameter  $|\mathcal{K}|_{\max}$  in assignment operator A5 is configured to 10. The stopping criteria for ARA-LNS algorithm are defined as  $N_{\max} = 300$  and  $U_{\max} = 15$  s. Unless stated otherwise, the parameters mentioned above will remain consistent across all numerical experiments.

### 3.5.2 Evaluation against the benchmark approach

To demonstrate the overall performance of our proposed algorithm regarding solution quality, we conduct a comparison of its results against those achieved by a benchmark method that follows the same rolling horizon framework proposed in Section 3.2 but employs a different algorithm to solve the static subproblem for each rolling horizon computation. Specifically, it constructs the initial solution for all vehicles based on their partial solutions by employing the insertion heuristic proposed in the study of Potvin and Rousseau (1993). Then, the LNS algorithm, utilizing the same removal and insertion operators as our proposed method, is directly implemented on all vehicle routes to explore improved routing solutions, disregarding the adaptive assignment procedures. Since the fleet size and the number of passenger request arrivals are expected to influence the computational efficiency of the solution method, we consider a total of 15 scenarios with different numbers of SAVs and numbers of passenger request combinations, i.e.,  $|\mathcal{H}| \in \{15, 30, 45\}$  and  $|\mathcal{R}| \in \{120, 150, 180, 210, 240\}$ . Given the specific SAV fleet size and request number combination, 5 test instances are randomly generated with the aforementioned parameters and average results are presented. As such, the proposed solution method will be evaluated using a total of 75 instances.

Table 3.1 presents a comparison of the performance between the proposed solution method and the benchmark method for various instances, differentiated by varying number of SAV (#SAV) and number of requests (#Request) combinations. For both approaches, we report the system profitability (Obj) and the number of served requests (#SR). We also provide the differences, i.e., the absolute gap (AbsGap) and the relative gap in percentage (RelGap), in terms of the objective value (Diff\_Obj) and served request number (Diff\_#SR) to have a more intuitive comparison. As we can observe, our proposed method constantly demonstrates superior performance compared to the benchmark approach by serving a greater number of requests and achieving better

solutions for all instances. Notably, differences in the served request number and the objective value reach as high as 9.2 and 52.0, respectively. This indicates that the proposed method with the well-designed adaptive assignment procedure can bring about apparent advantages over the benchmark approach. Furthermore, the gaps in objective value and the number of requests served between the two methods for instances with 15 vehicles are 1.8 and 8.6 on average, while those gaps are 5.2 and 38.2 on average for instances with 45 vehicles. This observation also works when it comes to their relative gaps. That is to say, the performance improvement between the proposed solution method and the benchmark method becomes more apparent, on average, under the scenarios involving a larger fleet size. This suggests that our proposed request assignment strategy integrated into the search process can notably achieve improvements in solution quality, especially for instances involving a greater number of vehicles. The primary reason is that the adaptive pre-assignment procedure effectively derives more feasible and potentially profitable request-SAV assignments for each vehicle, thus increasing the likelihood of generating higher-quality routing solutions. These findings collectively highlight the better performance of our proposed solution method compared to the benchmark method in solving the RT-SAVD problem.

Table 3.1. Comparison of the proposed solution method with the benchmark method

#SAV	#Request	Benchmark method		Our method		Comparison			
		Obj	#SR	Obj	#SR	Diff_Obj		Diff_#SR	
						AbsGap	RelGap	AbsGap	RelGap
15	120	351.5	98.4	361.9	99.6	10.4	3.0%	1.2	1.2%
15	150	420.7	121.8	428.7	122.6	8.0	1.9%	0.8	0.7%
15	180	453.2	134.8	461.4	136.0	8.2	1.8%	1.2	0.9%
15	210	504.6	143.2	506.5	146.0	1.9	0.4%	2.8	2.0%
15	240	525.6	158.4	540.2	161.4	14.6	2.8%	3.0	1.9%
30	120	425.5	116.3	437.9	119.0	12.4	2.9%	2.7	2.3%
30	150	521.1	142.5	541.9	146.4	20.8	4.0%	3.9	2.7%
30	180	591.0	168.0	610.2	170.4	19.2	3.2%	2.4	1.4%
30	210	659.4	186.2	693.5	191.0	34.1	5.2%	4.8	2.6%
30	240	718.4	207.2	749.6	212.2	31.2	4.3%	5.0	2.4%
45	120	445.3	119.4	457.7	120.0	12.4	2.8%	0.6	0.5%
45	150	525.7	147.2	552.9	149.8	27.2	5.2%	2.6	1.8%
45	180	599.7	172.8	647.1	178.6	47.4	7.9%	5.8	3.4%
45	210	702.5	198.4	752.6	207.6	50.1	7.1%	9.2	4.6%
45	240	780.1	220.8	834.1	228.6	54.0	6.9%	7.8	3.5%

### 3.5.3 Impact analysis

In this subsection, the benefit of the SAM service will first be evaluated by comparing it with the services without the ride-pooling option, referred to as SAMw/oP. Then, how the passengers' flexibility time affects the performance of the SAM service will be investigated.

#### Impact of SAM service model

We will assess the random instances discussed in Subsection 3.5.2 for the SAM and SAMw/oP services to evaluate the benefit of the ride-pooling option. Under each combination of SAV number and request number, average results for 5 randomly generated instances will be reported. To be specific, in addition to the profit and served request number, we will also evaluate the average waiting time (AvgWat) and served request ratio (SerRat). Differences in average waiting time, profit and served request

number are also reported in terms of both absolute gap and relative gaps in percentage. Table 3.2 presents the comparative results between these two services.

As expected, the ride-pooling option in an SAM service can significantly improve the total profit and served request ratio for all instances. For some instances, e.g., #SAV=30 and #Request=240, the total profit improvement is more than 60 and the average relative increment gap reaches up to 10%. Furthermore, the introduction of ride-pooling features does not affect the average passenger waiting time, with a relative reduction gap approximating 7% on average for all instances. This results from the ride-pooling feature, which facilitates the consolidation of multiple passengers to share the journey, effectively improving the utilization rate of vehicles and reducing each passenger's waiting time. In contrast, in non-pooling services, each request needs to wait for an SAV to pick them up, leading to a longer waiting time.

For a given SAV fleet size, the profit increment shows an increasing trend as the request number rises. For example, with 15 SAVs in operation, the profit gaps between the two types of services are only 24.4 and 27.8 for instances with 120 and 150 requests respectively, while the gap becomes 64.5 when the number of requests reaches 240. Additionally, larger instances generally exhibit a greater relative profit gap compared to smaller ones, which can be explained by the higher revenue generated from serving more requests. These findings suggest that SAM services tend to be more beneficial for larger instances. Another interesting finding is that the profit increment in SAM services diminishes as the fleet size grows. This observation is also evident in the relative gap of profit for most instances. It suggests that even without ride-pooling, requests can still have more opportunities to be feasibly accommodated in an SAV's route under a larger fleet size, thus making SAMw/oP services achieve more comparable results to SAM services. This aligns with the decreased difference in the served request number between the two service types. Hence, it is advisable for SAM service providers to carefully consider the SAV resources distribution to maximize ride-pooling benefits and enhance profitability.

Table 3.2. Comparison of the SAMw/oP and SAM service

#SAV	#Request	SAMw/oP				SAM				Comparison					
		Profit (\$)	AvgWat (min)	#SR	SerRat	Profit (\$)	AvgWat (min)	#SR	SerRat	Diff		Comparison			
										AbsGap	RelGap	AbsGap	RelGap	AbsGap	RelGap
15	120	337.5	11.90	96.0	80.0%	361.9	11.15	99.6	83.0%	24.4	7.2%	-0.75	-6.3%	3.6	3.7%
15	150	400.9	13.03	113.8	75.9%	428.7	11.89	122.6	81.7%	27.8	6.9%	-1.14	-8.7%	8.8	7.7%
15	180	422.5	13.07	125.8	69.9%	461.4	12.21	136.0	75.6%	38.9	9.2%	-0.86	-6.5%	10.2	8.1%
15	210	451.6	13.24	134.2	63.9%	506.5	12.42	146.0	69.5%	54.9	12.1%	-0.82	-6.2%	11.8	8.8%
15	240	475.7	13.61	148.2	61.8%	540.2	12.84	161.4	67.3%	64.5	13.5%	-0.77	-5.6%	13.2	8.9%
30	120	416.4	10.72	116.4	97.0%	437.9	10.10	119.0	99.2%	21.5	5.2%	-0.62	-5.8%	2.6	2.2%
30	150	514.6	11.16	141.8	94.5%	541.9	10.48	146.4	97.6%	27.3	5.3%	-0.68	-6.1%	4.6	3.2%
30	180	582.4	11.57	164.6	91.4%	610.2	10.44	170.4	94.7%	27.9	4.8%	-1.13	-9.8%	5.8	3.5%
30	210	638.7	11.96	182.6	87.0%	693.5	11.16	191.0	91.0%	54.8	8.6%	-0.79	-6.6%	8.4	4.6%
30	240	681.3	12.35	202.0	84.2%	749.6	11.38	212.2	88.4%	68.3	10.0%	-0.97	-7.8%	10.2	5.0%
45	120	443.0	10.41	119.8	99.8%	457.7	9.90	120	100.0%	14.7	3.3%	-0.51	-4.9%	0.2	0.2%
45	150	530.9	10.75	147.4	98.3%	552.9	10.19	149.8	99.9%	22.0	4.1%	-0.57	-5.3%	2.4	1.6%
45	180	614.7	11.35	174.0	96.7%	647.1	10.30	178.6	99.2%	32.4	5.3%	-1.05	-9.2%	4.6	2.6%
45	210	708.0	11.73	202.6	96.5%	752.6	10.93	207.6	98.9%	44.6	6.3%	-0.80	-6.8%	5.0	2.5%
45	240	785.0	11.74	222.4	92.7%	834.1	11.10	228.6	95.3%	49.1	6.3%	-0.65	-5.5%	6.2	2.8%

### Impact of flexibility time

To explore how the flexibility time of passenger requests influences the SAM service system performance, we will set up scenarios with five different flexibility time ranges. Specifically, we will vary the uniform distribution of the flexibility time parameter of passenger requests while keeping other parameters consistent with those given in Subsection 3.5.1. The average value of the flexibility time distribution varies incrementally, with each step adding 5 minutes, resulting in five distinct settings for the distribution:  $\{u(5,10), u(10,15), u(15,20), u(20,25), u(25,30)\}$ . The flexibility time parameter, i.e.,  $\chi(r)$ , will be drawn from these different uniform distributions. A distribution setting characterized by a higher average value indicates that, on average, passengers have greater temporal flexibility. This implies that passengers are more tolerant of longer detours during their journeys.

In order to have a comprehensive evaluation, we will examine the system performance of the SAM services for various scenarios involving different combinations of the SAV number (selected from  $\{15, 30, 45\}$ ) and the request number (selected from  $\{120, 180, 240\}$ ) under different flexibility time distribution settings. For a specific distribution setting, the average results for 5 randomly generated instances will be presented. Except for aforementioned performance metrics, we additionally present the pooled request number (#PR) and the pooled requests ratio (PolRat). Table 3.3 shows the overall results of the effect of the flexibility time on system performance. To facilitate comparison, we also visualize variations of different performance metrics based on a specific SAV number, i.e., #SAV=30, and all the request number combinations in Figure 3.3–Figure 3.8.

We can observe from Figure 3.3 that the profit steadily increases as the flexibility time increases for all tested scenarios. This finding aligns with the observations in the observed growth in both served request numbers and pooled request numbers under the growing flexibility time by looking further into Figure 3.5 and Figure 3.7. The results

are within our expectation: passengers' willingness to tolerate delays in their arrival times at their destinations allows the SAM services to serve more passengers before their latest departure time and have more flexibility to further create more cost-saving pooling arrangements, thus finding more profitable SAV routing plans and achieving profit improvement for the service provider. Nevertheless, the flexibility time increase will inevitably result in a prolonged average waiting time. Figure 3.4 illustrates that as flexibility time increases, the average waiting time steadily increases, showing a roughly linear trend. For example, the average waiting time increases from 4.17 min to 16.24 min for the scenarios of 120 requests when the flexibility time distribution increases from  $u(5,10)$  to  $u(25,30)$ . Interestingly, according to Table 3.3, the average waiting time increase rate generally follows a trend of first ascending and then declining, indicating the reduced adverse impact of the prolonged flexibility time on the average waiting time.

Particularly, it is encouraging to observe that the average increment of profit for every additional 5 min of the average value of the flexibility time distribution is relatively larger when the demand increases. For instance, in scenarios with 30 SAVs, the corresponding profit will averagely increase by 15.5 and 25.6 as the flexibility time distribution changes 5 min incremental step for scenarios with 120 to 240 requests. The results imply that the overall profit for the SAM services will be further boosted if a larger number of requests are happy to accept delayed arrival times. In comparison, the average increment of profit becomes lower under the increased number of SAVs. For example, for the scenarios of 240 requests, the profit will averagely increase by 39 and 17.4 with an additional 5 min of flexibility time for fleets of 15 and 45 SAVs respectively. This suggests a diminishing beneficial effect of the increasing flexibility time on the total profit when the SAV fleet size expands.

Figure 3.5 and Figure 3.6 show the variations in the served request number and the served request ratio with increasing passenger flexibility time, respectively. Notably, the served request number increases rapidly from 182.8 to 229.8 for the scenarios with

the number of requests being 240, and the corresponding served request ratio increases from 76.2% to 95.8%. If it is still the case with further increases in flexibility time, the served request number is expected to increase more than 95.8%, nearly up to 100.0% in the scenario of the request number and flexibility time distribution setting being 180 and  $u(25, 30)$ , respectively. Additionally, according to Figure 3.5 and Figure 3.6, when the market adoption of SAM services remains comparatively low level such as #Request=120, the increment is marginal (around 1.5 on average). More specifically, for the scenarios of 30 SAVs, the increase rate of the served request number for 120 requests keeps decreasing with the increase of passengers' flexibility time, while the increase rate for the request number of 180 or 240 first rises and then drops. This may be attributed to a large SAV fleet in relation to demand, i.e., 30 SAVs for 120 requests, enabling the SAM service to meet the majority of the demand with a relatively small flexibility time. Therefore, even if the time window becomes more flexible, the service capacity of the fleet is relatively stable and no significant growth for the served requests will be produced. Comparatively, when the ratio of the SAV fleet to demand is low, fully meeting all demand within the tight time window becomes impossible. As the flexibility time increases, the fleet gains more opportunities to match passenger requests, thereby increasing the number of passengers served. However, with continued increases in flexibility time, once the service capacity of the SAV fleet saturates, further changes in flexibility time no longer significantly affect the number of served requests, resulting in a diminished growth rate.

Regarding the variations in the number of pooled requests and pooled request ratio, Figure 3.7 and Figure 3.8 show that both variations show a similar upward trend to the variations of served request number and served request number ratio. Notably, the number of pooled requests averagely reaches as high as 69.2 with the corresponding pooled ratio of 30.1% when the request number is 240. Particularly, an average increase of 5 min in flexibility time for passengers is expected to significantly boost the pooled request number and this contribution is even more significant under large-scale

instances. For example, when the flexibility time distribution changes from  $u(5,10)$  to  $u(25,30)$ , the number of pooled requests will averagely increase by 4.6 for the scenario with 120 requests, while that average increments become 11.0 and 14.6 for the scenarios with additional 30 and 60 requests, i.e., 180 and 240, respectively. We also notice that the increase rates of both pooled request number and pooled request ratio become smaller when the flexibility time distribution exceeds  $u(15,20)$ . To be specific, the pooled request number and its ratio increase sharply when the flexibility time distribution changes from  $u(5,10)$  to  $u(15,20)$ , but the increment becomes smaller when the flexibility time distribution increases further. This finding is consistent with the changes in the increase rate observed in the number of requests served and their corresponding ratios. It also indicates that a prolonged flexibility time can positively affect the pooling probability among potential requests, but the improvement in pooling would become limited when the flexibility time exceeds approximately 20 min.

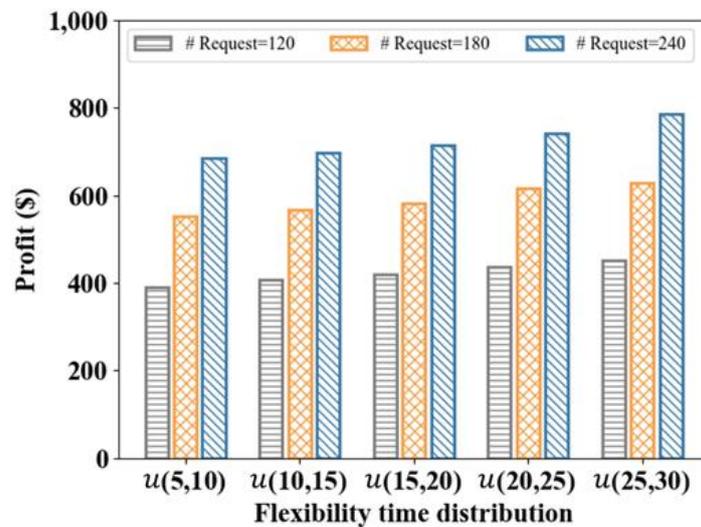


Figure 3.3. Impact of the flexibility time on the total profit

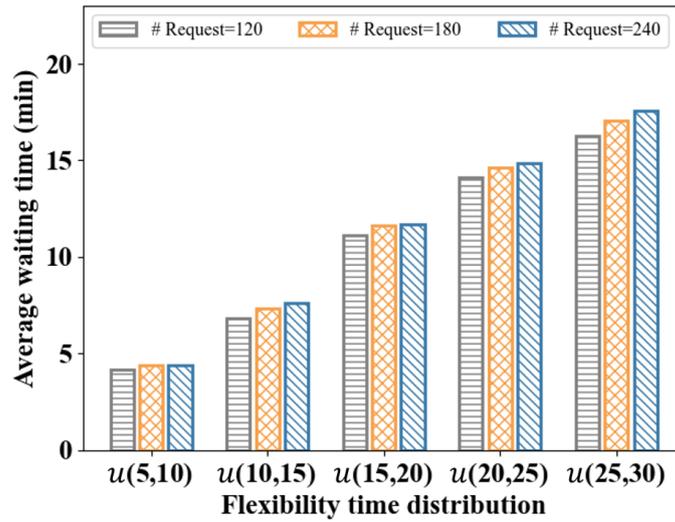


Figure 3.4. Impact of the flexibility time on the average waiting time

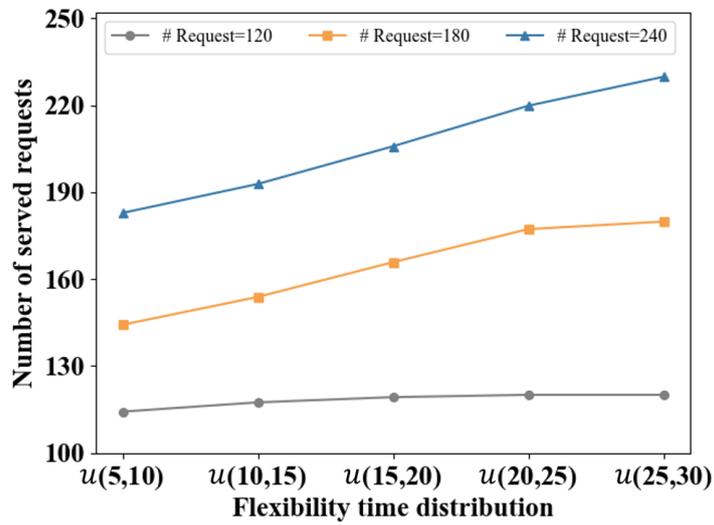


Figure 3.5. Impact of the flexibility time on the number of served requests

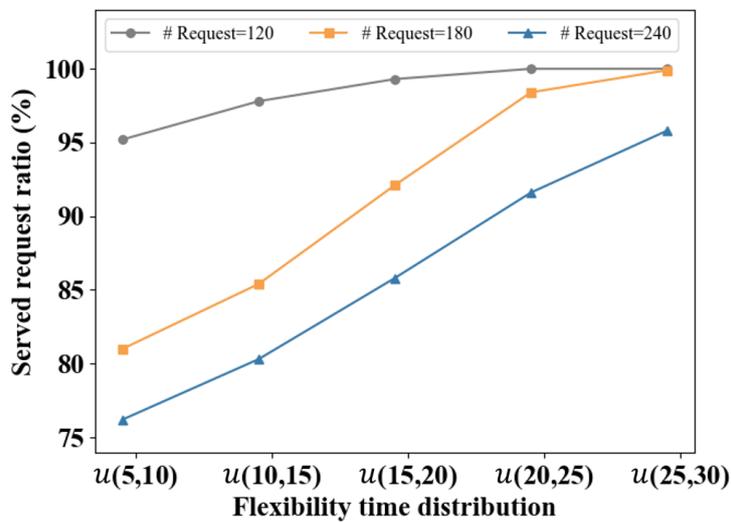


Figure 3.6. Impact of the flexibility time on the served request ratio

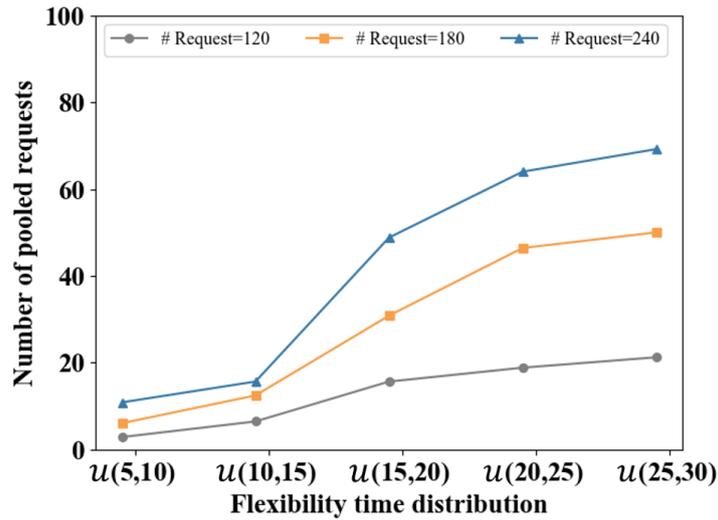


Figure 3.7. Impact of the flexibility time on the number of pooled requests

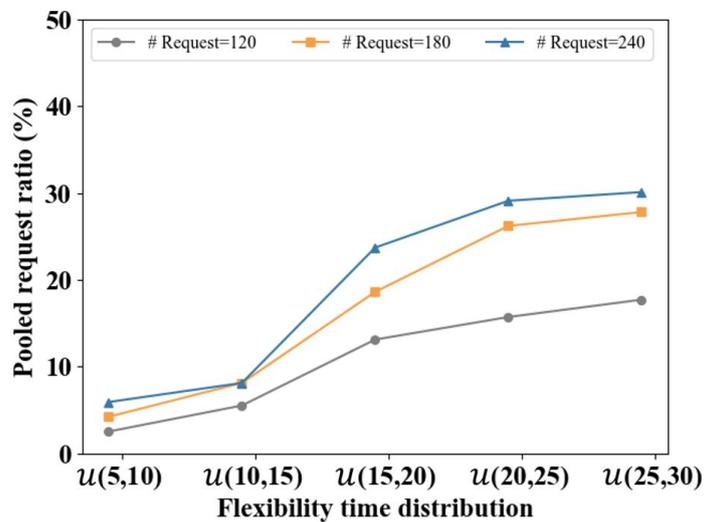


Figure 3.8. Impact of the flexibility time on the pooled request ratio

Table 3.3. Effect of flexibility time on system performance with different SAV and request number combinations

Distribution	#SAV	#Request	Profit (\$)	AvgWat (min)	#SR	SerRat	#PR	PolRat
$u(5,10)$	15	120	345.2	4.74	88.2	73.5%	4.8	5.4%
$u(10,15)$	15	120	365.4	7.58	93.6	78.0%	5.6	6.0%
$u(15,20)$	15	120	380.3	11.82	104.8	87.3%	18.0	17.2%
$u(20,25)$	15	120	396.1	15.19	110.8	92.3%	23.6	21.3%
$u(25,30)$	15	120	419.1	17.62	116.6	97.2%	27.2	23.3%
$u(5,10)$	15	180	368.3	5.00	101.4	56.3%	7.4	7.3%
$u(10,15)$	15	180	410.1	8.00	112.6	62.6%	13.6	12.1%

$u(15, 20)$	15	180	447.9	12.24	129.6	72.0%	35.6	27.5%
$u(20, 25)$	15	180	487.7	15.38	144.2	80.1%	44.8	31.1%
$u(25, 30)$	15	180	505.4	18.24	151.4	84.1%	52.8	34.9%
$u(5, 10)$	15	240	446.6	5.07	116.0	48.3%	8.4	7.2%
$u(10, 15)$	15	240	486.6	8.16	132.2	55.1%	20.8	15.7%
$u(15, 20)$	15	240	545.8	12.46	150.4	62.7%	50.8	33.8%
$u(20, 25)$	15	240	575.8	15.69	175.0	72.9%	64.0	36.6%
$u(25, 30)$	15	240	605.9	18.74	187.2	78.0%	71.2	38.0%
$u(5, 10)$	30	120	389.0	4.17	114.2	95.2%	2.8	2.5%
$u(10, 15)$	30	120	408.0	6.84	117.4	97.8%	6.4	5.5%
$u(15, 20)$	30	120	420.2	11.09	119.2	99.3%	15.6	13.1%
$u(20, 25)$	30	120	436.8	14.13	120.0	100.0%	18.8	15.7%
$u(25, 30)$	30	120	450.9	16.24	120.0	100.0%	21.2	17.7%
$u(5, 10)$	30	180	552.5	4.36	144.2	80.1%	6.0	4.2%
$u(10, 15)$	30	180	566.1	7.30	153.8	85.4%	12.4	8.1%
$u(15, 20)$	30	180	582.4	11.62	165.8	92.1%	30.8	18.6%
$u(20, 25)$	30	180	616.2	14.64	177.2	98.4%	46.4	26.2%
$u(25, 30)$	30	180	628.5	17.04	179.8	99.9%	50.0	27.8%
$u(5, 10)$	30	240	684.3	4.40	182.8	76.2%	10.8	5.9%
$u(10, 15)$	30	240	696.7	7.61	192.8	80.3%	15.6	8.1%
$u(15, 20)$	30	240	713.6	11.71	205.8	85.8%	48.8	23.7%
$u(20, 25)$	30	240	740.5	14.85	219.8	91.6%	64.0	29.1%
$u(25, 30)$	30	240	786.5	17.58	229.8	95.8%	69.2	30.1%
$u(5, 10)$	45	120	441.6	4.11	115.2	96.0%	2.4	2.1%
$u(10, 15)$	45	120	451.0	6.83	118.4	98.7%	5.2	4.4%
$u(15, 20)$	45	120	465.2	10.96	120.0	100.0%	13.6	11.3%
$u(20, 25)$	45	120	478.2	14.58	120.0	100.0%	18.2	15.2%
$u(25, 30)$	45	120	491.5	15.68	120.0	100.0%	19.6	16.3%
$u(5, 10)$	45	180	579.6	4.11	170.0	94.4%	4.8	2.8%
$u(10, 15)$	45	180	613.5	7.15	174.8	97.1%	9.6	5.5%
$u(15, 20)$	45	180	617.5	11.76	179.6	99.8%	20.0	11.1%
$u(20, 25)$	45	180	631.8	14.78	180.0	100.0%	36.8	20.4%
$u(25, 30)$	45	180	634.4	16.69	180.0	100.0%	37.2	20.7%
$u(5, 10)$	45	240	773.6	4.28	210.8	87.8%	6.0	2.8%
$u(10, 15)$	45	240	794.1	7.24	217.2	90.5%	14.9	6.9%
$u(15, 20)$	45	240	807.5	11.75	233.0	97.1%	32.8	14.1%
$u(20, 25)$	45	240	832.3	14.94	239.6	99.8%	60.4	25.2%
$u(25, 30)$	45	240	843.1	16.83	240.0	100.0%	65.6	27.3%

### 3.6 Concluding Remarks

This chapter investigates the RT-SAVD problem for SAM services in a dynamic environment considering maximum ride-pooling stranger number and passengers' satisfaction constraints. We develop a dynamic vehicle dispatching framework based on the rolling horizon approach, in which a series of S-SAVD subproblems are solved at a given set of consecutive time points using all known information up to the decision time point to periodically update the SAV dispatching solutions. The final demand serving and pooling arrangement decisions of each passenger request will not be made until necessitated by the deadline of the request. For each static subproblem, we propose an MIP model to optimize the vehicle dispatching plans by maximizing the total profit while respecting the maximum ride-pooling stranger number and passenger satisfaction constraints. The current states of SAVs, onboard requests and confirmed requests are considered in the formulation.

To efficiently solve the proposed S-SAVD model, we develop a customized iterative hybrid algorithm, named ARA-LNS, that integrates an ARA scheme into the LNS heuristic framework, allowing us to transform the original problem into several single-vehicle problems. The LNS is used to iteratively determine optimal routing plans for each SAV. If there is no improvement in the total profit over certain iterations, the ARA scheme will be invoked and reassign the requests to different SAVs by adaptively selecting the assignment operators. We have performed comprehensive numerical experiments to evaluate the efficacy of our proposed solution method. Our proposed ARA-LNS algorithm demonstrates superior performance compared to the benchmark approach. Moreover, we examine the impact of the SAM services by comparing it with the business model without ride-pooling and results indicate that the SAM services significantly improve the total profit. We also analyze how variations in the flexibility time of passenger requests impact the SAM service performance. The findings show that the time flexibility increase may result in larger profits, more the number of served requests and pooled requests, but prolonged average waiting time of passengers.

### 3.7 Appendix. Notation

---

$T$	Duration of the entire planning horizon
$\mathcal{R}$	Passenger request set
$\mathcal{R}_k$	Set of active passenger requests known up to decision time instant $t_k$ for static subproblem in rolling iteration $k$
$\mathcal{R}_k$	Set of active passenger requests known up to decision time instant $t_k$ for static subproblem in rolling iteration $k$
$\mathcal{O}_k$	Set of onboard requests for static subproblem in rolling iteration $k$
$\mathcal{C}_k$	Set of confirmed requests for static subproblem in rolling iteration $k$
$\mathcal{N}_k$	Set of unscheduled requests for static subproblem in iteration $k$
$v_r^o$	Pick-up location of request $r$
$v_r^d$	Drop-off location of request $r$
$t_r^r$	Request announcement time of request $r$
$\delta_r$	Maximum confirmation duration of request $r$
$t_r^{lp}$	Latest pick-up time from origin of request $r$
$t_r^{ld}$	Latest drop-off time at destination of request $r$
$w_r$	Passenger number in request $r$
$G_r$	Service charge of request $r$ in a solo trip
$\hat{G}_r$	Service charge of request $r$ in a pooled
$Q$	Maximum pooling stranger number
$p_r$	Value of time of request $r$
$g_r$	Privacy-sensitivity value of request $r$
$\hat{q}_r$	Ride-pooling stranger number of request $r$
$\hat{\varsigma}_r$	Ride-pooling duration (with other requests onboard) of request $r$
$\hat{\xi}_r$	Extra (additional) travel time in the pooled trip of request $r$
$\underline{F}$	Minimum passengers' satisfaction threshold
$v_h^0$	Initial location of SAV $h$ at start of the planning horizon
$t_k$	Time instant of decision for each rolling iteration $k$

$\lambda$	Computational time limit for each static subproblem
$\Delta t$	Rolling time step in the rolling horizon framework
$\mathcal{H}$	Set of all SAVs
$v_h$	Current location of SAV $h$ for the static subproblem
$\hat{t}_h$	Arrival time of SAV $h$ at the current location
$\mathcal{S}_{k,h}$	Pre-assigned onboard requests and confirmed requests of SAV $h$ for the static subproblem
$\Theta_{k,h}$	Scheduled partial route information of SAV $h$ for the static subproblem
$\mathcal{G}$	Directed network for the static subproblem
$\mathcal{V}$	Node set within the network for the static subproblem
$\mathcal{E}$	Edge set within the network for the static subproblem
$\mathcal{V}_1$	Subset of nodes of the current vehicle locations for the static subproblem
$\mathcal{V}_2$	Subset of nodes denoting drop-off locations for requests in set $\mathcal{O}_k$
$\mathcal{V}_3$	Subset of nodes denoting pick-up locations for requests in set $\mathcal{C}_k$ and $\mathcal{N}_k$
$\mathcal{V}_4$	Subset of nodes denoting drop-off locations for requests in set $\mathcal{C}_k$ and $\mathcal{N}_k$
$\tau_{ij}$	Travel time from location $i$ to $j$
$\kappa_{ij}$	Travel cost from location $i$ to $j$
$q_r^*$	Recorded number of strangers that have previously shared with the request $r$
$\zeta_r^*$	Recorded (ride-pooling) duration (with other requests onboard) of the implemented (shared) trip of request $r$
$t_r^*$	Past pick-up time instant in the (shared) trip of request $r$
$x_{ij}^{rh}$	Whether request $r$ traverses on edge $(i, j)$ onboard of vehicle $h$
$y_{ij}^h$	Whether the vehicle $h$ traverses on edge $(i, j)$
$d_{rh}$	Whether request $r$ is picked up by vehicle $h$
$z_r$	Whether request $r$ is pooled with any other request during its trip
$q_r$	Number of strangers in the shared trip of request $r$
$u_{hi}$	Time instant at which node $i$ is reached by vehicle $h$

$\Omega$	Possible request-SAV mapping
$s_0$	Initial solution composed of $s_0^h$ , $\forall h \in \mathcal{H}$
$s$	Incumbent solution composed of $s^h$ , $\forall h \in \mathcal{H}$
$s_b$	Best solution composed of $s_b^h$ , $\forall h \in \mathcal{H}$
$s'$	Newly generated candidate solution of $s^{h'}$ , $\forall h \in \mathcal{H}$
$N$	Iteration number counter
$\mu$	Counter of consecutive iterations during which the assignment operator fails to yield an improved solution
$\mu_{\max}$	Maximum continuous attempts that an assignment operator does not achieve a better solution
$\mathcal{A}$	Set of all assignment operators
$m$	Percentage of removed requests in the removal operator
$\mathcal{M}$	Set of all removed requests
$\mathcal{M}_h$	Set of removed requests for vehicle $h$
$\varphi$	Size of the segment for the ARA scheme
$\ell$	Temperature in the simulated annealing accept criterion
$\nu$	Cooling rate in the simulated annealing accept criterion
$N_{\max}$	Maximum iteration number stop criterion
$U_{\max}$	Elapsed CPU time stop criterion
$\theta_{rh}^v$	Violation level of assigning a request $r$ to the SAV $h$
$\theta_{rh}^m$	Matching level of assigning an unscheduled request $r$ to an SAV $h$
$\theta_{rh}^o$	Observation level of assigning request $r$ to an SAV $h$
$\tilde{\mathcal{S}}_h$	Currently assigned requests for SAV $h$ in assignment operators A3 or A4
$ \mathcal{K} $	Number of iterations for recording the previous obtained profit values in assignment operator A5
$ \mathcal{K} _{\max}$	Maximum number of recorded iterations in assignment operator A5
$\varpi_a$	Weight of the assignment operator $a \in \mathcal{A}$
$\pi_a$	Recorded score of assignment operator $a \in \mathcal{A}$
$\psi$	Importance parameter of the historical information on the algorithm performance in the adaptive weight adjustment strategy of the ARA scheme

---

## **Chapter 4 Optimal Compensation Scheme Design for Integrative Shared Mobility Services**

This chapter investigates the ISM (integrative shared mobility) services that use an on-demand shared vehicle fleet to offer transportation services in terms of passenger ride and parcel delivery simultaneously. To stimulate the acceptance of the ERD (extra ride duration) caused by pickups or drop-offs of parcels and other passengers, the ISM service operator will offer the on-board passengers a certain amount of compensation. The passengers' tolerance for ERD is elastic in relation to the compensation amount. The objective is to determine the tactical-level optimal compensation scheme under stochastic passenger and parcel transportation demands considering the operational vehicle routing and the passengers' nonlinear AERD (acceptable ERD) profile. The problem is formulated as a two-stage stochastic programming model and the sample average approximation (SAA) method is employed. A customized iterative ALNS-CSA algorithm that incorporates an ALNS (adaptive large neighborhood search) heuristic and an efficient CSA (compensation scheme adjustment) method is proposed to solve the SAA problem. The ALNS heuristic is used to determine the optimal DPV (demand serving, passenger compensation, and vehicle routing) solution of each scenario while relaxing the cumbersome AERD constraints. The CSA method is developed to further improve the compensation scheme to resume the feasibility of DPV solution in relation to the constraints imposed by the AERD while minimizing the compensation cost. Numerical experiments are carried out to evaluate the performance of the model and solution method. Impact analysis is also conducted to explore the effectiveness of this new business model and derive valuable managerial insights.

The remainder of this chapter is structured as follows. The assumptions, notations and problem description are introduced in Section 4.1. Section 4.2 proposes a two-stage nonlinear programming model formulation. A tailored ALNS-CSA solution method is developed in Section 4.3. The efficacy of the proposed solution method is demonstrated by the numerical experiments in Section 4.4. Section 4.5 provides concluding remarks

for this chapter. Finally, the notations used throughout this chapter are provided in Section 4.6 for readability.

#### 4.1 Assumptions, Notations and Problem Statement

Consider an ISM service provider that operates an on-demand mobility vehicle fleet in the designated set  $\mathbf{K}$ , offering daily door-to-door passenger ride and parcel delivery transportation services simultaneously within a predefined urban area. A depot is available for vehicle parking during periods of low demand in the service area. Throughout a typical operation day, the service provider handles numerous transportation requests from customers, which are distributed both spatially and temporally. The realization of uncertain demand is denoted by  $\omega$ . For each demand realization  $\omega$ , we have distinct sets of demands, one for passenger rides and another for parcel deliveries. Each passenger request specifies with the origin, pickup time window, destination, drop-off time window, and passenger count, whereas each parcel delivery request is characterized by the pickup location, pickup time window, loading duration, drop-off location, drop-off time window, unloading duration, and cargo weight. Let  $\mathbf{V}_\omega^{p,o}$  and  $\mathbf{V}_\omega^{p,d}$  denote the sets of passenger origins and destinations, respectively, and  $\mathbf{V}_\omega^{f,o}$  and  $\mathbf{V}_\omega^{f,d}$  denote the sets of pickup and drop-off positions for parcels, respectively. For ease of presentation, the location indices are organized in the sequence of  $\mathbf{V}_\omega^{p,o}$ ,  $\mathbf{V}_\omega^{f,o}$ ,  $\mathbf{V}_\omega^{p,d}$ , and  $\mathbf{V}_\omega^{f,d}$ . Then the destination of a passenger or parcel with an origin  $i \in \mathbf{V}_\omega^{p,o} \cup \mathbf{V}_\omega^{f,o}$  can be represented by location  $i + \sigma_\omega$ , where  $\sigma_\omega$  stands for the total number of passenger and parcel requests. For simplicity, the index of the origin or pickup location for each passenger or parcel request is used to represent the respective request. Furthermore, the time window for picking up passenger or parcel request  $i \in \mathbf{V}_\omega^{p,o} \cup \mathbf{V}_\omega^{f,o}$  is represented by  $[e_i, l_i]$ , where  $e_i$  and  $l_i$  denote the earliest and latest pickup times, respectively. In a similar manner, the time

window for dropping off a passenger or parcel  $i \in \mathbf{V}_\omega^{p,o} \cup \mathbf{V}_\omega^{f,o}$  is represented by  $[e_{i+\sigma_\omega}, l_{i+\sigma_\omega}]$ . The passenger count for request  $i \in \mathbf{V}_\omega^{p,o}$  is represented by  $q_i^p$ , while the load for parcel request  $i \in \mathbf{V}_\omega^{f,o}$  is indicated by  $q_i^f$ . Loading and unloading times for the service of parcel request  $i \in \mathbf{V}_\omega^{f,o}$  are represented by  $\delta_i$  and  $\delta_{i+\sigma_\omega}$ , respectively.

A vehicle  $k \in \mathbf{K}$  is capable of serving multiple requests for passenger transportation and parcel delivery services simultaneously (in the trunk), subject to specific capacities for carrying passengers and parcels, represented by  $Q_k^p$  and  $Q_k^f$ , respectively. In other words, in addition to ride-pooling among passengers, we also allow ‘ride-pooling’ of both passengers and parcels, allowing for the package deliveries by idle vehicles as well as the ones being in service for passengers. This means that, when carrying a passenger, a vehicle can detour to pick up or drop off additional passengers or parcels. The revenue obtained from fulfilling a request  $i \in \mathbf{V}_\omega^{p,o} \cup \mathbf{V}_\omega^{f,o}$  is denoted by  $R_i$ , while a penalty  $P_i$ , is applied if the request is rejected and not served. The travel time and associated cost from location  $i$  to location  $j$  are indicated by  $t_{ij}$  and  $\kappa_{ij}$ , respectively.

The detour to serve other requests, however, may incur ERD for the on-board passengers. To ensure service quality and customer satisfaction, we assume that the service operator will pay a certain amount of compensation to on-board passengers suffering ERD based on a compensation scheme. Each passenger has an AERD, which is defined as the maximum additional ride time he/she can tolerate, under a specific amount of compensation. The value of AERD is not fixed but rather varies with the amount of compensation offered by the service provider. Hence, each passenger will be associated with an AERD profile, i.e., the AERD variation with the amount of compensation, which could be nonlinear. To thoroughly present the problem of interest, the next two subsections will provide a detailed explanation of the compensation

scheme and passengers' AERD profiles.

#### 4.1.1 Compensation scheme

To encourage the approval of passengers for ride-pooling with parcels or other passengers, we assume that the service operator will compensate on-board passengers for the ERD incurred by the detours for additional pickups or drop-offs during their journeys in ISM services. Specifically, a fixed and a variable amount of compensation will be offered in a stepwise manner. The compensation will thus follow a piecewise linear function with respect to ERD, which can approximate any linear or nonlinear compensation function of general forms. As illustrated in Figure 4.1 (a), to characterize the compensation scheme, let  $[0, h^M]$  denote the ERD interval under consideration, which has been discretized into  $M$  arbitrary intervals with  $(M - 1)$  breakpoints represented by  $h^1, h^2, h^3, \dots, h^m, \dots, h^{M-1}$ . For ease of presentation, we define  $h^0 := 0$ . Then, the compensation scheme can be represented by a vector  $\mathbf{x} := \{x_0, x_1, x_2, \dots, x_m, \dots, x_M\}$ , where  $x_0$  denotes the fixed amount of compensation and  $x_m, \forall m = 1, 2, 3, \dots, M$ , represents the amount of compensation per unit ERD when the ERD  $h$  falls in the interval  $(h^{m-1}, h^m]$ . The amount of compensation  $c$  offered to a passenger suffering ERD  $h$  can thus be calculated by

$$c = FunC(h | \mathbf{x}) = \begin{cases} x_0 + \sum_{n=1}^{m-1} [x_n (h^n - h^{n-1})] + x_m (h - h^{m-1}), & \forall h \in (h^{m-1}, h^m], m = 1, 2, \dots, M \\ 0, & h = 0 \end{cases} \quad (4.1)$$

where  $FunC(\cdot)$  is the compensation function. Figure 4.1 (b) illustrates a piecewise compensation function with three linear segments.

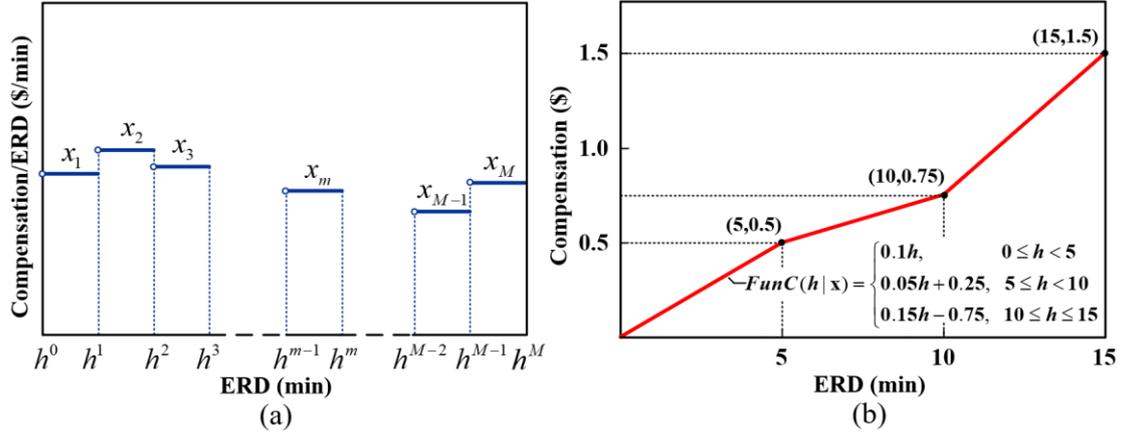


Figure 4.1. Compensation scheme illustration

#### 4.1.2 Passenger's AERD profile

As mentioned earlier, to characterize passengers' satisfaction with ISM services, we define AERD and assume that the value of AERD of each passenger varies with the amount of received compensation. The AERD profile of a passenger is determined by his/her demographics. For simplicity, the passengers in one ride request are assumed to be associated with one AERD profile. Let  $\bar{h}_i$  denote the AERD of passenger request  $i$  under the compensation  $c_i$ . Then the AERD profile of request  $i$  can be represented by the following generic function:

$$\bar{h}_i = FunH(c_i) \quad (4.2)$$

The AERD profile can be a nonlinear continuous function, like the indifference curve in utility theory, or a discontinuous function, such as the step function. Figure 4.2 (a) and Figure 4.2 (b) show two continuous and two stepwise AERD profiles with different combinations of individual-specific parameters, respectively. The function form and parameters of the AERD profile can be identified by stated-preference surveys. For example, passengers can be required to indicate their AERDs under various levels/amounts of compensation in the ride-hailing app when they opt for ISM services. The data will be used to calibrate AERD profiles.

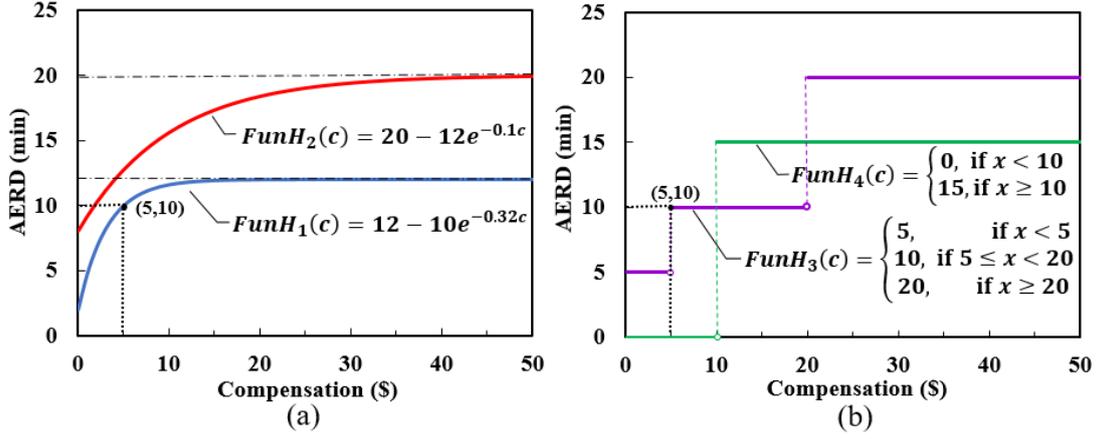


Figure 4.2. Stepwise compensation scheme illustration

Given the stochastic demands, our objective is to determine the optimal compensation scheme  $\mathbf{x}$  at the tactical level by maximizing the expected profit for the service operator across all the realizations of the uncertain demands such that, for each available realization  $\omega$ , the following conditions are satisfied: (i) Each request is accommodated by no more than one vehicle, (ii) the ERD incurred to each on-board passenger is not larger than his/her AERD under the compensation offered according to the compensation scheme, and (iii) constraints on vehicle capacity and requests' time windows are respected. For clarity, we refer to this problem as CSD problem.

#### 4.2 Two-Stage Stochastic Programming Model

In this section, we formulate a two-stage stochastic programming model for the CSD problem. The first stage involves optimizing the expected profit for the service operator by determining the optimal compensation scheme  $\mathbf{x}$ , while the second stage is to maximize the profit under a particular realization  $\omega$  by determining the served demands, the amount of compensation to each served passenger, and the routing plan of vehicles, given the compensation scheme determined in the first stage. For ease of model building, the CSD problem is defined on a directed network  $\mathbf{G}_\omega = (\mathbf{V}_\omega, \mathbf{E}_\omega)$  for a particular realization  $\omega$ , where  $\mathbf{V}_\omega = \mathbf{V}_\omega^{p,o} \cup \mathbf{V}_\omega^{p,d} \cup \mathbf{V}_\omega^{f,o} \cup \mathbf{V}_\omega^{f,d} \cup \{0, 2\sigma_\omega + 1\}$ , and

nodes 0 and  $2\sigma_\omega + 1$  serve as respective start and end depots, both situated at the same physical location. Each node  $i \in \bigcup_\omega \mathbf{V}_\omega$  is associated with a service time window  $[e_i, l_i]$ , a service duration  $\delta_i$ , a passenger load  $q_i^p$ , a parcel load  $q_i^f$ , a revenue  $R_i$ , and a penalty  $P_i$ . Each edge  $(i, j) \in \bigcup_\omega \mathbf{E}_\omega$  is lined to both a travel time  $t_{ij}$  and a corresponding cost  $\kappa_{ij}$  from node  $i$  to node  $j$ ,  $\forall i, j \in \bigcup_\omega \mathbf{V}_\omega$ , with the following details:

- Service duration:  $\delta_i = 0$ ,  $\forall i \in \bigcup_\omega [\mathbf{V}_\omega^{p,o} \cup \mathbf{V}_\omega^{p,d} \cup \{0, 2\sigma_\omega + 1\}]$ .
- Passenger number and parcel load:  $q_0^p = q_{2\sigma_\omega + 1}^p = q_0^f = q_{2\sigma_\omega + 1}^f = 0$ ;  $q_i^f = 0$ ,  $\forall i \in \bigcup_\omega \mathbf{V}_\omega^{p,o}$ ;  $q_i^p = 0$ ,  $\forall i \in \bigcup_\omega \mathbf{V}_\omega^{f,o}$ ;  $q_{i+\sigma_\omega}^p = -q_i^p$  and  $q_{i+\sigma_\omega}^f = -q_i^f$ ,  $\forall i \in \bigcup_\omega (\mathbf{V}_\omega^{p,o} \cup \mathbf{V}_\omega^{f,o})$ .
- Revenue:  $R_i = 0$ ,  $\forall i \in \bigcup_\omega [\mathbf{V}_\omega^{p,d} \cup \mathbf{V}_\omega^{f,d} \cup \{0, 2\sigma_\omega + 1\}]$ .
- Penalty:  $P_i = 0$ ,  $\forall i \in \bigcup_\omega [\mathbf{V}_\omega^{p,d} \cup \mathbf{V}_\omega^{f,d} \cup \{0, 2\sigma_\omega + 1\}]$ .

As for the decision variables, we introduce binary decision variables  $z_i$  to denote whether request  $i$  is served or not, binary decision variables  $y_{ij}^k$  to indicate whether vehicle  $k$  travels directly from node  $i$  to node  $j$ , continuous variables  $\tau_i^k$  to represent the specific time instant when vehicle  $k$  starts the service at node  $i$ , and continuous variables  $r_i^{pk}$  and  $r_i^{fk}$  to represent the passenger and parcel load of vehicle  $k$  after serving at node  $i$ , respectively. Let  $h_i$  and  $c_i$  be the continuous variables denoting the actual ERD of passenger request  $i$  and the compensation offered to him/her. Then, a two-stage stochastic programming model is formulated for the CSD problem, as follows:

[CSD]

First stage:

$$\max_{\mathbf{x} \geq \mathbf{0}} \mathbb{E} \left[ \hat{W}(\mathbf{x}, \omega) \right] \quad (4.3)$$

Second stage:

$$\begin{aligned} \hat{W}(\mathbf{x}, \omega) = \max_{\{y, z, \tau, r, h, c\}} & \sum_{i \in \mathbf{V}_\omega^{p,o} \cup \mathbf{V}_\omega^{f,o}} R_i z_i - \sum_{i \in \mathbf{V}_\omega^{p,o}} c_i - \\ & \sum_{k \in \mathbf{K}} \sum_{i \in \mathbf{V}_\omega} \sum_{j \in \mathbf{V}_\omega} \kappa_{ij} y_{ij}^k - \sum_{i \in \mathbf{V}_\omega^{p,o} \cup \mathbf{V}_\omega^{f,o}} P_i (1 - z_i) \end{aligned} \quad (4.4)$$

subject to

$$\sum_{i \in \mathbf{V}_\omega^{p,o} \cup \mathbf{V}_\omega^{f,o}} y_{0,i}^k = \sum_{i \in \mathbf{V}_\omega^{p,o} \cup \mathbf{V}_\omega^{f,o}} y_{i,2\sigma_\omega+1}^k = 1, \quad \forall k \in \mathbf{K} \quad (4.5)$$

$$\sum_{i \in \mathbf{V}_\omega^{p,o} \cup \mathbf{V}_\omega^{f,o}} y_{0,i}^k = \sum_{i \in \mathbf{V}_\omega^{p,o} \cup \mathbf{V}_\omega^{f,o}} y_{i,2\sigma_\omega+1}^k = 1, \quad \forall k \in \mathbf{K} \quad (4.6)$$

$$z_i = \sum_{k \in \mathbf{K}} \sum_{j \in \mathbf{V}_\omega} y_{ij}^k \leq 1, \quad \forall i \in \mathbf{V}_\omega^{p,o} \cup \mathbf{V}_\omega^{f,o} \quad (4.7)$$

$$\sum_{j \in \mathbf{V}_\omega} y_{ij}^k = \sum_{j \in \mathbf{V}_\omega} y_{i+\sigma_\omega, j}^k, \quad \forall i \in \mathbf{V}_\omega^{p,o} \cup \mathbf{V}_\omega^{f,o}, k \in \mathbf{K} \quad (4.8)$$

$$\tau_i^k \leq \tau_{i+\sigma_\omega}^k, \quad \forall i \in \mathbf{V}_\omega^{p,o} \cup \mathbf{V}_\omega^{f,o}, k \in \mathbf{K} \quad (4.9)$$

$$e_i \leq \tau_i^k \leq l_i, \quad \forall i \in \mathbf{V}_\omega, k \in \mathbf{K} \quad (4.10)$$

$$\tau_j^k \geq (\tau_i^k + \delta_i + t_{ij}) y_{ij}^k, \quad \forall i, j \in \mathbf{V}_\omega, k \in \mathbf{K} \quad (4.11)$$

$$r_j^{pk} \geq (r_i^{pk} + q_j^p) y_{ij}^k, \quad \forall i, j \in \mathbf{V}_\omega, k \in \mathbf{K} \quad (4.12)$$

$$r_j^{fk} \geq (r_i^{fk} + q_j^f) y_{ij}^k, \quad \forall i, j \in \mathbf{V}_\omega, k \in \mathbf{K} \quad (4.13)$$

$$\max \{0, q_i^p\} \leq r_i^{pk} \leq \min \{Q_k^p, Q_k^p + q_i^p\}, \quad \forall i \in \mathbf{V}_\omega, k \in \mathbf{K} \quad (4.14)$$

$$\max \{0, q_i^p\} \leq r_i^{pk} \leq \min \{Q_k^p, Q_k^p + q_i^p\}, \quad \forall i \in \mathbf{V}_\omega, k \in \mathbf{K} \quad (4.15)$$

$$h_i = \sum_{k \in \mathbf{K}} (\tau_{i+\sigma_\omega}^k - \tau_i^k) - t_{i, i+\sigma_\omega}, \quad \forall i \in \mathbf{V}_\omega^{p,o} \quad (4.16)$$

$$c_i = \text{FunC}(h_i | \mathbf{x}), \quad \forall i \in \mathbf{V}_\omega^{p,o} \quad (4.17)$$

$$\bar{h}_i = \text{FunH}(c_i), \quad \forall i \in \mathbf{V}_\omega^{p,o} \quad (4.18)$$

$$h_i \leq \bar{h}_i, \quad \forall i \in \mathbf{V}_\omega^{p,o} \quad (4.19)$$

$$y_{ij}^k \in \{0,1\}, z_i \in \{0,1\}, \tau_i^k, r_i^{pk}, r_i^{jk}, h_i, c_i \geq 0, \quad \forall i, j \in \mathbf{V}_\omega, k \in \mathbf{K} \quad (4.20)$$

The objective function in Eq. (4.3) for the first-stage aims to maximize the service provider's expected profit subject to a non-negative constraint. This objective is determined by the optimal value of the objective function for the second-stage problem, as specified in Eqs. (4.4)–(4.20). The objective function in Eq. (4.4) for the second-stage aims to maximize the service provider's profit under a particular scenario of passenger and parcel transportation requests. For each realization of the demand, the ISM service operator will further provide the optimal vehicle routing and scheduling to maximize overall profitability. Constraint (4.5) specifies the origin and destination of each vehicle. Eq. (4.6) expresses flow conservation equations applicable to each node, excluding the depot. Constraints (4.7) and (4.8) guarantee that no requests can be accommodated more than once by different vehicles. Constraint (4.9) enforces that the pickup time for each request must precede its drop-off time. Constraint (4.10) requires adherence to the designated time windows for each node. Constraint (4.11) updates every request's service time instant along the route of a vehicle, while constraints (4.12) and (4.13) update the passenger number and parcel load, respectively. Constraints (4.14) and (4.15) impose the limits of vehicle carrying capacity. Constraints (4.16) and (4.17) calculate the actual ERD and the corresponding compensation offered to passengers, which connect the decisions of the first-stage and second-stage problems. Constraint (4.18) defines the passenger's AERD profile. Constraint (4.19) guarantees that the ERD of a passenger request cannot exceed the AERD calculated by constraint. Constraint (4.20) defines the domains of second-stage decision variables.

### 4.3 Solution Algorithm Design

Owing to the curse of dimensionality, the two-stage stochastic programming model [CSD] presents significant computational difficulties. The consideration of nonlinear AERD profiles will induce additional complexity. To address this problem, we will first employ the SAA method to transform the two-stage stochastic programming model into an SAA problem by approximating the expected total profit by a sample average estimate, i.e.,  $\mathbb{E}[\hat{W}(\mathbf{x}, \omega)] = \sum_{\omega \in \Omega} \phi_{\omega} \hat{W}(\mathbf{x}, \omega)$ , where  $\Omega$  represents a sample set including random demand realizations and  $\phi_{\omega}$  is the occurrence probability of scenario  $\omega$ . The SAA problem aims to simultaneously optimize compensation scheme and DPV plans under each scenario that maximize the approximated total profit.

A customized iterative hybrid algorithm combining the ALNS algorithm and an efficient CSA method is developed for solving the SAA problem. The algorithm is referred to as ALNS-CSA algorithm. The ALNS algorithm is a widely-used solution method to address various VRP variants like PDP (Ropke and Pisinger, 2007). This method improves the solution by iteratively selecting and applying different removal and insertion operators, while adapting operators' selection probabilities based on past performance to identify near-optimal solutions in an extensive solution space. Despite its efficacy for vehicle routing-related problems, the incorporation of the compensation scheme design decision and AERD profiles make it difficult to apply ALNS directly for the SAA problem. To this end, we will first decompose the optimization of compensation scheme and DPV plans by relaxing the AERD constraint. The relaxation can be achieved by replacing the elastic AERD profile function with a fixed value of maximum AERD, i.e.,  $\bar{h}_i^{\max} \leftarrow \lim_{c_i \rightarrow \infty} FunH(c_i)$ . The optimization of DPV plans will thus become independent from the compensation scheme design, which can be further decomposed into several DPV sub-problems, each corresponding to a specific scenario. Each DPV sub-problem is a PDP variant that is effectively addressed by ALNS. The obtained DPV solution, however, may not be feasible due to the AERD constraint

relaxation procedure. We will then restore the feasibility of the DPV solution while minimizing the compensation cost by implementing a customized CSA method. The ALNS and CSA methods will be employed in an iterative process until a predetermined iteration count is achieved. The following three subsections will provide a detailed elaboration on the ALNS-CSA algorithm framework, the new DPV solution generation, and the compensation scheme determination in Subsections 4.3.1, 4.3.2 and 4.3.3, respectively.

### 4.3.1 ALNS-CSA heuristic algorithm framework

The iterative ALNS-CSA algorithm starts with an arbitrary compensation scheme. During each iteration, the ALNS heuristic is first employed to determine DPV plans while relaxing the AERD constraints under a specific compensation scheme. Based on the DPV solution, the CSA method will be employed later to update the compensation scheme and restore DPV solution feasibility while minimizing the compensation cost. The new DPV and compensation scheme solution will be evaluated to be potentially accepted as the incumbent solution and optimal solution. The probabilities for selecting ALNS operators are adaptively updated according to historical solution performance. This loop will continue until the specified threshold for the number of iterations is met. It is noteworthy that the adaptive improvement of DPV plans will lay the foundation for the subsequent compensation scheme update procedure.

**Algorithm 4.1** outlines the pseudocode of the ALNS-CSA solution method. It can be seen that an initial compensation scheme  $\mathbf{x}^0 := \{x_0^0, x_1^0, x_2^0, \dots, x_m^0, \dots, x_M^0\}$  will be first created, in which  $x_m^0$  is a randomly generated number. Given the current compensation scheme  $\mathbf{x}^0$ , an initial DPV solution  $\{\Psi_\omega^0\}_{\omega \in \Omega}$  is obtained by a parallel heuristic proposed by Potvin and Rousseau (1993). The criterion for determining the optimal position of each request in the routes will be the objective function value

derived from the second-stage problem in the [CSD] model, denoted by  $W(\Psi_\omega | \mathbf{x})$  for DPV solution  $\{\Psi_\omega\}_{\omega \in \Omega}$  given the compensation scheme  $\mathbf{x}$ . The incumbent solution  $(\mathbf{x}, \{\Psi_\omega\}_{\omega \in \Omega})$  and optimal solution  $(\mathbf{x}^*, \{\Psi_\omega^*\}_{\omega \in \Omega})$  are initialized to be the initial solution  $(\mathbf{x}^0, \{\Psi_\omega^0\}_{\omega \in \Omega})$  (see Line 2). We will also initialize the selection probability of the ALNS operators denoted by  $\mathbf{p}_\omega, \forall \omega \in \Omega$  (see Line 3).

After the initialization, a loop is launched for new DPV solution generation in multiple scenarios and the compensation scheme adjustment procedure (see Lines 6–7), the incumbent and optimal solutions updating (see Lines 8–13), and the ALNS operators' selection probability updating (see Line 14). Specifically,  $ALNS(\bullet)$  and  $CSA(\bullet)$  are the subfunctions used to generate a new DPV solution and corresponding new compensation scheme. ALNS operators will be selected based on their probability for each scenario  $\omega \in \Omega$  under the incumbent compensation scheme  $\mathbf{x}$ . The selected operators will be applied to define a move, which involves the removal and insertion of a certain number  $\rho_\omega$  of requests, to generate a new DPV solution  $\Psi_\omega^{n'}$  at the  $n^{th}$  iteration from the incumbent solution  $\Psi_\omega$  (see Line 6). After this, a new compensation scheme  $\mathbf{x}^{n'}$  will be determined based on the incumbent one and the DPV solution  $\{\Psi_\omega^{n'}\}_{\omega \in \Omega}$  (see Line 7).

For solution evaluation and operator selection probability update,  $AcpIcm(\bullet)$  and  $AcpOpt(\bullet)$  are two subfunctions employed to evaluate the acceptance of a new solution and to update the incumbent solution (see Lines 8–10) and optimal solution (see Lines 11–13) accordingly. The subfunctions will return 'true' if certain criteria are met. Specifically, subfunction  $AcpIcm(\bullet)$  is used for determining that a new solution

is accepted with a probability  $\exp[-(\sum_{\forall \omega \in \Omega} \phi_{\omega} W(\Psi_{\omega}^* | \mathbf{x}^*) - \sum_{\forall \omega \in \Omega} \phi_{\omega} W(\Psi_{\omega}^{n'} | \mathbf{x}^{n'})) / T]$  based on a simulated annealing acceptance criterion, where  $T > 0$  denotes the temperature. The starting temperature is defined as  $T_{start}$  and will be decreased in every iteration based on a cooling rate  $0 < \nu < 1$  such that  $T \leftarrow \nu T$ . The subfunction  $AcpOpt(\bullet)$  evaluates and accepts a new solution as optimal if it demonstrates superior performance compared to the current best solution, i.e.,  $\sum_{\forall \omega \in \Omega} \phi_{\omega} W(\Psi_{\omega}^{n'} | \mathbf{x}^{n'}) > \sum_{\forall \omega \in \Omega} \phi_{\omega} W(\Psi_{\omega}^* | \mathbf{x}^*)$ . As for selection probability,  $PrbUpd(\bullet)$  is the subfunction used to update ALNS operators' selection probabilities, drawing on their historical performance metrics according to roulette wheel selection principle proposed by Golberg (1981) (see Line 14). The loop will terminate after  $n_{max}$  iterations, and the algorithm will produce the current optimal solution  $(\mathbf{x}^*, \{\Psi_{\omega}^*\}_{\omega \in \Omega})$  identified as its output.

In subsequent subsections, we will provide a detailed explanation of the new DPV plan generation using ALNS operators and compensation scheme determination by an efficient CSA method in Subsections 4.3.2 and 4.3.3, respectively.

---

**Algorithm 4.1. Pseudocode of the proposed ALNS-CSA algorithm**

---

**Input:** On-demand mobility vehicle set and passenger and parcel demand set

**Output:** Optimal compensation scheme and DPV solution  $(\mathbf{x}^*, \{\Psi_\omega^*\}_{\omega \in \Omega})$

---

```
1  Generate an initial compensation scheme  $\mathbf{x}^0$  and DPV solution  $\{\Psi_\omega^0\}_{\omega \in \Omega}$ ;
   Initialize incumbent and optimal solutions:  $(\mathbf{x}, \{\Psi_\omega\}_{\omega \in \Omega}) \leftarrow (\mathbf{x}^0, \{\Psi_\omega^0\}_{\omega \in \Omega})$ ,
2   $(\mathbf{x}^*, \{\Psi_\omega^*\}_{\omega \in \Omega}) \leftarrow (\mathbf{x}^0, \{\Psi_\omega^0\}_{\omega \in \Omega})$ ;
3  Initialize the selection probability of operators  $\mathbf{p}_\omega, \forall \omega \in \Omega$ ;
4   $n \leftarrow 1$ ;
5  Repeat
6  |  $\{\Psi_\omega^{n'}\}_{\omega \in \Omega} \leftarrow ALNS(\mathbf{x}, \{\Psi_\omega\}_{\omega \in \Omega}, \{\mathbf{p}_\omega\}_{\omega \in \Omega}, \{\mathbf{O}_\omega\}_{\omega \in \Omega})$ ; // new DPV solution
7  |  $\mathbf{x}^{n'} \leftarrow CSA(\mathbf{x}, \{\Psi_\omega^{n'}\}_{\omega \in \Omega})$ ; // compensation scheme determination
8  | If  $AcpIcm((\mathbf{x}^*, \{\Psi_\omega^*\}_{\omega \in \Omega}), (\mathbf{x}^{n'}, \{\Psi_\omega^{n'}\}_{\omega \in \Omega}))$  is true, then
9  | |  $(\mathbf{x}, \{\Psi_\omega\}_{\omega \in \Omega}) \leftarrow (\mathbf{x}^{n'}, \{\Psi_\omega^{n'}\}_{\omega \in \Omega})$ ; // update incumbent solution
10 | EndIf
11 | If  $AcpOpt((\mathbf{x}^*, \{\Psi_\omega^*\}_{\omega \in \Omega}), (\mathbf{x}^{n'}, \{\Psi_\omega^{n'}\}_{\omega \in \Omega}))$  is true, then
12 | |  $(\mathbf{x}^*, \{\Psi_\omega^*\}_{\omega \in \Omega}) \leftarrow (\mathbf{x}^{n'}, \{\Psi_\omega^{n'}\}_{\omega \in \Omega})$ ; // update optimal solution
13 | EndIf
14 |  $\mathbf{p}_\omega \leftarrow PrbUpd(\mathbf{p}_\omega), \forall \omega \in \Omega$ ; // update selection probability of operators
15 |  $n \leftarrow n + 1$ ;
16 Until  $n > n_{\max}$ 
17 Return the optimal compensation scheme and DPV solution  $(\mathbf{x}^*, \{\Psi_\omega^*\}_{\omega \in \Omega})$ .
```

---

### 4.3.2 New DPV solution generation

As aforementioned, the ALNS heuristic is a widely applied solution method to address the PDP and its variants (Ropke and Pisinger, 2007). Given an initial routing solution, ALNS can generate a new solution by employing a series of operators at adaptive frequencies for both removal and insertion. These operators define different kinds of methods (neighborhoods), for generating a set of solutions (neighbors), thereby leading to a diversified search process. For instance, a removal operator can be defined as randomly removing several requests from a route, while an insertion operator can be

defined as inserting each removed request into ‘destroyed’ routes at a best position that can increase the objective function value the most (for maximization problem). In particular, during each iteration of ALNS, the choice of the operators is adaptively guided by their historical performance during the search and a well-performed operator will have a higher selection probability.

To apply the ALNS heuristic for the DPV sub-problem of a particular scenario, following Ropke and Pisinger (2006), we will employ five removal operators, including random removal, Shaw removal, worst removal, spatial-oriented removal, and temporal-oriented removal, and four insertion operators, including greedy, regret-2, regret-3, and regret- $\eta$  heuristics. In each iteration, the algorithm will use a single removal operator and a single insertion operator to generate the new DPV solution. These operators have demonstrated high effectiveness for general PDPs. Since the DPV sub-problem involves two types of requests and allows request rejection with the aim of maximizing the profit, we will refine these operators considering the differences between the DPV sub-problem and PDPs. In addition, for selection probability update, operator performance is usually evaluated based on the obtained solution. Given that the primary objective of the CSD problem is the optimal compensation scheme design, we propose new operator evaluation criteria: the performance of operators will be evaluated not only based on the solution quality for DPV sub-problem, but also on the potential to further improve the compensation scheme in the next step. The performance of the proposed evaluation criteria will be assessed in numerical experiments.

#### *4.3.2.1 Removal and insertion operators*

##### Removal operators

Five removal operators will be employed to decide  $\rho_\omega$  requests to be removed.

- **R1 (random removal):** This operator randomly chooses  $\rho_\omega$  requests, regardless of passenger or parcel requests, to be removed, for the diversification of the search.
- **R2 (worst removal):** This operator sequentially removes  $\rho_\omega$  requests, guided by the difference in the second-stage objective function value with the incumbent DPV plan and a derived DPV plan. The derived DPV plan is obtained by removing a particular request from the incumbent plan.

Ropke and Pisinger (2006) proposed a worst removal operator that evaluates the cost difference before and after the removal of an individual request, and the operator seeks to remove requests with high-cost difference. Considering the profit maximization objective of this study, we evaluate the profit difference and remove the ones with a low profit difference value. Specifically, let  $\Psi_{\omega(-i)}$  denote the derived DPV plan by removing request  $i$  from the DPV plan  $\Psi_\omega$ . In each iteration of the removal, we will rank all remaining candidate requests in the ascending order of the value of  $(W(\Psi_\omega | \mathbf{x}) - W(\Psi_{\omega(-i)} | \mathbf{x}))$  and remove the request at the  $\lfloor y_1^{\mu_1} |L_1| \rfloor^{\text{th}}$  position, where  $y_1$  represents a random number selected from the interval  $[0,1)$ ,  $\mu_1 \geq 1$  represents the determinism parameter used to introduce a degree of randomness into the selection process, and  $|L_1|$  is the number of remaining candidate requests.

- **R3 (Shaw removal):** This operator removes  $\rho_\omega$  requests, focusing on prioritizing requests that exhibit similarities. The rationale behind this operator is that it is easier to rearrange requests that share similarities, potentially leading to the creation of better solutions (Ropke and Pisinger, 2006).

Shaw (1998) proposed a removal heuristic based on the *relatedness measure* considering the similarities regarding distance, time, capacity and vehicle suitability of

two requests. This measure aims to identify similar requests that are not highly restricted in terms of being inserted back into their original positions or unfavorable ones. Removing more related requests provides more opportunities for the interchange of their positions within the routes, potentially generating more profitable reinsertions. In this study, we define the relatedness measure between two requests consisting of three components in terms of the spatial, temporal and capacity terms while considering the request type. To be specific, compared with two requests of the same type, one parcel request and one passenger request are much more different from each other and the value of relatedness measure of two different types of requests will thus be lower. Let  $\theta_{ij}^\omega$  denote the relatedness measure of requests  $i$  and  $j$  defined as follows:

$$\theta_{ij}^\omega = \chi_1 (\kappa_{ij} + \kappa_{\sigma_\omega+i, \sigma_\omega+j}) + \chi_2 (|\tau_i - \tau_j| + |\tau_{\sigma_\omega+i} - \tau_{\sigma_\omega+j}|) + \chi_3 \left[ 1 + \mathcal{G}_{ij} \left( \frac{|q_i^p - q_j^p| + |q_i^f - q_j^f|}{\max\{q_i^p, q_j^p, q_i^f, q_j^f\}} - 1 \right) \right] \quad (4.21)$$

where  $\mathcal{G}_{ij}$  denotes a binary parameter, which equals 1 when requests  $i$  and  $j$  are different types, and 0 otherwise; and  $\chi_1$ ,  $\chi_2$ , and  $\chi_3$  are the coefficients of the terms measuring spatial, temporal, and capacity relatedness, respectively. For implementation, the values of  $\kappa_{ij}$ ,  $\kappa_{\sigma_\omega+i, \sigma_\omega+j}$ ,  $\tau_i$ ,  $\tau_j$ ,  $\tau_{\sigma_\omega+i}$  and  $\tau_{\sigma_\omega+j}$  will be normalized such that they take on values in interval  $[0,1]$ . We will randomly select an initial request to be removed and iteratively remove other requests one by one based on the value of compatibility. In each iteration of the removal, we will rank all candidate requests in the ascending order of the compatibility between each candidate request and a request randomly selected from the removed request set. Then, the request at the  $\lfloor y_2^{\mu_2} |L_2| \rfloor^{\text{th}}$  place will be removed, where  $y_2$  represents a randomly selected number from the interval  $[0,1)$  and  $\mu_2 \geq 1$  represents the determinism parameter used to introduce a degree of randomness into the selection process, and  $|L_2|$  is the number of remaining

candidate requests.

- **R4 (spatial-oriented removal)**: This operator represents the particular case of **R3** when  $\chi_1 = 1$ ,  $\chi_2 = 0$ , and  $\chi_3 = 0$ .
- **R5 (temporal-oriented removal)**: This operator represents the particular case of **R3** when  $\chi_1 = 0$ ,  $\chi_2 = 1$ , and  $\chi_3 = 0$ .

### Insertion operators

We employ four insertion operators, i.e., greedy, regret-2, regret-3, and regret- $\eta$  heuristics, to construct a new DPV plan based on the destroyed one. Each request is associated with a *best route-specific insertion position* for each route of the vehicles, in which the insertion on the route brings the maximum profit improvement while ensuring the feasibility of the DPV solution. All the best route-specific insertion positions for a request will be ranked in the descending order of profit improvement and the position at the  $r^{th}$  place is referred to as the  *$r^{th}$  best insertion position* and the corresponding route is called the  *$r^{th}$  best insertion route*. Note that if all the best route-specific insertion positions for a request lead to the profit decrease, this request will be removed completely, indicating that the service of this request will be denied.

The greedy heuristic involves iteratively reinserting a request from the removed request set which will lead to the maximum profit improvement and insert it at the 1<sup>st</sup> best insertion position. The regret- $\eta$  heuristic is to iteratively insert a request based on the accumulated difference between the profit improvement brought by inserting it into the 1<sup>st</sup> best insertion route and that brought by inserting it into the second to the  $\eta^{th}$  best insertion routes. The request that exhibits the greatest accumulated difference will be chosen to be placed in the 1<sup>st</sup> best insertion position. The regret-2 and regret-3 heuristics are special cases of regret- $\eta$  heuristic with  $\eta = 2$  and  $\eta = 3$ , respectively.

### 4.3.2.2 Operator selection mechanism

We will select the operator for new DPV solution generation according to roulette wheel selection principle (Golberg, 1981). Specifically, the entire search is equally divided into several segments grouped in set  $S$ , and each segment  $g \in S$  consists of a number of iterations. Each operator is associated with a segment-specific weight and an iteration-specific score. All the operators are equally weighted by a randomly generated number in the first segment and its value will be updated in the subsequent segment based on the score of the operator at the last iteration of the current segment. Let  $\xi_o^g$  be the score of the operator  $o \in \mathbf{O}_\omega$  at the last iteration of segment  $g \in S$ , and  $w_o^g$  denote the weight of operator  $o$  in segment  $g$ . Then the weight of operator  $o$  in segment  $g+1$  is calculated by

$$w_o^{g+1} = w_o^g (1 - \ell) + \ell \cdot \frac{\xi_o^g}{\pi_o^g} \quad (4.22)$$

where  $\ell$  is the reaction factor that defines the rate at which the weight adjustment responds to the variations in the algorithm's effectiveness, and  $\pi_o^g$  denotes the frequency with which operator  $o$  is applied during segment  $g$ . Given the operator's weight, the probability of selecting operator  $o$  is determined by

$$p_o^g = \frac{w_o^g}{\sum_{o \in \mathbf{O}_\omega} w_o^g} \quad (4.23)$$

### Score determination

The score of the operator indicates its historical performance and a higher score means a better performance. The value of the score will be initialized to be zero at the first iteration of each segment and will be increased by  $\sigma_\omega^n$  in iteration  $n$  if it is used in that iteration. In the literature, the score increment  $\sigma_\omega^n$  is usually determined by the

solution quality achieved by the corresponding operator, which includes the following three cases: (1) the new solution is the global optima, (2) the new solution improves the current solution that was not previously accepted, and (3) the new solution, which was not accepted earlier, is worse than the current solution (Ropke and Pisinger, 2017). However, as the ultimate purpose of the ALNS-CSA algorithm is the optimal compensation scheme determination, the new DPV solution, with a high potential leading to compensation cost reduction in the step of compensation scheme update, should be desired. To incorporate the influence of a DPV solution on the compensation cost, we propose the notion of AERD deviation of a DPV solution  $\Delta_{\omega}^n$  to be detailed in the next subsection. A large value of AERD deviation means the current DPV solution largely deviates from the feasible and optimal one in terms of AERD constraint. Therefore, we define the score increment  $\sigma_{\omega}^n$ , as the sum of an objective-function-dependent term  $\sigma_{\omega,Obj}^n$  and a new AERD-deviation-related term  $\sigma_{\omega,\Delta}^n := \min\{\varepsilon / \Delta_{\omega}^n, \bar{\varepsilon}\}$  as below:

$$\sigma_{\omega}^n = \sigma_{\omega,Obj}^n + \sigma_{\omega,\Delta}^n \quad (4.24)$$

where the objective-function-dependent term  $\sigma_{\omega,Obj}^n$  is given by

$$\sigma_{\omega,Obj}^n = \begin{cases} \varepsilon_1, & \text{if } W(\Psi_{\omega}^{n'} | \mathbf{x}^{n'}) > W(\Psi_{\omega}^* | \mathbf{x}^*) \\ \varepsilon_2, & \text{if } W(\Psi_{\omega}^{n'} | \mathbf{x}^{n'}) > W(\Psi_{\omega} | \mathbf{x}) \text{ and} \\ & \text{AcpIcm}\left(\left(\mathbf{x}, \{\Psi_{\omega}\}_{\omega \in \Omega}\right), \left(\mathbf{x}^{n'}, \{\Psi_{\omega}^{n'}\}_{\omega \in \Omega}\right)\right) = 1 \text{ and} \\ & \left(\mathbf{x}^{n'}, \{\Psi_{\omega}^{n'}\}_{\omega \in \Omega}\right) \text{ is not in the table of visited solutions} \\ \varepsilon_3, & \text{if } W(\Psi_{\omega}^{n'} | \mathbf{x}^{n'}) < W(\Psi_{\omega} | \mathbf{x}) \text{ and} \\ & \text{AcpIcm}\left(\left(\mathbf{x}, \{\Psi_{\omega}\}_{\omega \in \Omega}\right), \left(\mathbf{x}^{n'}, \{\Psi_{\omega}^{n'}\}_{\omega \in \Omega}\right)\right) = 1 \text{ and} \\ & \left(\mathbf{x}^{n'}, \{\Psi_{\omega}^{n'}\}_{\omega \in \Omega}\right) \text{ is not in the table of visited solutions} \\ 0, & \text{otherwise} \end{cases} \quad (4.25)$$

and  $\varepsilon_1, \varepsilon_2, \varepsilon_3, \varepsilon, \bar{\varepsilon}$  are pre-specified parameters.

### 4.3.3 Compensation scheme determination

Given the new DPV solution, an efficient CSA method is developed to update the compensation scheme to restore the feasibility of the obtained DPV solution in relation to the AERD constraint, while minimizing the total compensation cost. To be more specific, each passenger served in the DPV solution will be associated with a gap between the ERD and the real AERD, referred to as the AERD deviation of that passenger. The received compensation amount and the corresponding real AERD of passenger  $i$  can be calculated first based on the current compensation scheme and ERD. Then the AERD deviation of passenger  $i$  will be given by

$$\Delta h_i = \bar{h}_i - h_i = \text{FunH}(\text{Func}(h_i | \mathbf{x})) - h_i \quad (4.26)$$

The value of  $\Delta h_i$  indicates the passenger-level AERD deviation with the following three cases:

- **Case 1:**  $\Delta h_i < 0$ . The passenger is under-compensated and the AERD constraint is violated. In this case, some compensation components  $x_m$  should be increased to ensure more compensation is offered to the concerned passenger so that the value of the AERD can be improved to eliminate the violation.
- **Case 2:**  $\Delta h_i = 0$ . The passenger is exactly-compensated and the AERD constraint is satisfied. The passenger's ERD is exactly equal to AERD under the current compensation scheme.
- **Case 3:**  $\Delta h_i > 0$ . The passenger is over-compensated and the AERD constraint is respected. Some compensation components  $x_m$  can be reduced to cut down the compensation cost while maintaining the validation of the AERD constraint.

With the above definition, the AERD deviation of a DPV solution in Subsection 4.3.2.2 will be the sum of the AERD deviation of all the served passengers by the DPV solution. In other words, we have

$$\Delta_{\omega}^n = \sum_{i \in \Upsilon_{\omega}^n} |\Delta h_i| \quad (4.27)$$

where  $\Upsilon_{\omega}^n$  denotes the set of passengers served by DPV solution  $\{\Psi_{\omega}^n\}_{\omega \in \Omega}$ .

To further determine the adjustment amount of compensation component  $x_m$ , we will first calculate the compensation deviation, i.e., the amount of undercompensation or overcompensation, of each passenger. Let  $\underline{c}_i$  be the minimum required compensation for passenger  $i$  to accept the current ERD. Then the compensation deviation  $\Delta c_i$  will be given by

$$\Delta c_i = \underline{c}_i - c_i = FunH^{-1}(h_i) - c_i \quad (4.28)$$

where  $FunH^{-1}(\cdot)$  denotes the inverse function of AERD profile function  $FunH(\cdot)$ .

For passengers with non-zero  $\Delta c_i$ , we can adjust the compensation components  $x_m$  based on undercompensation/overcompensation amount in Eq. (4.28), thereby increasing/decreasing the compensation received by that passenger. The purpose is to make passenger's AERD align with the ERD. Although this is a straightforward idea to restore the feasibility of DPV solutions while maintaining a low compensation cost, the implementation requires a careful design because the adjustment of the compensation scheme based on a particular passenger will inevitably affect the received compensation of the other passengers, probably making DPV solution infeasible again. To avoid this, we develop an iterative CSA method for compensation scheme determination, which includes a critical passenger identification mechanism and increase/decrease adjustment techniques.

### 4.3.3.1 CSA method

Given the current compensation scheme  $\mathbf{x}$  and DPV solution  $\{\Psi_\omega\}_{\omega \in \Omega}$ , we will first classify the passengers served by the DPV solution into three exclusive sets  $\Upsilon_1$ ,  $\Upsilon_2$  and  $\Upsilon_3$ , who are under-compensated, exactly-compensated, and over-compensated, respectively, based on  $\Delta c_i$ . To avoid the violation of the AERD constraint, we will focus on the under-compensated passengers in set  $\Upsilon_1$  first, and iteratively check each passenger and increase the compensation components until there is no under-compensated passengers. We will then examine the over-compensated passengers in set  $\Upsilon_3$ , and iteratively check each passenger and reduce the compensation components until no feasible reduction is allowed. Kindly note that each time we make any adjustment to the compensation scheme, the three passenger sets  $\Upsilon_1$ ,  $\Upsilon_2$  and  $\Upsilon_3$  will be updated as well. Furthermore, adjustments for both increase and decrease are constrained within a specific range.

#### Increase adjustment

For increase adjustment, let  $m_i$  denote the index of the minimum ERD breakpoint that is larger than passenger  $i$ 's ERD  $h_i$ , such that  $h_i$  falls in the interval  $(h^{m_i-1}, h^{m_i}]$  of the compensation scheme. The deficient compensation amount  $\Delta c_i$  can be complemented by increasing the compensation component  $x_m$  by  $\frac{\Delta c_i}{h_i}$  over the range  $(0, h^{m_i}]$ . Figure 4.3 illustrates the increase adjustment according to the compensation deviation of passenger  $i$ . To minimize the impact on the other passengers, in each iteration, we will identify a critical passenger  $i^*$  among the ones

in set  $\Upsilon_1$  that is associated with the minimal value of  $\frac{\Delta c_i}{h_i}$ , i.e.,  $i^* \leftarrow \arg \min_{i \in \Upsilon_1} \left\{ \frac{\Delta c_i}{h_i} \right\}$ ,

and implement the increase adjustment as follows:

$$x_m \leftarrow x_m + \frac{\Delta c_{i^*}}{h_{i^*}}, \quad \forall m = 1, 2, 3, \dots, m_{i^*}. \quad (4.29)$$

Note that before the adjustment in the next iteration, the passenger sets  $\Upsilon_1$ ,  $\Upsilon_2$  and  $\Upsilon_3$  need to be updated based on the newly adjusted compensation scheme.

### Decrease adjustment

Different from the increase adjustment that is made over the range  $(0, h^{m_i}]$ , the decrease adjustment is proposed for a more bounded interval  $(h^{\hat{m}}, h^{m_i}]$ , where the index  $\hat{m}$  of the ERD breakpoint  $h^{\hat{m}}$  is set to be the largest  $m_i$  of the exactly-compensated passengers in set  $\Upsilon_2$ , namely,  $\hat{m} \leftarrow \max_{i \in \Upsilon_2} \{m_i\}$ . This can avoid the AERD constraint violation of the passengers in set  $\Upsilon_2$ . Figure 4.4 illustrates the decrease adjustment based on the compensation deviation of passenger  $i$ . To minimize the impact on the other passengers in set  $\Upsilon_3$ , in each iteration, we will identify a critical passenger  $i^*$  in set  $\Upsilon_3$ , which is associated with a non-null adjustable interval and the minimal value of  $\frac{|\Delta c_i|}{h_i - h^{\hat{m}}}$ , that is,  $i^* \leftarrow \arg \min_{i \in \Upsilon_3, s.t. m_i > \hat{m}} \left\{ \frac{|\Delta c_i|}{h_i - h^{\hat{m}}} \right\}$ , and implement the decrease adjustment as follows:

$$x_m \leftarrow x_m - \frac{|\Delta c_{i^*}|}{h_{i^*} - h^{\hat{m}}}, \quad \forall m = \hat{m}, \hat{m} + 1, \hat{m} + 2, \dots, m_{i^*}. \quad (4.30)$$

Again, it is worth mentioning that the passenger sets  $\Upsilon_2$  and  $\Upsilon_3$  for the determination of  $\hat{m}$  and  $i^*$  are continuously updated based on the newly adjusted

compensation scheme.

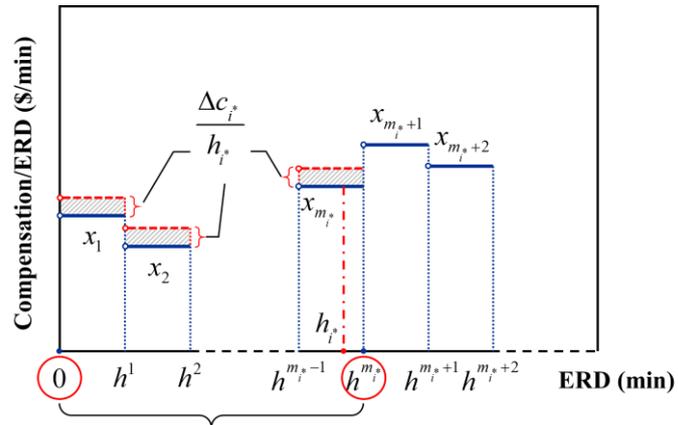


Figure 4.3. Increase adjustment illustration

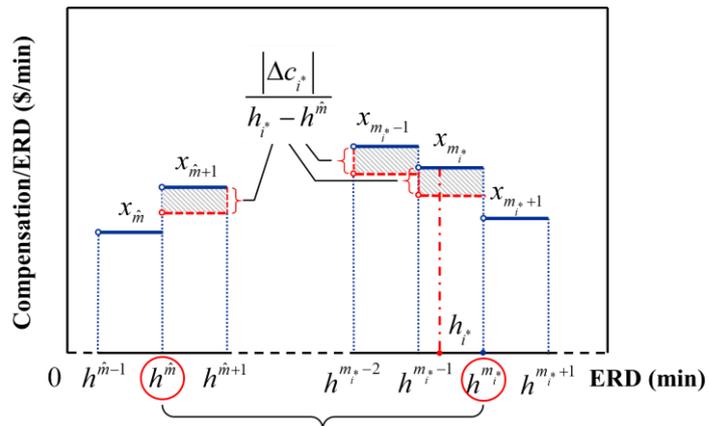


Figure 4.4. Decrease adjustment illustration

The overall flowchart of the CSA method with critical passenger identification and increase and decrease adjustment techniques is illustrated in Figure 4.5.

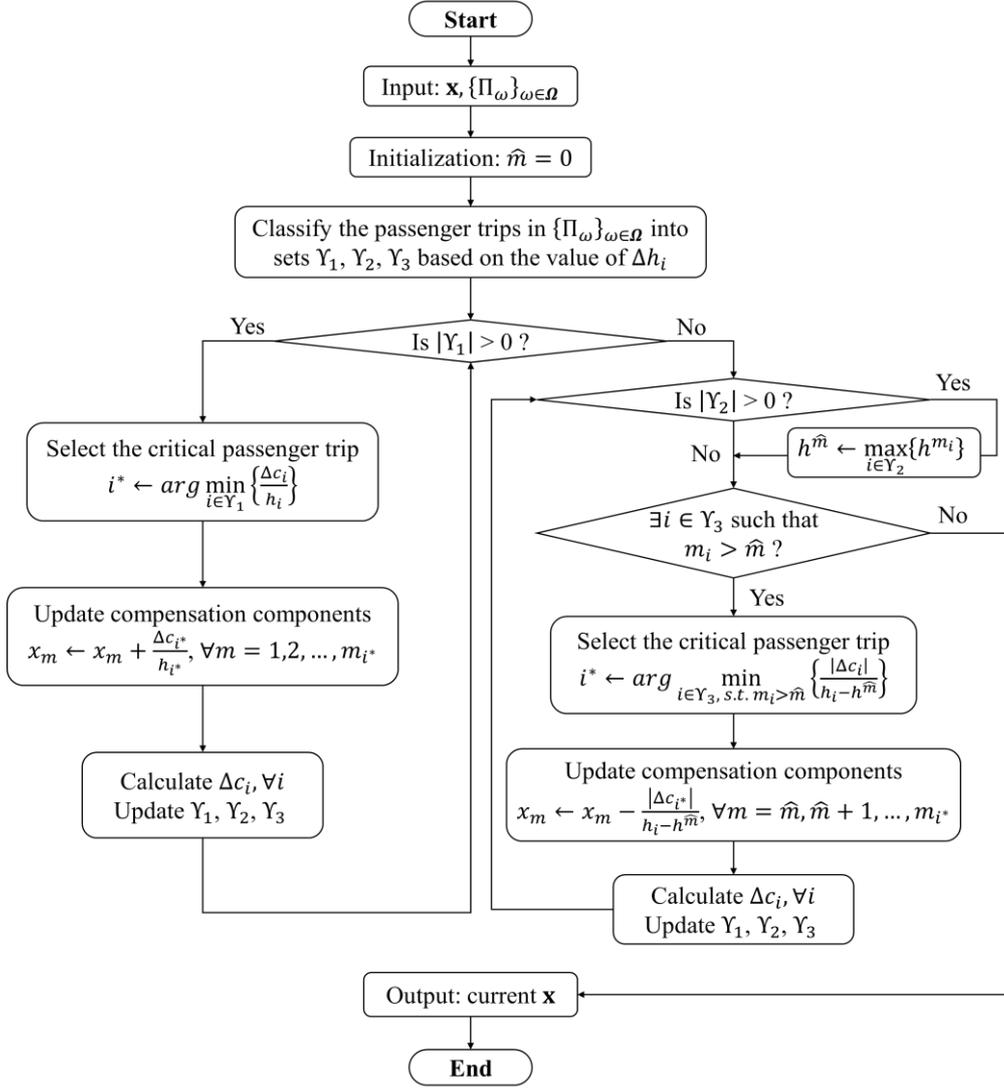


Figure 4.5. Flowchart of our proposed CSA method

#### 4.4 Numerical Experiments

This section reports on extensive numerical experiments involving randomly generated instances. First, the test instance generation and experimental settings will be introduced. Then, the performance of the proposed model and ALNS-CSA algorithm will be evaluated. Finally, we will explore the benefit of the ISM service and analyze the impact of the parcel delivery demand and passengers' AERD profile on the system performance. The solution algorithm is coded with Python on a personal computer with Intel (R) Core (TM) i7, 2.80GHz CPU, 16.0 GB RAM.

#### 4.4.1 Test instances and parameter setting

The test instances with various numbers of passengers and parcels are generated in a  $20 \text{ km} \times 20 \text{ km}$  square region in a study period from 7:00 a.m. to 7:00 p.m. For stochastic demand, we consider a total number of demand realizations  $|\Omega|=4$ , among which three are weekday scenarios and one is weekend scenario, each with an occurrence probability of 0.25.

To simulate real-world travel patterns of passengers, we begin by selecting a few representative locations within this geographical area. Specifically, three residential neighborhood centers at (5 km, 7 km), (10 km, 12 km) and (15 km, 5 km), three CBDs at (4 km, 17 km), (9 km, 15 km) and (14 km, 9 km), and six leisure centers at (3 km, 10 km), (11 km, 8 km), (7 km, 4 km), (6 km, 6 km), (12 km, 11 km) and (8 km, 7 km) are first selected. We will simulate different demand patterns on weekdays and weekends. For example, on weekdays, we will generate 40% of passenger requests traveling from origins in residential areas to destinations in CBD areas during the morning rush hour from 7:00 AM to 9:00 AM, while for the evening rush hour between 5:00 PM and 7:00 PM, we will generate 40% passenger requests with origins in CBD areas and destinations in residential areas. The specific coordinates for both origins and destinations for all passenger requests will be randomly selected within a 2 km radius of these representative locations. The origins and destinations of the other 60% passenger requests on weekdays are generated randomly within this region. Let the study period be defined as the interval from 0 to 720 minutes. The earliest drop-off time  $e_{i+\sigma_o}$  for each outbound request  $i$  from home to workplace during morning rush hours is randomly and uniformly selected from the interval  $[0, 120]$ , while the latest drop-off time  $l_{i+\sigma_o}$  is randomly chosen from the interval  $[e_{i+\sigma_o} + 15, e_{i+\sigma_o} + 45]$ . Similarly, for each inbound request  $i$  from the workplace to home during evening rush hours, the earliest and latest pickup time  $e_i$  and  $l_i$  are randomly selected from the

interval  $[600, 720]$  and  $[e_i + 15, e_i + 45]$ , respectively. The pickup time windows of outbound requests and time windows for dropping off inbound requests are defined as  $[0, 720]$ . For parcel requests, their pick-up and drop-off locations will be randomly chosen within the study area with a service time window  $[0, 720]$ . The location of the depot is also randomly selected from the region.

As for the AERD profile of passenger request, we employ a nonlinear concave function as follows:

$$\bar{h}_i = FunH(c_i) = 2t_{i,i+\sigma_o} - \alpha_i e^{-\beta_i c_i} \quad (4.31)$$

where  $\alpha_i$  and  $\beta_i$  are the individual-specific parameters that jointly describe the attitude of the passenger to the compensation. The parameter  $\alpha_i$  is randomly and uniformly selected from  $[15, 20]$  for weekdays and  $[5, 10]$  for weekends. The parameter  $\beta_i$  is randomly and uniformly selected from  $[0.05, 0.15]$ .

For the other parameters of the CSD problem, the passenger number of each passenger request  $q_i^p$  is randomly set to be 1 or 2, while each parcel request load  $q_i^f$  is set to be 1. Loading and unloading times for each parcel request are set to be 10 s. The vehicle number is defined to be half of the total request number. The capacities for carrying passenger and parcel are set to be 4 and 2, respectively. The Euclidean distance is employed to compute the direct distance from node  $i$  to  $j$ , denoted as  $d_{ij}$ . By assuming the constant vehicle traveling speed of 50 km/h, the travel time  $t_{ij}$  can be easily computed. Travel cost  $\kappa_{ij}$  is set to be  $0.5d_{ij}$ . The revenue obtained from each passenger and parcel request is set to be  $3d_{i,i+\sigma_o}$  and  $2d_{i,i+\sigma_o}$ , respectively. The penalty for denying the passenger request  $P_i$  is set  $1.5d_{i,i+\sigma_o}$ . For the compensation scheme design, we consider the total number of intervals  $M = 5$  with breakpoints at 5 min, 10

min, 15 min, 20 min, and 25 min.

Regarding the parameter setting for the proposed ALNS-CSA algorithm, the initial value of each compensation component  $x_m$ ,  $\forall m = 0, 1, 2, \dots, 5$  is set to 2. The other algorithmic parameters are determined by the tuning method introduced by Ropke and Pisinger (2007). The iteration number limit  $n_{\max}$  is defined as 200. The cooling rate in the acceptance criterion  $\nu$  is defined as 0.9999. In each iteration, the requests to be removed  $\rho_\omega$  is designated as 1/4 of the total request number. The parameter  $\mu_1$  in operator R2 is set to 5. Parameters  $\chi_1$ ,  $\chi_2$  and  $\chi_3$  in operator R3 are 9, 2, and 5 respectively. Parameters in score update,  $\varepsilon_1$ ,  $\varepsilon_2$ ,  $\varepsilon_3$ ,  $\varepsilon$ , and  $\bar{\varepsilon}$ , are 10, 5, 3, 5 and 10.

All the instances for the numerical experiments are available on GitHub at <https://github.com/JiangyanHuang/ISM>.

#### 4.4.2 Algorithm performance

This subsection assesses the performance of our proposed ALNS-CSA relative to the commercial solver Gurobi. We will also examine the efficacy of the proposed ALNS operators and the operator selection mechanism.

##### 4.4.2.1 Comparison of ALNS-CSA and Gurobi

In this subsection, we will evaluate the computational results generated by our proposed ALNS-CSA algorithm in comparison to those produced by Gurobi. The preliminary results show that Gurobi cannot find the optimal solution when both the passenger and parcel request numbers exceed 6. Hence, to evaluate the solution quality achieved by our proposed algorithm, we will consider several groups of small-size instances with the numbers of passenger and parcel requests ranging from 2 to 6. Each instance group is associated with a specific number of passengers and parcel requests.

For each instance group, we will generate five different instances with randomly generated parameters detailed in Subsection 4.4.1. We also examine the scalability of the ALNS-CSA algorithm for relatively large instances involving the passenger request number and parcel request number varying from 10 to 50. A limit of 2 h is imposed in Gurobi. For a particular instance, Gurobi will terminate when it either finds the optimum in 2 h or when the elapsed time exceeds 2 h.

Table 4.1 shows the results of the proposed ALNS-CSA algorithm and Gurobi in small-size instances. Each instance is named by ‘(#Passenger, #Parcel)-No.’, where ‘#Passenger’ and ‘#Parcel’ are the number of passenger requests and parcel requests, and ‘No.’ is the instance index given a passenger and parcel request number. For example, ‘(2,4)-1’ refers to the 1<sup>st</sup> instance with 2 passenger requests and 4 parcel requests. For each solution method, we present the best value of objective (Obj) and corresponding computation time (CPU time) for every test instance. We highlight the best objective values in bold while marking the optimal objective values with asterisks.

For a better presentation, the relative gaps (RelGap), computed as  $\frac{obj^* - obj}{obj} \times 100\%$ , are also reported, where  $obj^*$  and  $obj$  refer to the objective values achieved using our proposed ALNS-CSA algorithm and Gurobi, respectively.

We can see from Table 4.1 that the proposed ALNS-CSA method obtains the solutions to all instances within 60 s, while Gurobi takes much longer times. Not all instances can be solved by Gurobi within 2 hr. In fact, Gurobi fails to solve all the instances with 6 parcel requests, while the ALNS-CSA algorithm shows a much better performance in those unsolved instances, with 7.0%, 17.2%, and 24.5% better objective function values than that of Gurobi in instances groups (2,6), (4,6) and (6,6), respectively. The results demonstrate the computational complexity encountered when directly solving the proposed CSD problem by the mixed-integer programming solver. For the rest instances that are solved by Gurobi, the ALNS-CSA method shows

comparable performance with Gurobi regarding solution quality, exhibiting a small average relative solution gap of 0.58%. The optimal solutions to more than half of these instances have also been found by ALNS-CSA method, with much less times than Gurobi. Overall, the proposed ALNS-CSA method shows comparable performance with Gurobi in terms of solution quality while demonstrating much higher computational efficiency than Gurobi.

To examine the scalability of the proposed algorithm more thoroughly, we solve several relatively larger instances with passenger and parcel request numbers ranging from 10 to 50. For these instances, Gurobi fails to identify any feasible solution within 2 h. We report the average results using the ALNS-CSA method for the five instances in each instance group in Table 4.2. We can observe that our solution method finds reasonable solutions with objective values increasing with the growth in request number. We further illustrate the computational efficiency in Figure 4.6 based on the data presented in the table. Figure 4.6 (a) depicts how CPU times changes with an increasing number of parcels under specific numbers of passenger requests. Figure 4.6 (b) shows the how CPU times changes with an increasing number of passengers under specific numbers of parcel requests. It can be observed that the CPU times of our algorithm show a linear and exponential trend with the increase of parcel and passenger request numbers, respectively. This indicates good scalability of the proposed algorithm for instance with a greater parcel number. However, the results also demonstrate the high sensitivity of the computation time to the passenger request number. Since passenger requests have more constraints such as the passengers' ride-pooling acceptance, it would be more challenging to find feasible and good-quality DPV solutions. Therefore, further improvement of the computational efficiency of our algorithm is attained by reducing the times for handling passenger requests in DPV solution generation.

Table 4.1. Small-scale instances evaluation of ALNS-CSA algorithm versus Gurobi

Instance	ALNS-CSA		Gurobi		RelGap
	Obj	CPU time (s)	Obj	CPU time (s)	
(2,2)-1	<b>81*</b>	7	<b>81*</b>	686	0.0%
(2,2)-2	<b>96*</b>	7	<b>96*</b>	518	0.0%
(2,2)-3	66	10	<b>68*</b>	1,020	-2.9%
(2,2)-4	<b>66*</b>	7	<b>66*</b>	375	0.0%
(2,2)-5	<b>54*</b>	6	<b>54*</b>	418	0.0%
Average	73	7	73	603	-0.5%
(2,4)-1	103	16	<b>104*</b>	1,230	-1.0%
(2,4)-2	111	15	<b>113*</b>	761	-1.8%
(2,4)-3	<b>128*</b>	15	<b>128*</b>	1,092	0.0%
(2,4)-4	108	13	<b>111*</b>	898	-2.7%
(2,4)-5	<b>83*</b>	15	<b>83*</b>	1,135	0.0%
Average	107	15	108	1,023	-1.1%
(2,6)-1	<b>165</b>	20	151	7,200	9.3%
(2,6)-2	<b>131</b>	26	121	7,200	8.3%
(2,6)-3	<b>143</b>	22	135	7,200	5.9%
(2,6)-4	<b>142</b>	20	137	7,200	3.6%
(2,6)-5	<b>122</b>	16	113	7,200	8.0%
Average	141	21	131	7,200	7.0%
(4,2)-1	<b>123*</b>	12	<b>123*</b>	938	0.0%
(4,2)-2	116	15	<b>117*</b>	712	-0.9%
(4,2)-3	<b>96*</b>	12	<b>96*</b>	805	0.0%
(4,2)-4	129	10	<b>130*</b>	1,265	-0.8%
(4,2)-5	<b>98*</b>	16	<b>98*</b>	566	0.0%
Average	112	13	113	857	-0.4%
(4,4)-1	154	17	<b>155*</b>	2,230	-0.6%
(4,4)-2	123	26	<b>124*</b>	2,253	-0.8%
(4,4)-3	<b>157*</b>	26	<b>157*</b>	2,033	0.0%
(4,4)-4	<b>167*</b>	18	<b>167*</b>	1,961	0.0%
(4,4)-5	<b>152*</b>	18	<b>152*</b>	1,449	0.0%
Average	151	21	151	1,985	-0.3%
(4,6)-1	<b>170</b>	25	143	7,200	18.9%
(4,6)-2	<b>161</b>	40	132	7,200	22.0%
(4,6)-3	<b>185</b>	26	166	7,200	11.4%
(4,6)-4	<b>239</b>	25	202	7,200	18.3%
(4,6)-5	<b>211</b>	27	181	7,200	16.6%
Average	193	29	165	7,200	17.2%
(6,2)-1	<b>152*</b>	19	<b>152*</b>	1,067	0.0%
(6,2)-2	<b>108*</b>	18	<b>108*</b>	1,125	0.0%

(6,2)-3	<b>102*</b>	21	<b>102*</b>	1,123	0.0%
(6,2)-4	140	15	<b>141*</b>	1,105	-0.7%
(6,2)-5	144	18	<b>145*</b>	1,003	-0.7%
Average	129	18	130	1,085	-0.3%
(6,4)-1	186	30	<b>188*</b>	5,209	-1.1%
(6,4)-2	165	49	<b>166*</b>	6,266	-0.6%
(6,4)-3	186	49	<b>189*</b>	6,890	-1.6%
(6,4)-4	<b>154*</b>	35	<b>154*</b>	5,240	0.0%
(6,4)-5	220	32	<b>222*</b>	5,338	-0.9%
Average	182	39	184	5,789	-0.9%
(6,6)-1	<b>228</b>	62	189	7,200	19.4%
(6,6)-2	<b>235</b>	53	192	7,200	21.4%
(6,6)-3	<b>233</b>	49	178	7,200	27.4%
(6,6)-4	<b>230</b>	48	169	7,200	30.3%
(6,6)-5	<b>215</b>	52	167	7,200	23.9%
Average	228	53	179	7,200	24.5%

Table 4.2. Large-scale instances evaluation of ALNS-CSA algorithm

Instance group	Obj	CPU time (s)	Instance group	Obj	CPU time (s)	Instance group	Obj	CPU time (s)
(10,10)	338	121	(20,10)	356	392	(30,10)	382	889
(10,20)	424	288	(20,20)	466	805	(30,20)	517	1,246
(10,30)	545	706	(20,30)	595	1,154	(30,30)	682	1,705
(10,40)	721	1,083	(20,40)	770	1,455	(30,40)	830	2,199
(10,50)	831	1,580	(20,50)	871	1,864	(30,50)	1,029	2,557
(40,10)	465	1,452	(50,10)	624	2,852	-	-	-
(40,20)	647	1,863	(50,20)	792	3,153	-	-	-
(40,30)	785	2,222	(50,30)	924	3,419	-	-	-
(40,40)	973	2,954	(50,40)	1,149	4,308	-	-	-
(40,50)	1,203	3,460	(50,50)	1,530	5,162	-	-	-

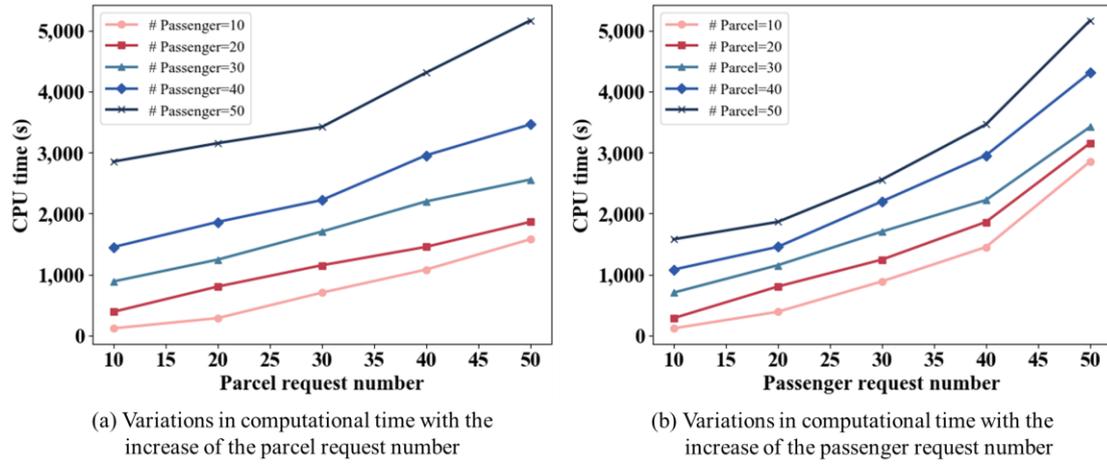


Figure 4.6. Variations in computation time of the ALNS-CAS algorithm with request numbers

#### 4.4.2.2 Performance of ALNS operators and operator selection mechanism

In this subsection, we begin by evaluating the effectiveness of five removal operators and four insertion operators. Figure 4.7 presents the average call count percentage of all the operators. For removal operators, it shows that Shaw and worst removal operators are more frequently used and contribute more to the solution improvement, followed by two special cases of Shaw removal, i.e., spatial-oriented removal and temporal-oriented removal. As for the insertion operator, regret-3 is most frequently called, followed by regret- and regret-2. In addition to the call count, we also calculate the contribution to solution improvement of each operator for instance group (10,10) in Table 4.3. The contribution is measured by the solution gap between the proposed algorithm with all operators included and a benchmark method excluding the concerned operator only (the other operators are included). The larger the solution gap, the more the contribution of the concerned operator. These results align with our earlier findings that Shaw removal and worst removal are the best removal operators while regret-3 is the best insertion operator.

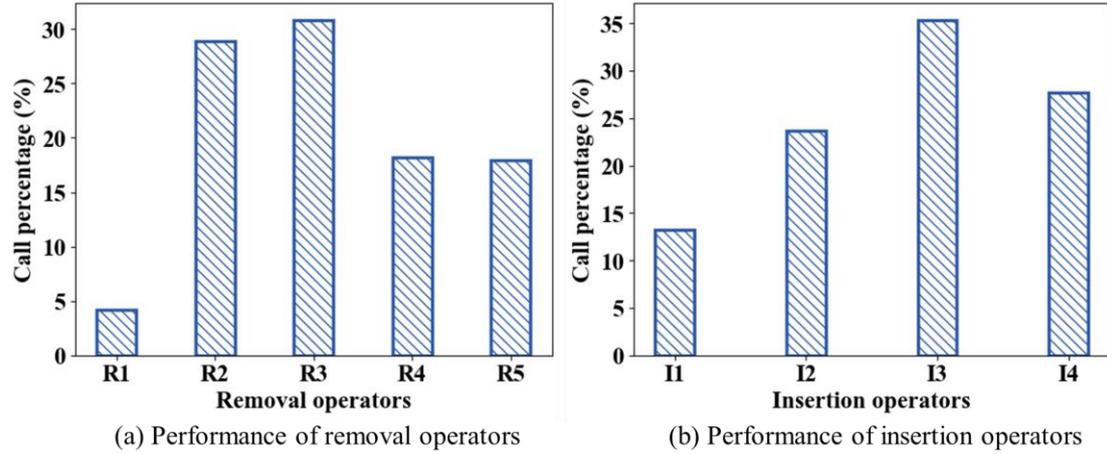


Figure 4.7. Average call percentage of removal and insertion operators

Table 4.3. Contribution of operators in terms of solution gap

	R1	R2	R3	R4	R5
Removal operators	2.9	14.5	18.1	8.1	10.0
Insertion operators	2.5	2.4	10.2	8.8	-

To explore the efficacy of the proposed operator selection mechanism, i.e., the critical score determination, in the proposed ALNS-CSA framework, the results of the algorithm with the proposed score determination method are evaluated against those obtained through the traditional benchmark approach, which does not include the AERD-deviation-related term. These two methods are referred to as ‘wAERD’ and ‘w/oAERD’. The average objective value (Obj) and relative gap (RelGap) of the two methods for each instance group are presented in Table 4.4. The results show that our proposed ALNS-CSA solution method outperforms the benchmark approach for all the instance groups with better solutions. The average relative gap can be as high as 8.52% for instance group (50,10). Additionally, the relative gap increases with the increase in the passenger request number and the decrease in the parcel request number. This implies that our proposed operator selection mechanism leads to a more significant advantage for the instances where the proportion of passenger requests is higher. This is because more passenger requests provide more information in terms of AERD

deviation to the score increment, and thus offering more potential to reduce compensation costs and determine favorable solutions.

Table 4.4. Performance of operator selection mechanism in the ALNS-CSA algorithm

Instance group	wAERD	w/oAERD	RelGap	Instance group	wAERD	w/oAERD	RelGap
(10,10)	338	322	4.97%	(20,10)	356	339	5.01%
(10,20)	424	411	3.16%	(20,20)	466	447	4.25%
(10,30)	545	531	2.64%	(20,30)	595	574	3.66%
(10,40)	721	711	1.41%	(20,40)	770	757	1.72%
(10,50)	831	820	1.34%	(20,50)	871	856	1.75%
(30,10)	382	362	5.52%	(40,10)	465	438	6.16%
(30,20)	517	493	4.87%	(40,20)	647	616	5.03%
(30,30)	682	657	3.81%	(40,30)	785	762	3.02%
(30,40)	830	810	2.47%	(40,40)	973	944	3.07%
(30,50)	1,029	1,010	1.88%	(40,50)	1,203	1,180	1.95%
(50,10)	624	575	8.52%	-	-	-	-
(50,20)	792	754	5.04%	-	-	-	-
(50,30)	924	887	4.17%	-	-	-	-
(50,40)	1,149	1,105	3.98%	-	-	-	-
(50,50)	1,530	1,495	2.34%	-	-	-	-

#### 4.4.3 Impact analysis

In this subsection, we will first examine the benefit of the ISM service by comparing it with passenger transportation services with and without pooling, referred to as Pw/P and Pw/oP, respectively. We will then analyze the impact of the parcel delivery demand and passengers' AERD profile on the performance of ISM services. Unless stated otherwise, we use the instances in group (50,30) with 5 vehicles, and the parameter  $\alpha_i$  of the AERD profile is set to be 20 and 10 for the weekday and weekend scenarios, respectively.

### Impact of ISM service model

We will compare the profit and service rate of passenger requests of the ISM, Pw/P, and Pw/oP services. For the ISM and Pw/P services, we will also evaluate the average ERD (AvgERD) and average compensation (AvgCom) received by the passengers with detours. Table 4.5 summarizes the comparison results of the three systems. As expected, it shows that the Pw/P service has a slightly higher total profit and service rate than the Pw/oP service. In addition, the profit of ISM is much higher than that of Pw/P, demonstrating the significance of parcel delivery in improving the profitability of mobility services. It's also encouraging to see that the high profit of ISM is achieved without sacrificing the coverage of passenger services as the average service rate of ISM and Pw/P services are the same. There are cases, for instance (50,30)-4, for example, the ISM results in a lower service rate than the Pw/P service. This is because some passenger requests are rejected in order to serve parcel requests with less travel costs. On the other hand, we also have cases in which the ISM service model leads to a higher service rate, such as instance (50,30)-3. It implies that some passenger requests rejected in Pw/P service become profitable to be served in ISM service, probably along with the service of parcel requests nearby. This is further supported by the increased average detour time and the corresponding increase in the compensation amount for ISM service. In summary, we can see that passenger pooling can potentially enhance both the profitability and service rate of mobility services, but the improvement is limited. By allowing parcel delivery, the profitability can be further greatly improved without much negative impact on the passenger service rate. This is achieved by longer detours in passenger rides supported by larger amounts of compensation.

Table 4.5. Result comparison of ISM services with two benchmark services

Instance	ISM				PwP				PwoP	
	Profit (\$)	SerRat	AvgERD (min)	AvgCom (\$)	Profit (\$)	SerRat	AvgERD (min)	AvgCom (\$)	Profit (\$)	SerRat
(50,30)-1	961	0.93	7.36	11.19	630	0.93	3.70	3.77	561	0.90
(50,30)-2	893	0.89	7.77	11.75	597	0.89	3.98	5.49	522	0.84
(50,30)-3	913	0.93	7.60	12.42	619	0.91	3.78	5.27	570	0.90
(50,30)-4	881	0.87	8.08	12.62	598	0.90	4.39	7.50	559	0.87
(50,30)-5	923	0.94	7.52	11.57	647	0.93	3.64	2.77	575	0.88
Avg.	914	0.91	7.67	11.91	618	0.91	4.01	4.96	557	0.88

Impact of parcel delivery demand

To examine how the parcel delivery demand affects the system performance, we will test the instances with a fixed passenger request number of 50 while varying the parcel number from 10 to 50. In addition to the aforementioned metrics, we also report the optimal compensation level (ComLev) of the compensation scheme, i.e., the average amount of compensation per unit time calculated by  $\sum_{n=1}^M x_n / M$ , and the percentage of passenger trips with detours among all passenger requests, referred to as detour rate (DetRat). Table 4.6 summarizes the results of the impacts of the number parcel requests on system performance. For a better presentation, the variations of compensation level and profit are also visualized in Figure 4.8. It can be seen that with the increase in parcel delivery demand, we should set a higher compensation level to encourage more passengers to accept longer detours for parcel delivery. This is verified by the increased detour rate and average detour time in Table 4.6. By doing so, we will achieve a significant profit improvement, with its value more than doubled when the parcel request number increases from 10 to 50, while maintaining a relatively stable and high service rate at around 90%. The average compensation amount also shows an apparent increase because of the collective effects of increased compensation level and detour time. It is noteworthy to mention that the increment of compensation level

actually decreases with the parcel request number, as shown in Figure 4.8. This implies that there may exist a threshold, beyond which the increase in the compensation level may not be helpful for improving profit further. Because passengers' AERD is supposed to be upperly bounded, a higher compensation level will not contribute to a longer passenger detour time tolerance, when the compensation has already been at a high level. If this is the case, offering more compensation will incur more costs instead.

Table 4.6. Impacts of the parcel request number

#Parcel	Profit (\$)	SerRat	AvgERD (min)	AvgCom (\$)	ComLev (\$/min)	DetRat
10	556	0.91	5.85	5.07	0.79	0.35
20	743	0.93	6.37	7.82	1.25	0.38
30	961	0.93	7.36	11.19	1.78	0.39
40	1,163	0.88	8.18	14.46	2.01	0.40
50	1,483	0.91	10.61	19.79	2.21	0.42

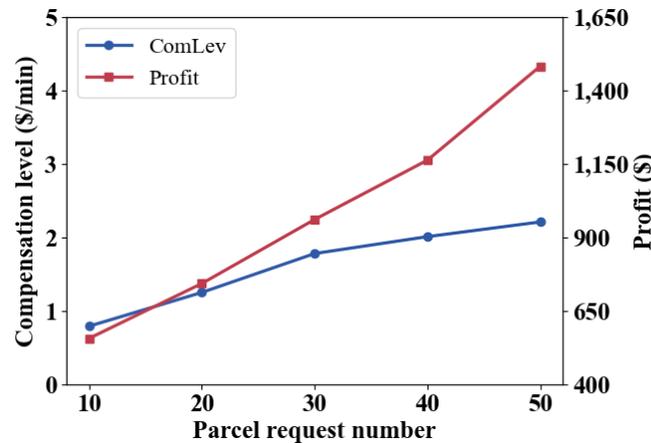


Figure 4.8. Variations of compensation level and profit with parcel number increase

### Impact of AERD profile

To explore the impact of AERD profile, we will analyze the solutions under increasing values of parameter  $\alpha_i$  in weekday and weekend scenarios, i.e.,  $(\alpha_i^D, \alpha_i^E) \in \{(12,2), (16,6), (20,10), (24,14), (28,18)\}$ , where  $\alpha_i^D$  and  $\alpha_i^E$  denote the

parameters on weekdays and weekends, respectively. The results are tabulated in Table 4.7. Again, we further visualize the variations in compensation level and profit in Figure 4.9. As indicated in Eq. (31), a higher value of parameter  $\alpha_i$  means a high value-of-time of the passengers and thus a greater reluctance to detour during trips. We can see from the figure that when passengers become more reluctant to detour, a higher compensation level should be set to incentivize them in order to create opportunities for ride pooling. It also shows that similar to the impact analysis of parcel delivery demand, there also exists an upper bound of the compensation level with the increase of passengers' detour reluctance, as the AERD may have approached the maximal values. Despite the efforts to provide more incentives, both the profit and passenger service rate will suffer with obvious drops when passengers' detour reluctance increases. As expected, the detour rate and average detour time of passengers decreases as well. The value of average compensation received by passengers fluctuates as it is jointly determined by the increasing compensation level and the decreasing detour time.

Table 4.7. Impacts of AERD parameter

$(\alpha_i^D, \alpha_i^E)$	Profit (\$)	SerRat	AvgERD (min)	AvgCom (\$)	ComLev (\$/min)	DetRat
(12,2)	1,046	0.94	9.59	9.96	1.10	0.44
(16,6)	984	0.93	8.05	10.68	1.57	0.44
(20,10)	961	0.93	7.36	11.19	1.78	0.39
(24,14)	940	0.91	7.19	11.52	1.98	0.39
(28,18)	868	0.86	6.12	11.36	2.02	0.35

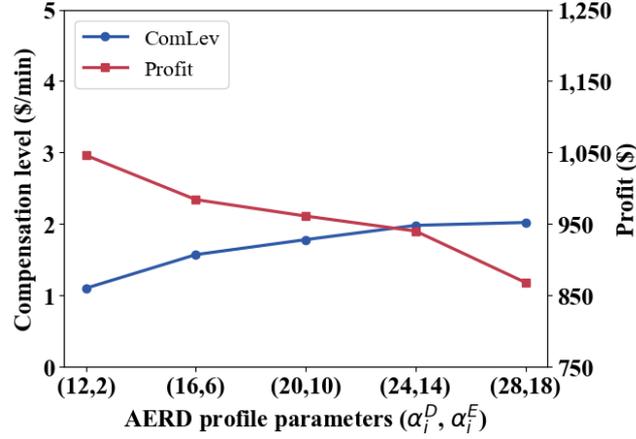


Figure 4.9. Variations in the compensation level and profit under different AERD parameters

#### 4.5 Concluding Remarks

This chapter investigates the compensation scheme design problem considering passengers' elastic tolerance for ERD for the on-demand mobility service-based ISM system considering stochastic demand. An on-demand mobility vehicle fleet is employed to handle simultaneous passenger ride and parcel delivery tasks subject to the constraints of their carrying capacity. Passengers will receive a certain amount of compensation determined by the compensation scheme for the ERD caused by the shared rides with parcels and other passengers. The passenger's nonlinear AERD profile is used for describing the elasticity of the tolerance for ERD to compensation.

To optimize the tactical compensation scheme while considering the operational vehicle routing and passengers' AERD profiles that maximize the expected profit of the service provider, we formulated a two-stage stochastic programming model incorporating nonlinear AERD profiles of passengers. The first stage determines the optimal compensation scheme and the second stage deals with a VRP variant considering passengers' elastic tolerance for ERD upon the realization of the stochastic demands. Considering the intricate characteristics of the CSD model, we develop a tailored metaheuristic algorithm ALNS-CSA that is adapted from the ALNS framework

with the integration of an efficient CSA method to obtain the optimal compensation scheme. Finally, numerical experiments were carried out to demonstrate the efficacy of our proposed solution method and to explore the impacts of several key factors on system performance. Based on a well-designed compensation scheme, the OMS-based ISM services can facilitate profit improvement for service providers. In real-world implementations, the compensation scheme should be determined before the realization of the different scenarios. Therefore, the optimal solution considering the stochastic demand information could provide insights and guidance for the planning stage in the operation of such a coupled system.

## 4.6 Appendix. Notation

---

<i>Indices and sets</i>	
<hr/>	
$\Omega$	Demand scenario set
$\mathbf{K}$	Vehicle set $\mathbf{K} = \{1, 2, \dots, k, \dots,  \mathbf{K} \}$
$\mathbf{G}_\omega = (\mathbf{V}_\omega, \mathbf{E}_\omega)$	Directed network with node set $\mathbf{V}_\omega$ and edge set $\mathbf{E}_\omega$ in scenario $\omega \in \Omega$
$\mathbf{V}_\omega^{p,o}$	Passenger requests' origin set in scenario $\omega \in \Omega$
$\mathbf{V}_\omega^{p,d}$	Passenger requests' destination set in scenario $\omega \in \Omega$
$\mathbf{V}_\omega^{f,o}$	Parcel requests' origin set in scenario $\omega \in \Omega$
$\mathbf{V}_\omega^{f,d}$	Parcel requests' destination set in scenario $\omega \in \Omega$
$(i, \sigma_\omega + i)$	Index for the origin node $i$ and destination node $\sigma_\omega + i$ of request $i$
$\{0, 2\sigma_\omega + 1\}$	Index for the origin and destination depots
<hr/>	
<i>Known parameters</i>	
<hr/>	
$\phi_\omega$	Occurrence probability for scenario $\omega \in \Omega$
$[e_i, l_i]$	Time window for node $i \in \bigcup_{\omega \in \Omega} \mathbf{V}_\omega$
$\delta_i$	Service time of node $i \in \bigcup_{\omega \in \Omega} \mathbf{V}_\omega$
$q_i^p$	Passenger load at node $i \in \bigcup_{\omega \in \Omega} \mathbf{V}_\omega$
$q_i^f$	Parcel load at node $i \in \bigcup_{\omega \in \Omega} \mathbf{V}_\omega$
$R_i$	Revenue for serving the passenger or parcel request at node $i \in \bigcup_{\omega \in \Omega} \mathbf{V}_\omega$
$P_i$	Penalty for denying service of the passenger or parcel at node $i \in \bigcup_{\omega \in \Omega} \mathbf{V}_\omega$
$t_{ij}$	Travel time between node $i$ and node $j$ , $\forall i, j \in \bigcup_{\omega \in \Omega} \mathbf{V}_\omega$
$\kappa_{ij}$	Travel cost incurred when moving from node $i$ to node $j$ , $\forall i, j \in \bigcup_{\omega \in \Omega} \mathbf{V}_\omega$
$Q_k^p$	Capacity of a vehicle $k \in \mathbf{K}$ to carry passengers
$Q_k^f$	Capacity of a vehicle $k \in \mathbf{K}$ to carry parcels

---

---

*Decision variables*

---

$z_i$	Binary decision variable indicating whether request $i$ is served
$y_{ij}^k$	Binary decision variable indicating whether vehicle $k$ travels from node $i$ to node $j$ directly
$\tau_i^k$	Time point at which vehicle $k$ initiates service at node $i$
$r_i^{pk}$	Passenger load in vehicle $k$ after serving at node $i$
$r_i^{fk}$	Parcel load in vehicle $k$ after serving at node $i$
$h_i$	Excess ride duration of passenger $i$ on-board of the vehicle that serves him/her
$c_i$	Compensation received by the passenger $i$
$\mathbf{x}$	Compensation design corresponding to each breakpoint of excess ride time interval in the compensation design

---

*Parameters in ALNS-CSA algorithm*

---

$\Psi^\omega$	The DPV solution of scenario $\omega \in \Omega$
$\Pi^\omega$	Set of all the passenger trips in the DPV solution $\Psi^\omega$ in scenario $\omega \in \Omega$
$\Delta h_i$	The difference between the real AERD under the received compensation and the ERD of passenger $i$
$\Upsilon_1$	Set of passenger trips with $\Delta h_i < 0$
$\Upsilon_2$	Set of passenger trips with $\Delta h_i = 0$
$\Upsilon_3$	Set of passenger trips with $\Delta h_i > 0$
$\underline{c}_i$	The threshold of the minimum required compensation of passenger $i$ which is computed by the inverse function of the passenger's AERD profile
$\Delta c_i$	The difference between $\underline{c}_i$ and the compensation received by passenger $i$
$h^{m_i}$	The specific ERD breakpoint that indicates the value of $h_i$ falls in an ERD range such that $h_i \in (h^{m_i-1}, h^{m_i}]$
$i^*$	The critical passenger identified in each iteration used to implement the increase or decrease adjustments
$h^{\hat{m}}$	The ERD breakpoint $h^{\hat{m}}$ with the index $\hat{m}$ of set to be the largest $m_i$ of the exactly-compensated passengers in set $\Upsilon_2$ , namely, $\hat{m} \leftarrow \max_{i \in \Upsilon_2} \{m_i\}$

---

## **Chapter 5 Simulation-Based Optimization of Public Transit Line Design Considering Bike-Sharing Integration**

This chapter addresses the PTLP (public transit line planning) problem of a single bus route with two directions integrated with the bike-sharing feeder services under an uncertain environment. The goal is to determine the optimal cost-efficient bus stop location and service frequency by minimizing the total system cost comprised of both passenger and operator aspects in the context of the public transit line with integrated shared bikes and walking as feeder modes. The complexity of the PTLP problem is heightened by the disaggregate behaviors and interactions of various entities, coupled with stochastic elements and inherent nonlinearities, which are challenging to tackle in the analytical models. Consequently, a simulation-based optimization framework is established for PTLP problem, utilizing a specially designed MABS (multi-agent-based simulation) system for the public transit line to measure system performance and derive total system cost. To address the computational challenge of the black-box simulation-based PTLP problem, an SBO (surrogate-based optimization) solution method is developed. This method iteratively approximates the response surface that maps public transit line planning input to the total system cost output to obtain high-quality solutions by a few simulation evaluations. Numerical experiments on a tested bus route and a real-world case to evaluate the efficacy of our proposed SBO method. The impact analysis is also carried out to explore the effects of passenger demand and bus capacity on system performance.

The remainder of this chapter is organized as follows. Section 5.1 introduces the problem statement and optimization framework. An MABS system for the public transit line is developed in Section 5.2. Section 5.3 describes the proposed SBO solution method. Numerical experiments are conducted in Section 5.4. Section 5.5 presents the conclusions and future research directions. Section 5.6 offers a detailed overview of the notations used in this chapter for readability.

## 5.1 Problem Statement and Optimization Framework

### 5.1.1 Problem statement

Let us consider a single bus route in a corridor operated by a public transit operator using a homogenous bus fleet in an urban area to provide transit services for passenger demand according to the daily service frequency over the operational period  $[0, T]$ . We assume that the route has two directions, denoted by  $\mathcal{R}$ , to cope with asymmetrical geographical and traffic conditions. The bus route is operated in both directions between two fixed bus terminals, denoted by  $\mathcal{S}$ . Each bus terminal  $s \in \mathcal{S}$  functions as both the starting and ending point (i.e., bus terminal) in the route, depending on the traveling direction. Each direction  $r \in \mathcal{R}$  expands for a length of  $L$  and consists of a series of bus stops. Let  $\mathcal{I}$  denote all the intermediate stops and  $\mathcal{I}^r \in \mathcal{I}$  denote the set of stops comprised in route  $r$ . Each bus stop  $i \in \mathcal{I}^r$  is associated with a location  $x_i$ . The distance between the deployed bus stops, i.e., the stop spacing, should fall in the interval  $[\Delta^{\min}, \Delta^{\max}]$ . The minimum and maximum bus stop numbers for both directions are  $I^{\min}$  and  $I^{\max}$ , respectively. We assume that, given these limitation constraints, a bus stop could be deployed at any location along the corridor in the designated direction. The typical one-day operation horizon is divided into several periods, denoted by set  $\mathcal{M}$ . The bus service frequency for departures varies throughout different periods to accommodate the fluctuating passenger demand as observed in actual scenarios. Let  $h_{mr}$  denote the service frequency for period  $m \in \mathcal{M}$  in each direction  $r$  of the route.

Let  $\mathcal{P}$  and  $\mathcal{B}$  denote the set of passengers and available buses respectively. All the passengers will arrive dynamically over time and space during the planning horizon  $[0, T]$ . Particularly, within each time period, passenger arrivals are represented using a Poisson process, which accounts for different demand rates at each bus stop within the catchment area. As previously noted, the passenger demand will be modeled on an individual basis for the better representation of passenger traveling behaviors. Each

passenger  $p \in \mathcal{P}$  is associated with the known origin, destination, boarding stop and alighting stop. They will experience the entire journey, starting from the origin and proceeding through the boarding stop, boarding the bus, traveling within the vehicle, alighting the bus, and finally completing the journey from the alighting stop to the destination. Considering the penetration of bike-sharing in urban mobility services, we assume that both walking and shared-bike modes can be used in conjunction with public transit (i.e., the first-mile travel from origin to boarding bus stop or the last-mile travel from alighting bus stop to destination). The mode choice is assumed to be determined based on the acceptable first or last-mile distance as well as the time sensitivity for each passenger. In addition, the traveling speed is largely affected by factors like weather, traffic and road conditions. Therefore, all these considerations bring about uncertainties in passenger demand. Notably, we consider a fixed demand scenario, where passengers do not cancel their requests for the public transit mode and switch to other modes of transportation.

All the buses will be located at two terminals at the start of the entire planning period  $[0, T]$ . Each bus  $b \in \mathcal{B}$  will be dispatched to take several bus trips based on the service frequency, serving all the intermediate bus stops on this trip. The bus dwell times at bus stops will be dynamically determined based on real-time passenger boarding and alighting considering loading and unloading durations. Thus, bus stop-to-stop travel time and dwell time at stops are critical factors influencing the travel time of each bus trip. Drivers exhibit varying preferences in operating the bus, involving different phases of motion such as acceleration, maintaining constant speed, coasting, and braking during stop-to-stop movements. Furthermore, complex urban conditions along the corridor will actually contribute to fluctuating bus traveling speeds during transit. All these factors will bring about uncertainties in each bus trip travel time. Note that once a bus completes a trip, it will prepare at the terminal and wait to be dispatched for the next trip in the opposite direction of the bus route.

Given the passenger demand information, the PTLP problem aims to simultaneously determine the optimal bus stop location deployment and service frequency solution in pursuit of minimizing the total system expenses, which are associated with service operator expenses and those incurred by passengers. This problem incorporates the dynamics and stochasticity of the passenger demand and the bus movement with corresponding realistic constraints. More specifically, both the passenger traveling behavior considering the varying traveling speed during the first- and last-mile journeys, and the unpredictable bus motion phases and travel speed throughout bus trips are precisely described. The real-time fashion, complex uncertainty and inherent realistic constraints facilitate the development of a more robust public transit line plan. However, these considerations also present significant challenges in building optimization models and designing algorithms to address the PTLP problem effectively.

### 5.1.2 Optimization framework

To address these real-time dynamics and stochastic complexities, we propose a conceptual optimization framework for the PTLP problem considering uncertainty. To mathematically set up the modeling framework, let  $\mathbf{x}$  denote the decision vector of all the intermediate stop locations with each element  $x_i, \forall i \in \mathcal{I}^r, r \in \mathcal{R}$ , referring to the specific location of the bus stop  $i$ ; let  $\mathbf{h}$  denote the decision vector of the service frequency with each element  $h_{mr}, \forall m \in \mathcal{M}, r \in \mathcal{R}$ , representing the specific service frequency of the route direction  $r$  within the operational period  $m$ . In addition, for ease of description, all the uncertain data arising from passengers and buses are consolidated into  $\xi$ , which contain all the uncertain elements relevant to the problem. Note that  $\xi$  represents a random vector with a known distribution  $\mathbb{Z}$ . This distribution corresponds to the joint distribution of the complex mixture distributions derived from the uncertainties in passenger travel and bus movement, as previously

discussed. Therefore, the optimization framework of the PTLP problem can be represented by:

**[PTLP]**

$$\min_{\{\mathbf{x}, \mathbf{h}\}} \{f(\mathbf{x}, \mathbf{h}) = \mathbb{E}_{\mathbb{Z}}[F(\mathbf{x}, \mathbf{h}, \xi)]\} \quad (5.1)$$

subject to

$$\mathbf{x}^l \leq \mathbf{x} \leq \mathbf{x}^u \quad (5.2)$$

$$\mathbf{h}^l \leq \mathbf{h} \leq \mathbf{h}^u \quad (5.3)$$

$$\Delta^{\min} \leq x_{i+1} - x_i \leq \Delta^{\max}, \quad \forall i \in \mathcal{I}^r, r \in \mathcal{R} \quad (5.4)$$

$$I^{\min} \leq |\mathcal{I}^r| \leq I^{\max}, \quad \forall r \in \mathcal{R} \quad (5.5)$$

$$g(\mathbf{x}, \mathbf{h}, \xi) \leq 0 \quad (5.6)$$

The objective function defined in Eq. (5.1) seeks the overall system cost minimization. The objective value  $f$  is expressed as an expectation of the stochastic public transit line system performance measure, i.e., the total system cost  $F$ , which is determined by the decision vectors  $(\mathbf{x}, \mathbf{h})$  and realizations of the exogenous uncertain data  $\xi$  under the distribution  $\mathbb{Z}$ . Constraint (5.2) defines the interval constraint of the bus stop location decisions, where  $\mathbf{x}^l(\mathbf{x}^u)$  denotes the lower (upper) bounds of  $\mathbf{x}$ . Constraint (5.3) defines the interval constraint of the service frequency decisions, where  $\mathbf{h}^l(\mathbf{h}^u)$  denotes the lower (upper) bounds of  $\mathbf{h}$ . Constraints (5.4) and (5.5) ensure the stop-spacing and stop number restriction respectively. Constraint (5.6) summarizes other realistic constraint sets of decision vectors in the transit system.

Given the complexities of the public transit line system, micro-simulators can be utilized to estimate the expected total system cost output by incorporating individual-specific characteristics and practical factors, which are extremely hard to formulate in an analytical model. As such, the proposed [PTLP] model essentially functions as a

simulation-based optimization modeling framework. Specifically, the microscopic agent-based simulator enables the proposed model to embed the most detailed behavioral models that describe how agents make travel decisions to provide a detailed representation of the dynamic and fine-grained passenger traveling and bus movement in a real-world transit system. Additionally, the complex stochasticity of the PTLP problem can be efficiently handled by replicating the evaluation of objective functions in different scenarios. Then, the objective function value associated with the public transit line design can be set as the average of all evaluations. To be more specific, in order to handle the uncertain information captured by  $\xi$ , we adopt the Monte Carlo sampling method. Every simulation run can lead to multiple realizations of  $F$  and involve sampling from the numerous probability distributions that account for uncertainty in demand generation and agent behaviors (i.e., passenger traveling and bus movement). For a given bus transit line design  $(\mathbf{x}, \mathbf{h})$ , assuming that we have observed  $\Omega$  independent realizations of  $F$ , denoted by  $F_1(\mathbf{x}, \mathbf{h}, \xi_1), \dots, F_\Omega(\mathbf{x}, \mathbf{h}, \xi_\Omega)$ . Then, the objective function  $f(\mathbf{x}, \mathbf{h})$  can be estimated using the average of these samples, i.e.,  $\hat{f}(\mathbf{x}, \mathbf{h}) = \frac{1}{\Omega} \sum_{i=1}^{\Omega} F_i(\mathbf{x}, \mathbf{h}, \xi_i)$ . In this approach, the sample size  $\Omega$  remains consistent for all the public transit line planning solution evaluations.

As we can see, the various disaggregate models embedded within the simulator can yield an explicit description of the public transit operation system and provide accurate estimates of the total system cost. Nevertheless, addressing the simulation-based black-box PTLP model remains a challenging task. For given  $(\mathbf{x}, \mathbf{h})$ ,  $f(\mathbf{x}, \mathbf{h})$  is measured by  $\hat{f}(\mathbf{x}, \mathbf{h})$  using micro-simulators. According to the earlier discussion,  $f(\mathbf{x}, \mathbf{h})$  is estimated by incorporating interactions between the buses and all passengers traveling within the dynamic transit system, while also addressing complex uncertainty and realistic conditions. Therefore, the mapping relationship between the  $f(\mathbf{x}, \mathbf{h})$  and  $(\mathbf{x}, \mathbf{h})$  are inexplicit and highly non-linear without closed-form expressions, which

makes the objective function of the PTLP problem non-differentiable, non-convex and intractable to obtain the global minimum. Additionally, evaluating a potential public transit line design solution accurately is computationally expensive due to the numerous replications of running the simulation and it is unrealistic to enumerate all possible solutions. In the subsequent two sections, we will first offer a comprehensive explanation on the MABS system for the public transit line in Section 5.2. Then, in Section 5.3, we will address the aforementioned difficulties by developing a customized solution method to efficiently solve the proposed simulation-based PTLP model.

## **5.2 MABS System for Public Transit Line**

This section develops the design of an MABS system for the public transit line, aimed at providing an estimated expectation for the total system cost. The agent-based simulation is a powerful approach for understanding complex systems through the operations and interactions among various agents, typically governed by decision-making rules (Hatzenbühler et al., 2020; Shen et al., 2018; Wen et al., 2018). It offers explanatory insights into the emergent collective properties of the overall system, which are not simply the sums of individual actions but are more complex and interconnected phenomena at a higher-level or macro-scale (Rieser, 2010). This methodology is particularly suitable for simulating systems like public transit lines. Agents can represent various entities such as passengers, buses, and bus stops, and the behavior of each component contributes to the overall system dynamics. Specifically, the travel mode choice of passengers in first-/last-mile journeys, the phases of motion decision of bus drivers, and the real-time bus operation control can be effectively and flexibly modeled by setting rules on the activities of involved agents. Thus, some realistic considerations that are different to address in the analytical models can be incorporated such as the bus fleet dispatch process, stop-to-stop movement process, dynamic passenger demand arriving process, and passenger traveling process. Through a multi-agent-based public transit line simulation system, we can observe the overall system

performance that arises from the behaviors and interactions of lower-level agents that might not be apparent through traditional modeling approaches. This allows for evaluating the total system cost with more accuracy and authenticity, thus providing better guidance in the public transit line designs. In what follows, we will first introduce the modeling of passenger travel behavior and bus motion by explicitly incorporating real-life features and the total system cost formulation considering both passenger and operator aspects. Then, we will develop an agent-based public transit line simulation system consisting of multiple interactive agents. The passenger and bus agent behaviors and interactions will be elaborated, followed by the explanation of primary principles for passenger arrival and bus dispatch process, and the introduction to the overall simulation workflow.

### 5.2.1 Passenger travel behavior, bus motion and cost function formulation

As previously mentioned, bus movement and passenger travel as well as some realistic constraints cannot be explicitly considered in the analytical optimization models but can be easily demonstrated using the simulation method. In this subsection, we begin by discussing the passenger travel behavior and bus motion models embedded within the simulator in Subsection 5.2.1.1 and then elaborate on the detailed total system cost function formulation in Subsection 5.2.1.2.

#### Assumptions and notations

Each passenger  $p \in \mathcal{P}$  is described by a tuple  $(x_p^o, x_p^d, i_p^o, i_p^d, e_p^{walk}, e_p^{bike}, \tilde{v}_p^a, \tilde{v}_p^e)$ , where  $x_p^o$  denotes the predetermined origin,  $x_p^d$  denotes the predetermined destination,  $i_p^b$  stands for the boarding stop,  $i_p^a$  denotes the alighting stop,  $e_{walk}^p$  and  $e_{bike}^p$  denote the parameter indicating the personal preference of choosing walking and shared bikes respectively during the first- and last-mile journey,  $\tilde{v}_p^a$  denotes the speed

of accessing boarding bus stops from the origin, and  $\tilde{v}_p^e$  denotes the speed of egressing bus stop to the destination. Note that passengers would like to select the closest bus stop to their origins (destinations) for boarding (or alighting).

In addition to these individual-specific parameters, let  $\beta$  and  $\alpha$  denote the boarding and the alighting time of each passenger. We assume that the speed of walking and using a shared bike, denoted by  $\tilde{v}_p^{walk}$  and  $\tilde{v}_p^{bike}$ , are random variables following certain distributions  $\mathbb{P}_1$  and  $\mathbb{P}_2$  with the probability density functions  $f_1(v)$  on an interval  $[\underline{v}_p^{walk}, \bar{v}_p^{walk}]$  and  $f_2(v)$  on an interval  $[\underline{v}_p^{bike}, \bar{v}_p^{bike}]$  respectively. Therefore, the passenger accessing and egressing speed, i.e.,  $\tilde{v}_p^a$  and  $\tilde{v}_p^e$ , exhibit inherent variability. This randomness is determined by the speeds of different travel modes and the preference (i.e., probability) that passengers will select each mode. This will be illustrated in detail in Subsection 5.2.1.1.

As for the bus operation, the bus fleet  $\mathcal{B}$  will carry out a number of bus trip tasks over the operational period, and all the bus trips (departures) with respect to the service frequency are organized into set  $\mathcal{K}$ . Each bus trip  $k \in \mathcal{K}$  is associated with the departure terminal  $s_k^d \in \mathcal{S}$ , the arrival terminal  $s_k^a \in \mathcal{S}$ , the direction of the route  $r_k \in \mathcal{R}$ , the departure time  $t_k^d$  from the departure terminal of the trip, the estimated arrival time  $\tilde{t}_k^a$  and the trip travel time  $\tilde{t}_k$  that can be computed by  $\tilde{t}_k^a - t_k^d$ . Each bus trip will be taken by a bus  $b \in \mathcal{B}$ , the assigned bus will be dispatched from the predetermined departure terminal, serve all the intermediate bus stops associated in this trip subject to its carrying capacity  $W$ , and end at the predetermined arrival terminal. For ease of presentation, we arrange the index of bus stop locations for each bus trip  $k$  following the sequence of  $s_k^d$ ,  $\mathcal{I}^{r_k}$  and  $s_k^a$  to represent the start bus terminal, intermediate bus stops and end bus terminal. For bus  $b$  on trip  $k$ , let  $a_{ki}^b$  and  $d_{ki}^b$

represent the arrival and departure time at stop  $i$ . Particularly, the stop-to-stop movement can be explicitly modeled as a cycle of a series of phases of motion including acceleration, constant speed, coasting and braking (Vuchic, 2007). Let  $\hat{\partial}_{acc}$ ,  $\hat{\partial}_{coa}$ , and  $\hat{\partial}_{bre}$  denote the acceleration, deceleration in coasting and deceleration in braking during the different phases respectively. In practice, the bus traveling speed between stops will be significantly affected by various factors like road conditions, passenger demand and weather affecting the bus movement between bus stops. Therefore, we assume that the constant travel speed for the stop-to-stop motion from stop  $i$  to the successive stop  $i+1$ , represented by  $\tilde{v}_{ki}^b$ , follows a certain distribution  $\mathbb{P}_3$  characterized by a probability function  $f_3(v)$  defined over an interval  $[\underline{v}_b, \bar{v}_b]$ . The corresponding stop-to-stop travel time is denoted by  $\Delta \tilde{t}_{ki}^b$ , which is a random variable affected by the stochastic bus traveling speed and phases of motion in the bus operation.

Additionally, each stop-to-stop motion during the trip will be followed by a dwelling process at the next arriving stop to serve the passengers who will get on and off the bus, and the dwell time at stop  $i$  is represented by  $\Delta \tilde{w}_{ki}^b$ . The number of boarding passengers at stop  $i$  is denoted by  $B_{ki}^b$ , while the number of alighting passengers at stop  $i$  is denoted by  $A_{ki}^b$  respectively. After the dwelling process, we also assume that the holding control strategy is implemented for each bus before starting its next stop-to-stop motion, possibly leading to an additional holding time  $\Delta \tilde{g}_{ki}^b$  at stop  $i$ . Notably, both the stop-to-stop movement and the dwelling process considering the potential holding strategy bring complex stochasticity to the total travel time of each bus trip  $\tilde{t}_k$ . Furthermore, after completing the services of a bus trip, the bus will be prepared at the terminal with a minimum driver shifting time  $\varpi$ , ready to be assigned for the next trip in the opposite direction.

For other model parameters associated with the cost function formulation, the unit time cost coefficients of passenger walking, in-vehicle, and waiting times are denoted by  $\theta^a$ ,  $\theta^v$  and  $\theta^w$ , respectively. As for bus operation, the unit distance cost at cruising speed, unit time cost in acceleration and deceleration, and unit time cost of the vehicle dwelling and holding at stops are represented by  $\theta^t$ ,  $\theta^l$  and  $\theta^s$ , respectively. The unit time salary for each bus driver and investment of each bus stop are denoted by  $\theta^h$  and  $\theta^c$  respectively. Furthermore, the fixed construction cost and maintenance cost for each bus stop are denoted by  $\theta^c$  and  $\theta^m$  respectively.

### 5.2.1.1 Passenger travel and bus motion

#### Passenger travel behavior model

To facilitate more realistic passenger traveling behaviors within the bus operation system where the choices between walking and shared bikes during the first- and last-mile journey are considered, we will introduce a probabilistic model incorporating utility-based decision-making with the randomness of speed. Specifically, we assume that the selection of either mode by a passenger depends on the travel distance, the monetary cost associated with shared bikes, and personal preferences which influence their respective utilities. Therefore, the utility functions of each mode for passenger  $p$  can be calculated based on these factors as follows:

$$U_{walk} = -\chi_1 \cdot d + \chi_3 \cdot e_{walk}^p \quad (5.7)$$

$$U_{bike} = -\chi_1 \cdot d - \chi_2 \cdot c_{bike} + \chi_3 \cdot e_{bike}^p \quad (5.8)$$

where  $\chi_1$ ,  $\chi_2$  and  $\chi_3$  are coefficients that quantify the sensitivity to travel distance of different modes, monetary cost and personal preference respectively,  $d$  is the travel distance of the first- and last-mile journey, and  $c_{bike}$  is the additional monetary cost of choosing the mode of shared bikes. Note that the travel distance for the first-mile

journey is determined by the origin  $x_p^o$  to the boarding stop  $i_p^o$ , while for the last-mile journey, it will be calculated from the alighting stop  $i_p^d$  to the destination  $x_p^d$ .

The walking or shared-bike mode selection probability can be determined using the logistic regression model, which is typical in discrete choice theory, as detailed below:

$$\Pr(v = \tilde{v}_p^{walk}) = \frac{e^{U_{walk}}}{e^{U_{walk}} + e^{U_{bike}}} \quad (5.9)$$

$$\Pr(v = \tilde{v}_p^{bike}) = \frac{e^{U_{bike}}}{e^{U_{walk}} + e^{U_{bike}}} \quad (5.10)$$

Either the access speed  $\tilde{v}_p^a$  or the egress speed  $\tilde{v}_p^e$  of passenger  $p$  will be described as a random variable, with its distribution influenced by the mode choice that is probabilistically determined. Given the mode choice, the distribution of  $\tilde{v}_p^a$  or  $\tilde{v}_p^e$  follows the different density functions according to the selected mode. Specifically, in practical applications, we can first determine the mode based on the computed probabilities  $\Pr(v = \tilde{v}_p^{walk})$  and  $\Pr(v = \tilde{v}_p^{bike})$ , and then perform a Bernoulli trial to simulate the mode choice. Once the mode is determined probabilistically, the corresponding speed for that mode is sampled from the specific speed distribution for that mode, i.e.,  $f_1(v)$  or  $f_2(v)$  for walking or biking respectively. Overall, this model effectively captures the inherent uncertainties and realistic decision-making process of individuals based on calculated utilities which are calculated by using external information.

### Bus motion model

The bus motion model involves the formulation of the departure time at bus stops, travel time between stops, arrival time at bus stops and dwell time, taking into account

the bus phases of motion and the passenger boarding and alighting behaviors.

The arrival time of bus  $b$  at stop  $i$  is determined by the departure time at the preceding stop  $i-1$  and the bus travel time from stop  $i-1$  to stop  $i$ . The departure time of bus  $b$  at given stop  $i$  is computed by summing the arrival time at the stop, the total dwell time and, potentially, an additional holding time. Therefore, for bus  $b$  taking bus trip  $k$ , the arrival time and departure time at stop  $i$  can be calculated as follows:

$$a_{ki}^b = d_{k,i-1}^b + \Delta \tilde{t}_{k,i-1}^b, \quad \forall i \in \{\mathcal{I}^k, s_k^a\} \quad (5.11)$$

$$d_{ki}^b = a_{ki}^b + \Delta \tilde{w}_{ki}^b + \Delta \tilde{g}_{ki}^b, \quad \forall i \in \mathcal{I}^k \quad (5.12)$$

where the departure time of bus from the start terminal  $s_k^d$  depends on the departure time of the trip  $t_k^d$ .

The headway-based holding strategy adopted by Wu et al. (2017) is considered in the bus operation to mitigate bunching. Specifically, the bus will remain at the stop for a specified duration based on the inter-departure headway deviation. During a given service period  $m$ , the holding time of bus  $b$  taking trip  $k$  at stop  $i$  can be defined as follows:

$$\Delta \tilde{g}_{ki}^b = \min \left\{ \max \left[ 0, \delta h_{m_k} - (\bar{d}_{ki}^b - d_{k'i}^{b'}) \right], \Delta g_{\max} \right\}, \quad \forall i \in \mathcal{I}^k \quad (5.13)$$

where  $\bar{d}_{ki}^b$  stands for the departure time in the absence of a holding strategy,  $d_{k'i}^{b'}$  denotes the departure time of the previously adjacent bus  $b'$  in trip  $k'$  and can be derived by  $d_{k'i}^{b'} = \bar{d}_{k'i}^{b'} + \Delta \tilde{g}_{k'i}^{b'}$ , and  $\delta h_{m_k}$  denotes the minimum allowable headway with parameter  $0 < \delta \leq 1$  representing the ratio to determine this headway threshold for the holding criterion. Additionally,  $\Delta g_{\max}$  denotes the predetermined holding time threshold designed to avoid excessively prolonged holding times and the potentially

consequent domino effect.

The bus dwell time is dependent on the volume of passengers boarding and alighting while taking into account the constraints of bus capacity. Additionally, we assume that boarding occurs through the front door of the bus, whereas the back door is reserved for passengers alighting. Therefore, the dwell time of bus  $b$  in trip  $k$  at stop  $i$  can be calculated by:

$$\Delta\tilde{w}_{ki}^b = \max\{\beta B_{ki}^b, \alpha A_{ki}^b\}, \quad \forall i \in \mathcal{I}^k \quad (5.14)$$

Here, it is worth noting that the successfully boarded passenger number, i.e.,  $B_{ki}^b$ , is likely to be less than the total number of waiting passengers accumulated during the inter-service frequency between the departure time at stop  $i$  and the immediately preceding bus.

As previously mentioned, each stop-to-stop movement involves several phases: acceleration, cruising at a constant speed, coasting, and braking. The combination of these phases will be significantly influenced by the distance between stops and the driver's handling of the vehicle. Specifically, whether a bus can reach the cruising speed  $\tilde{v}_{ki}^b$  will depend on the stop-to-stop distance. We define a *critical distance*  $l_c$  between stops, derived under the assumption that the bus reaches its maximum cruising speed  $\tilde{v}_{ki}^b$ , accounting for only acceleration and braking. This critical distance is vital in determining the motion phases a bus undergoes between stops. If the actual distance between stops, i.e.,  $x_{i+1} - x_i$ , is smaller than  $l_c$ , the bus will primarily accelerate to its maximum attainable speed  $\tilde{v}_{ki}^{b'} (\leq \tilde{v}_{ki}^b)$  before shifting to the braking phase as it prepares to stop (see Figure 5.1 (a)). Otherwise, if the distance exceeds  $l_c$ , the motion of the bus will possibly include an additional cruising phase at a constant speed and/or a coasting phase. These scenarios manifest in four distinct cases (see Figure 5.1 (b)–

(d): (b) accelerating to reach the maximum cruising speed  $\tilde{v}_{ki}^b$  and maintaining it until braking; (c) accelerating to reach the maximum cruising speed  $\tilde{v}_{ki}^b$ , followed by a coasting phase to decelerate to speed  $\tilde{v}_{ki}^{b''}$ , and then braking; (d) accelerating to reach the maximum cruising speed  $\tilde{v}_{ki}^b$ , traveling at the speed, coasting to speed  $\tilde{v}_{ki}^{b''}$ , and then braking. Note that for the case  $x_{i+1} - x_i > l_c$ , the possible combinations will be randomly applied handling by the driver in practice. Therefore, the stop-to-stop motion will further add to the uncertainty in bus stop-to-stop travel time, thus significantly affecting the total bus trip time due to cumulative effects.

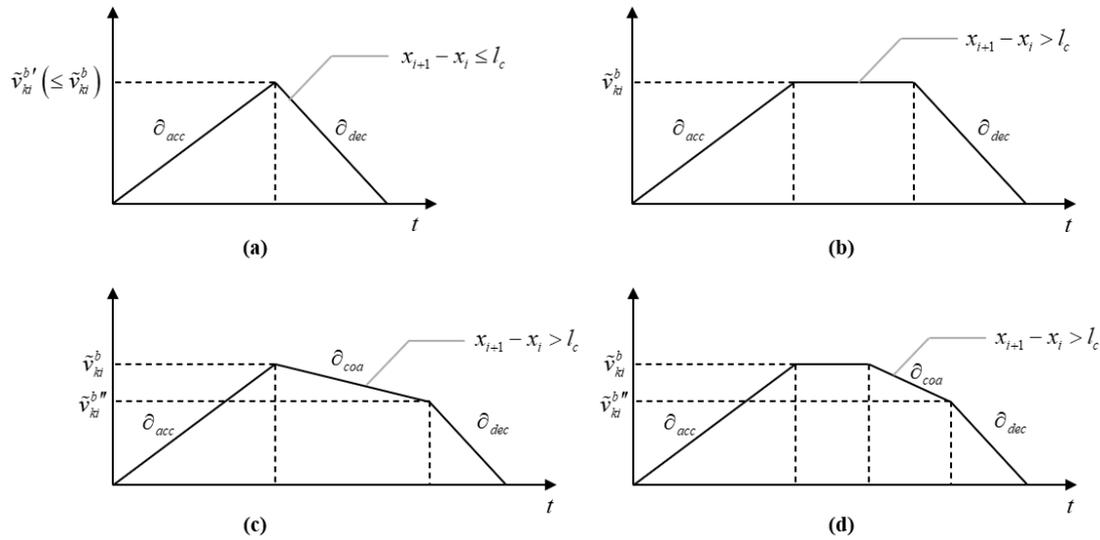


Figure 5.1. Different bus stop-to-stop movement patterns

### 5.2.1.2 Cost function formulation

The total system cost is composed of two primary components: (1) the passenger cost and (2) the operator cost. Specifically, the passenger cost involves the access and egress cost associated with the walking process between the origins and the boarding stops as well as from the alighting stops to the final destinations, waiting time costs incurred by passengers waiting at these bus stops, and the time cost associated with passengers traveling within the vehicle. The operator cost includes both operational

expenditures for the bus fleet throughout the study period and fixed expenses for bus stop construction and maintenance.

### Passenger cost

The total passenger cost ( $C_{TP}$ ) is comprised of three primary elements: the cost of passenger access and egress ( $C_a$ ), the cost associated with passenger waiting time ( $C_w$ ) and the cost related to time spent in-vehicle ( $C_v$ ).

$$C_{TP} = C_a + C_w + C_v \quad (5.15)$$

where each cost component can be derived from the amount of associated time of all passengers during the operational period by considering their traveling process.

For each passenger, the accessing time stands for the time spent during the first-mile journey traveling between the origin and the boarding stop, while the egressing time represents the time spent during the last-mile journey traveling from the alighting stop to the final destination. The waiting time indicates the duration from the instant of arriving at the boarding bus stop to the time instant when the passenger can successfully board the bus. The in-vehicle time is the period from the moment a passenger boards the bus to the time they get off. For the sake of clarity, let  $\tau_p^{fm}$ ,  $\tau_p^{lm}$ ,  $\tau_p^w$  and  $\tau_p^v$  represent the accessing, egressing, waiting and in-vehicle time of passenger  $p$ . Then, the three cost components can be expressed as follows:

$$C_a = \theta^a \sum_{p \in \mathcal{P}} (\tau_p^{fm} + \tau_p^{lm}) \quad (5.16)$$

$$C_w = \theta^w \sum_{p \in \mathcal{P}} \tau_p^w \quad (5.17)$$

$$C_v = \theta^v \sum_{p \in \mathcal{P}} \tau_p^v \quad (5.18)$$

### Operator cost

The total operator cost ( $C_{TO}$ ) comprises two components: the vehicle operating cost ( $C_f$ ) and bus stop cost ( $C_s$ ):

$$C_{TO} = C_f + C_s \quad (5.19)$$

where the vehicle operating cost can be further decomposed into five components including fleet cost  $C_f^f$ , driver salary cost  $C_f^h$ , cruising time cost for bus fleet  $C_f^t$ , deceleration and acceleration time cost for bus fleet  $C_f^l$ , and dwell time cost at bus stops for bus fleet  $C_f^d$ .

The fleet cost is determined by the fixed unit cost for each bus and the total number of buses needed for effective operation in order to meet the service frequency across all the different time periods over the whole operational period:

$$C_f^f = \theta^b |\mathcal{B}| \quad (5.20)$$

The cost associated with the bus driver is determined by their fixed salary per unit time and all the bus trips to be completed over the operational period:

$$C_f^h = \theta^h \sum_{k \in \mathcal{K}} \tilde{t}_k \quad (5.21)$$

The bus cruising time cost can be computed based on the cumulated duration for all bus travels at the constant speed to take all the bus trips, while the acceleration and deceleration time cost is computed according to the cumulated duration for all the acceleration and deceleration motions. Similarly, the dwell time cost is calculated based on the cumulated dwell time for passenger loading and unloading and holding time at all bus stops. These three cost components can be represented as follows:

$$C_f^t = \theta^t \sum_{b \in \mathcal{B}} \sum_{k \in \mathcal{K}} \sum_{i \in \{s_k^d, \mathcal{I}^k\}} \Delta \tilde{t}(c)_{ki}^b \tilde{v}_{ki}^b \delta_k^b \quad (5.22)$$

$$C_f^l = \theta^l \sum_{b \in \mathcal{B}} \sum_{k \in \mathcal{K}} \sum_{i \in \{s_k^d, \mathcal{I}^k\}} \Delta \tilde{t}(l)_{ki}^b \delta_k^b \quad (5.23)$$

$$C_f^s = \theta^s \sum_{b \in \mathcal{B}} \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{I}^k} (\Delta \tilde{w}_{ki}^b + \Delta \tilde{g}_{ki}^b) \delta_k^b \quad (5.24)$$

where  $\Delta \tilde{t}(c)_{ki}^b$  denote the duration for the bus traveling at the constant speed and  $\Delta \tilde{t}(l)_{ki}^b$  denote the duration in acceleration and deceleration respectively during each bus stop-to-stop movement from stop  $i$  to stop  $i+1$  for bus  $b$  in trip  $k$ , and  $\delta_k^b$  denotes whether bus  $b$  is allocated to undertake trip  $k$  revealed in the simulation.

The stop deployment cost is determined by the expenses associated with fixed construction and maintenance, along with the overall deployed bus stop number:

$$C_s = (\theta^c + \theta^m) \cdot |\mathcal{I}| \quad (5.25)$$

Overall, the expensive-to-evaluate simulation-based objective function considering both passenger and operator perspectives can be expressed by:

$$\begin{aligned} F(\mathbf{x}, \mathbf{h}, \xi) = & \theta^a \sum_{r \in \mathcal{P}} (\tau_p^{fm} + \tau_p^{lm}) + \theta^w \sum_{r \in \mathcal{P}} \tau_p^w + \theta^v \sum_{r \in \mathcal{P}} \tau_p^v + \theta^b |\mathcal{B}| + \theta^h \sum_{k \in \mathcal{K}} \tilde{t}_k + \\ & \theta^t \sum_{b \in \mathcal{B}} \sum_{k \in \mathcal{K}} \sum_{i \in \{s_k^d, \mathcal{I}^k\}} \Delta \tilde{t}(c)_{ki}^b \tilde{v}_{ki}^b \delta_k^b + \theta^l \sum_{b \in \mathcal{B}} \sum_{k \in \mathcal{K}} \sum_{i \in \{s_k^d, \mathcal{I}^k\}} \Delta \tilde{t}(l)_{ki}^b \delta_k^b + \\ & \theta^s \sum_{b \in \mathcal{B}} \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{I}^k} (\Delta \tilde{w}_{ki}^b + \Delta \tilde{g}_{ki}^b) \delta_k^b + (\theta^c + \theta^m) \cdot |\mathcal{I}| \end{aligned} \quad (5.26)$$

## 5.2.2 Simulation system development

The previously discussed modeling details lay the foundation for establishing the multi-agent public transit line simulation system. The key for the simulation system development is to explicitly describe the dynamic behaviors and interactions of agents and the rules followed by them, which will be illustrated in Subsections 5.2.2.1 and 5.2.2.2, respectively.

### 5.2.2.1 *Agent behaviors and interactions*

#### Agents

The MABS system provides an explicit description of how discrete agents exhibit dynamic behaviors and interact with one another over time. Specifically, four types of agents constitute our proposed system and are introduced in detail as follows:

- **Passenger agent:** Each passenger agent simulates the behavior and interaction of a passenger. Each passenger agent engages in a series of activities to complete the intended travel journey by considering its current location and state.
- **Bus agent:** Each bus agent mimics the behavior and interaction of a bus (driver). Each bus agent is responsible for completing some bus trips according to the service frequency with compulsory shifting between two consecutive trips in different directions. When taking each bus trip, each bus engages in a cycle of activities to serve each bus stop considering its current location and state in driving, pausing, dwelling, and holding.
- **Bus stop agent:** Each passenger agent represents each bus stop. The bus stop agent is responsible for managing the queues for passengers waiting at locations of bus stops, identifying the arrived buses, and passing information to assist in the boarding and alighting process.
- **Terminal agent:** Each terminal agent acts like the central controller to dispatch proper buses to take bus trip tasks according to the departure time aligned with the current service frequency. If there is more than one bus at the bus terminal, the agent also needs to manage the waiting queue of buses preparing to take future trips tasks.

#### Behaviors and interactions between agents

The behaviors and interactions between different agents are realized based on the

information exchange and state update. The mutual relationship between different agents is described in Figure 5.2. The passengers arrive along the route according to the passenger arrival principle. Each newly arrived passenger agent enters the “accessing” state and chooses to walk or ride a shared bike to the boarding stop from the origin. Once arriving at the boarding stop, the passenger agent transfers to the “waiting” state and will be added to the waiting queue at the corresponding stop agent. Passengers will receive the information of the stopped bus shared by the bus stop agent to decide whether to board the bus. Once the bus agent dwells at the stop with available seats, the passenger agent will interact with the bus agent and change to the “boarding” state. Meanwhile, the bus stop agent will update the waiting queue by removing those passenger agents who have successfully boarded the bus. Onboard passenger agents receive the location information of the bus agent and will turn to “egressing” state once the bus arrives at their alighting bus stops. In the last state, each passenger agent also chooses to walk or ride a shared bike and will exit the system when reaching the destination.

The terminal agents constantly assign proper bus agents at the bus terminals to take bus trips based on bus dispatch strategy. Upon departure from the terminal, each bus agent enters the “driving” state and approaches the next bus stop. After completing the stop-to-stop movement, the bus agent searches for an available bay at the bus stop by the bus stop agent. The bus agent will enter the “pausing” state, waiting briefly for a bay to open if no bay is available. Otherwise, the bus agent will share its occupation information with the stop agent and initiate “dwelling” state either when there are passengers at the stop awaiting boarding or when passengers on the bus need to alight. The “dwelling” state will be maintained until all passengers have boarded or alighted. Before departing from the current stop, the bus might enter a holding state to ensure a scheduled headway as detailed in Eq. (5.13). This movement process of the bus agent is visualized in Figure 5.3. The bus agent that has finished a trip task will enter the “shifting” state at the terminal to meet the minimum time for the shift work and prepare

for the next dispatch. Note that the uncertainties of passenger travel and bus movement will be applied in the “driving” state of the bus agent and the “accessing” and “egressing” state of the passenger agents. The duration spent in each state for both passenger and bus agents is tracked by a timer in the simulation to control the transition through various states.

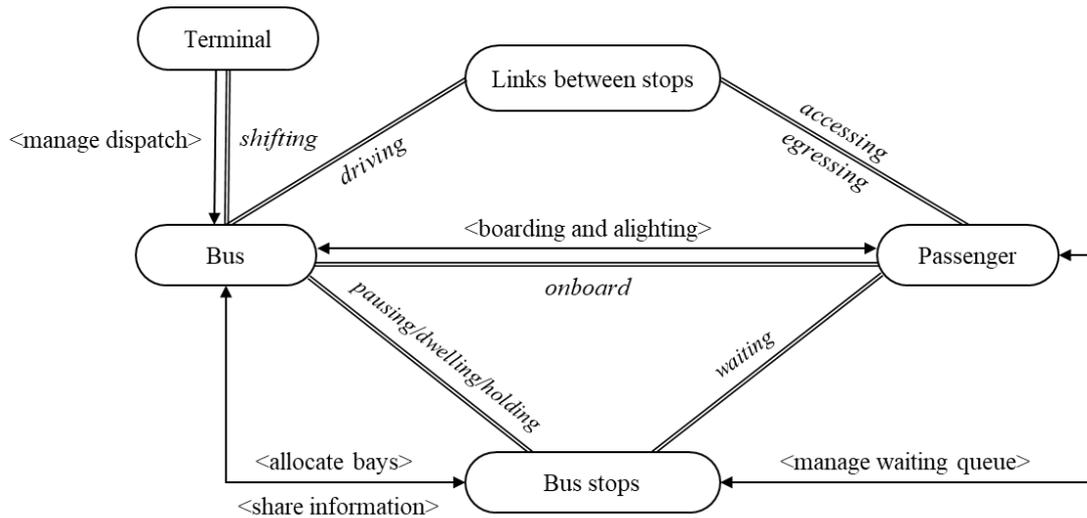


Figure 5.2. Relationship between different agents

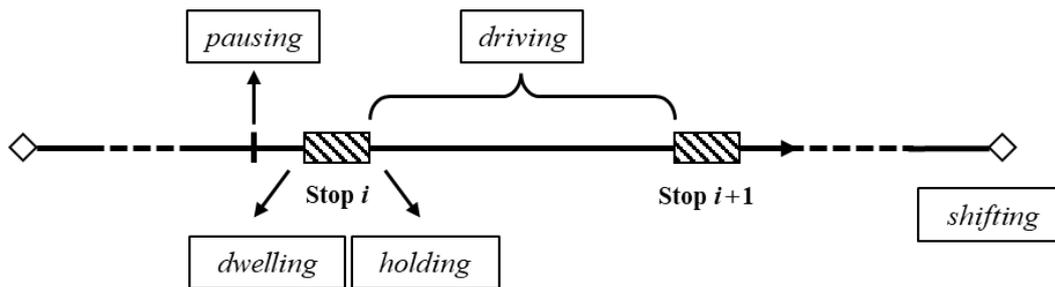


Figure 5.3. State transitions of bus agent

### 5.2.2.2 Principles and simulation workflow

#### Passenger arrival principle

The passenger arrival principle describes a detailed process for simulating passenger arrivals on a bus route, including the determination of their random origin and destination that reflect varying demand rates along different locations of the bus route. Assuming that we can obtain the cumulative boarding and alighting rate functions across the entire bus route in both directions. In the simulation of passenger arrivals, a Poisson process is utilized, incorporating the specified demand rates along the route to model the inflow of passengers. The pseudocode of the passenger arrival principle is provided in **Algorithm 5.1**. Specifically, let  $B_{mr}(x)$  and  $A_{mr}(x)$  denote cumulative boarding and alighting rate functions for each specified observation period  $m \in \mathcal{M}$  in direction  $r \in \mathcal{R}$  respectively. Let  $\lambda_{mr}$  denote the total passenger arrival rate along the route during the period  $m$  in direction  $r$ , which can be evaluated the cumulative boarding rate function at the end of the route. We initialize by creating an empty passenger set  $\mathcal{P}$ , starting from the start epoch of the operational period (see Line 1). In every simulation epoch  $t_{sim}$ , we can calculate the probability of a new passenger arrival and identify if a new passenger agent arrives using the random number generator according to the Poisson process principle, which is represented by the subfunction  $GenPas(\lambda_{mr})$  (see Lines 2–4). For each new passenger agent, we can determine the origin and destination (see Lines 5–7). Specifically, we first generate the origin  $x_p^o$  based on  $B_{mr}(x)$ , which can be achieved by setting a random number  $rnd_b$  within the interval  $[0,1]$  and solving  $B_{mr}(x) = rnd_b \times B_{mr}(L_r)$ . Similarly, we can determine the destination  $x_p^d$  based on  $A_{mr}(x)$  while ensuring the passenger will get off at one of the following bus stops. As such, the newly generated passenger with the determined origin and destination can then be added to the arrived passenger set (see Line 8).

---

**Algorithm 5.1. Pseudocode of the passenger arrival principle.**

---

```
1 Initialization: arrived passenger set  $\mathcal{P} \leftarrow \emptyset$ 
2 For  $t_{sim}$  from  $start\_epoch$  to  $end\_epoch$  do
3   Identify the specified observation period  $m$ 
4   If  $GenPas(\lambda_{mr})$ , then
5     Generate a new passenger  $p$ 
6     Determine the origin  $x_p^o$  based on  $B_{mr}(x)$ 
7     Determine the destination  $x_p^d$  based on  $A_{mr}(x)$ 
8     Append the newly generated  $p$  in set  $\mathcal{P}$ 
9   EndIf
10 EndFor
```

---

### Bus dispatch principle

The bus dispatch principle aims to assign a proper bus to take the bus trips given by  $\mathcal{K}$  considering the availability of the bus fleet. The basic principle can be presented as **Algorithm 5.2**. Specifically, we initialize by creating an empty bus fleet set  $\mathcal{B}$  and proceeding with the simulation in every simulation epoch  $t_{sim}$  to check bus trip information to decide whether to dispatch a bus (Lines 2–4). The candidate buses should be in the “shifting” state at the terminal that is consistent with the departure location of the concerned trip  $k$ . In addition, a bus is available only if the duration in the “shifting” state meets the minimum threshold  $\varpi$ . If there exist multiple available buses, the first-in-first-out (FIFO) strategy is utilized to determine which bus will undertake the trip (see Lines 5–6). If there is not any bus that can be dispatched, a new bus will be generated to take the trip and added to the bus fleet (see Line 7). We continue this process until the current simulation time reaches the end epoch for the operational period. Notably, the number of required buses  $|\mathcal{B}|$  can be determined during the bus dispatch procedure.

---

**Algorithm 5.2. Pseudocode of the bus dispatch principle.**

---

```
1 Initialization: bus fleet  $\mathcal{B} \leftarrow \emptyset$ 
2 For  $t_{sim}$  from start_epoch to end_epoch do
3   For  $k \in \mathcal{K}$  do
4     If  $t_{sim} == t_k^d$ , then
5       If  $\exists b \in \mathcal{B}$  available at bus terminal  $s_k^d$ , then
6         Select a bus  $b \in \mathcal{B}$  based on the FIFO strategy to take trip  $k$ 
7       Else Generate a new bus  $b$  to take trip  $k$ ; add  $b$  in set  $\mathcal{B}$ 
8       EndIf
9     EndIf
10  EndFor
11 EndFor
```

---

Computerized workflow of the MABS system

According to the interactive agents with behavior rules and introduced principles, we can establish the MABS system for public transit lines based on a comprehensive computerized workflow, as explicitly depicted in Figure 5.4. Given input data of the passenger demand and public transit line settings, the simulation will proceed by updating all the agents within the system at each simulation time epoch. New passenger agents are constantly inflowed into the system according to the passenger arrival principle. Based on the current simulation time, the terminal agents check the bus trip information and assign the buses to perform the trip tasks according to the bus dispatch principle. Meanwhile, the status of both bus agents and passenger agents will be updated while considering the mutual interactions with the bus stop agents. Note that the inflowed bus agents will be in recycled use while the passenger agents that have completed their individual trips will be expired and removed from the system. The simulation procedure will be terminated until reaching the time epoch representing the end of the operational period. System performance will be evaluated to derive the total system cost output.

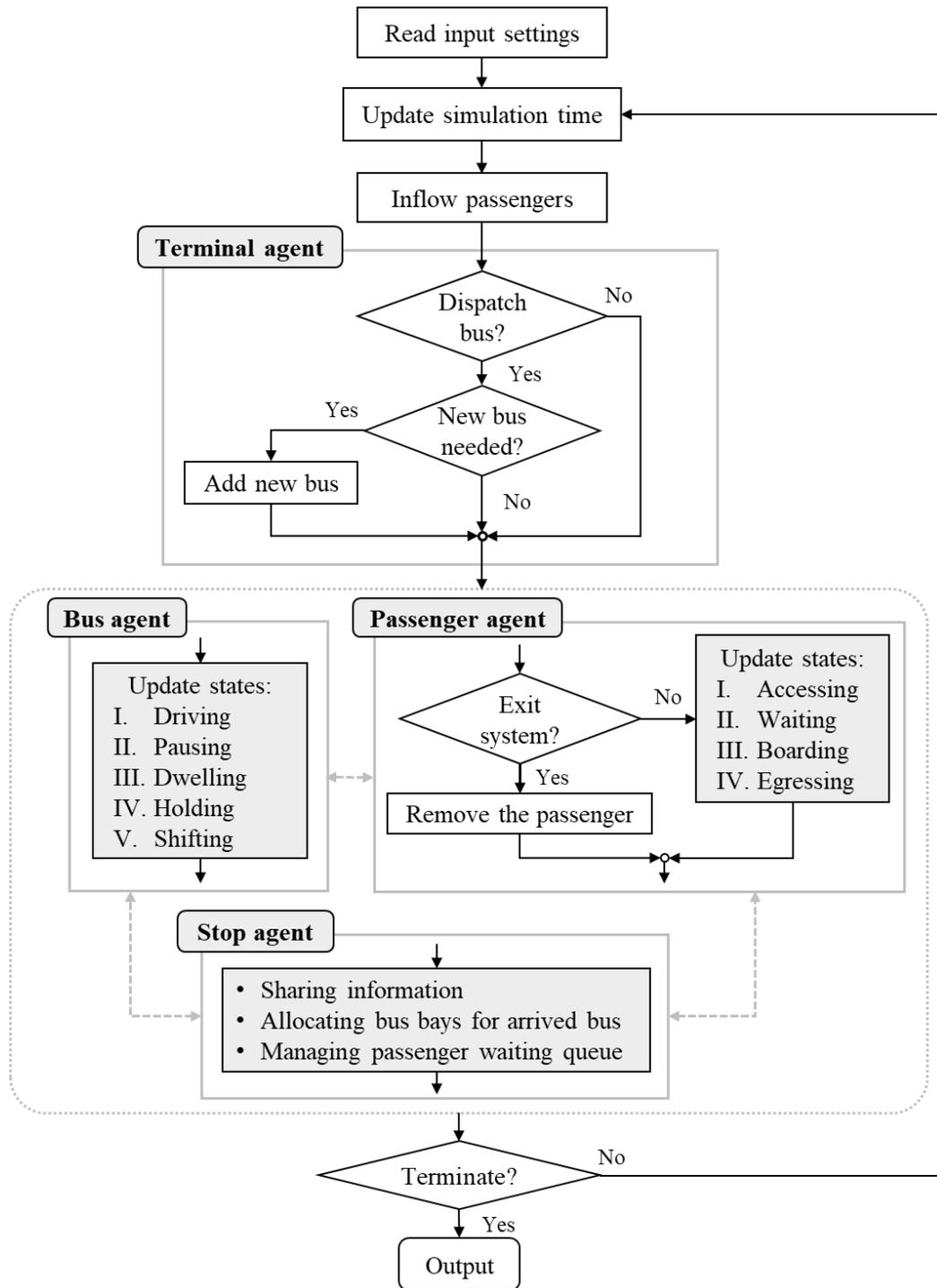


Figure 5.4. Workflow of the agent-based public transit line simulation system

### 5.3 Solution Method

Although the simulation model offers a comprehensive description of the disaggregate behaviors, inherent stochastic and nonlinear elements as well as realistic

constraints in the public transit operation system, the simulation-based optimization process poses significant challenges due to the totally implicit nature of the relationship between inputs and outputs. To address these challenges, we introduce a heuristic method, known as SBO, to address the computationally expensive simulation-based PTLP problem.

The SBO method aims to approximate a surrogate model that effectively maps the input–output relationship, thereby facilitating a more efficient search for optimal solutions with considerably reduced computational efforts (Bhosekar and Ierapetritou, 2018; Chen et al., 2014; Forrester and Keane, 2009). The proposed SBO method is developed based on the Bayesian optimization framework, which is designed as a sequential approach used to address the global optimization characterized by black-box functions or problems involving computationally expensive objective functions (Frazier, 2018; Jones et al., 1998). It consists of two primary components: a mathematically tractable surrogate model and an easily solvable acquisition function. The basic idea of the SBO method involves iteratively constructing (updating) the surrogate model based on currently sampled points and their corresponding function values, and then further generating new infill samples to be incorporated with the previous samples for further refinement of the surrogate model. To be more specific, the non-parametric Gaussian process (GP) is employed to construct the surrogate response surface, which is capable of iteratively allocating the evaluations leverages both prior knowledge and continuously updated posterior distributions derived from observed data. The process of identifying new, informatively significant samples involves optimizing a carefully designed acquisition function. This optimization is critical in its balance of the dual objectives of exploration—aiming to investigate less understood regions of space—and exploitation—focusing on regions promising high performance based on current model predictions.

The algorithmic framework of the proposed SBO method ensures that the PTLP problem can be solved by iteratively sampling new feasible sets. Each iteration requires

only the public transit line design input, i.e.,  $(\mathbf{x}, \mathbf{h})$ , and its associated total system cost output  $f(\mathbf{x}, \mathbf{h})$  evaluated by simulation to construct GP. To accommodate constraints on bus stop locations and service frequency, this method incorporates the specific interval constraints (i.e., constraints (5.2) – (5.4)) into the optimization of the acquisition function. The subsequent two subsections will provide a detailed explanation of the aforementioned procedures involved. Specifically, Subsection 5.3.1 will elaborate on the construction of the surrogate using the Gaussian process and Subsection 5.3.2 will discuss the selection of new points using the acquisition function respectively.

### 5.3.1 Gaussian process surrogate

The GP surrogate is developed through GP prediction (regression). Specifically, this begins with a prior function distribution and incorporates data from currently identified samples to transition into a GP posterior. The posterior represents an updated predictive distribution characterized by a mean function and a covariance function reflecting the observed data. Essentially, the GP surrogate in our proposed method can be designed to approximate the mapping relationship between decision variable input  $(\mathbf{x}, \mathbf{h})$  and simulation output  $\hat{f}(\mathbf{x}, \mathbf{h})$ . This is achieved by iteratively adding newly sampled points and corresponding objective function values, thereby providing a more accurate and refined estimate of the function with a quantifiable measure of uncertainty.

Consider that we have evaluated the objective function at  $n$  sample points. Let  $(\mathbf{x}, \mathbf{h})_i$  and  $\hat{f}_i$  denote the sampled point and the associated function value respectively of sample  $i$ ,  $\forall i \in \{1, 2, \dots, n\}$ . For GP regression, based on the sample set  $\{((\mathbf{x}, \mathbf{h})_1, \hat{f}_1), \dots, ((\mathbf{x}, \mathbf{h})_n, \hat{f}_n)\}$ , the output vector  $\mathbf{F}_n = [\hat{f}_1, \dots, \hat{f}_n]^T$  conforms to an  $n$ -dimensional multivariate Gaussian distribution defined by its mean vector and covariance matrix:

$$\mathbf{F}_n \sim \mathcal{N}(\boldsymbol{\mu}_n, \mathbf{K}_n) \quad (5.27)$$

where  $\boldsymbol{\mu}_n$  is an  $n \times 1$  column vector with each entry denoted by  $m((\mathbf{x}, \mathbf{h})_i)$ ,  $\forall i \in \{1, 2, \dots, n\}$ , representing the mean function, and  $\mathbf{K}_n$  is an  $n \times n$  matrix with each entry denoted by  $Cov(\hat{f}_i, \hat{f}_j)$ ,  $\forall i, j \in \{1, 2, \dots, n\}$ . Here, the mean function  $m(\cdot)$  is expected to describe the average or central tendency of the output values generated by the model across different inputs. For simplicity,  $\boldsymbol{\mu}_n$  is often assumed to be a zero vector since the expected value does not influence the optimization directly. The matrix  $Cov(\hat{f}_i, \hat{f}_j)$  represents the covariance (or similarity) between the function values at any two sample points  $(\mathbf{x}, \mathbf{h})_i$  and  $(\mathbf{x}, \mathbf{h})_j$ , which is measured by the kernel function  $\kappa(\cdot)$ . In our method, we employ the widely-used radial basis function (RBF) kernel which is expressed as:

$$Cov(\hat{f}_i, \hat{f}_j) = \kappa((\mathbf{x}, \mathbf{h})_i, (\mathbf{x}, \mathbf{h})_j) = \exp\left(-\frac{\|(\mathbf{x}, \mathbf{h})_i - (\mathbf{x}, \mathbf{h})_j\|_2^2}{2\ell^2}\right) \quad (5.28)$$

where  $\|(\mathbf{x}, \mathbf{h})_i - (\mathbf{x}, \mathbf{h})_j\|_2^2$  denotes the squared Euclidean distance between two sample vectors and  $\ell$  is recognized as the length scale of the kernel impacting the smoothness of the function. The parameter  $\ell$  can be tuned via the maximum likelihood estimation as shown in Eq. (5.29):

$$\ell = \arg \max_{\ell > 0} -\frac{1}{2} \mathbf{F}_n^T \mathbf{K}_n^{-1} \mathbf{F}_n - \frac{1}{2} \ln |\mathbf{K}_n| - \frac{n}{2} \ln(2\pi) \quad (5.29)$$

where  $\mathbf{F}_n^T$  denotes the transpose of matrix  $\mathbf{F}_n$ .

With the GP regression framework established, the joint distribution of the observed function value output  $\mathbf{F}_n$  and the function value  $\hat{f}_{n+1}$  at the new  $(n+1)$ -th sample point  $(\mathbf{x}, \mathbf{h})_{n+1}$  can be expressed in Eq. (5.30) as below. Typically, the mean

of  $\hat{f}_{n+1}$ , denoted by  $\mu_*$ , is set to zero as its value does not hinder reaching the optimal solution.

$$\mathbf{F}_{n+1} = \begin{bmatrix} \mathbf{F}_n \\ \hat{f}_{n+1} \end{bmatrix} \sim \mathcal{N} \left( \begin{bmatrix} \boldsymbol{\mu}_n \\ \mu_* \end{bmatrix}, \begin{bmatrix} \mathbf{K}_n & \mathbf{K}_* \\ \mathbf{K}_*^T & \mathbf{K}_{**} \end{bmatrix} \right) \quad (5.30)$$

where  $\mathbf{K}_* = [\text{Cov}(\hat{f}_1, \hat{f}_{n+1}), \text{Cov}(\hat{f}_2, \hat{f}_{n+1}), \dots, \text{Cov}(\hat{f}_n, \hat{f}_{n+1})]^T$ ,  $\mathbf{K}_{**} = \text{Cov}(\hat{f}_{n+1}, \hat{f}_{n+1})$ ,

According to Bayes' theorem, the conditional (posterior) distribution of  $\hat{f}_{n+1}$  at given  $(\mathbf{x}, \mathbf{h})_{n+1}$  of the sample  $n+1$  can be computed by:

$$\hat{f}_{n+1} \Big| (\mathbf{x}, \mathbf{h})_{n+1} \sim \mathcal{N} \left( \mathbf{K}_*^T \mathbf{K}_n^{-1} (\mathbf{F}_n - \boldsymbol{\mu}_n) + \mu_*, \mathbf{K}_{**} - \mathbf{K}_*^T \mathbf{K}_n^{-1} \mathbf{K}_* \right) \quad (5.31)$$

where  $\mathbf{K}_*^T \mathbf{K}_n^{-1} (\mathbf{F}_n - \boldsymbol{\mu}_n) + \mu_*$  indicates the posterior mean that provides the estimated objective function value at sample point  $(\mathbf{x}, \mathbf{h})_{n+1}$  and  $\mathbf{K}_{**} - \mathbf{K}_*^T \mathbf{K}_n^{-1} \mathbf{K}_*$  stands for the posterior covariance that reflects the uncertainty of the estimation, thereby guiding subsequent sampling and optimization efforts.

### 5.3.2 Acquisition function

Following the determination of the posterior mean and covariance of  $\hat{f}_{n+1}$  for any new sample  $n+1$  obtained from the previous step, we can then design an acquisition function to suggest where to select new samples that are promising to achieve an improvement on the objective function estimation. The acquisition function, which is computationally inexpensive, aids in the search for the next sample point to be evaluated by emphasizing both exploitation and exploration (i.e., sampling where the estimation with both high expected performance and uncertainty). One commonly used acquisition function is based on the expected improvement (EI) criterion (Zhan and Xing, 2020; Yin et al., 2022). This criterion aims to identify the next sample point that is likely to maximize the expected improvement over the current best observed

objective function value. Specifically, denote  $\hat{f}^{\min}$  as the optimal value of current best observed value of the objective function  $\hat{f}(\mathbf{x}, \mathbf{h})$ . The EI can be defined as:

$$EI(\mathbf{x}, \mathbf{h}) = \mathbb{E} \left[ \max(0, \hat{f}^{\min} - \hat{f}(\mathbf{x}, \mathbf{h})) \right] \quad (5.32)$$

where the right-hand-side expectation can be further expressed as an integral, which is computable in a closed form by applying integration by parts:

$$EI(\mathbf{x}, \mathbf{h}) = \left( \mu_n(\mathbf{x}, \mathbf{h}) - \hat{f}^{\min} \right) \Phi \left( \frac{\hat{f}^{\min} - \mu_n(\mathbf{x}, \mathbf{h})}{\sigma_n(\mathbf{x}, \mathbf{h})} \right) + \sigma_n(\mathbf{x}, \mathbf{h}) \phi \left( \frac{\hat{f}^{\min} - \mu_n(\mathbf{x}, \mathbf{h})}{\sigma_n(\mathbf{x}, \mathbf{h})} \right) \quad (5.33)$$

Here,  $\mu_n(\mathbf{x}, \mathbf{h}) = \mathbf{K}_*^T \mathbf{K}_n^{-1} (\mathbf{F}_n - \boldsymbol{\mu}_n)$  and  $\sigma_n(\mathbf{x}, \mathbf{h}) = \mathbf{K}_{**} - \mathbf{K}_*^T \mathbf{K}_n^{-1} \mathbf{K}_*$  indicated in Eq. (5.31).  $\Phi(\cdot)$  is the cumulative distribution function and  $\phi(\cdot)$  is the probability density function of the standard normal distribution respectively.

To ensure that sampled solutions are feasible, the interval constraints from [PTLP] model are integrated into the optimization of the EI-based acquisition function. Consequently, the next sample point is chosen by minimizing the acquisition function, i.e.,  $(\mathbf{x}, \mathbf{h})_{n+1} = \arg \max_{(\mathbf{x}, \mathbf{h})} EI(\mathbf{x}, \mathbf{h})$ , subjected to constraints (5.2)-(5.4). The acquisition function can be efficiently optimized by applying continuous first- or second-order optimization techniques, e.g., quasi-Newton method, L-BFGS-B (Frazier, 2018).

Overall, the SBO framework repeatedly alternates between constructing the GP surrogate model and optimizing the acquisition function. Notably, our proposed SBO solution method is specifically designed to solve the PTLP problem by examining various configurations of bus stops. By systematically adjusting the bus stop number  $|\mathcal{I}^r|$  within a specified range, from a predefined minimum value  $I^{\min}$  to a maximum value  $I^{\max}$ , this method can pinpoint the most effective solution. Below is a concise and coherent summary of the step-by-step procedure:

**Step 0: (Initialization)** Start by defining the bus stop number  $|\mathcal{I}^r|$ ,  $\forall r \in \mathcal{R}$ ,

ranging from  $I^{\min}$  to  $I^{\max}$ . Set the parameters and counters for the iterative algorithm.

**Step 1: (Initial sample generation)** Generate a set of initial public transit line design solutions within the feasible region, represented by  $\{(\mathbf{x}, \mathbf{h})_1, \dots, (\mathbf{x}, \mathbf{h})_n\}$ , using the Latin hypercube sampling method. This technique ensures a near-uniform coverage that all dimensions across all dimensions, providing a comprehensive initial sample set.

**Step 2: (Function evaluation)** Evaluate the function under each input of public transit line design and derive the total system cost output through simulation to obtain the objective function value set. Let this collection of evaluated samples be denoted as  $\Theta_n = \{((\mathbf{x}, \mathbf{h})_1, \hat{f}(\mathbf{x}, \mathbf{h})_1), \dots, ((\mathbf{x}, \mathbf{h})_n, \hat{f}(\mathbf{x}, \mathbf{h})_n)\}$ .

**Step 3: (Constructing GP surrogate)** Apply GP regression to update the predictive posterior probability distribution based on the observed samples to model the mapping relationship between  $(\mathbf{x}, \mathbf{h})$  and  $f(\mathbf{x}, \mathbf{h})$ , providing both a prediction and a measure of uncertainty for any new unsampled point.

**Step 4: (Sampling new points)** Identify the next best sample point  $(\mathbf{x}, \mathbf{h})_{n+1}$  with the maximum acquisition function value. Evaluate this new sample through further simulation.

**Step 5: (Updating sample set)** Expand the sampled set by incorporating the newly evaluated sample, i.e.,  $\Theta_n = \Theta_n \cup \{((\mathbf{x}, \mathbf{h})_{n+1}, \hat{f}(\mathbf{x}, \mathbf{h})_{n+1})\}$ .

**Step 6: (Checking stop criteria)** Check whether the stop criteria are met, which include either reaching the maximum evaluation time criterion or the maximum iterations without any improvement. If these stopping criteria are fulfilled, proceed to the final step; otherwise, return to **Step 3**.

**Step 7: (Searching for the optimal solution)** Given observed samples  $\Theta_n$ , formulate a GP regression model. Solve  $(\mathbf{x}, \mathbf{h})_{opt} = \arg \min_{\mathbf{x}, \mathbf{h}} \mu_n(\mathbf{x}, \mathbf{h})$  and return the

optimal public transit line design solution  $(\mathbf{x}, \mathbf{h})_{opt}$ .

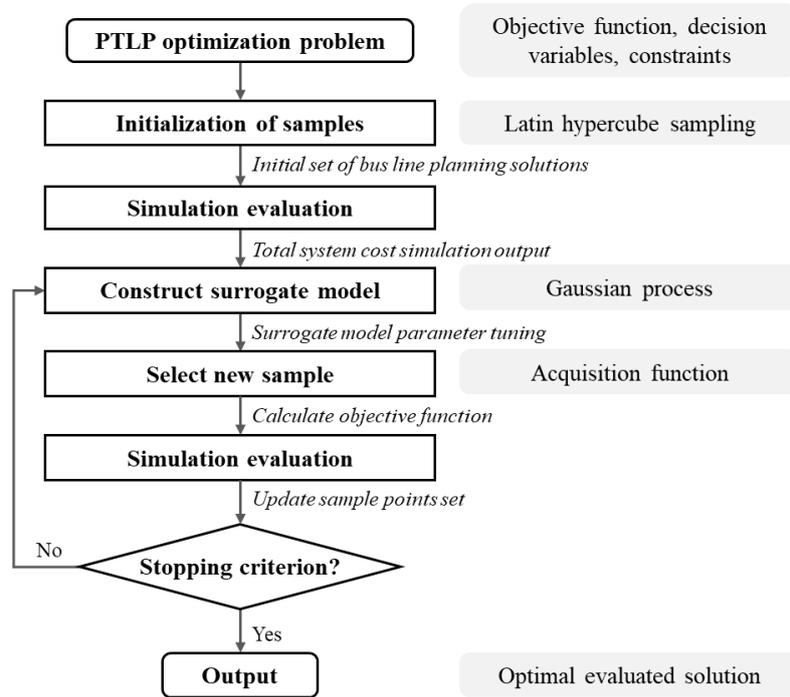


Figure 5.5. Flowchart of the iterative SBO solution method

## 5.4 Numerical Experiments

In this section, we will first evaluate the performance of the proposed solution method against the benchmark approach based on tested instances. Then, we will proceed to conduct a real-world bus route case study to further examine the efficiency of the proposed solution method and explore how the passenger demand level and bus capacity affect the system performance. The MABS system for the public transit line is developed based on the opensource programmable modeling environment NetLogo 6.1.1, and the proposed solution algorithm is coded in Python on a personal computer with Intel (R) Core (TM) i7, 2.80GHz CPU, 16.0 GB RAM.

### 5.4.1 Computational performance

In this subsection, a modified ABC (artificial bee colony) benchmark approach will

first be introduced. Then, the parameter settings of several representative random instances will first be introduced for both the simulation development and the solution method. Finally, the efficiency of our proposed solution method will be evaluated based on these instances by comparing its algorithm performance against that of the benchmark approach.

#### 5.4.1.1 *Benchmark approach*

In this subsection, we further develop a tailored heuristic benchmark approach modified from the traditional ABC algorithm for algorithm performance comparison and evaluation. The ABC algorithm draws inspiration from the behavior of bees in nature, leveraging their collective intelligence to find optimal solutions (Szeto et al., 2011; Szeto and Jiang, 2014). It involves three distinct categories of artificial bees: employed bees, onlooker bees, and scout bees, and each type of bees performs a specific phase in exploring and exploiting the nectar source (solution) space through neighborhood search or random exploration. The proposed modified ABC benchmark approach is developed based on a simulation and optimization framework with the fitness value (expected system cost) of each nectar source (solution) evaluated by the simulation proposed in Section 5.2. Additionally, we modified the traditional ABC algorithm to improve the competitiveness of the benchmark approach. Specifically, we first propose several customized neighborhood search operators to simultaneously identify new bus stop location and service frequency solutions. Then, we modify the traditional ABC method by adopting an adaptive operator selection technique inspired by Ropke and Pisinger (2006) to apply competing operators with frequencies based on their historical performances. The iterative adapted ABC algorithm is outlined in detail through the following step-by-step procedure:

***Step 0: (Input and initialization).*** Input the solution (nectar source) size  $Q$ , i.e., the employed/onlooker bee colony size. Initialize scores and selection probabilities of all service frequency operators  $\mathbf{O}_L$  and bus stop operators  $\mathbf{O}_F$ . Set *limit\_counter* to

0 for each solution to track number of iterations of the solution without system cost decrease (fitness value improvement). Randomly generate the initial solutions  $\{(\mathbf{x}^0, \mathbf{h}^0)_q\}_{q \in \mathbf{Q}}$ , where  $\mathbf{Q} = \{1, \dots, Q\}$ . Initialize the current solutions  $\{(\mathbf{x}, \mathbf{h})_q\}_{q \in \mathbf{Q}} \leftarrow \{(\mathbf{x}^0, \mathbf{h}^0)_q\}_{q \in \mathbf{Q}}$  and optimal solutions  $(\mathbf{x}^*, \mathbf{h}^*) \leftarrow \emptyset$ .

***Step 1: (Employed bee phase)***

- ***Step 1.1: (Neighborhood search)*** Each employed bee searches the vicinity of the currently associated solution  $(\mathbf{x}, \mathbf{h})_q$  by selecting operators based on probabilities to generate new solution  $(\mathbf{x}', \mathbf{h}')_q$ .
- ***Step 1.2: (Solution evaluation)*** If the newly generated solution results in a reduced system cost, then update the current solution, i.e.,  $(\mathbf{x}, \mathbf{h})_q \leftarrow (\mathbf{x}', \mathbf{h}')_q$ , and increase the score of the selected operator by a predefined score parameter; otherwise, the *limit\_counter* associated to the current solution will be increased by 1.

***Step 2: (Onlooker bee phase)***

- ***Step 2.1: (Solution selection and neighborhood search)*** Employed bees communicate solution information to onlooker bees, and each onlooker bee chooses an associated solution based on roulette wheel selection method to prioritize high-quality solutions for further exploration.
- ***Step 2.2: (Solution evaluation)*** Current solutions may either be replaced by new ones or retained depending on whether superior solutions are identified. The operator's score and *limit\_counter* for the existing solution will be updated as outlined in *Step 1.2*.

***Step 3: (Scout bee phase)*** If a solution fails to show improvement for *max\_trial* iterations, a scout bee will intervene by replacing the unpromising solution with a completely new, randomly generated solution to introduce diversity in the search.

**Step 4: (Best solution memorization)** Memorize the best-found solution during each iteration.

**Step 5: (Operator probability update)** If a segment-specific iteration number is reached, the score associated with each operator will be reset and the weight will be updated, which will be elaborated later; otherwise, proceed to **Step 6**.

**Step 6: (Stopping criterion)** If the pre-defined maximum iteration number criterion is not met, return to **Step 1**; otherwise proceed to **Step 7**.

**Step 7: (Output)** Return the optimally memorized solution  $(\mathbf{x}^*, \mathbf{h}^*)$ .

As for the modifications to the ABC algorithm, the neighborhood search operators and the adaptive technique are detailed below.

#### Neighborhood search operators

Several customized operators for the neighborhood search are proposed to generate new bus stop location and service frequency solutions in the employed bee phase and onlooker bee phase. In what follows, we will introduce three operators for generating new bus service frequencies and four operators for modifying the bus stop locations in the predefined intervals respectively.

##### (1) Service frequency-related operators

F1 (random frequency adjustment): This operator randomly chooses a new service frequency as the new solution.

F2 (frequency increase): This operator increases the current service frequency by a specific increment.

F3 (frequency decrease): This operator decreases the current service frequency by a specific amount.

##### (2) Stop location-related operators

L1 (random stop location adjustment): This operator adjusts the stop locations randomly for both directions without adding or removing bus stops.

L2 (access and egress cost-based stop location adjustment): This operator adjusts the spacing between specific consecutive bus stops selected based on a criterion, which is measured by the following access and egress cost associated with each bus stop:

$$C_{a,i} = \theta^a \left( \sum_{p \in \mathcal{P}|r_p^a=i} (\tau_p^{fm}) + \sum_{p \in \mathcal{P}|r_p^d=i} (\tau_p^{lm}) \right) \quad (5.34)$$

The bus stops in each direction  $r \in \mathcal{R}$  are ranked based on the descending order of the associated cost value  $C_{a,i}$ . The first bus stop is selected to reduce the stop spacing around these stops, while the last bus stop is selected to increase the adjacent stop spacing. Specifically, for each selected stop  $i$ , the locations of adjacent bus stops  $i-1$  and  $i+1$ , i.e.,  $x_{i-1}$  and  $x_{i+1}$ , are adjusted closer to (or away from) the location of bus stop  $i$ , i.e.,  $x_i$ , by a random adjustment distance while considering the minimum (or maximum) stop spacing constraints.

L3 (bus stop number decrease): This operator removes specific bus stops selected based on the criterion measured by  $C_{a,i}$ . Specifically, the bus stops in each direction  $r \in \mathcal{R}$  are ranked based on the descending order of  $C_{a,i}$ , and the last bus stop is removed from the public transit line while considering the maximum stop spacing constraint.

L4 (bus stop number increase): This operator adds additional bus stops at specific bus stop locations selected based on the criterion measured by  $C_{a,i}$ . Specifically, the bus stops in each direction  $r \in \mathcal{R}$  are ranked based on the descending order of  $C_{a,i}$ , and the first bus stop is selected to add more stops around them. Within the catchment distance of each selected stop  $i$ , one additional bus stop is added, and the previous bus

stop location is adjusted to be located at the three equinoxes together with the newly introduced bus stops while considering the minimum stop spacing constraint.

### Adaptive operator selection

Different from the traditional ABC algorithm, we further adopt an adaptive operator selection technique inspired by Ropke and Pisinger (2006) to search for solutions with the competing neighborhood structures with frequency derived from their historic performance. Specifically, each operator  $o \in (\mathbf{O}_b \cup \mathbf{O}_d)$  is assigned a score  $\pi_o$  that will be incremented according to the performance of the new solution when this operator is employed, and a weight  $\omega_o$  that will be updated periodically and computed based on the cumulative score. The whole iteration is divided into several segments. The score  $\pi_o$  of operator  $o$  will be reset to 0 at the beginning of each segment and will be increased by a score parameter  $\mathcal{G}$  when the search by the neighborhood operator yields an improved solution. The weight  $\omega_o$  of operator  $o$ , initially assigned equally for each operator, and the weight updates will be made after each segment. Considering the weight  $\omega_o$  of operator  $o$  from the previous segment, the weight  $\omega_o'$  in the next segment can be computed by  $\omega_o' = \omega_o(1 - \rho) + \rho \cdot \frac{\pi_o}{\zeta_o}$ ,

where  $\rho$  is the parameter indicating how the adjustment of weight responds to the operator's effectiveness,  $\pi_o$  denotes the cumulative score in the previous segment of operator  $o$ , and  $\zeta_o$  represents the number of times of operator  $o$  applied in the previous segment. Both the service frequency-related operator and bus stop-related operator will be selected based on the route wheel selection scheme based on their selection probabilities calculated by  $\frac{\omega_o}{\sum_{o \in \mathbf{O}_F} \omega_o}$  and  $\frac{\omega_o}{\sum_{o \in \mathbf{O}_L} \omega_o}$  respectively.

#### 5.4.1.2 Parameter settings and instance generation

We consider a public transit line with a route length of 10 km in both directions in a bus corridor. The minimum and maximum spacing between adjacent stops are set to 250 m and 1000 m. Leaving out the start and end bus terminals, the minimum and maximum intermediate bus stop numbers are set to be 16 and 18. The service frequency ranges from 3 min to 10 min. To accurately simulate the demand for passengers boarding and alighting along the bus route, we conduct experiments on three different spatial distribution models: uniform distribution, normal distribution and bimodal distribution. Each demand generation model provides a different perspective on how passengers might board and alight across the route, reflecting various real-world scenarios. For simplicity, we consider the two-hour period (e.g., the peak hour period), and the cumulative passenger number for each scenario is set to be 800/h. We assume that the demand in two directions follows the same distribution and detailed settings are elaborated as follows:

**Scenario 1:** Uniform distribution. The passenger demand is assumed to be evenly distributed across the entire route, typical of areas with uniform population density and consistent demand. The density functions for boarding and alighting are  $b_{mr}^U(x) = \frac{1}{10}$

and  $a_{mr}^U(x) = \begin{cases} 0, & \text{if } x \leq 3 \\ \frac{1}{10-3}, & \text{if } 3 < x \leq 10 \end{cases}$  respectively. Note that the alighting density

function is set to start from 3 km rather than at the beginning of the route. This ensures a more realistic distribution of passenger alighting, reflecting the delay in alighting as passengers approach their destinations.

**Scenario 2:** Gaussian distribution. The passenger demand is assumed to be concentrated around specific points reflecting scenarios where major hubs or points of interest such as transit centers or business districts attract higher passenger volumes. For boarding, the demand peaks at 3 km with a standard deviation of 3 km, while for

alighting, it peaks at 7 km. The density functions for boarding and alighting are

$$b_{mr}^G(x) = \frac{1}{\sqrt{2\pi} \cdot 3} e^{-\frac{(x-3)^2}{2 \cdot 3^2}} \quad \text{and} \quad a_{mr}^G(x) = \frac{1}{\sqrt{2\pi} \cdot 3} e^{-\frac{(x-7)^2}{2 \cdot 3^2}} \quad \text{respectively.}$$

**Scenario 3:** Bimodal distribution. This demand distribution is utilized to represent areas with two significant points of attraction. It consists of two Gaussian peaks, reflecting high passenger demand at different specific locations along the route. For boarding, the demand splits into two peaks, at 2 km and 4 km, with a narrower standard deviation of 1 km for each, while for alighting, the demand splits into two peaks, at 7 km and 8 km, with a narrower standard deviation of 1 km for each. The density

$$\text{functions for boarding and alighting are } b_{mr}^B(x) = \frac{1}{\sqrt{2\pi} \cdot 1} e^{-\frac{(x-2)^2}{2 \cdot 1^2}} + \frac{1}{\sqrt{2\pi} \cdot 1} e^{-\frac{(x-4)^2}{2 \cdot 1^2}} \quad \text{and}$$

$$a_{mr}^B(x) = \frac{1}{\sqrt{2\pi} \cdot 1} e^{-\frac{(x-7)^2}{2 \cdot 1^2}} + \frac{1}{\sqrt{2\pi} \cdot 1} e^{-\frac{(x-8)^2}{2 \cdot 1^2}} \quad \text{respectively.}$$

Based on the density rate function, we can easily obtain the cumulative boarding and alighting rate function, which can be used to generate passengers according to the proposed arrival principle. In addition to the determined origin, destination, boarding stop and alighting stop of each newly arrived passenger, the personal preference values of choosing walking and shared bikes are randomly and uniformly selected from the set  $\{0.1, 0.2, 0.3, \dots, 1\}$ . The boarding and alighting time parameters of each passenger are taken as 1.55 s and 0.99 s. The passenger walking speed will be realized ranging from 0.4 m/s to 1.7 m/s, while the passenger riding bike speed will be realized ranging from 3.0 m/s to 6.0 m/s. The parameters  $\chi_1$ ,  $\chi_2$  and  $\chi_3$  in the passenger travel behavior model are assumed to be 0.3, 0.1 and 0.5 respectively. The cost of selecting the shared bike mode is set to \$ 3. The acceleration, coasting deceleration and braking deceleration are set to 1.0 m/s<sup>2</sup>, 1.0 m/s<sup>2</sup> and 2.0 m/s<sup>2</sup>, respectively. The bus cruising speed will fall in the interval from 15 km/h to 25 km/h for peak-hour demand period, while it ranges from 25 km/h to 35 km/h for off peak-hour demand period. The bus capacity is set to

110. The optimal threshold parameter and the maximum holding time in the headway-based holding constraint are set to 0.7 and 1.5 min (Cats et al., 2011; Fu and Yang, 2002).

As for cost function calculation parameters, the coefficients of passenger walking time cost, in-vehicle time cost, and waiting time cost are set to 8 \$/h, 4 \$/h and 3 \$/h. The fixed unit cost per bus is set to \$ 300 per day and the fixed driver salary per unit time are set to 100 \$/h. The unit distance cost incurred in the cursing phase is set to 20 \$/km, and the unit time cost of bus acceleration/deceleration can be estimated as 500 \$/h. The dwell time cost coefficient is set to 0. The fixed construction cost and maintenance cost for each bus stop are set to be \$ 30 and \$ 10 respectively.

For the simulation settings, the entire simulation in each call will repeat 15 runs to generate multiple scenarios and obtain an estimated expected total system cost. The total simulation time is set as 14400 s, with the first 7200 s being the simulation stabilization time, and the remaining 7200 s period mimics the considered period. As for the SBO method-related parameters, the upper and lower bounds of stop locations can be obtained through the route length and the minimum stop spacing, the maximum number of simulation evaluations for each specific stop number setting is set to 100, and the stopping criterion for iterations without improvement in the best-found solution is set at 15. As for the parameters in simulation-based ABC algorithm, both employed and onlooker bee numbers are set to 8, the trial time limit is set to 20, the score parameter increment of operator is fixed at 1, the reaction degree parameter is defined as 0.08, and the maximum iteration is set to 300.

#### *5.4.1.3 Computational results and algorithm performance*

Three instances corresponding to three different demand scenarios are generated and the average results for 5 trials are tabulated in Table 5.1 for both proposed solution method and benchmark approach. We report the total system cost (Obj), user cost (Obj<sup>pas</sup>), operator cost (Obj<sup>opt</sup>), the bus fleet size needed in the operational period and

the computational time (Time). In order to have a more intuitive comparison, we also report the absolute and relative gaps in total system cost (Gap\_Obj), passenger cost (Gap\_Obj<sup>pas</sup>) and operator cost (Gap\_Obj<sup>opt</sup>). As we can see, our proposed method achieves lower system costs for all scenarios, with corresponding reductions in both user and operator costs, compared to those obtained by using the ABC heuristic method. Notably, for the scenario of passenger demand in the bimodal distribution pattern, our method highlights the most significant improvements against the benchmark approach, with the difference in system cost and the corresponding relative gap reaching as high as -1900 and -15.0%. The comparative results suggest the obvious advantages of the proposed method over the benchmark approach.

It is also encouraging to see that the required bus fleet size to guarantee the operations identified by our proposed method is less than the benchmark approach. This finding is consistent with the more remarkable cost saving achieved in the operator aspect than that achieved in the passenger aspect, e.g., -5.7% and -21.2% in passenger cost reduction and operator cost reduction respectively for the bimodal scenario. This indicates that our method is more effective in finding better public transit line planning solutions that can yield a more efficient use of the resources while also fulfilling the passenger demand as well. Additionally, we can observe that our method achieves the optimal solution with significantly reduced computational time. We further visualize the convergence performance of two methods for the Gaussian scenario under given bus stop number in Figure 5.6. The x-axis represents the simulation evaluation numbers, while the y-axis indicates values of the objective function. The results illustrate that our method attains faster convergence toward a lower system cost in comparison to the benchmark approach. Overall, the above analyses demonstrate the efficacy and robustness of our proposed method in handling diverse urban mobility patterns.

Table 5.1. Comparison of the proposed solution method and the benchmark approach for different scenarios on test instances

Scenario	Benchmark method				Our method				Comparison							
	Obj	Obj <sup>pas</sup>	Obj <sup>opt</sup>	#FS	Time (s)	Obj	Obj <sup>pas</sup>	Obj <sup>opt</sup>	#FS	Time (s)	Gap_Obj		Gap_Obj <sup>pas</sup>		Gap_Obj <sup>opt</sup>	
											Abs	Rel	Abs	Rel	Abs	Rel
Uniform	10,483	4,360	6,123	12.3	21,290	9,564	4,229	5,335	9.9	15,181	-0,919	-8.8%	-131	-3.00%	-788	-12.9%
Gaussian	11,415	4,688	6,727	13.6	22,405	10,221	4,481	5,740	10.8	14,939	-1,194	-10.5%	-207	-4.42%	-987	-14.7%
Bimodal	12,662	5,066	7,596	14.4	21,602	10,762	4,777	5,985	11.6	14,965	-1,900	-15.0%	-289	-5.70%	-1611	-21.2%

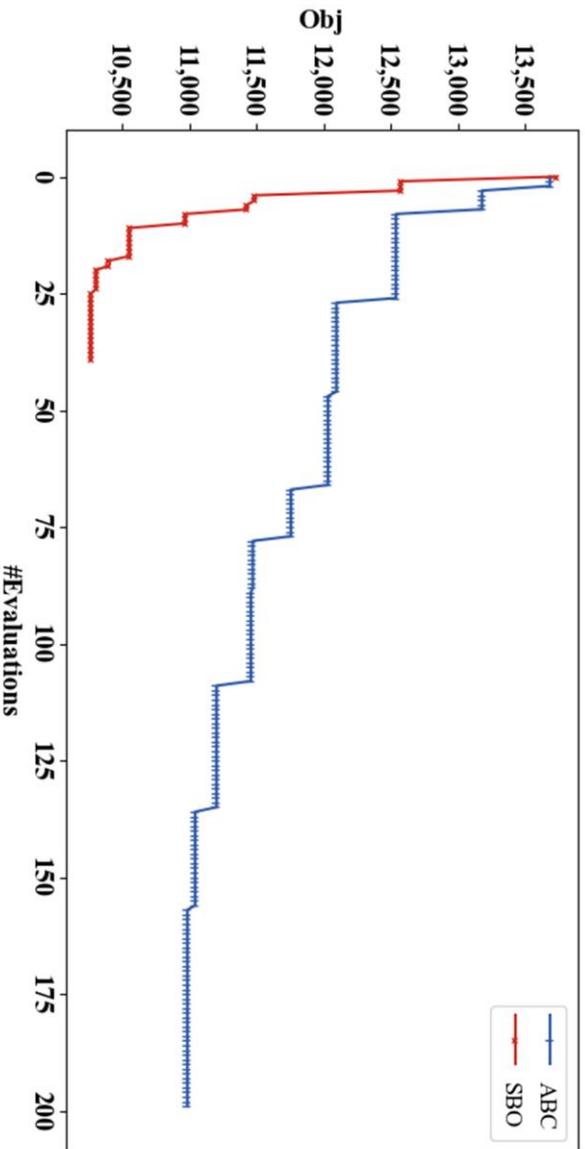


Figure 5.6. Computational efficacy comparison between proposed solution method and benchmark approach

### 5.4.2 Real-world case study

We will proceed to investigate the real-world public transit line case of the 3D bus route operated by the Kowloon Motor Bus (KMB) company in Hong Kong. The 3D bus route operates in two opposite directions, serving the Wong Tai Sin district and Kwun Tong district. There are two bus terminals, i.e., Tsz Wan Shan (TWS) and Kwun Tong (KT). The TWS-KT direction extends for 7.35 km and comprises 15 bus stops, while the KT-TWS direction extends for 7.5 km and traverses 17 bus stops, as shown in Figure 5.7. Detailed bus stop location information for both directions, obtained through Google Maps measurements, is provided in Table 5.2. In addition, we consider the total operational horizon representing a typical working day with period-specific demand patterns. Specifically, we opt for the critical period simulation approach and focus on a four-hour operational horizon that consists of a one-hour period during morning peak times, a two-hour period during off-peak times and a one-hour period during afternoon peak times. Therefore, the period characterized by the most stressed public transit systems and that under normal traffic conditions can be incorporated in the public transit line planning process. Note that although we only focus on the working-day situation, the resulting optimal public transit line design by considering these peak and off-peak conditions remains effective and adequate for accommodating weekend scenarios, which typically experience a reduction in passenger demand.

To estimate the passenger boarding and alighting information used for generating passenger arrivals in simulation, we adopt a straightforward approach that involves onboard surveys for every concerned period. We conduct the on-site surveys during three periods: three hours of morning peak from 6:30 a.m. to 9:30 a.m., the off-peak transition period from 9:30 a.m. to 4:30 p.m., and three hours of afternoon peak from 4:30 p.m. to 7:30 p.m. For each period, by counting the number of passengers boarding and alighting at each stop, we can approximately determine the boarding and alighting rates for every bus stop within a given service frequency, which can be used to derive the cumulative boarding and alighting rate information for the entire route. The passenger boarding and alighting demand information is available on GitHub at (<https://github.com/JiangyanHuang/TLP>). An example of the cumulative boarding and alighting and the corresponding bus loading profiles in both route directions during the morning peak times is depicted in Figure 5.8. Current bus stop locations are marked in

the figure. Current service frequency intervals for different operational periods are obtained from the official website of KMB. The service frequency will be determined within the range from 4 min to 7 min for two peak-hour periods, while the service frequency during the off-peak-hour period will be realized within the range from 5 min to 8 min. The minimum and maximum intermediate bus stop numbers are set to be 13 and 16 for TWS-KT direction and 15 and 18 for KT-TWS direction. Unless stated otherwise, the parameter settings are consistent with those outlined in Subsection 5.4.1.

Table 5.2. Bus stop locations of No. 3D bus route in both directions

Bus stops (TWZ→KT)	Location (km)	Bus stops (KT→TWZ)	Location (km)
1 Tsz Wan Shan (Central) Bus Terminus	0	1 Kwun Tong (Yue Man Square) Bus Terminus	0
2 Ching Fai House	0.36	2 Kwun Tong BBI - Millennium City (Platform B4)	0.55
3 Ching Hong House	0.7	3 Ting Fu Street, Kwun Tong	1.05
4 Tak Oi Secondary School	1.05	4 Lower Ngau Tau Kok Estate	1.51
5 Po Leung Kuk No.1 W. H. Cheung College	1.82	5 Telford Gardens	1.81
6 Diamond Hill Crematorium	2.32	6 Kowloon Bay Station	2.11
7 Fu Shan Estate	2.72	7 Kai Tai Court	2.60
8 Hammer Hill Road Sports Ground	3.27	8 Ngau Chi Wan BBI - Tan Fung House (Platform J2)	3.50
9 Choi Hung Bbi - Pik Hoi House (Platform M4)	4.25	9 Kai Yip Estate	4.25
10 Kai Yip Estate	4.87	10 Choi Hung BBI - Choi Hung Bus Terminus (Platform N6)	4.95
11 Kowloon Bay Station	5.21	11 Chun Tok School	5.30
12 Lower Ngau Tau Kok Estate	5.71	12 Grandview Garden	5.65
13 Kwun Tong Road Sitting-Out Area	6.16	13 Po Kong Village Road School Village	6.00
14 Kwun Tong Bbi - Millennium City (Platform U3)	6.86	14 Tak Oi Secondary School	6.34
15 Kwun Tong (Yue Man Square) Bus Terminus	7.35	15 Tsz Man Estate	6.75
		16 Tsz On Court	7.10
		17 Tsz Wan Shan (Central) Bus Terminus	7.50

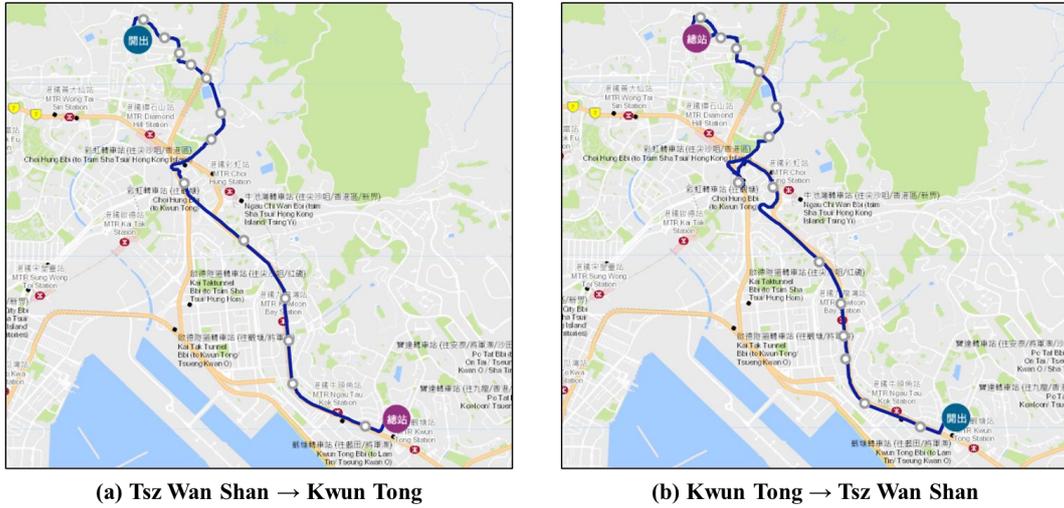


Figure 5.7. The KMB 3D bus route

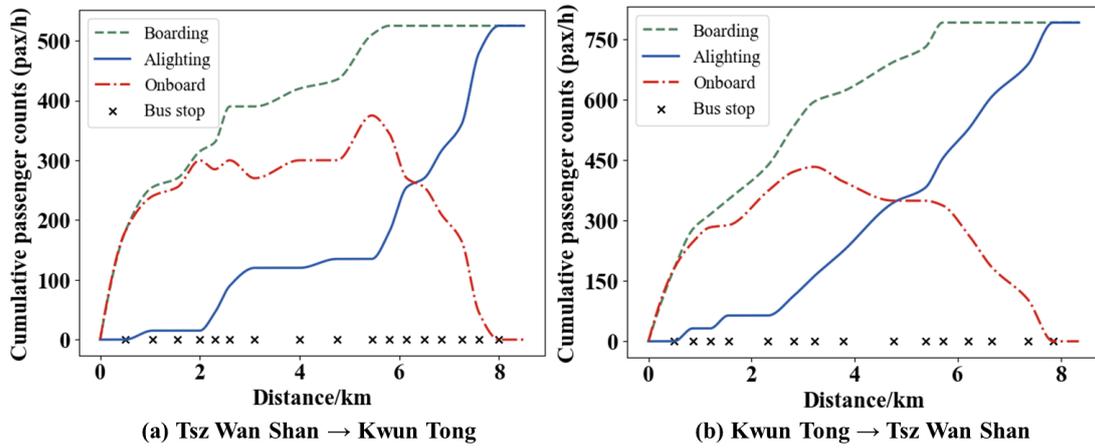


Figure 5.8. Cumulative passenger counts of the KMB 3D bus route during the morning peak-hour period

#### 5.4.2.1 Assessment of the proposed method in KMB 3D bus route

This subsection evaluates the performance of our method in the real-world KMB 3D bus route in Hong Kong by comparing it with the adapted ABC benchmark approach. Unlike Table 5.1, which summarizes the average results without presenting the hidden variability, results for all tested instances are tabulated in Table 5.3. We can observe that the proposed SBO method yields better solutions in much less computational time for all the real-world instances. Specifically, the benchmark approach obtains a total system cost of around 20,000, whereas our method can produce a greatly lower total system cost of around 17,000. As a result, the relative gap of the total system cost is larger than 12.9% among all 5 instances and the maximum reaches as high as 17.1 % with the

absolute gap being -3,502. By looking further into the details of the separate costs, we can find that the passenger cost savings achieved by our method are comparatively higher than operator cost savings, which is similar to the results observed in tests on the randomly generated instances. As for the comparison of the bus fleet size, we can observe that our method consistently requires a maximum of 12 buses for operations, which is fewer than that obtained by using the benchmark approach among all the instances. Overall, the findings prove that our method demonstrates higher computational efficiency and produces more favorable public transit line planning solutions that can achieve better resource utilization in real-world applications.

#### 5.4.2.2 *Impact analysis*

In this subsection, we will conduct extensive numerical experiments to investigate the impact of the demand level and bus capacity on the system performance of the public transit line operation. These experiments aim to examine the potential concerns regarding how these critical factors affect the system costs, service quality and operational efficiency.

##### *Impact of demand level*

To investigate how the varying passenger demand affects system performance, we set up scenarios with different demand levels and each scenario specifies a ratio that compares the demand of the tested scenario to the real demand. Table 5.4 tabulates the results under varying demand levels ranging from 0.5 to 2 and the average results for 5 trials are reported. We report the average stop density (StopDst), defined as the average

stop number per unit route length in a direction calculated by  $\frac{|\mathcal{I}^r|}{L}$ , the average service

frequency (AvgFreq), i.e., the average service frequency across all periods computed by

$\frac{\sum_{m \in \mathcal{M}} h_{mr}}{|\mathcal{M}|}$ , the total system cost (TotCost), the operator cost (OptCost), and the passenger

cost (PasCost), respectively. To better represent the bus operation performance, we also report the bus fleet size (FS) needed to meet the demand and the average bus dwell time at stops (AvgDel) during the operational horizon. Furthermore, we report the average waiting time (AvgWat), the average access and egressing time (AvgAne), and the average in-vehicle time (AvgInv) of passengers to show the potentially affected time

duration in different stages of their journey. Table 5.4 illustrates that as the demand level increases, we should deploy more bus stops and set a higher bus departure frequency with a larger bus fleet size to guarantee service quality. Accordingly, all three types of costs, i.e., total system cost, passenger cost and operator cost, display an upward trend as the demand level increases. Notably, the increment in passenger cost is much higher than that of operator cost as shown in Figure 5.9. As the demand level exceeds 1.25, the proportion of passenger cost within the overall system cost becomes higher than that of the operator cost. The rise in operator costs can be verified by the increase in the number of buses, i.e., from 8.9 to 20.1. Additionally, despite the effort of deploying more bus stops, the average bus dwell time also obviously increases from 2.89 min to 4.03 min as the demand level grows from 0.5 to 2. It is noteworthy to mention that the increment of stop density actually decreases with the demand level, as shown in Figure 5.10. This implies that there may exist a threshold, beyond which the increase of stop density may not be helpful for regulating the system cost further. Because the bus capacity is upperly bounded, denser bus stop deployment will not contribute to accommodating more passengers along the route, when the demand level already becomes excessively large and exceeds the bus service capability. If this is the case, improving the bus stop density will incur more operational and construction costs instead.

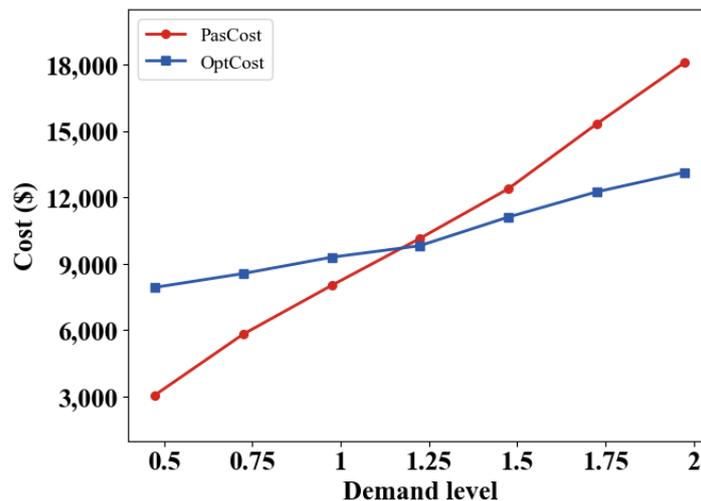


Figure 5.9. Variations of the passenger and operator costs with an increase in the demand level

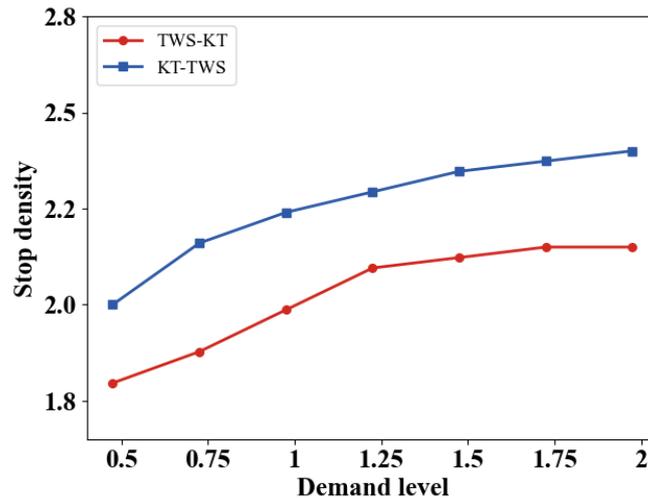


Figure 5.10. Variations of the stop density with an increase in the demand level

For a better presentation, we further depict the variations of the average accessing and egressing time, average waiting time and average in-vehicle time of the passengers in Figure 5.11. It is evident that as demand levels rise, passengers' average accessing and egressing time decreases, while the average waiting time first drops and then rises. The result is within our expectation, as the increased stop density suggests that passengers' accessing and egressing distance will be reduced, thus resulting in a decrease in the duration needed for these procedures. The higher service frequency with a larger bus fleet size helps decrease the passenger waiting time when the level of demand increases from 0.5 to 1.5. However, as passenger demand grows further, up to a level of 2, the average waiting time obviously increases to 3.78. This can be explained by an increased number of waiting passengers who might fail to get on a bus and need to wait for latter arrivals due to the bus capacity limit, thus leading to additional waiting times. Comparatively, as the demand level grows, there is a noticeable increase in passengers' average in-vehicle time. For example, the in-vehicle time is only 10.79 on average when the demand level is 0.5, while it increases to 12.98 for the scenarios of the demand level being 2. This means that although the denser stop deployment can effectively reduce costs associated with passenger access and egress, it would result in more frequent stopovers of the bus fleet during the services. Consequently, the accessing and egressing time cost savings are significantly outweighed by the negative impact on in-vehicle time costs for passengers. As a matter of fact, in practice, the determination of the optimal stop location and service frequency results from balancing the costs incurred by passengers with the expenses of the service operator for planning

and operations. As the demand increases to an excessively high level, the marginal effect of the passenger waiting and accessing/egressing time costs on the public transit line planning, regarding the deployment of bus stops and service frequency in both directions, becomes greater than effects of in-vehicle time cost for passengers and operator expenses.

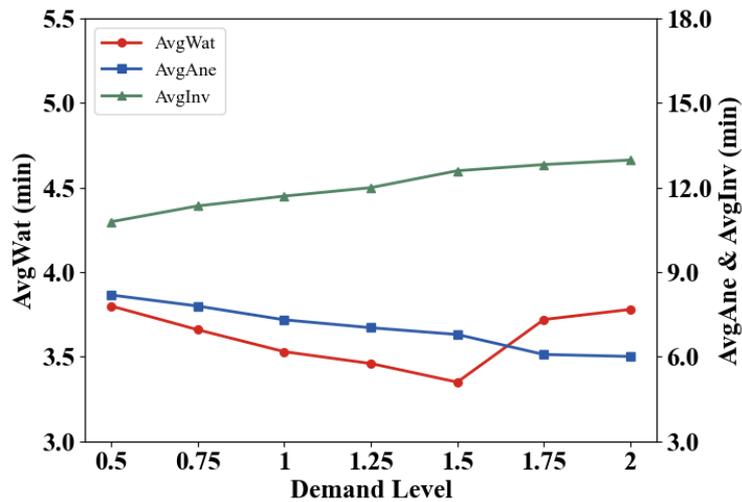


Figure 5.11. Variations of the average waiting time, accessing and egressing time and in-vehicle time of passengers with an increase in the demand level

### Impact of bus capacity

To examine how bus capacity affects system performance, we conduct extensive numerical experiments under different bus capacities varying from 30 to 150. The average results for 5 trials are tabulated in Table 5.5 and the same performance indicators as the impact analysis of the demand level are reported. Again, we further visualize the variations of the passenger cost and operator cost in Figure 5.12, and the variations in average waiting time, accessing and egressing time and in-vehicle time of passengers in Figure 5.13. We can see from Figure 5.12 that when the bus capacity increases the lower bus stop density and service frequency could be considered in order to cut down the system costs while satisfying the passenger demand. By doing so, we will achieve a system cost saving as high as 1,855 when the bus capacity grows from 30 to 150 with a significant decrease in operator cost coupled with a slight increase in the passenger cost. Specifically, the increased passenger cost could be attributed to the collective effects of the increased stop spacing and service frequency, which can be verified by the increased average accessing and egressing time and waiting time of passengers (see Figure 5.13). Comparatively, the operator cost is reduced due to the

decreased service frequency setting with the obvious cut in the bus fleet. As expected, an increase in bus capacity leads to a corresponding rise in bus dwell time, since a higher capacity indicates that the buses are capable of serving more passengers at bus stops and thus will experience prolonged dwell time. Moreover, Figure 5.13 also indicates that passengers' average in-vehicle time shows no obvious variation with the increase in the bus capacity. This can be attributed to the fact that although the larger capacity may facilitate better stop spacing and potentially reduce delays due to less frequent stopovers, the longer dwell time resulting from more boarding and alighting passengers will lead to an obvious negative effect, thus resulting in the fluctuation of the passengers' in-vehicle time on average.

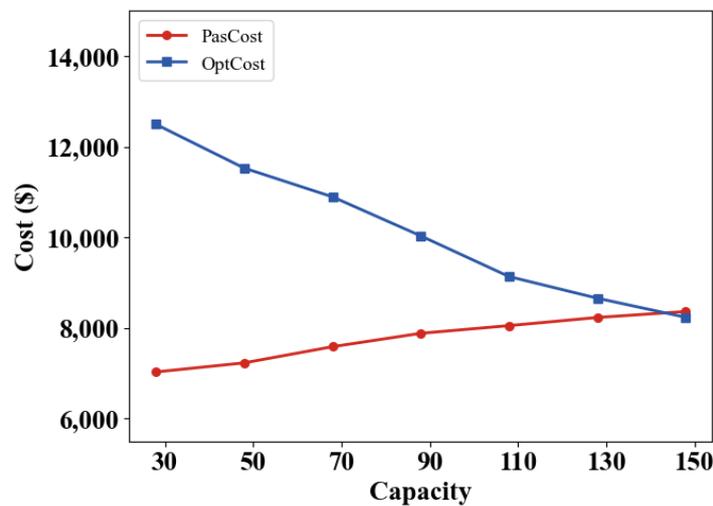


Figure 5.12. Variations of the passenger and operator costs with an increase in the bus capacity

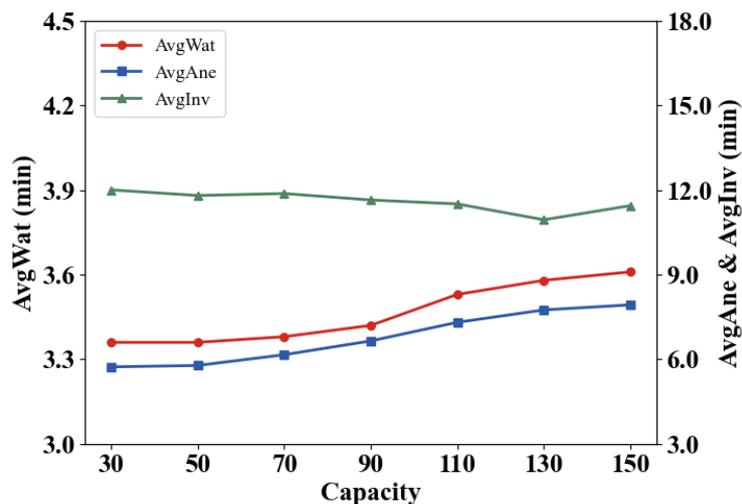


Figure 5.13. Variations of the average waiting time, accessing and egressing time and in-vehicle time of passengers with an increase in the bus capacity

Table 5.3. Comparison of the proposed solution method and the benchmark approach on KMB 3D bus route

Instance No.	Benchmark method				Our method				Comparison							
	Obj	Obj <sup>pas</sup>	Obj <sup>opt</sup>	#FS	Time (s)	Obj	Obj <sup>pas</sup>	Obj <sup>opt</sup>	#FS	Time (s)	Gap_Obj Abs	Rel	Gap_Obj <sup>pas</sup> Abs	Rel	Gap_Obj <sup>opt</sup> Abs	Rel
1	20,116	8,794	11,322	15	71,932	17,320	7,843	9,477	12	47,128	-2,796	-13.9%	-951	-10.8%	-1,845	-16.3%
2	20,510	8,995	11,515	15	69,001	17,008	8,001	9,007	11	47,143	-3,502	-17.1%	-994	-11.1%	-2,508	-21.8%
3	19,910	8,609	11,301	14	69,310	17,034	7,953	9,081	12	46,116	-2,876	-14.4%	-656	-7.6%	-2,220	-19.6%
4	20,475	8,983	11,492	15	69,805	17,592	8,102	9,490	12	45,994	-2,883	-14.1%	-881	-9.8%	-2,002	-17.4%
5	20,195	8,882	11,313	14	72,013	17,584	8,051	9,533	12	46,730	-2,611	-12.9%	-831	-9.4%	-1,780	-15.7%

Table 5.4. Effect of demand levels on system performance

Demand level	StopDst			AvgFeq(min)			TotCost (\$)			PasCost (\$)			OptCost (\$)			FS			AvgDel (min)			AvgWat (min)			AvgAne (min)			AvgInv (min)		
	TWS-KT	KT-TWS	KT-TWS	TWS-KT	KT-TWS	KT-TWS	TWS-KT	KT-TWS	KT-TWS	TWS-KT	KT-TWS	KT-TWS	TWS-KT	KT-TWS	KT-TWS	TWS-KT	KT-TWS	KT-TWS	TWS-KT	KT-TWS	KT-TWS	TWS-KT	KT-TWS	KT-TWS	TWS-KT	KT-TWS	KT-TWS	TWS-KT	KT-TWS	KT-TWS
0.5	1.80	2.00	2.00	7.91	7.99	7.99	11,642	3,703	7,939	8.9	2.89	3.80	8.19	10.79																
0.75	1.88	2.16	2.16	7.75	7.88	7.88	14,389	5,825	8,564	10.6	3.22	3.66	7.80	11.35																
1	1.99	2.24	2.24	7.60	7.66	7.66	17,326	8,025	9,301	11.8	3.40	3.53	7.31	11.70																
1.25	2.10	2.29	2.29	7.47	7.57	7.57	19,971	10,150	9,821	12.7	3.66	3.46	7.03	12.00																
1.5	2.12	2.35	2.35	7.23	7.40	7.40	23,505	12,391	11,114	17.1	3.84	3.35	6.79	12.60																
1.75	2.15	2.37	2.37	7.22	7.24	7.24	26,870	15,317	12,253	19.5	3.95	3.72	6.08	12.82																
2	2.15	2.40	2.40	7.18	7.15	7.15	29,651	18,110	13,142	20.1	4.03	3.78	6.01	12.98																

Table 5.5. Effect of bus capacity on system performance

Bus capacity	StopDst			AvgFeq(min)			TotCost (\$)			PasCost (\$)			OptCost (\$)			FS			AvgDel (min)			AvgWat (min)			AvgAne (min)			AvgInv (min)		
	TWS-KT	KT-TWS	KT-TWS	TWS-KT	KT-TWS	KT-TWS	TWS-KT	KT-TWS	KT-TWS	TWS-KT	KT-TWS	KT-TWS	TWS-KT	KT-TWS	KT-TWS	TWS-KT	KT-TWS	KT-TWS	TWS-KT	KT-TWS	KT-TWS	TWS-KT	KT-TWS	KT-TWS	TWS-KT	KT-TWS	KT-TWS	TWS-KT	KT-TWS	KT-TWS
30	2.18	2.40	2.40	6.89	6.85	6.85	19,521	7,029	12,492	22.4	2.53	3.36	5.73	12.01																
50	2.15	2.35	2.35	7.22	7.21	7.21	18,755	7,229	11,526	18.2	2.71	3.36	5.78	11.81																
70	2.10	2.29	2.29	7.36	7.38	7.38	18,478	7,586	10,892	13.6	2.91	3.38	6.16	11.88																
90	2.04	2.27	2.27	7.51	7.58	7.58	17,913	7,881	10,031	12.0	3.17	3.42	6.65	11.65																
110	1.99	2.24	2.24	7.60	7.66	7.66	17,384	8,051	9,133	11.8	3.40	3.53	7.31	11.51																
130	1.88	2.19	2.19	7.68	7.79	7.79	16,887	8,230	8,656	10.2	3.50	3.58	7.75	10.95																
150	1.80	2.16	2.16	7.71	7.82	7.82	16,597	8,362	8,235	8.9	3.57	3.61	7.93	11.45																

## 5.5 Concluding Remarks

This chapter investigates the joint optimization of bus stop location and service frequency for a single bus route with bike-sharing feeder mode integration. We propose a conceptual optimization modeling framework for the PTLP problem in pursuit of system cost minimization while considering uncertainty. In light of the disaggregate behaviors (movements) of involved entities and the inherent stochasticity and nonlinearity as well as realistic constraints, the PTLP model is hard to formulate with a closed-form analytical model. Hence, we design an MABS system to simulate the public transit line operation process and derive the estimate of the objective function. The simulation system is constructed with explicit consideration of the passengers' first- and last-mile travel mode selection, vehicles' different regimes, and stochasticity and dynamics in traveling for all moving agents. To efficiently solve the expensive-to-evaluate simulation-based PTLP model, we develop an SBO method to approximate the GP surrogate to derive the mapping relation between decision input and simulation output and identify the optimal stop location and service frequency solution in the bus operation system. We conduct extensive numerical experiments based on a tested bus route with different demand distribution scenarios and a real-world scenario to investigate the SBO method performance against an adapted ABC benchmark approach. Lastly, we examine how variations in demand levels and bus capacity affect the bus operation system performance.

## 5.6 Appendix. Notation

---

$\mathcal{R}$	Direction set in the bus route, with each direction denoted by $r$
$L^r$	The bus route length in direction $r$
$\mathcal{S}$	Bus terminal set, with each terminal denoted by $s$
$\mathcal{I}$	Bus stop set, with each bus stop denoted by $i$
$\mathcal{I}^r$	Bus stop set in direction $r$
$x_i$	The location of the bus stop $i$
$\Delta^{\min}$	The minimum bus stop spacing
$\Delta^{\max}$	The maximum bus stop spacing
$I^{\min}$	The minimum number of stops
$I^{\max}$	The maximum number of stops
$\mathcal{M}$	Time period set in the operational horizon, with each period denoted by $m$
$h_{mr}$	The service frequency for period $m$ in each direction $r$ of the bus route
$\mathcal{P}$	The set of all passengers, with each passenger denoted by $p$
$\mathcal{B}$	The set of required buses, with each bus denoted by $b$
$T$	Duration of the entire planning horizon
$\mathbf{x}$	Decision vector of all the intermediate stop locations
$\mathbf{h}$	Decision vector of the service frequency
$\xi$	A random vector containing all the uncertain data in the problem
$\mathbb{Z}$	The joint distribution of the uncertain data
$\mathbf{x}^l(\mathbf{x}^u)$	The lower (upper) bounds of $\mathbf{x}$
$\mathbf{h}^l(\mathbf{h}^u)$	The lower (upper) bounds of $\mathbf{h}$
$x_p^o$	The predetermined origin of passenger $p$
$x_p^d$	The predetermined destination of passenger $p$
$i_p^b$	The boarding bus stop of passenger $p$
$i_p^a$	The alighting bus stop of passenger $p$
$e_{walk}^p$ ( $e_{bike}^p$ )	The personal preference indicator of passenger $p$ choosing walking (shared bikes) during the first- and last-mile journey

$\tilde{v}_p^a$	The speed of passenger $p$ accessing boarding bus stops from the origin
$\tilde{v}_p^e$	The speed of passenger $p$ egressing bus stop to the destination
$\beta$	The average passenger boarding time
$\alpha$	The average passenger alighting time
$\mathcal{K}$	The set of bus trips (departures) with respect to the service frequency, with each departure denoted by $k$
$s_k^d$	The departure terminal of bus trip $k$
$s_k^a$	The arrival terminal of bus trip $k$
$r_k$	The route direction of bus trip $k$
$t_k^d$	The time of departure from the terminal for trip $k$
$\tilde{t}_k^a$	The arrival time of the trip $k$ at the arrival terminal
$\tilde{t}_k$	The travel time during the trip $k$
$W$	The bus capacity
$a_{ki}^b$	The arrival time at bus stop $i$ in trip $k$ taken by bus $b$
$d_{ki}^b$	The departure time at bus stop $i$ in trip $k$ taken by bus $b$
$\partial_{acc}$	The bus acceleration
$\partial_{coa}$	The bus deceleration in coasting
$\partial_{dec}$	The bus deceleration in braking
$\tilde{v}_{ki}^b$	The constant travel speed for the stop-to-stop motion from bus stop $i$ to the successive stop $i+1$ in trip $k$ taken by bus $b$
$\Delta\tilde{t}_{ki}^b$	The stop-to-stop travel time from bus stop $i$ to the successive stop $i+1$ in trip $k$ taken by bus $b$
$\Delta\tilde{w}_{ki}^b$	The dwell time at bus stop $i$ in trip $k$ taken by bus $b$
$B_{ki}^b$	The successfully boarded passenger number on bus $b$ at stop $i$ in trip $k$
$A_{ki}^b$	The alighting passenger number at stop $i$ in trip $k$ taken by bus $b$
$\Delta\tilde{g}_{ki}^b$	The additional holding time of bus $b$ at stop $i$ in trip $k$
$\tilde{t}_k$	The bus trip travel time
$\varpi$	The minimum driver shifting time at terminals
$\theta^a$	The unit time cost for passenger walking

$\theta^v$	The unit time cost for passenger in-vehicle
$\theta^w$	The unit time cost for passenger waiting
$\theta^b$	The fixed cost of each bus
$\theta^h$	The value of salary of each bus driver per unit time
$\theta^t$	The value of vehicle unit distance cost in cruising speed
$\theta^l$	The value of vehicle unit time cost in acceleration and deceleration
$\theta^s$	The value of vehicle cost per unit time in dwelling (and holding) at bus stops
$\theta^c$	The value of construction cost of each bus stop
$\theta^o$	The value of maintenance cost of each bus stop
$B_{mr}(x)$	The cumulative boarding rate functions for each specified observation period $m$ in direction $r$
$A_{mr}(x)$	The cumulative alighting rate functions for each specified observation period $m$ in direction $r$
$\lambda_{mr}$	The total passenger arrival rate along the route during the period $m$ in direction $r$
$\boldsymbol{\mu}_n$	The mean vector of GP
$\mathbf{K}_n$	The covariance matrix of GP
$\ell$	The length scale of the kernel function
$\mu_n(\mathbf{x}, \mathbf{h})$	The estimated value at new sampled point $(\mathbf{x}, \mathbf{h})$
$\sigma_n(\mathbf{x}, \mathbf{h})$	The posterior distribution of the prediction at new sampled point $(\mathbf{x}, \mathbf{h})$
$\Theta_n$	Sample set in SBO method
$\mathbb{Z}$	The joint distribution of the passenger walking speed and bus cruising speed

---



## Chapter 6 Conclusions and Future Research Recommendations

### 6.1 Overview and Research Contributions

This thesis is dedicated to solving new challenges in the shared mobility services faced by service operators and governments by considering three research topics: (i) RT-SAVD problem for the SAM services, (ii) CSD problem for ISM services under stochastic demand, (iii) PTLP problem with integration of bike-sharing.

For the first research topic, we first investigate a RT-SAVD problem for SAM services considering ride-pooling strangers and passengers' satisfaction, which has not been explored in previous literature. Second, we design a look-ahead rolling horizon framework and propose a tailored ARA-LNS algorithm to solve the large-scale dynamic SAV dispatching problem. The traditional LNS algorithm iteratively identifies the optimal routing solution for each SAV, with an embedded ARA scheme invoked if the overall profit for routing solutions fails to improve after certain iterations to adaptively reassign requests to different vehicles. The adaptive selection of the different assignment operators will guide the algorithm to efficiently search for good-quality SAV dispatching solutions. Third, we have performed numerical experiments to demonstrate the efficacy of our proposed solution approach by comparing it with the benchmark approach. Additionally, we examine the effects of incorporating the ride-pooling option in SAM services and the flexibility of time windows of passenger requests to derive managerial insights.

For the second research topic, we first propose an interesting CSD problem for ISM services considering the nonlinear AERD profiles of passengers under uncertain demand, which has not been explored in previous literature. Second, we develop an efficient ALNS-CSA solution algorithm that combines an ALNS heuristic with an efficient CSA method to obtain good-quality solutions by iteratively finding the optimal DPV solution for each scenario and improving the compensation scheme accordingly. Third, extensive numerical experiments on adapted benchmark instances have been conducted to evaluate the proposed solution method, assess the performance of ISM services, and derive valuable managerial insights.

For the third research topic, we address the PTLP problem to determine the optimal design of a single bus route with integrated bike-sharing feeder services in an uncertain environment. To the best of our knowledge, no previous literature has investigated a

simulation-based optimization problem to jointly optimize the bus stop location and service frequency for the public transit line supported by the complementary bike-sharing feeder service considering uncertainties. Additionally, we develop an agent-based simulation system that incorporates the disaggregate issues and captures uncertainties in both passenger traveling and bus operational processes at the micro-level, compared to the aggregated modeling and rough approximation approaches in most existing studies. Furthermore, we develop an SBO solution method based on the Bayesian optimization framework that builds the link between simulation and optimization to find the optimal decision variable set of the stop location and service frequency through the exploration and exploitation of the computationally efficient surrogate. Last, we have performed numerical experiments on the tested bus route with multiple demand scenarios and the real-world bus route in Hong Kong and conducted impact analysis to derive managerial insights.

## **6.2 Recommendations for Future Research**

For the SAM services, future research can be explored in several aspects. First, this research topic investigated in this thesis a dynamic and deterministic SAV dispatch problem without considering probabilistic or forecasted features of future demand. Incorporating stochastic information, such as uncertain passenger requests, into the SAV dispatching would enhance its alignment with real-world scenarios. Second, as EV battery prices drop and charging facilities become more convenient, it would be also interesting to explore the deployment of the electric SAV fleet or a mixed fleet with both types of vehicles considering the charging strategy. Third, designing a reasonable compensation scheme that can encourage more passengers to opt for pooled trips and reduce the costs for the SAM service provider presents another research avenue for future exploration. Lastly, the integration of the SAV fleet into existing urban transport networks necessitates the consideration of congestion effects within SAM service operations. Additionally, considering the economic, social, and environmental aspects will offer valuable research opportunities to improve the sustainability of SAM services.

Future research on the ISM services can be conducted in three key areas. First, the research topic explored in this thesis has focused on the tactical compensation scheme design. It would be interesting to further investigate the operational level issues such as dynamic routing and request pooling under the given compensation scheme. Second,

we have only considered the homogeneous gasoline-powered fleet; however, with the rise of transportation electrification, it is highly expected to study the OMS-based ISM service employing an electric vehicle fleet or a mixed fleet with both types of vehicles to determine routing plans considering vehicle charging. Last but not least, the proposed methodology used in this research problem can be extended to other integrated mobility services. For instance, in the realm of individual-based ISM service, we can also determine the optimal compensation scheme while accounting for individual elastic tolerance.

For the public transit design with bike-sharing integration, the following research directions are recommended for future exploration. First, the simulation-based PTLP model could be further improved by incorporating future passenger demand to produce a more robust public transit line design. Second, it is also worth trying to extend the single bus route to a bus network with multiple lines considering passenger transfers to find a more reliable and comprehensive public transit design solution. Third, in the PTLP problem, shared-bike service is currently considered only to complement and fit the pre-existing public transit. Investigate the collaboration design between the shared-bike feeder service and public transportation service could enhance the role of shared micromobility, making it become a component rather than just a complement to public transit. Last, given the economic and environmental advantages of electric buses over traditional diesel buses, it would be interesting to explore the public transit line planning with charging facility deployment problem based on the electric bus fleet or the mixed fleet while considering the charging strategy. Moreover, one may also investigate the realistic EB fleet replacement scheme by considering real-life bus operations.

In addition to these specific research directions for each topic discussed in this thesis, some general recommendations could further enhance the shared mobility research domain. First, as the implementation and operations of shared mobility services become increasingly intricate, it is crucial to conduct deeper empirical analyses. This involves collecting and analyzing real-world data to validate and refine existing theoretical models, thus uncovering latent patterns and insights that enhance our understanding of shared mobility dynamics. Second, with the rise of big data, adopting data-driven methodologies revolutionizes how we deal with various decision-making issues like operational-level customer request matching, delivery task assignment, and route optimization, etc. This approach can outperform conventional methods to some

extent by leveraging vast datasets to improve problem-solving accuracy and efficiency. Furthermore, the rapidly evolving field of artificial intelligence, particularly machine learning, has introduced powerful tools for addressing challenges in the studies of shared mobility. These technologies enable researchers to tackle computationally complex problems beyond the scope of traditional optimization techniques and boost more sophisticated analyses and solutions in the shared mobility domain.

## References

- Agatz, N., Erera, A. L., Savelsbergh, M. W. P., & Wang, X. (2011). Dynamic ride-sharing: A simulation study in metro Atlanta. *Procedia - Social and Behavioral Sciences*, *17*, 532–550.
- 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.
- Al Hla, Y. A., Othman, M., & Saleh, Y. (2019). Optimizing an eco-friendly vehicle routing problem model using regular and occasional drivers integrated with driver behavior control. *Journal of Cleaner Production*, *234*, 984–1001.
- Alonso-Mora, J., Samaranayake, S., Wallar, A., Frazzoli, E., & Rus, D. (2017). On-demand high-capacity ride-sharing via dynamic trip-vehicle assignment. *Proceedings of the National Academy of Sciences*, *114*(3), 462–467.
- American Public Transportation Association (APTA). (2024). *Ridership Report*. APTA. <https://www.apta.com/research-technical-resources/transit-statistics/ridership-report/>.
- 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.

- Attanasio, A., Cordeau, J.-F., Ghiani, G., & Laporte, G. (2004). Parallel Tabu search heuristics for the dynamic multi-vehicle dial-a-ride problem. *Parallel Computing*, 30(3), 377–387.
- Azadeh, S. S., Atasoy, B., Ben-Akiva, M. E., Bierlaire, M., & Maknoon, M. Y. (2022). Choice-driven dial-a-ride problem for demand responsive mobility service. *Transportation Research Part B: Methodological*, 161, 128–149.
- Berbeglia, G., Cordeau, J.-F., & Laporte, G. (2010). Dynamic pickup and delivery problems. *European Journal of Operational Research*, 202(1), 8–15.
- Berbeglia, G., Cordeau, J.-F., & Laporte, G. (2012). A hybrid tabu search and constraint programming algorithm for the dynamic dial-a-ride problem. *INFORMS Journal on Computing*, 24(3), 343–355.
- Berbeglia, G., Cordeau, J.-F., Gribkovskaia, I., & Laporte, G. (2007). Static pickup and delivery problems: A classification scheme and survey. *TOP*, 15(1), 1–31.
- Bhatti, A., Akram, H., Basit, H. M., Khan, A. U., Raza, S. M., & Naqvi, M. B. (2020). E-commerce trends during COVID-19 Pandemic. *International Journal of Future Generation Communication and Networking*, 13(2), 1449–1452.
- Bhosekar, A., & Ierapetritou, M. (2018). Advances in surrogate based modeling, feasibility analysis, and optimization: A review. *Computers & Chemical Engineering*, 108, 250–267.
- Bouton, S., Hannon, E., Haydamous, L., Heid, B., Knupfer, S., Naucner, T., ..., Ramanathan, S. (2017). An integrated perspective on the future of mobility, part 2: Transforming urban delivery. *McKinsey Center for Business and Environment*.

- Bozzi, A. D., & Aguilera, A. (2021). Shared E-scooters: A review of uses, health and environmental impacts, and policy implications of a new micro-mobility service. *Sustainability*, *13*(16), 8676.
- Bruzzo, F., Cavallaro, F., & Nocera, S. (2021). The integration of passenger and freight transport for first-last mile operations. *Transport policy*, *100*, 31-48.
- Buldeo Rai, H., Verlinde, S., Merckx, J., & Macharis, C. (2017). Crowd logistics: an opportunity for more sustainable urban freight transport? *European Transport Research Review*, *9*, 1–13.
- Burger, L. (2019) *1 billion riders*. Medium. <https://leonard-burger.medium.com/1-billion-riders-8430cd2f306b>
- Campbell, K. B., & Brakewood, C. (2017). Sharing riders: How bikesharing impacts bus ridership in New York City. *Transportation Research Part A: Policy and Practice*, *100*, 264–282.
- Cats, O., & Glück, S. (2019). Frequency and vehicle capacity determination using a dynamic transit assignment model. *Transportation Research Record*, *2673*(3), 574–585.
- Cats, O., Larijani, A. N., Koutsopoulos, H. N., & Burghout, W. (2011). Impacts of holding control strategies on transit performance: bus simulation model analysis. *Transportation Research Record*, *2216*(1), 51–58.
- Ceder, A. A., Butcher, M., & Wang, L. (2015). Optimization of bus stop placement for routes on uneven topography. *Transportation Research Part B: Methodological*, *74*, 40–61.

- Ceder, A., & Wilson, N. H. M. (1986). Bus network design. *Transportation Research Part B: Methodological*, 20(4), 331–344.
- Ceylan, H., & Ozcan, T. (2018). Optimization of headways and departure times in urban bus networks: A case study of Çorlu, Turkey. *Advances in Civil Engineering*, 2018, 1–12.
- Chang, S. K., & Schonfeld, P. M. (1991). Multiple period optimization of bus transit systems. *Transportation Research Part B: Methodological*, 25(6), 453–478.
- Chen, X., Wang, Y., & Ma, X. (2021). Integrated optimization for commuting customized bus stop planning, routing design, and timetable development with passenger spatial-temporal accessibility. *IEEE Transactions on Intelligent Transportation Systems*, 22(4), 2060–2075.
- Chen, X., Zhang, L., He, X., Xiong, C., & Li, Z. (2014). Surrogate-based optimization of expensive-to-evaluate objective for optimal highway toll charges in transportation network. *Computer-Aided Civil and Infrastructure Engineering*, 29(5), 359–381.
- Cheng, G., Zhao, S., & Zhang, T. (2019). A bi-level programming model for optimal bus stop spacing of a bus rapid transit system. *Mathematics*, 7(7), 625.
- Cheng, R., Jiang, Y., & Nielsen, O. A. (2023). Integrated people-and-goods transportation systems: from a literature review to a general framework for future research. *Transport Reviews*, 43(5), 997–1020.
- Chien, S. I., & Qin, Z. (2004). Optimization of bus stop locations for improving transit accessibility. *Transportation Planning and Technology*, 27(3), 211–227.
- Cleophas, C., Cottrill, C., Ehmke, J. F., & Tierney, K. (2019). Collaborative urban transportation: Recent advances in theory and practice. *European Journal of*

*Operational Research*, 273(3), 801–816.

Cleophas, C., Cottrill, C., Ehmke, J. F., Tierney, K. (2019). Collaborative urban transportation: Recent advances in theory and practice. *European Journal of Operational Research*, 273(3), 801–816.

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.

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

Cordeau, J.-F., & Laporte, G. (2007). The dial-a-ride problem: Models and algorithms. *Annals of Operations Research*, 153(1), 29–46.

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

Daganzo, C. F. (2009). A headway-based approach to eliminate bus bunching: Systematic analysis and comparisons. *Transportation Research Part B: Methodological*, 43(10), 913–921.

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.

dell’Olio, L., Moura, J. L., & Ibeas, A. (2006). Bi-level mathematical programming model for locating bus stops and optimizing frequencies. *Transportation Research Record*, 1971(1), 23–31.

- Elbert, R., & Rentschler, J. (2022). Freight on urban public transportation: A systematic literature review. *Research in Transportation Business & Management*, 45, 100679.
- Fagnant, D. J., & Kockelman, K. M. (2018). Dynamic ride-sharing and fleet sizing for a system of shared autonomous vehicles in Austin, Texas. *Transportation*, 45(1), 143–158.
- Farhan, J., & Chen, T. D. (2018). Impact of ridesharing on operational efficiency of shared autonomous electric vehicle fleet. *Transportation Research Part C: Emerging Technologies*, 93, 310–321.
- 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.
- Fehn, F., Engelhardt, R., Dandl, F., Bogenberger, K., Busch, F. (2023). Integrating parcel deliveries into a ride-pooling service—An agent-based simulation study. *Transportation Research Part A: Policy and Practice*, 169, 103580.
- Forrester, A. I., & Keane, A. J. (2009). Recent advances in surrogate-based optimization. *Progress in Aerospace Sciences*, 45(1–3), 50–79.
- Frazier, P. I. (2018). A tutorial on Bayesian optimization. *arXiv preprint arXiv:1807.02811*.
- Fu, L., & Yang, X. (2002). Design and implementation of bus-holding control strategies with real-time information. *Transportation Research Record*, 1791(1), 6–12.
- Furuhata, M., Dessouky, M., Ordóñez, F., Brunet, M. E., Wang, X., Koenig, S., 2013. Ridesharing: The state-of-the-art and future directions. *Transportation Research Part B: Methodological*, 57, 28–46.

- Ge, Q., Han, K., & Liu, X. (2021). Matching and routing for shared autonomous vehicles in congestible network. *Transportation Research Part E: Logistics and Transportation Review*, 156, 102513.
- Gendreau, M., Guertin, F., Potvin, J.-Y., & Séguin, R. (2006). Neighborhood search heuristics for a dynamic vehicle dispatching problem with pick-ups and deliveries. *Transportation Research Part C: Emerging Technologies*, 14(3), 157–174.
- Ghilas, V., Demir, E., & Van Woensel, T. (2016a). The pickup and delivery problem with time windows and scheduled lines. *INFOR: Information Systems and Operational Research*, 54(2), 147–167.
- Ghilas, V., Demir, E., & Van Woensel, T. (2016b). An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows and scheduled lines. *Computers & Operations Research*, 72, 12-30.
- Ghilas, V., Demir, E., & Woensel, T. V. (2016c). A scenario-based planning for the pickup and delivery problem with time windows, scheduled lines and stochastic demands. *Transportation Research Part B: Methodological*, 91, 34–51.
- Gkiotsalitis, K., Schmidt, M., & van der Hurk, E. (2022). Subline frequency setting for autonomous minibusses under demand uncertainty. *Transportation Research Part C: Emerging Technologies*, 135, 103492.
- Gleason, J. M. (1975). A set covering approach to bus stop location. *Omega*, 3(5), 605–608.

- Golberg, D. E. (1989). Genetic algorithms in search, optimization, and machine learning. *Addion Wesley*, 1989(102), 36.
- Grab. (2021). 7 reasons to choose GrabExpress as a delivery partner for your online business. *Grab*. <https://www.grab.com/ph/community/7-reasons-to-choose-grabexpress-as-delivery-partner-for-your-online-business/>
- Häll, C. H., Lundgren, J. T., & Voss, S. (2015). Evaluating the performance of a dial-a-ride service using simulation. *Public Transport*, 7, 139–157.
- Häme, L. (2011). An adaptive insertion algorithm for the single-vehicle dial-a-ride problem with narrow time windows. *European Journal of Operational Research*, 209(1), 11–22.
- Häme, L., & Hakula, H. (2015). A maximum cluster algorithm for checking the feasibility of dial-a-ride instances. *Transportation Science*, 49(2), 295–310.
- Hartmans, A., & Leskin, P. (2019). The history of how Uber went from the most feared startup in the world to its massive IPO. *Business Insider*, 11.
- Hatzenbühler, J., Cats, O., & Jenelius, E. (2020). Transitioning towards the deployment of line-based autonomous buses: Consequences for service frequency and vehicle capacity. *Transportation Research Part A: Policy and Practice*, 138, 491–507.
- Haupt, R. L., & Haupt, S. E. (2004). *Practical Genetic Algorithms*. John Wiley & Sons.
- Ho, S. C., Szeto, W. Y., Kuo, Y.-H., Leung, J. M. Y., Petering, M., & Tou, T. W. H. (2018). A survey of dial-a-ride problems: Literature review and recent developments. *Transportation Research Part B: Methodological*, 111, 395–421.
- Hörcher, D., & Tirachini, A. (2021). A review of public transport economics. *Economics of Transportation*, 25, 100196.

- Hosni, H., Naoum-Sawaya, J., & Artail, H. (2014). The shared-taxi problem: Formulation and solution methods. *Transportation Research Part B: Methodological*, 70, 303–318.
- Hou, L., Li, D., & Zhang, D. (2018). Ride-matching and routing optimization: Models and a large neighborhood search heuristic. *Transportation Research Part E: Logistics and Transportation Review*, 118, 143–162.
- Hua, S., Zeng, W., Liu, X., & Qi, M. (2022). Optimality-guaranteed algorithms on the dynamic shared-taxi problem. *Transportation Research Part E: Logistics and Transportation Review*, 164, 102809.
- Ibarra-Rojas, O. J., Delgado, F., Giesen, R., & Muñoz, J. C. (2015). Planning, operation, and control of bus transport systems: A literature review. *Transportation Research Part B: Methodological*, 77, 38–75.
- Ibeas, Á., dell’Olio, L., Alonso, B., & Sainz, O. (2010). Optimizing bus stop spacing in urban areas. *Transportation Research Part E: Logistics and Transportation Review*, 46(3), 446–458.
- Institute for Transportation and Development Policy (ITDP). (2021). *Maximizing potential by connecting micromobility and transit*. ITDP. <https://itdp.org/2021/06/30/maximizing-potential-by-connecting-micromobility-and-transit/>
- Jones, D. R., Schonlau, M., & Welch, W. J. (1998). Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, 13, 455–492.
- Jung, J., Jayakrishnan, R., & Park, J. Y. (2016). Dynamic shared-taxi dispatch algorithm with hybrid-simulated annealing. *Computer-Aided Civil and Infrastructure Engineering*, 31(4), 275–291.

- Kaan, L., & Olinick, E. V. (2013). The vanpool assignment problem: Optimization models and solution algorithms. *Computers & Industrial Engineering*, *66*(1), 24–40.
- Katrakazas, C., Quddus, M., Chen, W.-H., & Deka, L. (2015). Real-time motion planning methods for autonomous on-road driving: State-of-the-art and future research directions. *Transportation Research Part C: Emerging Technologies*, *60*, 416–442.
- Ke, J., Yang, H., Li, X., Wang, H., & Ye, J. (2020). Pricing and equilibrium in on-demand ride-pooling markets. *Transportation Research Part B: Methodological*, *139*, 411–431.
- Kim, K. (2023). Investigation of modal integration of bike-sharing and public transit in Seoul for the holders of 365-day passes. *Journal of Transport Geography*, *106*, 103518.
- Kızıl, K. U., & Yıldız, B. (2023). Public transport-based crowd-shipping with backup transfers. *Transportation Science*, *57*(1), 174–196.
- Kong, H., Jin, S. T., & Sui, D. Z. (2020). Deciphering the relationship between bikesharing and public transit: Modal substitution, integration, and complementation. *Transportation Research Part D: Transport and Environment*, *85*, 102392.
- Lavieri, P. S., & Bhat, C. R. (2019). Modeling individuals' willingness to share trips with strangers in an autonomous vehicle future. *Transportation Research Part A: Policy and Practice*, *124*, 242–261.

- Le Pira, M., Tavasszy, L., Correia, G., Ignaccolo, M., Inturri, G. (2021). Opportunities for integration between Mobility as a Service (MaaS) and freight transport: a conceptual model. *Sustainable Cities and Society*, 74, 103212.
- Le, T. V., Stathopoulos, A., Van Woensel, T., & Ukkusuri, S. V. (2019). Supply, demand, operations, and management of crowd-shipping services: A review and empirical evidence. *Transportation Research Part C: Emerging Technologies*, 103, 83–103.
- Levin, M. W. (2017). Congestion-aware system optimal route choice for shared autonomous vehicles. *Transportation Research Part C: Emerging Technologies*, 82, 229–247.
- 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.
- 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.
- Li, B., Krushinsky, D., Van Woensel, T., Reijers, H. A. (2016a). An adaptive large neighborhood search heuristic for the share-a-ride problem. *Computers & Operations Research*, 66, 170-180.
- Li, B., Krushinsky, D., Van Woensel, T., & Reijers, H. A. (2016b). The share-a-ride problem with stochastic travel times and stochastic delivery locations. *Transportation Research Part C: Emerging Technologies*, 67, 95-108.

- Lien, T. (2015). *Sidecar to expand package delivery service*. Los Angeles Times. <https://www.latimes.com/business/la-fi-0210-sidecar-delivery-service-20150210-5-story.html>
- Liu, L., Sun, L., Chen, Y., & Ma, X. (2019). Optimizing fleet size and scheduling of feeder transit services considering the influence of bike-sharing systems. *Journal of Cleaner Production*, 236, 117550.
- Loeb, B., Kockelman, K. M., & Liu, J. (2018). Shared autonomous electric vehicle (SAEV) operations across the Austin, Texas network with charging infrastructure decisions. *Transportation Research Part C: Emerging Technologies*, 89, 222–233.
- Lokhandwala, M., & Cai, H. (2018). Dynamic ride sharing using traditional taxis and shared autonomous taxis: A case study of NYC. *Transportation Research Part C: Emerging Technologies*, 97, 45-60.
- Lu, C.-C., Diabat, A., Li, Y.-T., & Yang, Y.-M. (2022). Combined passenger and parcel transportation using a mixed fleet of electric and gasoline vehicles. *Transportation Research Part E: Logistics and Transportation Review*, 157, 102546.
- Ma, J., Li, X., Zhou, F., & Hao, W. (2017). Designing optimal autonomous vehicle sharing and reservation systems: A linear programming approach. *Transportation Research Part C: Emerging Technologies*, 84, 124–141.
- Maalouf, M., MacKenzie, C. A., Radakrishnan, S., & Court, M. (2014). A new fuzzy logic approach to capacitated dynamic Dial-a-Ride problem. *Fuzzy Sets and Systems*, 255, 30–40.

- Marković, N., Nair, R., Schonfeld, P., Miller-Hooks, E., & Mohebbi, M. (2015). Optimizing dial-a-ride services in Maryland: Benefits of computerized routing and scheduling. *Transportation Research Part C: Emerging Technologies*, 55, 156–165.
- Martinez, L. M., & Viegas, J. M. (2017). Assessing the impacts of deploying a shared self-driving urban mobility system: An agent-based model applied to the city of Lisbon, Portugal. *International Journal of Transportation Science and Technology*, 6(1), 13–27.
- Masson, R., Trentini, A., Lehuédé, F., Malhéné, N., Péton, O., Tlahig, H., 2017. Optimization of a city logistics transportation system with mixed passengers and goods. *EURO Journal on Transportation and Logistics*, 6(1), 81-109.
- Medina, M., Giesen, R., & Muñoz, J. C. (2013). Model for the optimal location of bus stops and its application to a public transport corridor in Santiago, Chile. *Transportation Research Record*, 2352(1), 84–93.
- Mourad, A., Puchinger, J., & Chu, C. (2019). A survey of models and algorithms for optimizing shared mobility. *Transportation Research Part B: Methodological*, 123, 323–346.
- Mousavi, K., Bodur, M., & Roorda, M. J. (2022). Stochastic last-mile delivery with crowd-shipping and mobile depots. *Transportation Science*, 56(3), 612–630.
- movmi. (2023). *How to integrate micromobility with our public transportation: Cheat sheet.* movmi. <https://movmi.net/blog/how-to-integrate-micromobility-with-our-public-transportation-cheat-sheet/>

- Mulley, C., Nelson, J. D., & Wright, S. (2018). Community transport meets mobility as a service: On the road to a new a flexible future. *Research in Transportation Economics*, 69, 583–591.
- Murray, A. T. (2003). A Coverage model for improving public transit system accessibility and expanding access. *Annals of Operations Research*, 123(1), 143–156.
- Narayanan, S., Chaniotakis, E., & Antoniou, C. (2020). Shared autonomous vehicle services: A comprehensive review. *Transportation Research Part C: Emerging Technologies*, 111, 255–293.
- National Association of City Transportation Officials (NACTO). (2022). *Shared Micromobility in 2022*. NACTO. <https://nacto.org/publication/shared-micromobility-in-2022/>
- Nicholson, W., & Snyder, C. M. (2012). *Microeconomic Theory: Basic Principles and Extensions*. Cengage Learning.
- Organization for Economic Co-operation and Development (OECD). (2023). *ITF transport outlook*. OECD. [https://www.oecd-ilibrary.org/transport/itf-transport-outlook-2023\\_b6cc9ad5-en](https://www.oecd-ilibrary.org/transport/itf-transport-outlook-2023_b6cc9ad5-en)
- Otto, B., & Boysen, N. (2014). A dynamic programming based heuristic for locating stops in public transportation networks. *Computers & Industrial Engineering*, 78, 163–174.
- Perumal, S. S. G., Lusby, R. M., & Larsen, J. (2022). Electric bus planning & scheduling: A review of related problems and methodologies. *European Journal of Operational Research*, 301(2), 395–413.

- Pisinger, D., & Ropke, S. (2007). A general heuristic for vehicle routing problems. *Computers & Operations Research*, 34(8), 2403–2435.
- Ionescu, D. (2024). *Micromobility operators call for better links to transit*. Planetizen. <https://www.planetizen.com/news/2024/03/127724-micromobility-operators-call-better-links-transit>
- Potvin, J.-Y., & Rousseau, J.-M. (1993). A parallel route building algorithm for the vehicle routing and scheduling problem with time windows. *European Journal of Operational Research*, 66(3), 331–340.
- Precedence Research. (2024). *Shared Mobility Market Size, Share, and Trends 2024 to 2034*. Precedence Research. <https://www.precedenceresearch.com/shared-mobility-market>
- Prins, C. (2004). A simple and effective evolutionary algorithm for the vehicle routing problem. *Computers & Operations Research*, 31(12), 1985–2002.
- Radzimski, A., & Dzięcielski, M. (2021). Exploring the relationship between bike-sharing and public transport in Poznań, Poland. *Transportation Research Part A: Policy and Practice*, 145, 189–202.
- 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.
- Ren, T., Jiang, Z., Cai, X., Yu, Y., Xing, L., Zhuang, Y., Li, Z., 2021. A dynamic routing optimization problem considering joint delivery of passengers and parcels. *Neural Computing and Applications*, 33, 10323–10334.
- Rieser, M. (2010). *Adding transit to an agent-based transportation simulation: concepts and implementation*. PhD. Thesis.

- Ropke, S., & Pisinger, D. (2006). An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. *Transportation Science*, 40(4), 455–472.
- Sadrani, M., Tirachini, A., & Antoniou, C. (2022). Optimization of service frequency and vehicle size for automated bus systems with crowding externalities and travel time stochasticity. *Transportation Research Part C: Emerging Technologies*, 143, 103793.
- Sampaio, A., Savelsbergh, M., Veelenturf, L., Van Woensel, T. (2019). Crowd-based city logistics. In *Sustainable Transportation and Smart Logistics* (pp. 381–400). Elsevier.
- Santos, D. O., & Xavier, E. C. (2013). Dynamic taxi and ridesharing: A framework and heuristics for the optimization problem. *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence*, 2885–2891.
- Santos, D. O., & Xavier, E. C. (2015). Taxi and ride sharing: A dynamic dial-a-ride problem with money as an incentive. *Expert Systems with Applications*, 42(19), 6728–6737.
- Sayarshad, H. R., & Chow, J. Y. J. (2015). A scalable non-myopic dynamic dial-a-ride and pricing problem. *Transportation Research Part B: Methodological*, 81, 539–554.
- Sayarshad, H. R., & Oliver Gao, H. (2018). A scalable non-myopic dynamic dial-a-ride and pricing problem for competitive on-demand mobility systems. *Transportation Research Part C: Emerging Technologies*, 91, 192–208.
- Schaller, B. (2023). *The new automobility: Lyft, Uber and the future of American cities*. Schaller Consulting.

<http://www.schallerconsult.com/rideservices/automobility.htm>

- Schilde, M., Doerner, K. F., & Hartl, R. F. (2014). Integrating stochastic time-dependent travel speed in solution methods for the dynamic dial-a-ride problem. *European Journal of Operational Research*, 238(1), 18–30.
- Schittekat, P., Kinable, J., Sörensen, K., Sevaux, M., Spieksma, F., & Springael, J. (2013). A metaheuristic for the school bus routing problem with bus stop selection. *European Journal of Operational Research*, 229(2), 518–528.
- Shaheen, S. A., & Cohen, A. P. (2013). Carsharing and personal vehicle services: worldwide market developments and emerging trends. *International Journal of Sustainable Transportation*, 7(1), 5–34.
- Shaheen, S., & Cohen, A. (2019). Shared ride services in North America: Definitions, impacts, and the future of pooling. *Transport Reviews*, 39(4), 427–442.
- Shaheen, S., Cohen, A., & Jaffee, M. (2018). *Innovative mobility: Carsharing outlook*. UC Berkeley: Transportation Sustainability Research Center.
- Shaheen, S., Cohen, A., & Zohdy, I. (2016). *Shared mobility: current practices and guiding principles* (No. FHWA-HOP-16-022). United States. Federal Highway Administration. <https://rosap.ntl.bts.gov/view/dot/42193>
- Shaw, P. (1998). Using constraint programming and local search methods to solve vehicle routing problems. In *International conference on principles and practice of constraint programming* (pp. 417-431). Springer.
- Shelat, S., Huisman, R., & van Oort, N. (2018). Analysing the trip and user characteristics of the combined bicycle and transit mode. *Research in Transportation Economics*, 69, 68–76.

- Shen, Y., Zhang, H., & Zhao, J. (2018). Integrating shared autonomous vehicle in public transportation system: A supply-side simulation of the first-mile service in Singapore. *Transportation Research Part A: Policy and Practice*, *113*, 125–136.
- Simonetto, A., Monteil, J., & Gambella, C. (2019). Real-time city-scale ridesharing via linear assignment problems. *Transportation Research Part C: Emerging Technologies*, *101*, 208–232.
- Souza, A. L., Bernardo, M., Penna, P. H., Pannek, J., & Souza, M. J. (2022). Bi-objective optimization model for the heterogeneous dynamic dial-a-ride problem with no rejects. *Optimization Letters*, *16*(1), 355–374.
- Sun, L., Wang, S., Liu, S., Yao, L., Luo, W., & Shukla, A. (2018). A complete research on the feasibility and adaptation of shared transportation in megacities—A case study in Beijing. *Applied Energy*, *230*, 1014–1033.
- Sun, Y., Chen, Z.-L., & Zhang, L. (2020). Nonprofit peer-to-peer ridesharing optimization. *Transportation Research Part E: Logistics and Transportation Review*, *142*, 102053.
- Sun, Y., Goh, B. (2020). *China's Didi launches delivery service after coronavirus hammers ride-hailing demand*. Reuters. <https://www.reuters.com/article/us-health-coronavirus-Didi-idUSKBN2130ZB>
- Sun, Y., Guo, Q., Schonfeld, P., & Li, Z. (2017). Evolution of public transit modes in a commuter corridor. *Transportation Research Part C: Emerging Technologies*, *75*, 84–102.
- Szeto, W. Y., & Jiang, Y. (2014). Transit route and frequency design: Bi-level modeling and hybrid artificial bee colony algorithm approach. *Transportation Research Part B: Methodological*, *67*, 235–263.

- Szeto, W. Y., Wu, Y., & Ho, S. C. (2011). An artificial bee colony algorithm for the capacitated vehicle routing problem. *European Journal of Operational Research*, *215*(1), 126–135.
- Tafreshian, A., Abdolmaleki, M., Masoud, N., & Wang, H. (2021). Proactive shuttle dispatching in large-scale dynamic dial-a-ride systems. *Transportation Research Part B: Methodological*, *150*, 227–259.
- Teixeira, J. F., Silva, C., & Moura e Sá, F. (2021). Empirical evidence on the impacts of bikesharing: a literature review. *Transport Reviews*, *41*(3), 329–351.
- Thombre, A., Agarwal, A. (2021). A paradigm shift in urban mobility: policy insights from travel before and after COVID-19 to seize the opportunity. *Transport Policy*, *110*, 335–353.
- Triki, C. (2021). Using combinatorial auctions for the procurement of occasional drivers in the freight transportation: A case-study. *Journal of Cleaner Production*, *304*, 127057.
- United Nations. (2021). COVID-19 and e-commerce: a global review. [https://unctad.org/system/files/official-document/dtlstict2020d13\\_en\\_0.pdf](https://unctad.org/system/files/official-document/dtlstict2020d13_en_0.pdf)
- Van Duin, R., Wiegmans, B., Tavasszy, L., Hendriks, B., He, Y. (2019). Evaluating new participative city logistics concepts: The case of cargo hitching. *Transportation Research Procedia*, *39*, 565–575.
- van Gelder, K. (2024). *Online shopping — statistics & facts*. Statista. <https://www.statista.com/topics/871/online-shopping/#topicOverview>
- van Mil, J. F., Leferink, T. S., Annema, J. A., & van Oort, N. (2021). Insights into factors affecting the combined bicycle-transit mode. *Public Transport*, *13*(3), 649–673.

- Vuchic, V. R. (2007). *Urban Transit Systems and Technology*. John Wiley & Sons.
- Wang, H., & Yang, H. (2019). Ridesourcing systems: A framework and review. *Transportation Research Part B: Methodological*, 129, 122–155.
- Wen, J., Chen, Y. X., Nassir, N., & Zhao, J. (2018). Transit-oriented autonomous vehicle operation with integrated demand-supply interaction. *Transportation Research Part C: Emerging Technologies*, 97, 216–234.
- Wirasinghe, S. C., & Ghoneim, N. S. (1981). Spacing of bus-stops for many to many travel demand. *Transportation Science*, 15(3), 210–221.
- Wu, L., Gu, W., Fan, W., & Cassidy, M. J. (2020). Optimal design of transit networks fed by shared bikes. *Transportation research part B: Methodological*, 131, 63–83.
- Wu, W., Liu, R., & Jin, W. (2017). Modelling bus bunching and holding control with vehicle overtaking and distributed passenger boarding behaviour. *Transportation Research Part B: Methodological*, 104, 175–197.
- Xiang, Z., Chu, C., & Chen, H. (2008). The study of a dynamic dial-a-ride problem under time-dependent and stochastic environments. *European Journal of Operational Research*, 185(2), 534–551.
- Yang, X.-H., Cheng, Z., Chen, G., Wang, L., Ruan, Z.-Y., & Zheng, Y.-J. (2018). The impact of a public bicycle-sharing system on urban public transport networks. *Transportation Research Part A: Policy and Practice*, 107, 246–256.
- Yin, R., Liu, Z., & Zheng, N. (2022). A simulation-based model for continuous network design problem using Bayesian optimization. *IEEE Transactions on Intelligent Transportation Systems*, 23(11), 20352–20367.

- Yin, Z., Rybarczyk, G., Zheng, A., Su, L., Sun, B., & Yan, X. (2024). Shared micromobility as a first-and last-mile transit solution? Spatiotemporal insights from a novel dataset. *Journal of Transport Geography*, *114*, 103778.
- Yu, V. F., Purwanti, S. S., Redi, A. A. N. P., Lu, C.-C., Suprayogi, S., & Jewpanya, P. (2018). Simulated annealing heuristic for the general share-a-ride problem. *Engineering Optimization*, *50*(7), 1178–1197.
- Zhan, D., & Xing, H. (2020). Expected improvement for expensive optimization: a review. *Journal of Global Optimization*, *78*(3), 507–544.
- Zhan, X., Szeto, W. Y., & Chen, X. M. (2022). A simulation–optimization framework for a dynamic electric ride-hailing sharing problem with a novel charging strategy. *Transportation Research Part E: Logistics and Transportation Review*, *159*, 102615.
- Zhan, X., Szeto, W. Y., Shui, C. S., & Chen, X. M. (2021). A modified artificial bee colony algorithm for the dynamic ride-hailing sharing problem. *Transportation Research Part E: Logistics and Transportation Review*, *150*, 102124.
- Zhang, J., Wang, D. Z. W., & Meng, M. (2020). Optimization of bus stop spacing for on-demand public bus service. *Transportation Letters*, *12*(5), 329–339.
- Zhang, W., Guhathakurta, S., Fang, J., & Zhang, G. (2015). The performance and benefits of a shared autonomous vehicles based dynamic ridesharing system: An agent-based simulation approach. *Transportation Research Board 94th Annual Meeting*, *15*, 2919.
- Zhou, Z., & Roncoli, C. (2022). A scalable vehicle assignment and routing strategy for real-time on-demand ridesharing considering endogenous congestion. *Transportation Research Part C: Emerging Technologies*, *139*, 103658.