Value of Information in Dispatching Shared Autonomous Mobility-on-Demand Systems: a Simulation Framework

by

Jian Wen

B.S., Tsinghua University (2013)
M.S., École Polytechnique (2016)

Submitted to the Department of Civil and Environmental Engineering
in partial fulfillment of the requirements for the degree of

Master of Science in Transportation

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

February 2018

Author .... Signature redacted ....
Department of Civil and Environmental Engineering
Jan 19, 2018

Certified by .... Signature redacted ...
Jinhua Zhao
Associate Professor of City and Transportation Planning
Thesis Supervisor

Certified by ...... Signature redacted ....
Neema Nassir
Senior Postdoctoral Associate of Civil and Environmental Engineering
Thesis Supervisor

Signature redacted

Accepted by ....
Jesse Kroll
Professor of Civil and Environmental Engineering
Chair, Graduate Program Committee

Value of Information in Dispatching Shared Autonomous Mobility-on-Demand Systems: a Simulation Framework

by

Jian Wen

Submitted to the Department of Civil and Environmental Engineering on Jan 19, 2018, in partial fulfillment of the requirements for the degree of Master of Science in Transportation

Abstract

The concept of shared mobility-on-demand (MoD) systems describes an innovative mode of transportation in which rides are tailored as per the immediate requests in a shared manner. Convenience of hailing, ease of transactions, and economic efficiency of crowd-sourcing the rides have made these services very attractive today. It is anticipated that autonomous vehicle (AV) technology may further improve the economics of such services by reducing the operational costs. The design and operation of such an shared autonomous mobility-on-demand (AMoD) system is therefore an important research direction that requires significant investigation.

This thesis mainly addresses three issues revolving around the dispatching strategies of shared AMoD systems. First, it responds to the special dispatching need that is critical for effective AMoD operation. This includes a dynamic request-vehicle assignment heuristic and an optimal rebalancing policy. In addition, the dispatching strategies also reflect transit-oriented designs in two ways: (a) the objective function embodies the considerations of service availability and equity through the support of various hailing policies; and (b), the service facilitates first-mile connections to public transportation. Second, this thesis models the interaction between demand and supply through simulation. Using the level of service as interface, this mechanism enables feedback between operators and travelers to more closely represent the choices of both parties. A fixed-point approach is then applied to reach balance iteratively, estimating both the demand volume and the system performance at equilibrium. The results from the simulation support decision-making with regard to comprehensive system design problems such as fleet sizing, vehicle capacities and hailing policies. Third, the thesis evaluates the value of demand information through simulation experiments. To quantify the system performance gain that can be derived from the demand information, this thesis proposes to study two dimensions, level of information and value of information, and builds up the relationship between them. The numerical results help rationalize the efforts operators should spend on data collection, information inference and advanced dispatching algorithms.

This thesis also implements an agent-based modeling platform, `amod-abm`, for simulating large-scale shared AMoD applications. Specifically, it models individual travelers and vehicles with demand-supply interaction and analyzes system performance through various metrics of indicators. This includes wait time, travel time, detour factor and service rate at the traveler's side, as well as vehicle distance traveled, load and profit at the operator's side. A case study area in London is selected to support the presentation of methodology. Results show that encouraging ride-sharing and allowing in-advance requests are powerful tools to enhance service efficiency and equity. Demand information from in-advance requests also enables the operator to plan service ahead of time, which leads to better performance and higher profit.

The thesis concludes that the demand-supply interaction can be effective for defining and assessing the roles of AV technology in our future transportation systems. Combining efficient dispatching strategies and demand information management tools is also important for more affordable and efficient services.

Thesis Supervisor: Jinhua Zhao
Title: Associate Professor of City and Transportation Planning

Thesis Supervisor: Neema Nassir
Title: Senior Postdoctoral Associate of Civil and Environmental Engineering

Acknowledgments

This thesis is made possible through the help and support of many people who have been with me during the past one year and half. My time at MIT would not have been so fabulous without any of you guys.

I would like to thank Professor Jinhua Zhao, my supervisor, for sharing his wisdom and guiding my thinking at every stage of the research. I've enjoyed the deep discussions and sparkling moments with him so much.

Thanks to Transport for London, who has generously funded this research and provided data. I appreciate the constructive conversations with Michael Hurwitz, David Christie and many of their colleagues that have inspired and shaped this thesis.

Thanks to Professor Patrick Jaillet and Professor Nigel Wilson. They have been always willing to listen, question, advise and push my research forward.

Thanks to Neema Nassir and Yuxin Leo Chen. It was a great pleasure for me to be their co-author in research and friend in life.

Thanks to the Transit Lab and JTL Mobility Lab members, for their comments and suggestions, energy and encouragement.

Thanks to all my dear friends across different time zones. They have spent countless days with me, sharing my joy when I was high and lifting my spirits when I was down.

Thanks to my roommates, Xuenan Ni and Xiaoyan Shen, for the dinners and laughters we had together.

Thanks to Jihan Liang. I would have never realized that topology could be so fun until I met her.

Last of all, thanks to my parents for their endless love and unconditional support.

THIS PAGE INTENTIONALLY LEFT BLANK

Contents

List of Figures

THIS PAGE INTENTIONALLY LEFT BLANK

List of Tables

THIS PAGE INTENTIONALLY LEFT BLANK

Chapter 1

Introduction

1.1 Research Questions

We are seeing three emerging trends in the field of urban transportation: mobility on demand, sharing and autonomous vehicles. The concept of mobility-on-demand (MoD) systems describes an innovative mode of transportation in which services are tailored as per the immediate requests. In many applications, non-commercial vehicles from the independent drivers are used and the MoD service providers only operate an online-enabled platform to pair requests with drivers for compensation. Despite many of the debates with regards to regulation and societal impact, getting private vehicles shared ("ride-sourcing") in an on-demand manner does improve the availability of service and enable more affordable trips. In addition, sharing also takes the form of ride-sharing, which makes a way to better utilize the empty seats and therefore reduces the cost per traveler even further. In this thesis, we refer to an MoD system as "shared MoD" only when ride-sharing is enabled[1].

Researchers have been seeking solutions to optimize the design and operation of MoD and shared MoD systems. The efforts have led to system design evaluation [1, 2, 3, 4, 5, 6], advanced dispatching algorithms [7, 8, 9], demand prediction tools [10, 11], dynamic pricing [12, 13], and the pursuit of full autonomy. Powered by the autonomous vehicles (AV) technologies, the conceptual shared autonomous mobility-on-demand (AMoD) systems represent the potentially disruptive and innovative changes to urban transportation in the future. The

[1] Some papers use "shared" for vehicle-sharing (e.g. in car-rental systems and taxi-like hailing systems). In the sense of vehicle-sharing, "shared system" is the opposite to personal trips based on traveler's private car ownership. In this thesis, in order to avoid confusion, we do not adopt this definition. Moreover, "MoD system" itself implies the sharing of vehicles by nature.

design and operation of such a system is therefore an important research direction that requires significant investigation.

The core research questions for this thesis revolve around the dispatching strategies of shared AMoD systems. Based on the defined dispatching methodology, topics such as system design and value of information will also be expanded upon. Collectively, the contribution of this thesis is therefore threefold:

- It addresses the special dispatching need that is critical for effective AMoD operation. This includes a dynamic request-vehicle assignment heuristic for pairing rides on the fly and an optimal rebalancing policy to offset the imbalance between vehicle supply and travel demand. In order to align with the goals for sustainable and integrated urban mobility, the dispatching strategies also reflect transit-oriented designs in two ways. First, the objective function embodies the considerations of service availability and equity through the support of various hailing policies. Second, the service facilitates first-mile connections to public transportation (PT). Dispatching strategies are also customized to deal with the time constraints accordingly.

- It models the interaction between demand and supply through simulation. Using the level of service as interface, this mechanism enables feedback between operators and travelers to more closely represent the choices of both parties. A fixed-point approach is then applied to reach balance iteratively, estimating both the demand volume and the system performance at equilibrium. The results from the simulation support decision-making with regard to comprehensive system design problems such as fleet sizing, vehicle capacities, fare schemes and hailing policies.

- It evaluates the value of demand information through simulation experiments. The different levels of knowledge of demand reflect the uncertainty operators face in collecting travel data, predicting future demand and dispatching vehicles in response to it. To understand the value of information, this part of research quantifies the system performance gain that can be derived from the demand information, thus helps rationalize the efforts operators should spend on data collection, information inference and advanced dispatching algorithms. It eventually contributes to the long-term goal of designing and managing information flow for shared AMoD systems.

This thesis also proposes an agent-based modeling platform, `amod-abm`, for simulating large-scale shared AMoD applications [14]. Agent-based framework has been popular in AMoD research for its advantages in capturing individual behaviors, enabling dynamic operations and accounting for stochasticity. The `amod-abm` platform, specifically, models individual travelers and vehicles with demand-supply interaction and analyzes system performance through a metrics of indicators. This includes wait time, travel time, detour and service rate at the traveler's side, as well as vehicle distance traveled, load and profit at the operator's side. A case study area in London is selected in this thesis to support the presentation of methodology, although the general simulation-based framework can be applied to a variety of urban settings with different operational objectives.

1.2 Organization

The remainder of the thesis is structured as follows.

Chapter 2 reviews some of the most relevant works in the literature and identifies the research areas to which we can contribute. It begins with a presentation of the state-of-the-art large-scale agent-based simulations. The discussion on dynamic dispatching strategies follows, in which existing studies on assignment, rebalancing and integrated AV+PT systems are examined. Chapter 2 also presents applications to information theory to support the research on value of information.

Chapter 3 proposes a demand-supply interaction mechanism for AMoD systems and formulates it as a fixed-point problem. Based on the agent-based simulation platform `amod-abm`, iterative methods are applied to the problem to reach the system balance. This mechanism also reflects the system design needs in operating the shared AMoD service in a real urban setting, for which the case study area of Orpington, London comes in.

Chapter 4 responds to each of the technological challenges in dispatching shared AMoD systems. Request-vehicle assignment, as a variant of the vehicle routing problem, is solved using insertion heuristics and simulated annealing. As for rebalancing, both network-based optimization approach and reinforcement learning approach are presented and compared with each other. The transit-oriented considerations are discussed in the end.

Chapter 5 applies the proposed methodology to the case study and makes recommendations to system design decisions. Decision variables here include hailing policy, fleet size and vehicle capacity and preference to service. A subset of scenarios with ranging variables are deliberately selected for our simulation experiments in order to test the viability of the proposed service in the area.

Chapter 6 presents the role demand information plays in conceptual AMoD systems and connects it to dependent factors such as data collection, information inference and advanced dispatching algorithms. It then proposes two dimensions, level of information and value of information, and builds up the relationship between them through simulation experiments. The results quantify the system performance gain that can be derived from aggregated and individual demand information.

Chapter 7 draws the conclusion and points the directions to future works. It also presents a careful rethink of the system design of the simulation platform.

1.3 Relation with Papers

The author also has two papers that are closely related to this research.

The content in Chapter 3 and Chapter 5 is based on the paper "Transit-Oriented Autonomous Vehicle Operation with Integrated Demand-Supply Interaction" by Jian Wen, Yu Xin Chen, Neema Nassir and Jinhua Zhao [15]. This paper is currently under review at Transportation Research Part C.

The content in Section 4.2 is based on the paper "Rebalancing Shared Mobility-on-Demand Systems: a Reinforcement Learning Approach" by Jian Wen, Jinhua Zhao and Patrick Jaillet [16]. This paper has been published in the conference proceedings of IEEE ITSC 2017 International Conference on Intelligent Transportation.

Chapter 2

Literature Review

2.1 AMoD Simulation

Spieser et al. [17] are among the first to conceptualize the AMoD system as an enabling technology for future urban mobility. Based on the analytical models and actual data, they prove that ideally the total number of vehicles in Singapore could be reduced to one third, assuming all modes of personal transportation are replaced by non-shared AMoD service. They also argue that AMoD reduces the trip cost by half since it eliminates the time consumed for active driving, parking and maintenance in the case of a private vehicle ownership.

Agent-based simulation has recently become popular in AMoD research for its advantages in capturing individual behaviors, enabling dynamic operations and accounting for stochasticity. It also provides analytical tools for evaluating the performance of the defined systems. The successive works by Fagnant and Kockelman are representative for agent-based simulation applications, in which issues such as dynamic ride-sharing, fleet sizing and operational costs have been discussed using a case study in Austin [18, 19]. Similar simulation frameworks could also be found in the applications in Lisbon [4, 5], Shanghai [6], Singapore [3, 7], Seoul [8] and New York [9]. As shown in Table 2.1, the scale and scope of research are being expanded gradually to include fleet management problems, emissions and energy consumptions, and implications on city-level traffic.

Under simplifications with regard to both supply and demand, the aforementioned research papers are able to demonstrate the potential productivity of AMoD systems. On the supply side, early works often assume station-based system. By discretizing the decision

Table 2.1: Large-scale Agent-based Simulation Applications in Literature

Paper	Assumptions					Research Questions
	Study Area	Supply	Demand	Dispatching		
				Assignment	Rebalancing	
[18]	gridded map	AMoD	random trips	nearest	block balance	fleet size, system performance, emissions, energy consumption
[19]	Austin	AMoD (station-based)	2%-10% of all trips	nearest	block balance	fleet size, system performance
[4]	Lisbon	Shared MoD (station-based)	all taxi trips	insertion heuristics	/	fleet size, vehicle capacity, system performance, cost
[5]	Lisbon	Shared AMoD (station-based)	mode choice model	insertion heuristics	/	fleet size, system performance, modal shift, emissions
[6]	Shanghai	AMoD (station-based)	all taxi trips	nearest	block balance	system performance, charging strategy
[3]	Singapore	AMoD	car trips in CBD	nearest	/	fleet size, system performance
[7]	Singapore	AMoD (station-based)	car trips in CBD	nearest	optimal (online/offline)	fleet size, system performance
[8]	Seoul	Shared AMoD	all taxi trips	simulated annealing	/	fleet size, system performance
[9]	New York	Shared AMoD	all taxi trips	optimal (online)	/	fleet size, vehicle capacity, system performance

* "Shared AMoD" follows the definition made in Section 1.1. It could be inconsistent with the definition used in the referred papers.

space, station-based systems reduce the complexity in dispatching and routing. They can also take advantage of the existing studies in graph theories. The latest simulations are gradually moving to free-floating system with door-to-door service. Dynamic ride-sharing has been enabled as well. Some also assume using electronic autonomous vehicles (EAV). In this case, the discussion on charging strategies comes high on the list of priorities.

On the demand side, most of the works only assume arbitrary shift from existing modes. The work by Martinez and Viegas [5] is exceptional. They take time to investigate travel behavior, include non-shared AV as a competing mode and build a nested Logit mode choice model to predict its mode share. Similar method can also be found in [20], which concludes that private AV will result in sharp decline in transit ridership and road congestion will increase consequently. Childress et al. [21] use activity-based travel demand simulation and reach the same conclusion. In response to the induced traffic, many researchers insist on the necessity of ride-sharing. Dynamic pricing has also come into play. Chen and Kockelman [22] and Qiu et al. [23] argue that, if shared AV is used instead of non-shared AV and pricing strategies are designed deliberately, AMoD could capture significant market share to be profitable without inducing extra vehicle miles traveled.

However, none of the existing papers has modeled the interaction between demand and supply as traveler behavior changes in reaction to the system performance. Some assumes private AV ownership [20, 21]. Others study AMoD systems, but their models still ignore travelers' sensitivity to service availability and travel time [5]. Yap et al. [24] survey the preference of travelers to AMoD systems. Based on the mode choice model, they successfully estimate the sensitivity towards different services. However, the connection with supply is still missing in the study. This thesis will address the identified issues with the motivation of developing a systematic approach for demand prediction and system evaluation. The proposed demand-supply interaction will be discussed in full details in Chapter 3.

2.2 Dispatching Strategies

2.2.1 Request-Vehicle Assignment

The assignment of requests to vehicles in real time is critical to the operation of AMoD systems. When ride-sharing is not permitted in the system, simulations often use "nearest" method, which assigns the nearest idle vehicle to the incoming request. If rides are shared, heuristic methods are often used to speed up the problem-solving. One example is the "insertion heuristics", which "inserts" the new request to the job list of an available vehicle while minimizing the total cost imposed on the entire system. Both nearest method and insertion heuristics prevail in large-scale real-time applications for reasons combined with low cost in computation and satisfactory accuracy in solutions.

Researchers are also looking for dynamic algorithms that give better solutions. Jung et al. [8] propose a hybrid simulated annealing algorithm for dynamic request-vehicle assignment. Alonso-Mora et al. [9] devise a more general mathematical model for optimal assignment and use metaheuristics for solutions. The results show that ideally taxi fleet size in New York City could be reduced by 75% when shared AV takes over. However, both dynamic algorithms consume large amount of computational resource. It is proposed that when scale is large, they should be used with limited frequency (e.g. every 10 to 30 seconds) and only when computational capacity allows.

In this thesis, the insertion heuristics and the hybrid simulated annealing defined in [8] are implemented.

2.2.2 Rebalancing

The problem of rebalancing stems from the spatial and temporal mismatch between demand and supply. Within the realm of urban transportation, the existing research works have been largely focused on rebalancing car rental systems [25, 26] and public bike sharing systems [27, 28]. Rebalancing AMoD, on the contrary, is a relatively new topic. Early AMoD applications often adopt naive "block balance" approach, which balances the number of vehicles in each block in an empirical manner.

State-of-the-art works draw on the experience of the car rental and bike sharing counterparts and adopt network-based optimization approaches. Pavone et al. [29] are among the first. Based on the fluid model, the paper proposes an optimal rebalancing model and simulates it on a 12-station AMoD system. In this system, every station will reach equilibrium so that there are excess vehicles and no waiting customers. However, under the influence of its car-rental predecessors, the proposed method is only limited to station-based systems. In addition, it does not address the issue of stochasticity in the demand-supply interplay. In continuation to this work, Spieser et al. [17] transform the model into an analytical guideline for AMoD fleet sizing and validate it in a Singapore case study. This strategical work still remains static and provides little insights to real-time operation.

Zhang and Pavone [30] extend the idea of the fluid model and present a queueing-theoretical approach within the framework of Jackson networks. Many efforts in their work have been made to prove that, as a closed Jackson network, the system is most efficient when inward and outward vehicle flows (including rebalancing flows) are equal at each station. The solution to an offline optimal rebalancing problem is given. They continue that, if taking only current information at a specific time point, the problem could be adopted to online applications. A case study in New York City with around 8,000 non-shared vehicles demonstrates the effectiveness of the method.

Marczuk et al. [7] test both offline and online policies with an agent-based simulation platform using Singapore travel data as input. The results show that about 28% and 23% less vehicles are required to guarantee the same service rate when offline and online rebalancing are in use respectively. Moreover, online policy outperforms the offline one by reducing the average wait time from 11 minutes to 9. Using a similar approach, Spieser et al. [31] tackle the rebalancing issues from the perspective of the fleet operators. They evaluate the operational cost as a function of fleet size, service rate and vehicle utilization and demonstrate that rebalancing can reduce the cost significantly.

Existing works adopting network-based optimization approaches are usually computationally demanding. In addition, as far as the author is aware, all of the works are limited to station-based systems and ride-sharing has also been omitted for the sake of simplicity. However, free-floating and ride-sharing are indeed two key elements that ensure the connectivity and affordability of the AMoD service. The thesis extends the online model in [30] to incorporate both door-to-door service and ride-sharing. It also introduces a reinforcement learning approach for comparison.

2.2.3 Transit-oriented Considerations

As shared AMoD grows, it will take some of the market share of public transportation (PT), unless planned on a mutually complementary basis. The idea of integrated AV+PT systems is first illustrated in [32] as "broadening service options of public transport" by providing multimodal service in less dense areas. Liang et al. [33] use integer programming models to study AV as a last-mile connection to train trips. Vakayil et al. [34] then develop an AV+PT hybrid system and emphasize its potential for reducing total vehicle miles traveled and the corresponding negative externalities such as congestion and emissions. Shen et al. [35] use agent-based simulation to explore the idea of supporting bus operation and planning with AV service as complement. In their paper, high-demand bus routes are preserved while low-demand ones are repurposed and shared AV comes in as an alternative. Results indicate that the integrated system would benefit both AV and PT operators.

In this thesis, the system design reflects transit-oriented designs in two ways. First, the design framework embodies the considerations of transit agencies including service availability and equity. Second, the service includes transit-specific characteristics such as first-mile connections to public transportation. Dispatching strategies are also customized to deal with the in-advance requests and time constraints. It shows that in-advance requests will largely improve the service availability if hailing policy is designed properly. The time-constrained connections are important for seamless transfers to public transportation as well.

2.3 Value of Information

The operation of the mobility-on-demand systems involves the management of the requests in an online manner. On-demand services are essentially desirable for travelers because it provides absolute flexibility both in time and space. However, the dispatching of vehicles would be more efficient and economical if the requests are known in advance. The operators are therefore motivated to understand travel demand, as well as the value that resides in the information.

Borodin and El-Yaniv [36] present the competitive ratio criteria as a metric to evaluate the performance of online algorithms. The competitive ratio is defined as the worst case ratio of the cost of the online (dynamic) algorithm to the cost of an optimal offline (static) algorithm. An algorithm is c-competitive if the competitive ratio of the algorithm is at most c. The c-competitiveness is important to evaluate the gain in performance of an algorithm when moving from zero knowledge to full knowledge of the demand. Jaillet and Wagner [37] then demonstrate that, the best possible competitive ratio of an online algorithm is 2 for the vehicle routing problem with m vehicles, infinite capacity and no precedence constraints. They also give an algorithm that is $(2-\alpha/(1+\alpha))$-competitive when request information is known in advance. α is a measure that is proportional to the reaction time, defined as the period between the time a request is known, and the time it is supposed be picked up. The online algorithms are almost surely asymptotically optimal when α is large enough, that is to say, when all requests are known a priori.

Pillac et al. [38] argue that the competitive ratio criteria has drawbacks when it's applied to real-world applications, since it requires the proof of the c-competitiveness. The value of information presented by Mitrović-Minić et al. [39] constitutes a more flexible and practical metric. This metric indicates the performance of an algorithm based on empirical results and captures the impact of the dynamism on the solution yielded by the algorithm. The larger the value is, the bigger the performance gap of the algorithm will be when a static system becomes completely dynamic. Gendreau et al. [40] report a value of information between 2.5% and 4.1% using their tabu search algorithm for the vehicle routing problem without capacity constraints. Tagmouti et al. [41] report a value larger than 10% for a neighborhood search descent heuristic when arcs in the graph are capacitated.

Given a specific algorithm, the value of information metric can also be used to study the benefits of partial demand information. Larsen et al. [42] define the degree of dynamism as the proportion of requests that are sent on-demand. The simulation results show that increasing the degree of dynamism results in a linear increase in operational cost. They also extend the concept to include temporal attributes such as request time and pick-up window. Diana [43] studies the impact of fleet size and assignment interval (cycle time) on dynamic and partially dynamic systems under different degrees of dynamism. The paper concludes that, when fleet size is small, the performance is more susceptible to the lack of a priori information. The cycle time does not significantly affect the solution.

As far as the author is aware, the scale and scope of the relevant studies have been limited to algorithmic progress and theoretical analysis. None of the existing agent-based simulation applications have addressed the operational needs of determining the value of demand information. In this thesis, the value of information metric incorporates a cost-benefit analysis based on real operational settings. It also extends the degree of dynamism with the motivation of developing a comprehensive approach for the evaluation of various request patterns.

THIS PAGE INTENTIONALLY LEFT BLANK

Chapter 3

Simulation Framework

3.1 System Design

Depending on existing transportation systems, demand patterns, traveler behaviors, social norms, and culture, AMoD services may take different forms to fit the unique travel needs of each individual city. In Wen et al. [15], a guideline to AMoD system design has been proposed. This guideline addresses system design issues in three parts:

- operating modes, including sharing policy, hailing policy and service availability;
- operational variables, which involve determining the best fleet size, vehicle capacity and dispatching strategies;
- pricing scheme, including regular fare structure, discounts and surge.

The system design decisions should reflect the considerations of each party in the system. To that end, Table 3.1 has identified the key stakeholders - travelers, AV operator, PT operator and government - as well as their interests. Note, this table is designed only for evaluating the system performance in this thesis. It is not intended to present a comprehensive list of interests for all stakeholders.

Performance metrics should be selected to evaluate the extent to which the interests of each stakeholder are met. In this thesis, we evaluate the performance using a series of indicators listed in the rightmost column in Table 3.1. For travelers, the level-of-service indicators include availability and total travel time. Availability is represented by service rate, the percentage of travelers being served given their time constraints. Total travel time

Table 3.1: Stakeholders, Interests, Performance Metrics and Indicators

Stakeholder	Interests	Performance Metrics	Indicators
Traveler	level of service	availability	service rate
		total travel time	wait time detour factor
	travel cost	pricing scheme	N/A
AV Operator	financial viability	cost	vehicle distance traveled distance-based load
		revenue	AV mode share
PT Operator	performance	ridership	PT mode share
		availability	N/A
		punctuality	N/A
	financial viability	cost/revenue	N/A
Government	public equity	availability	service rate
		accessibility	N/A
	sustainability	motorized traffic	vehicle distance traveled
		non-motorized trips	active mode share

* "N/A" implies indicators not applicable in the scope of this simulation. These indicators (and those omitted in this table) are important and will be explored more in-depth in following studies.
** We assume PT operator to be not profit-driven.

consists of wait time and in-vehicle travel time. The in-vehicle travel time, when shared, is proportional to the detour factor, defined as the ratio of actual in-vehicle travel time to shortest travel time. As for the AV operator, the supply performance is evaluated by the operational cost. Specifically, the cost of operating one vehicle is represented by average vehicle distance traveled, which indicates the service and rebalancing distance traveled by a single vehicle within an hour. Another indicator related to supply performance is the distance-based average load which measures the average load (number of travelers on board) weighted by the distance it travels. Based on these indicators, the design decisions such as fleet size, vehicle capacity, fare scheme and sharing/hailing policies can be simulated and evaluated. The indicators and jointly the shares of existing modes also shed light on the transport performance of the city as a whole. As such, the stakes of PT operator as well as active mode users will be taken into account.

3.2 Simulation Platform

3.2.1 Demand-Supply Interaction

In this section, we model the interaction between demand and supply in AMoD systems. As shown in Figure 3-1, the interaction mechanism consists of two loops representing the choices of travelers and operators respectively. The closed loop on the left side contributes the fixed-point problem for demand prediction equilibrium, even if level-of-service indicators are unknown beforehand. The open loop on the right side supports the shared AMoD system design by providing feedback to supply performance analysis. The overlapping part of the loops situates the agent-based simulation platform.

Using the level of service as interface, the iterative loop on the left side enables explicit feedback for travelers. To start, the level-of-service indicators are set to arbitrary values. Based on the historical data on all-mode trips, the level of service as well as assumptions with regard to fare and preference to service, the mode choice model predicts the demand volumes for each OD pair. The simulation platform then evaluates the system performance based on the predicted demand matrix, predefined supply settings as well as other system assumptions including dispatching strategies. It outputs both the level-of-service indicators for travelers and the supply performance indicators for operators. The former is returned to the mode choice model as feedback. The results of the demand prediction are updated

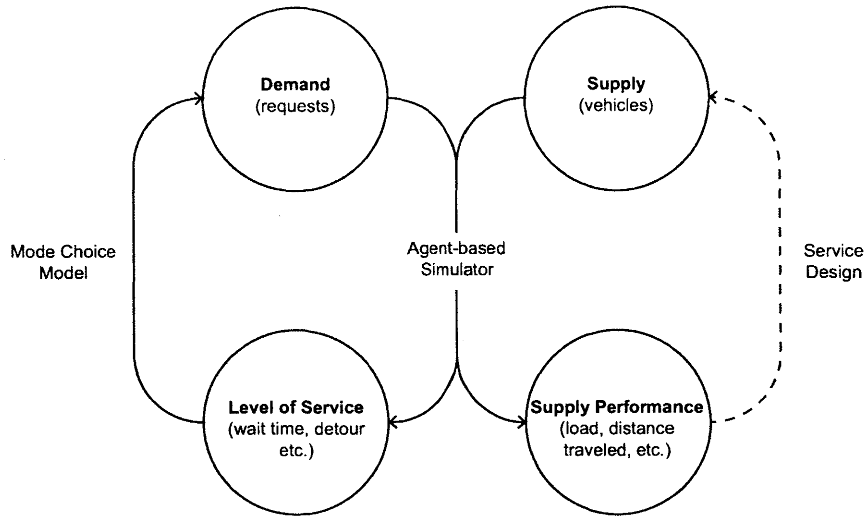


Figure 3-1: The demand-supply interaction mechanism.

accordingly and we establish the iterative loop.

The formulation of the problem is presented as below:

$$\begin{cases} \mathbf{D} = \textsc{ModeChoice}(\mathbf{T}, \mathbf{L}, \mathbf{V}_d, \mathbf{S}_d) \\ \mathbf{L}, \mathbf{P} = \textsc{Simulation}(\mathbf{D}, \mathbf{V}_s, \mathbf{S}_s) \end{cases} \tag{3.1}$$

ModeChoice is the demand prediction subproblem and Simulation is the simulation subproblem. ModeChoice takes total current trips $\mathbf{T}$, level-of-service indicators $\mathbf{L}$, demand decision variables $\mathbf{V}_d$ (e.g. fare) and other demand assumptions $\mathbf{S}_d$ as input. The output $\mathbf{D}$ is a vector of predicted OD-specific demand for AMoD service. Symmetrically, Simulation takes in predicted demand $\mathbf{D}$, supply decision variables $\mathbf{V}_s$ (e.g. vehicle capacity, fleet size, hailing policy) and system assumptions $\mathbf{S}_s$ and gives estimates of level of service $\mathbf{L}$ and supply performance $\mathbf{P}$.

Assuming both sub-problems have been solved, the solution to the interaction problem could be found by applying fixed-point iteration approach. The pseudo code is shown below in Algorithm 1. To deal with the stochasticity inside the simulation, the method of successive averages (MSA) is used to guarantee the convergence as shown in line 9 of the algorithm [44]. The procedure keeps updating $\mathbf{D}$, $\mathbf{L}$ and $\mathbf{P}$ iteratively unless $\mathbf{D}$ has converged by definition. At this time, the demand-supply interaction reaches balance and SolveFixedPoint will return indicators with regard to travelers and operators.

Algorithm 1 Fixed-point Solution

procedure SOLVEFIXEDPOINT($\mathbf{T}$, $\mathbf{V}_d$, $\mathbf{S}_d$, $\mathbf{V}_s$, $\mathbf{S}_s$)
  let level-of-service indicators be arbitrary values $\mathbf{L}^{(0)}$
  $\mathbf{D}^{(0)} = \text{MODECHOICE}(\mathbf{T}, \mathbf{L}^{(0)}, \mathbf{V}_d, \mathbf{S}_d)$
  let step counter $i = 0$
  do
    $i = i + 1$
    $\mathbf{L}^{(i)}, \mathbf{P}^{(i)} = \text{SIMULATION}(\mathbf{D}^{(i-1)}, \mathbf{V}_s, \mathbf{S}_s)$
    $\mathbf{D}^{(i)} = \text{MODECHOICE}(\mathbf{T}, \mathbf{L}^{(i)}, \mathbf{V}_d, \mathbf{S}_d)$
    $\mathbf{D}^{(i)} = \frac{1}{i}\mathbf{D}^{(i)} + \frac{i-1}{i}\mathbf{D}^{(i-1)}$
  while $\|\mathbf{D}^{(i)} - \mathbf{D}^{(i-1)}\| > \delta$
  return $\mathbf{D}^{(i)}$, $\mathbf{L}^{(i)}$, $\mathbf{P}^{(i)}$

On the right side of Figure 3-1, the system design decisions are represented by a dashed line as we do not model the loop explicitly. According to the discussion in Section 3.1, designing an AMoD service involves multiple stakeholders and the system design decisions should be made to reflect the interest of each party. In practice, the comprehensive performance metrics should be defined and examined on a case-by-case basis. Due to limited space, this thesis only evaluates the most important system design decisions from the perspective of key stakeholders in an empirical fashion (as in Table 3.1).

Subsections 3.2.2 and 3.2.3 will discuss the sub-problem SIMULATION. A short presentation to MODECHOICE can be found in Subsection 3.3.2. For full details, please see [15]. The parameters and variables used in the rest of the thesis are listed in Table 3.2 and classified as "input", "output", "decision", "assumption" or "intermediate". Specifically, the supply decision variables are system design decisions that we will discuss in Chapter 5 with various scenarios. The values of the "assumption" variables as well as the demand decision variables are set in Table 5.2 before we start the simulation experiments.

3.2.2 Simulation Framework

The level-of-service indicators in $\mathbf{L}$ that have impact on the mode choice behavior are service rate SR, wait time WT and detour factor DF. $\mathbf{L}$ is largely dependent on system design and operational strategies and should be consequently studied together with the supply side. For that reason, we cast the sub-problem $\mathbf{L}, \mathbf{P} = \text{SIMULATION}(\mathbf{D}, \mathbf{V}_s, \mathbf{S}_s)$ into a continuous-time agent-based simulation platform. The platform is able to simulate door-to-door shared AMoD service with a fixed-size fleet of dedicated autonomous vehicles (defined

Table 3.2: Parameters and Variables

Vector	Parameter/Variable	Type
level of service ($\mathbf{L}$)	service rate (SR)	output
	wait time (WT)	output
	detour factor (DF)	output
supply performance ($\mathbf{P}$)	vehicle distance traveled (VMT)	output
	distance-based load (L)	output
supply decision variables ($\mathbf{V}_s$)	vehicle capacity (K)	supply decision
	fleet size (V)	supply decision
supply assumptions ($\mathbf{S}_s$)	maximum wait time (MWT)	assumption
	maximum detour factor (MDF)	assumption
	period of simulation (T)	assumption
	period of study (T_s)	assumption
	period of warm-up (T_w)	assumption
	period of cool-down (T_c)	assumption
	interval of assignment (T_a)	assumption
	interval of rebalancing (T_r)	assumption
	(others, see Subsection 3.2.2)	assumption
demand decision variables ($\mathbf{V}_d$)	base fare (c_{base})	demand decision
	per-unit-time fare (c_{time})	demand decision
	per-unit-distance fare (c_{dist})	demand decision
	discount for sharing (DC)	demand decision
predicted demand matrix ($\mathbf{D}$)	(see Subsection 3.3.2)	intermediate
total current trips ($\mathbf{T}$)	(see Subsection 3.3.2)	input
demand assumptions ($\mathbf{S}_d$)	preference to AV (ASC)	assumption
	penalty wait time (PWT)	assumption
	(others, see Subsection 3.3.2)	assumption
supporting variables	cost of an AMoD trip (C)	intermediate
	actual travel time (TT)	intermediate
	shortest travel time (ST)	intermediate
	shortest travel distance (SD)	intermediate
	adjusted wait time (AWT)	intermediate

* Types "input" and "output" represent the inputs and outputs of the model respectively. Type "intermediate" represents the intermediate variables in the formulation. Type "demand decision" and "supply decision" for decision variables. Type "assumption" for assumptions.

** SR, WT and DF require being initialized to start the iteration.

by vehicle capacity K and fleet size V in $\mathbf{V}_s$), requests (reflecting the predicted demand $\mathbf{D}$ from ModeChoice) and the necessary operational models and dispatching strategies at operator's disposal (as assumptions in $\mathbf{S}_s$). The pseudo code for Simulation is shown in Algorithm 2.

Algorithm 2 Agent-based Simulation

```
procedure SIMULATION(D, V_s, S_s)
    initialize the system according to V_s and S_s
    t = 0, t_a = 0, t_r = 0
    while t < T do
        generate next request with arrival interval Δt and push into queue
        t = t + Δt
        if t > t_a then
            assign pending requests in queue to vehicles
            t_a = t_a + T_a
        if t > t_r then
            rebalance the idle vehicles
            t_r = t_r + T_r
        route the vehicles to t
    return L, P based on service performance during T_s
```

Three modules are fundamental in the agent-based simulation platform: request generator, vehicle dispatcher and routing server.

- The demand generator draws requests from the predefined OD list $\mathbf{D}$. The arrival of the requests follows a Poisson process of constant arrival rate, which is proportional to the OD-specific demand volume. Depending on hailing policy, a request can be either on-demand or in-advance and have specific constraints including maximum wait time (MWT) and maximum detour factor (MDF). Earliest possible departure time and latest possible arrival time also apply accordingly. The latest possible arrival time is important especially when travelers are making first-mile trips and transfers to PT are necessary.

- On-demand requests are then dynamically assigned to vehicles by the central dispatcher based on insertion heuristics. If a request cannot be served within the wait time window due to vehicle availability and request constraints, it times out and the traveler is assumed to "walk away"; i.e., leave the system. The service rate is then defined as the percentage of requests being served. The insertion heuristics also apply to in-advance requests. The only distinguishing characteristic is that in-advance requests are known to the dispatcher beforehand and travelers will be notified of their assignment 30 minutes before the earliest departure time. In addition, the dispatcher rebalances idle vehicles periodically to regain the balance between demand and supply and plan for the coming requests in the short future. In order to balance the trade-off between optimality and computational efficiency, the request-vehicle assignment is performed every T_a simulated seconds and rebalancing is performed every T_r seconds. Details on both assignment and rebalancing strategies will be discussed in Chapter 4.

- After dispatching, the routing server updates the shortest routes in real-time. Each vehicle, behaving as an individual agent, moves accordingly to pickup/drop-off travelers as well as to rebalance. The travel time between two locations is static, based on the average morning peak-hour travel time provided by OpenStreetMap.

The simulation runs for T seconds, of which T_s seconds makes the period of study. Requesting generated during T_s are used in the evaluation of level of service $\mathbf{L}$ and supply performance $\mathbf{P}$. $\mathbf{L}$ includes service rate, wait time and detour factor. $\mathbf{P}$ consists of vehicle distance traveled and distance-based load. The rest of the simulation time before and after T_s are warm-up and cool-down buffers. $T = T_w + T_s + T_c$.

To start the interaction and solve the fixed-point problem, we initiate service rate, wait time and detour factor in $\mathbf{L}$ with arbitrary values. ModeChoice takes $\mathbf{L}$ as input and outputs $\mathbf{D}$. Simulation then takes $\mathbf{D}$ for $\mathbf{L}$ and $\mathbf{P}$. The demand-supply interaction continues iteratively. It reaches the balance and stops when MSA condition is satisfied.

3.2.3 Implementation

The simulation platform is implemented in Python 3 and C++. An open-source version of the code can be found in the GitHub repository `amod-abm` [14]. The implementation follows the object-oriented design approach so that data structures are encapsulated in multiple classes and interfaces are provided to define how an object can interact with others. As of today, the following parts have been ready:

- class `Model` for shared AMoD systems, with a fleet of autonomous vehicles and a central dispatcher;
- the demand matrix $\mathbf{D}$ from ModeChoice sub-problem, with time-invariant demand volumes for a list of OD pairs;

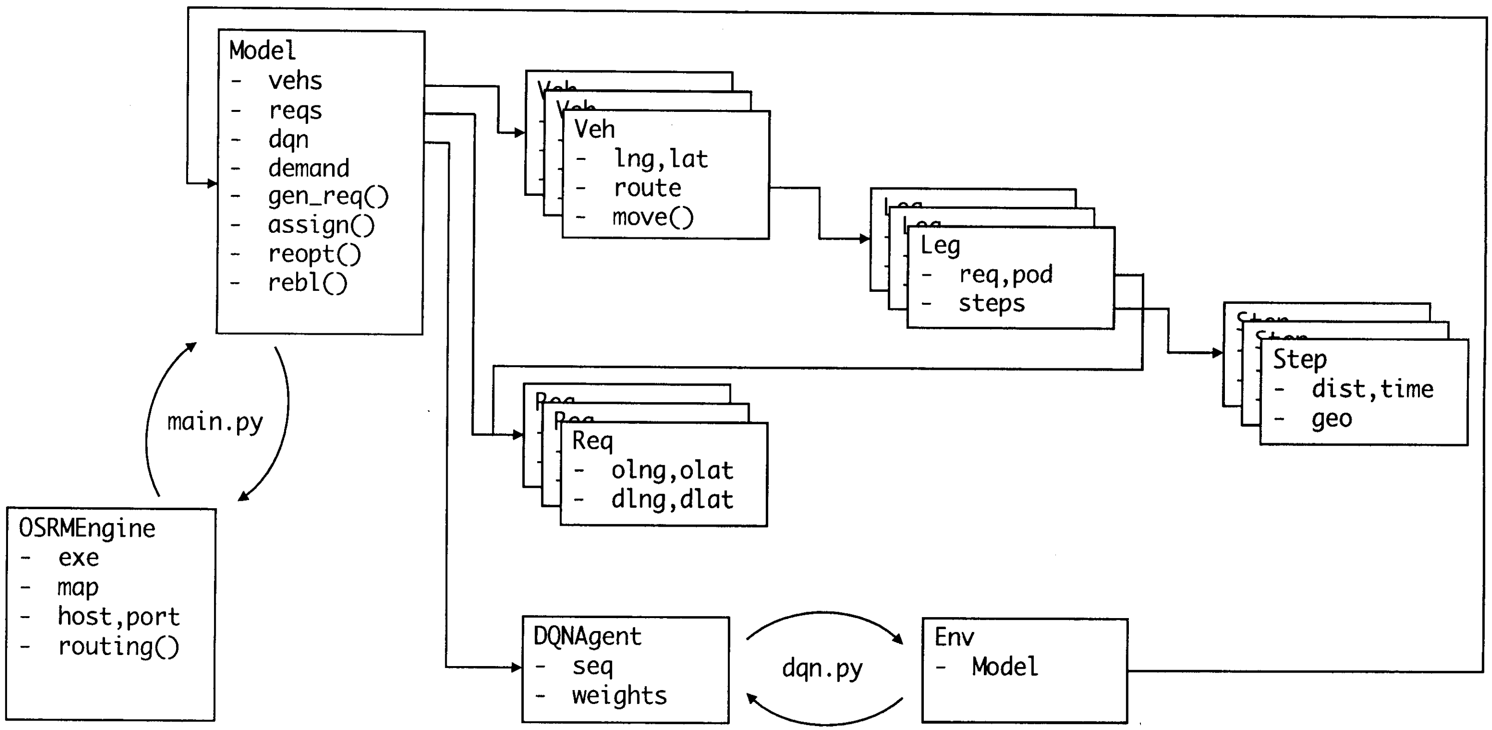


Figure 3-2: The class dependency in project `amod-abm`.

- class `Veh` for (shared) autonomous vehicles, the capacity of which can be set to 1 (no sharing), 2 (at most 2 travelers sharing at a time) or more;
- class `Req` for requests generated based on the demand matrix; requests can be either on-demand or in-advance depending on hailing policy;
- class `OSRMEngine` for connecting to the Open Source Routing Machine (OSRM), an internal routing server linked to OpenStreetMap database [45];
- class `Leg` and class `Step` for formating route information; the route of a vehicle may include one or several legs and each leg consists of multiple steps; each step is a connected sequence of straight line segments (polyline) and the travel speed within a step is constant;
- class `DQNAgent` and class `RebalancingEnv` for a deep Q network (DQN), which provides a reinforcement learning environment for the rebalancing problem; DQN works as a complement of the network-based rebalancing approach.

The dependency between classes is illustrated in Figure 3-2.

3.3 Case Study

3.3.1 Case Study Area Selection

In order to demonstrate the capability of the simulation platform in a real urban setting, we select a case study area in the southeast of London. The Greater London Area (GLA) has an extensive and developed transportation network in which public transport has a high mode share (45% in 2015). Commuter rail has shown strong growth over the past decade, providing good service from the outskirt of GLA to central London ("downtown").

The case study area, Orpington, is a spread-out residential area located about 25 kilometers outside of central London. It is centered around a commuter rail station (Orpington station) with frequent and high-speed train service to downtown. However, bus service in this area is infrequent and not economically efficient as a result of the low residential density. Consequently, local trips are particularly car-dependent. The area is chosen as the case study area because (a) it possesses a significant first-mile demand to the train station, (b) the inefficient bus service requires improvement, and (c) it has appropriate density and road infrastructure for initial AV trials. In this thesis, the local administrative boundary of Orpington is enlarged to include all significant bus routes originating from the rail station. This enlarged area, 15km×10km and home to around 159 thousand people, will be referred to as the Case Study Area (CSA).

3.3.2 Mode Choice Model

The mode choice sub-problem $\mathbf{D} = \textsc{ModeChoice}(\mathbf{T}, \mathbf{L}, \mathbf{V}_d, \mathbf{S}_d)$ is built upon a Nested Logit model. It is driven by historical trip data $\mathbf{T}$ and responsive to the level of service $\mathbf{L}$ as well as supply characteristics defined in $\mathbf{V}_d$ and $\mathbf{S}_d$.

Current Trips

$\mathbf{T}$ relies on an annually sampled travel survey of local households. Pseudonymized trips made by households in CSA during the morning peak (6:30am to 9:30am) from 2005 to 2014 are used. This proportion contains 2709 respondents and accounts for around 74,000 trips made by all residents after expansion. The all-mode demand is around 28,000 trips/h.

Table 3.3 outlines the current mode share of the seven modes: walk, bike, car, taxi, bus, rail and park/kiss+ride (P/K+R). "Trips to Downtown" represents trips that have origins in CSA and destinations in downtown (account for 11% of all trips). "Intrazonal Trip" represents trips that have both origins and destination in CSA (account for 54% of all trips). Note that "All Trips" consists of trip segments other than"Intrazonal Trips" and "Trips to Downtown", e.g. the trips from CSA to another area.

It can be noticed that CSA is very car-dependent with 58% of all trips and 57% of intrazonal trips using car. Trips to downtown are dominated by rail service because of the strong rail connection. 74% of them are using rail as main mode, another 16% rely on car as the first-mile access. The mode share of bus is very small due to the inefficient service. Walk is popular for short intrazonal trips. Bike and taxi trips are very rare for all trips.

AMoD Demand Analysis

We suppose that individual traveler chooses among all available transportation modes for a specific trip and the discrete choice behavior follows the Nested Logit model. When AMoD service becomes available, the probability of shifting from any existing mode to AV depends

Table 3.3: Current Mode Share in CSA

Mode	All Trips	Trips to Downtown	Intrazonal Trips
Walk	20%	0%	32%
Bike	1%	0%	1%
Car	58%	10%	57%
Taxi	0%	0%	0%
Bus	8%	0%	10%
Rail	11%	74%	0%
P/K+R	2%	16%	0%

on the utility of the new mode as well as the competence of the existing modes. We also assume no latent demand in this research.

As for the utility, assuming **L** is ready, average wait time and detour factor can jointly determine the travel time of AMoD service. When service rate goes below 100%, a penalty wait time is also introduced for those who walk away. Equations are therefore:

$$AWT = WT \times SR + PWT \times (1 - SR) \tag{3.2}$$

and

$$TT = AWT + ST \times DF \tag{3.3}$$

AWT represents the adjusted wait time, which is calculated using the average wait time WT for those served, the service rate SR, and the penalty wait time PWT for those rejected. We also have detour factor DT and ST for shortest travel time. TT is therefore the actual travel time for an AV leg.

Another assumption that should be made is ASC. ASC reveals the intrinsic preference of travelers to the proposed AMoD service when all independent variables such as travel time and travel cost are controlled. However, due to the lack of existing services and the uncertainty in system design, the knowledge of the preference is still limited. In this thesis, we propose to test a wide range of ASC in the simulation. Since the new mode inherently takes characteristics of both car and transit modes, in this case, three ASCs are chosen to reflect this range: -3.58 (lower bound benchmarked by P/K+R), -2.35 (upper bound benchmarked by car) and -3.00 (the midpoint case, average of bus and rail).

Decision variables $\mathbf{V}_d$ in the utility function includes fare, which is important for AV operator as well as other stakeholders as discussed in Section 3.1. In this thesis, the fare scheme using base fare, per-unit-distance, and per-unit-time is adopted:

$$C = (c_{base} + c_{time}ST + c_{dist}SD)(1 - DC) \tag{3.4}$$

C is the cost. ST and SD are shortest travel time and shortest travel distance. c_{base}, c_{time} and c_{dist} are base, per-unit-distance, and per-unit-time parameters respectively. To promote service goals relating to traffic congestion and sustainability, we also assume every trip has a chance to be shared and travelers cannot reject sharing. In that case, a universal sharing discount DC is applied to remunerate travelers.

Based on the utility function of AMoD service for each of the OD pairs, the mode choice model is able to predict its demand volume. The output list of OD-specific demand volumes $\mathbf{D}$ is used in the simulation.

THIS PAGE INTENTIONALLY LEFT BLANK

Chapter 4

Dispatching Strategies

4.1 Request-Vehicle Assignment

4.1.1 Vehicle Routing Problem

The static version of the request-vehicle assignment problem has been intensively studied in the literature as the vehicle routing problem (VRP). A typical VRP describes an integer programming problem for designing optimal collection/delivery routes from one or several depots to a number of geographically scattered customers, subject to side constraints. The most common side constraints include vehicle capacity, time windows as well as the precedence relations between collections and deliveries. Exact algorithms for the VRP can be classified into three broad categories: (a) direct tree search methods; (b) dynamic programming, and (c) integer linear programming [46]. Depending on formulation, constraints and size of problem, many variants of the aforementioned algorithms have been ready to use for fast and optimal solutions.

When it comes to dynamic ride-sharing problem, heuristic methods are often used in the literature for the purpose of speeding up the assignment process. Fundamental heuristic methods such as the nearest neighbor algorithm, insertion algorithms and tour improvement procedures can be applied to various VRPs almost without modifications. Metaheuristic algorithms including simulated annealing and tabu search may also follow up to "reoptimize" a feasible solution in the pursuit of better accuracy. Due to the limitation in time and computational capacity, metaheuristics in large-scale applications are often interrupted before optimality can be achieved.

Most studies on dynamic ride-sharing consider one of the following specific objectives when assigning requests to vehicles:

- minimize the system-wide vehicle miles traveled for the operator;
- minimize the system-wide travel time for the travelers;
- maximize the profit/benefit for the entire system assuming profit-driven or publicly-owned operator [47].

The last objective function is most comprehensive, which usually takes the form of a monetary combination of the first two items. However, due to the uncertainty in many aspects, the cost and revenue of operating a shared AMoD system still remain unknown. For simplicity and measurability, many state-of-the-art agent-based simulations mentioned in Section 2.1 continue with either travel time or vehicle miles as the objective.

In this thesis, we propose to use system-wide travel time. Consider a shared AMoD system with a fleet of shareable autonomous vehicles. $\mathcal{V}$ represents the entire fleet of V vehicles; each vehicle $v \in \mathcal{V}$ can be shared by at most K travelers at a time. s_v and e_v are the start and end locations of vehicle v. The group of all start locations s_v for all $v \in \mathcal{V}$ makes the set $\mathcal{N}_s$; the group of e_v makes $\mathcal{N}_e$. At the time of study, vehicle v has n_v travelers on board. A traveler r on board of vehicle v (noted as $r \in \mathcal{R}_{o,v}$) has an drop-off node d_r in the graph. An incoming traveler pending to be paired (noted as $r \in \mathcal{R}_i$) has both pick-up node p_r and drop-off node d_r. The pick-up and drop-off nodes are grouped into two sets $\mathcal{N}_p$ and $\mathcal{N}_d$. We also let $\mathcal{R}_o = \cup_{v \in \mathcal{V}} \mathcal{R}_{o,v}$, $\mathcal{R} = \mathcal{R}_o \cup \mathcal{R}_i$, and $\mathcal{N} = \mathcal{N}_s \cup \mathcal{N}_e \cup \mathcal{N}_p \cup \mathcal{N}_d$. The formulation to the request-vehicle assignment problem is therefore as below:

$$
\begin{aligned}
\min \quad & \textstyle\sum_{r \in \mathcal{R}_o} t_{d_r} + \sum_{r \in \mathcal{R}_i} \left(\beta t_{p_r} + (t_{d_r} - t_{p_r})\right) \qquad (4.1)\\
\text{s.t.} \quad & t_n \geq 0, \forall n \in \mathcal{N} \\
& x_{i,j,v} \in \{0,1\}, \forall i,j \in \mathcal{N}, i \neq j, \forall v \in \mathcal{V} \\
& n_{i,j,v} \in \{0,1,\cdots,K\}, \forall i,j \in \mathcal{N}, i \neq j, \forall v \in \mathcal{V}
\end{aligned}
$$

The decision variables are the visit times of all nodes, in which t_n represents the visit time of node n. The objective of the problem is therefore to minimize the total travel time of all travelers in the system. This includes the remaining in-vehicle travel times for those on board, as well as the sum of the wait times (weighted by coefficient $\beta \geq 1$) and the in-vehicle travel times for those pending to be paired.

We also have $x_{i,j,v}$ and $n_{i,j,v}$ for all pairs of $i, j \in \mathcal{N}, i \neq j$ and all vehicles in $\mathcal{V}$ as the supporting variables. $x_{i,j,v}$ is binary, indicating the existence of a leg from i to j in the planned route of vehicle v. $n_{i,j,v}$ is an integer not greater than K, representing the number of travelers on board of vehicle v in the journey from i to j. The optimization problem is therefore a mixed integer linear programming problem (MILP).

The problem is also subject to the following constraints.

$$t_j \geq t_i + t_{i,j}, \exists v \in \mathcal{V}, x_{i,j,v} = 1 \tag{4.2}$$

Constraint 4.2 states that, the visit time of node j should be at least $t_{i,j}$ time units later than the visit time of node i if j is visited by vehicle v right after i. $t_{i,j}$ is the static travel time from i to j in a predefined road network. We also assume there is no depot and a vehicle ends its service just at the location it drops off the last traveler. Consequently, $t_{i,j}$ is zero for all $i \in \mathcal{N}$ if $j \in \mathcal{N}_e$.

$$\hat{T}_n \leq t_n \leq \bar{T}_n, \forall n \in \mathcal{N}_p \cup \mathcal{N}_d \tag{4.3}$$

$$t_{p_r} \leq t_{d_r}, \forall r \in \mathcal{R}_i \tag{4.4}$$

Constraint 4.3 assures that the pick-up and drop-off times of any node n should satisfy the earliest service time $\hat{T}_n$ and the latest service time $\bar{T}_n$ defined by the hailing policy. Constraint 4.4 states that for any traveler in the system, the pick-up should be always made before the drop-off.

$$\sum_{i' \neq j} \sum_{v} x_{i',j,v} = \begin{cases} 0, & j \in \mathcal{N}_s \\ 1, & \text{otherwise} \end{cases}, \forall j \in \mathcal{N}, v \in \mathcal{V} \tag{4.5}$$

$$\sum_{j' \neq i} \sum_{v} x_{i,j',v} = \begin{cases} 0, & i \in \mathcal{N}_e \\ 1, & \text{otherwise} \end{cases}, \forall i \in \mathcal{N}, v \in \mathcal{V} \tag{4.6}$$

Constraints 4.5 to 4.6 assure that each vehicle should start and end its route at its predefined start and end locations. Also, each pick-up node and drop-off node in the graph should be visited exactly once.

$$\sum_{j' \neq s_v} \sum_{v' \neq v} x_{s_v, j', v'} = 0, \forall v \in \mathcal{V} \tag{4.7}$$

$$\sum_{i' \neq e_v} \sum_{v' \neq v} x_{i', e_v, v'} = 0, \forall v \in \mathcal{V} \tag{4.8}$$

Constraints 4.7 to 4.8 state that the start and end locations of any vehicle must not be visited by the rest of the fleet.

$$\sum_{i' \neq d_r} x_{i', d_r, v} = \sum_{j' \neq d_r} x_{d_r, j', v} = 1, \text{ if } \exists v \in \mathcal{V}, r \in \mathcal{R}_{o,v} \tag{4.9}$$

$$\sum_{i' \neq p_r} x_{i', p_r, v} = \sum_{j' \neq p_r} x_{p_r, j', v} = \sum_{i' \neq d_r} x_{i', d_r, v} = \sum_{j' \neq d_r} x_{d_r, j', v}, \forall v \in \mathcal{V}, \text{ if } r \in \mathcal{R}_i \tag{4.10}$$

Constraints 4.9 to 4.10 guarantee that, if a traveler is already on board of vehicle v, its drop-off node should also be visited by vehicle v; otherwise, a traveler should be picked up and dropped off by the same vehicle.

$$n_{i,j,v} = 0 \text{ if } x_{i,j,v} = 0, \forall i, j \in \mathcal{N}, i \neq j, \forall v \in \mathcal{V} \tag{4.11}$$

$$\sum_{j' \neq s_v} n_{s_v, j', v} = n_v, \forall v \in \mathcal{V} \tag{4.12}$$

$$\sum_{i' \neq e_v} n_{i', e_v, v} = 0, \forall v \in \mathcal{V} \tag{4.13}$$

$$\sum_{i' \neq j} \sum_{v} n_{i', j, v} + 1 = \sum_{k' \neq j} \sum_{v} n_{j, k', v}, \text{ if } j \in \mathcal{N}_p \tag{4.14}$$

$$\sum_{i' \neq j} \sum_{v} n_{i', j, v} - 1 = \sum_{k' \neq j} \sum_{v} n_{j, k', v}, \text{ if } j \in \mathcal{N}_d \tag{4.15}$$

Constraints 4.11 to 4.15 set the rules for loading. A vehicle v should start at s_v with n_v travelers on board and finishes the service at e_v being vacant. Also, the number of travelers on board should increment or decrement as the vehicle picks up or drops off travelers along the planned route respectively.

4.1.2 Algorithms

The aforementioned formulation describes a shared AMoD system at any time point. As discussed, heuristic methods are appropriate for solving such an MILP problem. This part of the thesis will present two methods that are popular in the literature: insertion heuristics (IH) for fast and feasible solutions, and simulated annealing (SA) for reoptimization. Both methods have been tested in this study using the agent-based simulation platform. Results show that, insertion heuristics outperforms simulated annealing for its computational efficiency. The performance of IH is also close to that of SA. Since large-scale real-time applications should always guarantee feasible solutions within short time constraints, we propose to use IH in the following chapters of the thesis.

Insertion Heuristics

Insertion heuristics finds an approximate solution to the problem by considering each new request individually and independently from other passenger requests. In this case, the global optimality is traded in exchange of a series of local optima for computational speed. In this thesis, the proposed IH serves the requests on a first-come-first-serve basis, and it searches for the best available vehicle for each of the new requests unless none of the vehicles satisfies its time and capacity constraints.

The algorithm for IH is as below:

Algorithm 3 Insertion Heuristics

```
procedure AssignRequest(R_i, V)
    c = +∞
    pop all pending requests from queue R_i
    for each request do
        for each vehicle in V do
            l = length of the vehicle's job list
            for i from 0 to l do
                for j from i to l do
                    insert pick-up and drop-off into the job list at positions i and j
                    if new job list satisfies all constraints then
                        if incremental cost < c then
                            update c with the incremental cost
                            update v*, i*, j* accordingly
        if c remains +∞ then
            if request is still in the wait time window then
                push the request back to queue
            else
                request times out
        else
            insert request to v*: pick-up at position i*, drop-off at j*
```

The job list of a vehicle is defined as "a series of pick-ups and drop-offs with scheduled time". Each vehicle then follows its individual job list to provide service to travelers. For each of the new requests, Algorithm 3 inserts its pick-up and drop-off jobs to the best positions by minimizing the incremental cost it imposes on all the travelers including itself. The cost defined in Equation 4.1 is used. If none of the vehicles is able to serve, Algorithm 3 pushes the request back to the pending queue. Request times out and is removed from the queue as long as the system time goes beyond its latest pickup time.

Insertion heuristics shows reasonable computational efficiency with the time complexity of $\mathcal{O}(R \cdot V \cdot K^2)$, in which R is the total number of requests, V is the fleet size and K is the vehicle capacity. Since IH does not consider all new requests at the same time, it may potentially lead to a sub-optimal solution. This leads to the reoptimization discussion in the next part.

Simulated Annealing

Simulated annealing is a generic probabilistic metaheuristic method which improves the accuracy of existing solutions in systems with large search space. The name of SA comes from annealing in metallurgy, a technique involving heating and controlled cooling of a material to improve its performance. SA implements controlled cooling by denoting a temperature $Temp$. It then explores the search space and accepts better solution with probability 1 and worse solution with probability proportional to $\exp(-1/Temp)$. During the search, the temperature is progressively decreased and SA ends with an accepted solution other than the original one.

Simulated annealing provides a reoptimization scheme to the solution of IH. The algorithm for SA is as below. Typical perturbations of a solution include swap and move. A "swap" exchanges two random requests from the job lists of two vehicles; a "move" removes a random request from its vehicle's job list and inserts it into a new vehicle.

The complexity of the simulated annealing depends on the speed of cooling down. Low start temperature leads to low probability of accepting worse solutions, therefore makes it less probable to jump out of the local optimum. Unnecessarily high start temperature, on the other hand, may result in totally random search in the beginning. The selection of the parameters also has significant impact on the quality of the solution, thus should be performed with great caution. In order to achieve better accuracy, we may also call

Algorithm 4 Simulated Annealing

procedure REOPTIMIZE($\mathcal{V}$)
  initialize $Temp$, $\Delta Temp$, k
  let S_0 be the initial solution according to the job lists from $\mathcal{V}$
  let new solution $S = S_0$
  while $Temp >$ cool-down temperature **do**
    randomly select a perturbation of S as S'
    if S' is better than S **then**
      accept and $S = S'$
    else
      accept and $S = S'$ with probability $\exp(-k(S' - S)/Temp)$
    decrease $Temp$ by $\Delta Temp$
  if S is better than S_0 **then**
    return new solution S
  else
    return original solution S_0

Algorithm 4 iteratively until the solution converges.

4.2 Rebalancing

4.2.1 Algorithms

In this section, we incorporate ride-sharing into the proposed free-floating shared AMoD system and propose three rebalancing approaches. The first approach extends the network-based online model in [30], to which a branch-and-bound heuristic is applied for fast solution. The second approach is based on reinforcement learning, which adopts a deep Q network (DQN) and adaptively moves idle vehicles to regain balance. This approach takes a very different perspective from the state-of-the-art network-based methods since it's model-free and able to cope with real-time systems with partial or full data availability. The last one is a simple anticipatory approach for performance comparison.

The contribution of this part is therefore twofold: (1) it makes a transition from the station-based systems (discrete space) to free-floating systems (continuous space) and adds ride-sharing features to the model; (2) the proposed DQN-based approach for rebalancing is real-time, demand predictive, computationally scalable and has good performance in terms of both level of service and operational cost.

Consider a shared AMoD system covering a predefined service area. For operational purposes, the area is divided into S transport analysis zones $\mathcal{S} = \{1, 2, \ldots, S\}$. The set of fleet $\mathcal{V}$ consists of V shareable vehicles of capacity K. Travelers with origins and destinations in $\mathcal{S}$ send requests and queue up. Travelers are then assigned to vehicles dynamically by the central dispatcher on a first-come-first-serve basis and no traveler will "walk away" despite the possible long wait.

A vehicle having been assigned to travelers is "in service". Otherwise, it is "idle" and available to be rebalanced. We assume that the online rebalancing algorithm is run every T_r seconds and at time $t = T$, one run is triggered. The period of study is therefore $[T, T + T_r]$. We also assume that, over the period, the number of incoming requests A_i in zone i follows the Poisson process with predicted arrival intensity λ_i, that is to say, $A_i \sim \text{Poisson}(\lambda_i T_r)$. Knowing both the demand distribution and the vehicle status, the objective of rebalancing is therefore to maximize the service availability while limiting the rebalancing cost. Service availability is evaluated by the average wait time of requests emerging within the period; rebalancing cost is represented by total vehicle rebalancing distance traveled, that is, the total distance covered by all vehicles in the fleet due to rebalancing.

Optimal Rebalancing Problem

The decision variables in the rebalancing problem are $r_{i,j}$ for $i, j \in \mathcal{S}$. $r_{i,j}$ represents the number of rebalancing vehicles sent from zone i to zone j at $t = T$, i.e., the beginning of the period. Specially, if $i = j$, it represents the number of idle vehicles in zone i that remain unmoved during the period. The decision variables satisfy $\sum_{j \in \mathcal{S}} r_{i,j} = r_i$, in which r_i is the number of idle vehicles available to be rebalanced in zone i when $t = T$.

The cost of rebalancing one vehicle from i to j is noted as $c_{i,j}$. In this thesis, $c_{i,j}$ is defined as below:

$$c_{i,j} = \begin{cases} c d_{i,j} & \text{if } j \text{ is accessible from } i \text{ within } T_r \quad (4.16a) \\ \bar{c} & \text{otherwise} \quad (4.16b) \end{cases}$$

In Equation 4.16a, $d_{i,j}$ is the distance from i to j and the cost is proportional to the distance with a multiplier c; in Equation 4.16b, the cost is set to be a large constant $\bar{c}$ when j is too far away from i. This is to discourage long rebalancing and guarantee that all rebalanced vehicles can arrive at their destinations before the period ends.

We assume that both in-service and idle vehicles follow the shortest routes when picking up/dropping off travelers and rebalancing. We also assume that the planned routes are not influenced by the incoming requests over the period and travel time is deterministic. Consequently, the level of supply at $t = T + T_r$, i.e., the end of the period, could be measured by the availability of both idle and in-service vehicles at that time. Note, a vehicle is classified as "idle" or "in-service" only according to its status before rebalancing starts.

The availability of idle vehicles at $t = T + T_r$ is measured by r'_j, the number of idle vehicles that zone j will receive at the end. $r'_j \triangleq \sum_{i \in \mathcal{S}} r_{i,j}$. An in-service vehicle may also become available (if all on-board passengers have been dropped off) or partially available (if it still has passenger(s) on board but admits ride-sharing) at the end. Similarly, we define s'_j as the availability of in-service vehicles in zone j at $t = T + T_r$. s'_j could be expressed as:

$$s'_j = \sum_{v \in \mathcal{V}} I'_j(v) w(l'_v), \forall j \in \mathcal{S} \tag{4.17}$$

$I'_j(v)$ is the indicator function that equals 1 if v is in zone j when $t = T + T_r$ and 0 otherwise. $w(l'_v)$ is the load-availability factor, defined as:

$$w(l'_v) = \frac{\hat{p}(l'_v)}{\hat{p}(0)}, \text{ for } 0 \leq l'_v \leq K \tag{4.18}$$

l'_v is the load of v when $t = T + T_r$. In a shared AMoD system with fixed settings, $\hat{p}(l'_v)$ is the empirical probability that a vehicle of load l'_v will be assigned to the new request based on simulation results. Similarly, $\hat{p}(0)$ is the empirical probability that an empty vehicle will be assigned to the new request. For a vehicle of capacity 4, we have $w(0) = 1.0$ and $w(\cdot) = 0.4, 0.2, 0.1, 0.0$ when load is $1, 2, 3, 4$ respectively. The estimated total supply v'_j when $t = T + T_r$ is therefore:

$$v'_j = \lfloor r'_j + s'_j \rfloor, \forall j \in \mathcal{S} \tag{4.19}$$

The objective of maximizing the areawide service availability is translated to maximizing the total expected number of requests b that can be served by vehicles from the same zone at the end of the period. $b = \sum_{j \in \mathcal{S}} b_j(v'_j)$ and $b_j(v'_j)$ represents the expected number of served requests in zone j during T_r, knowing that v'_j vehicles are in supply:

$$b_j(v'_j) = \sum_{k=0} \min(k, v'_j) \mathbb{P}(A_j = k) \tag{4.20}$$

Based on the definitions and assumptions, the optimal rebalancing problem (ORP) is as follows:

$$\begin{aligned} \max_{r_{i,j}} \quad & \textstyle\sum_{j\in\mathcal{S}} b_j(v'_j) - \sum_{i,j\in\mathcal{S}} c_{i,j} r_{i,j} \\ \text{s.t.} \quad & \textstyle\sum_{j\in\mathcal{S}} r_{i,j} = r_i, \forall i \in \mathcal{S} \\ & r_{i,j} \in \mathbb{N}, \forall i,j \in \mathcal{S} \end{aligned} \tag{4.21}$$

The equation 4.21 is a Mixed Integer Nonlinear Programming (MINLP) problem. When problem size is large, solving MINLP is extremely computationally burdensome and in this thesis, we use a combination of incremental-optimal and branch-and-bound methods to give a close approximation to the optimum. This approximated solution is referred to as heuristic optimal rebalancing (HOR).

Simple Anticipatory Rebalancing

Despite the good quality in solutions, the wide use of optimal rebalancing policies is still constrained by the limit of computational capacity. Large-scale applications and simulations often opt for locally executable algorithms which decentralize the decision-making and solve the problem empirically. By bounding the problem to a small area, it reduces the complexity in real-time vehicle operation and also naturally caps the induced rebalancing distance without explicitly describing the cost. One representative example is [19], which gives an intuitive method for local rebalancing called "block balance": if a block's supply exceeds its expected demand or vice versa, system pushes or pulls idle vehicles to or from adjacent blocks. The paper then justifies this empirical method through simulation.

We extend this method and develop here a simple anticipatory rebalancing (SAR) approach for free-floating systems: a vehicle makes decision based on local knowledge within its neighboring area $\bar{\mathcal{S}}$ of $\bar{S}$ zones; the probability that it moves to zone j is proportional to the number of predicted requests in that zone, that is:

$$\mathbb{P}(\text{vehicle moves to } j) = \frac{\lambda_j}{\sum_{j'\in\bar{\mathcal{S}}} \lambda_{j'}} \tag{4.22}$$

Under this policy, a vehicle could rebalance itself with its local knowledge and avoid causing increased workload on the central dispatcher. Decisions can be made in parallel.

Deep Q Network

Solving the rebalancing problem requires modeling the shared AMoD system deliberately. However, delicate models are barely solvable and transferable from system to system, which unfortunately discourages its use in practice.

Reinforcement learning provides a very different approach to tackling the rebalancing problem since it's model-free and requires little adjustment of the generic architecture. Recent advances in deep Q networks [48, 49] have also made it possible to handle the delay between actions and rewards as well as the sequences of highly correlated states. Research works have demonstrated that DQN has the ability to master difficult control policies including traffic controls and taxi dispatching [50, 51].

In this thesis, the neural network is trained with a variant of the Q-learning algorithm [48]. The neighboring area $\bar{\mathcal{S}}$ of a specific vehicle is divided into grids and thus makes the reinforcement learning environment. The distribution of idle vehicles, in-service vehicles together with the predicted demand around it (i.e. r_j, s'_j, λ_j for all $j \in \bar{\mathcal{S}}$) build up the state. Based on an action-value function, the DQN agent takes the current state as input and returns the policy by simply selecting the action with the highest value from the set of {noop, ne, e, se, s, sw, w, nw, n}. "noop" indicates no rebalancing operation; "ne" indicates rebalancing to the northeast adjacent zone and so on and so forth. The vehicle then executes the action and returns the reward.

The rewards is evaluated under the following rules: (1) if the vehicle is assigned to traveler(s) during the rebalancing period, we compare the environment to the one without rebalancing and calculate the save in wait time as reward; the episode terminates and the system moves on until the vehicle becomes idle again; (2) if the vehicle remains idle during the rebalancing period, we use a penalty (a negative constant) as reward to "punish" this rebalancing action; the episode continues. The penalty is designed as such to discourage empty-running rebalancing distance and limit the operational cost. The rewarded episodes are stored into a replay memory and we update action-value function using batched samples drawn from the memory. After the training, this action-value function is used to guide the rebalancing actions of all idle vehicles in the online algorithm.

The framework of DQN is illustrated in Figure 4-1. The architecture and parameterization are described in the next subsection together with the simulation experiments, with

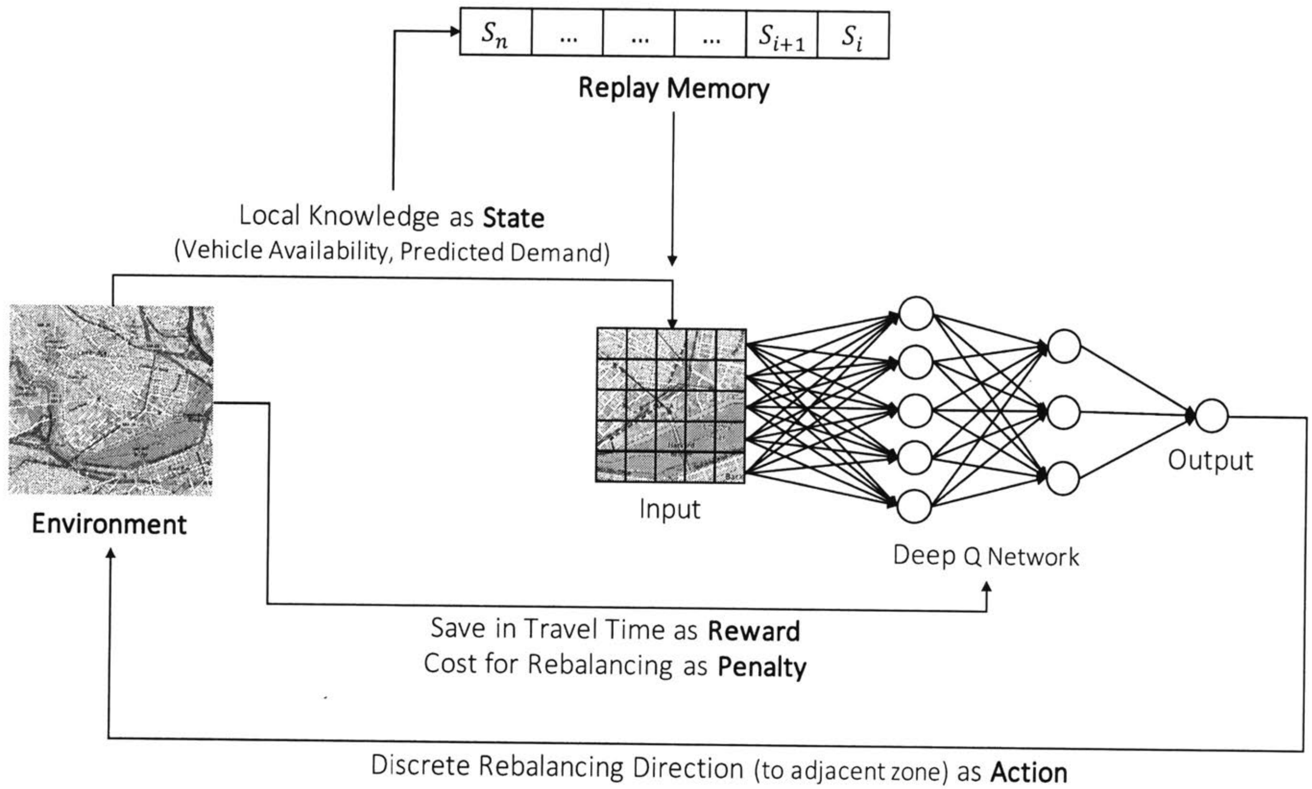


Figure 4-1: The framework of the Deep Q Network.

which we'll compare the effectiveness of DQN with HOR and SAR based on the simulation experiments.

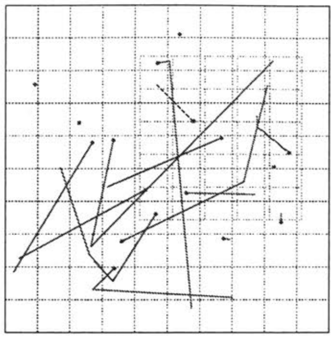
4.2.2 Performance Comparison

Benchmarking

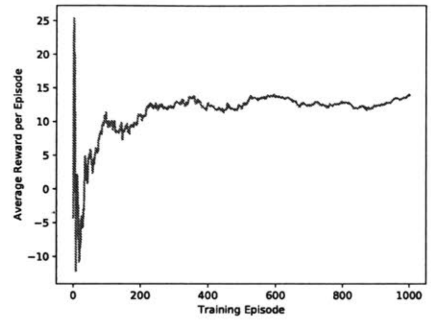
For the sake of training efficiency, we begin the test with an abstract 5km$\times$5km map with no road networks. Requests are drawn from a pool of origin-destination pairs with arrival rate of 100 trips/h, following the exponential distribution. 20 vehicles of capacity 4 forms the fleet, moving straight from point to point based on Euclidean distance with a constant speed of 21.6 km/h. Despite its simplification in describing the road network and the traffic, this presentation is sufficient to evaluate the effectiveness of the algorithms.

As shown in Figure 4-2a, HOR discretizes the entire map into 10$\times$10 fixed cells of 0.5km$\times$0.5km. As for the local algorithms, each vehicle in both SAR and DQN has knowledge of a neighboring area of 2.5km$\times$2.5km, which is centered at the location of the vehicle and discretized into 5$\times$5 moving cells of the same size. Points on the figure represent vehicles; lines (dashed line) originating from points represent planned routes (rebalancing routes); dotted grids represent the discretized cells for both HOR (10$\times$10) and SAR/DQN (5$\times$5 with red vehicle as center). For each of the rebalancing methods, simulation runs 50 times with a simulation time of 3 hours. Rebalancing is performed every 150 seconds. The DQN is trained with a three-layer neural network beforehand for 6000 steps, using ϵ-greedy behavior policy and a replay memory of 2000 most recent steps in batches of size 32. The learning rate is set to be 0.001 with no decay for the Adam optimizer [52] and the penalty for empty rebalancing is -5. Figure 4-2b shows that the average reward increases and reaches its height after around 400 episodes during one typical training (irregular ups and downs are due to randomness in training). The best DQN in terms of the shortest average wait time is used in the following analysis.

We distinguish three demand scenarios: (1) the balanced scenario, in which the trip ODs are uniformly distributed on the map at random; (2) the imbalanced scenario, in which the trip origins are concentrated to two production areas and destinations to two attraction areas; and (3) the first-mile scenario, in which trip origins are uniformly distributed while the destinations are fixed to one point (e.g. an access station). The level of imbalance



(a) Discretization.



(b) Average reward per episode during one typical training.

Figure 4-2: Discretizing and training the Deep Q Network.

increases from scenarios 1 to 3 as the distributions of origins (where requests are sent, i.e., demand) and destinations (where vehicles become idle, i.e., supply) mismatch each other to a greater and greater extent. The necessity of rebalancing is therefore expected to increase.

Performance Comparison

Figure 4-3 compares the performance of the rebalancing methods under the balanced scenario. When HOR is applied, the average traveler wait time according to 50 three-hour runs is 146.2 seconds. This is a great leap from 170.6 seconds (+16.7%) in the case with no rebalancing policy as shown in the rightmost box. With the same system settings, SAR scores 155.6 (+6.4%) and DQN scores 150.1 (+2.7%). It indicates that DQN has superiority over its local counterpart SAR with regard to high service accessibility, yet it falls behind HOR. The suboptimality of DQN could be explained by its design of individual rewarding. Without coordination with other vehicles in the training the agent (vehicle) tends to overreact to the imbalance. It is also noticed that both HOR and DQN produce smaller wait time variance. SAR, in contrast, performs in a rather random manner and the results are more dispersed. When it comes to the vehicle distance traveled, the simulation points out that all rebalancing methods would induce average vehicle distance traveled by around 30% to 35%. The rebalancing distance could be controlled by adjusting $c_{i,j}$ in HOR and the size of the neighboring area in SAR and DQN.

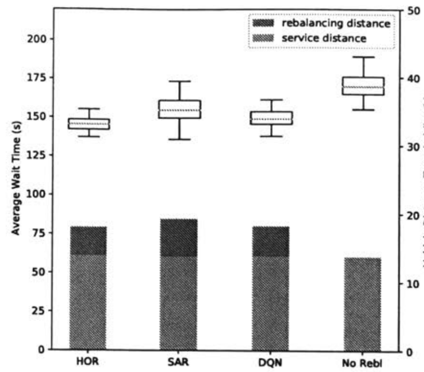


Figure 4-3: Comparison of rebalancing methods: the balanced scenario.

Various Demand Patterns

Table 4.1 shows how rebalancing responds to different demand patterns. As the level of imbalance increases from scenario 1 to scenario 3, the wait time with no rebalancing soars up from 170.6 to 232.8. If HOR is in action, the system performs surprisingly better when productions/attractions are more unevenly distributed, owing to the fact that agglomerated trip distribution reduces the routing difficulties in ride-sharing. The performance of DQN still resides in between HOR and SAR. However, limited to the local knowledge, SAR and DQN gradually loss their competitiveness when the demand-supply imbalance is significant only at the areawide level.

Table 4.1: Comparison of Wait Times under Different Demand Scenarios

	HOR	SAR	**DQN**	No Rebl
Scenario 1	146.2	155.6	**150.1**	170.6
(balanced)		(+6.4%)	(+2.7%)	(+16.7%)
Scenario 2	138.5	172.5	**155.8**	211.4
(imbalanced)		(+24.5%)	(+12.5%)	(+52.6%)
Scenario 3	129.2	161.9	**151.6**	232.8
(first-mile)		(+25.3%)	(+17.3%)	(+80.2%)

* Wait times (in seconds) on top; changes compared to HOR (in percentage) in the middle.

Scalability

We enlarge the map from 5km×5km to 10km×10km and 20km×20km and augment the demand intensity proportionally under the scenario 1. The fleet size also increases from 20 to 125 and 810 to maintain the same level of service under HOR. As shown in Table 4.2, the increasing wait time from 170.6 to 240.7 shows a manifest necessity for vehicle rebalancing when map is large. DQN stays very close to the optimal solution, demonstrating its robust performance over different map sizes.

The computational times are also shown in Table 4.2. Each value represents the average running time for one rebalancing solution using a machine with 2.7 GHz Intel Core i5 and 8 GB memory. HOR is undoubtedly the most computationally demanding one. The computational time increases drastically as the map expands since its complexity is proportional to the product of the fleet size and the number of cells in the area. This might raise a challenge when the scale of application grows. SAR and DQN, in contrast, perform much faster when the area is large and could be further distributed and computed in parallel. This structure evidently facilitates the application to large-scale networks, especially when autonomous vehicles are used.

Table 4.2: Comparison of Wait Times and Computational Times with Different Map Sizes

	HOR	SAR	**DQN**	No Rebl
Map 1	146.2	155.6	**150.1**	170.6
(5km×5km)		(+6.4%)	(+2.7%)	(+16.7%)
	0.033	*0.027*	*0.034*	
Map 2	147.9	164.3	**154.4**	176.3
(10km×10km)		(+11.1%)	(+4.4%)	(+19.2%)
	1.893	*0.682*	*0.822*	
Map 3	147.2	182.4	**158.2**	240.7
(20km×20km)		(+23.9%)	(+7.5%)	(+63.5%)
	259.185	*42.976*	*41.209*	

* Wait times (in seconds) on top; changes compared to HOR (in percentage) in the middle; computational times (in seconds) on bottom in italics.

Wrap-up

Results using an abstract map show that DQN performs effectively by reducing the wait time of travelers and limiting the distance traveled by vehicles. With these simulation experiments, the model-free DQN has revealed its potential in this field originally dominated by operation research models, particularly when scales are large and stochasticity hinders the quality of the optimal formulation.

However, its performance is still second to the network-based optimal problem with heuristic solution (HOR). Several directions might be worthy to follow up to further improve its performance, including:

- moving from discrete action space to continuous one to avoid discretizing rebalancing actions;
- extending the deep Q network to multi-agent models to correct overreacting;
- redefining the reward to represent different performance metrics from the perspective of both operators and travelers.

We choose to use HOR in the following chapters of the thesis.

4.3 Transit-oriented Considerations

The dispatching strategies should also reflect transit-oriented designs, especially when the shared AMoD system is run by a non-profit-driven government-owned operator and AV and PT are integrated as a multimodal public service. In this thesis, we embody the considerations of service availability and equity through the support of various hailing policies. Allowing in-advance requests with prioritized service and ahead-of-time notification will be especially important as it fundamentally changes the degree to which daily commuters and other first-mile trips can rely on the service to access the rail station.

For first-mile travelers, by setting constraints on latest drop-off time at the PT station, seamless connections to public transportation can be ensured. Symmetrically, last-mile travelers can take advantage of the in-advance requests and customize their earliest pick-up time at the station. In addition, we also observe that trips in areas distant from the main service zone are underserved. These trips usually have little attraction to the operator due to low probability of getting shared and costly empty running for picking up. Travelers from there experience long wait as well. In-advance requests can improve the availability in such areas by encouraging operator to plan ahead. The impact of different hailing policies on the level of service will be discussed in Section 5.2.3.

From the operator's perspective, although service priority constraints associated to in-advance requests do bring burden on operation, the knowledge of demand beforehand has more benefits. On one hand, the demand information improves operator's ability to optimize both request-vehicle assignment and rebalancing. On the other hand, the mechanism that communicates travel needs to the operator reduces the efforts required for inferring information and predicting future trips from noisy data. Chapter 6 will discuss the value of demand information for AMoD operators in great details.

Chapter 5

System Design Experiments

5.1 System Settings and Simulation Scenarios

In this chapter, the shared AMoD system's performance is studied through the simulation platform `amod-abm`. The system has a fixed-size fleet of taxi-like vehicles to provide service. Each vehicle may have more than one traveler on board at a time and every trip has a chance to be shared. The focus of this research is to test the viability of a first-mile AMoD service. As a result, for intrazonal trips only those with currently chosen modes as bus and car are selected for the simulation. For trips to downtown, all modes are included.

Table 5.2 is a list of values that has been applied to the assumption variables and demand decision variables in the simulation. These values are determined empirically and remain constant throughout the chapter.

In addition, service rate, wait time and detour factor are initiated to be 100%, 300 seconds and 1.00 for the demand-supply interaction. We also define that the balance is reached when the relative change of $\mathbf{D}$ is less than 0.5% of the total volume.

For simulating one scenario, the necessary system settings for the simulation experiments are the ASC for the service, fleet size and vehicle capacity. Decision variables also include the choice of hailing policies. In fact, the combinations of all variable levels for all possible scenarios are exponential. This prohibits exhaustion of all combinations. Systematic optimization of service variables is not in the scope of this thesis. Therefore, for illustration purposes, we select only a subset of scenarios as in Table 5.1 for our simulation experiments.

Table 5.1: System Settings and Simulation Scenarios (System Design)

Simulation Scenario	Parameters/Variable					Scenarios for Target of Interest
	Fleet Size	Vehicle Capacity	Fare Policy	Hailing Policy	ASC_{av+pt}	
Fleet Sizing (5.2.1)	target	4	as in 5.1	as in 5.2.3	-3.00	200, 220, 240, 260, 280
Vehicle Capacity (5.2.2)	dependent	target	as in 5.1	as in 5.2.3	-3.00	capacity=1: fleet=520, 560, 600, 640, 680 capacity=2: fleet=280, 300, 320, 340, 360 capacity=3: fleet=220, 240, 260, 280, 300 capacity=4: fleet=200, 220, 240, 260, 280
Hailing Policy (5.2.3)	220	4	as in 5.1	target	-3.00	on-demand requests vs. in-advance requests
Preference to AV (5.3)	dependent	4	as in 5.1	as in 5.2.3	target	ASC=-3.58: fleet=100, 120, 140, 160, 180 ASC=-3.00: fleet=200, 220, 240, 260, 280 ASC=-2.35: fleet=350, 400, 450, 500, 550

* A parameter/variable being "target" indicates it's isolated and tested with a range of scenarios.
** A parameter/variable being "dependent" indicates the choice of its value is dependent on that of the target.

Table 5.2: Assumed Values for Assumption Variables and Demand Decision Variables

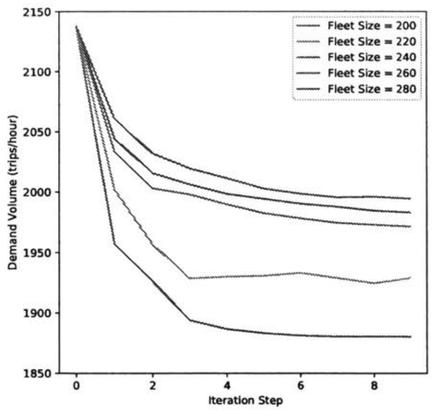
Variable	Value
maximum wait time (MWT)	10 minutes
maximum detour factor (MDF)	1.5
period of simulation (T)	7200 seconds
period of study (T_s)	3600 seconds
period of warm-up (T_w)	1800 seconds
period of cool-down (T_c)	1800 seconds
interval of assignment (T_a)	30 seconds
interval of rebalancing (T_r)	150 seconds
penalty wait time (PWT)	20 minutes
base fare (c_{base})	\$0.83
per-unit-time fare (c_{time})	\$0.11/min
per-unit-distance fare (c_{dist})	\$0.55/km
discount for sharing (DC)	25%

5.2 Impact of Service Design

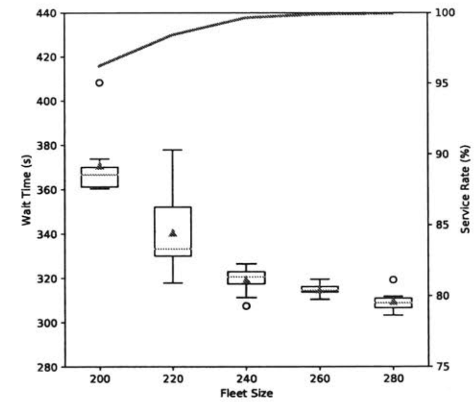
5.2.1 Fleet Sizing

Service rate, wait time and detour factor are unknown parameters in the utility function. Starting with initial values, the demand-supply interaction is able to reach balance in about ten steps. As illustrated in Figure 5-1a, demand volume converges to different levels when fleet size varies. A larger fleet leads to higher service rates and shorter wait times. This is shown in Figure 5-1b. Consequently, travelers are more likely to choose the service and demand grows. In contrast, the fleet size has little impact on detour factor. This stabilizes at a very low level (1.15), shown by the orange curve in Figure 5-1d. Average travel time is also steady (around 520s) regardless of the changing fleet size.

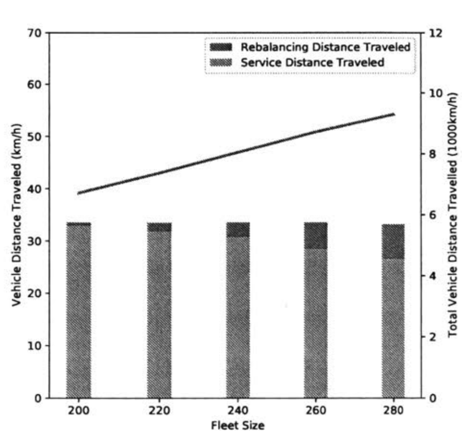
From the operator's point of view, providing more vehicles will result in increased total vehicle service distance traveled, this is shown in Figure 5-1c. The average load also drops from 1.30 to 1.16 as shown in the blue curve in Figure 5-1d. This indicates that, although a larger fleet size does improve the level of service, it also boosts the operational costs and decreases the vehicle occupancy. The nature of the fleet sizing problem is therefore the trade-off between the benefits to travelers and operators. Assuming the operator is contracted to guarantee high service rates (above 99%) and the operator is cost-averse to large capital



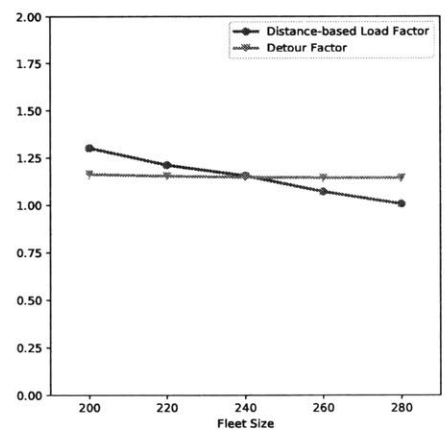
(a) Demand volume converges to different levels when fleet size varies.



(b) A larger fleet leads to higher service rates and shorter wait times.



(c) Providing more vehicles results higher total vehicle service distance traveled.



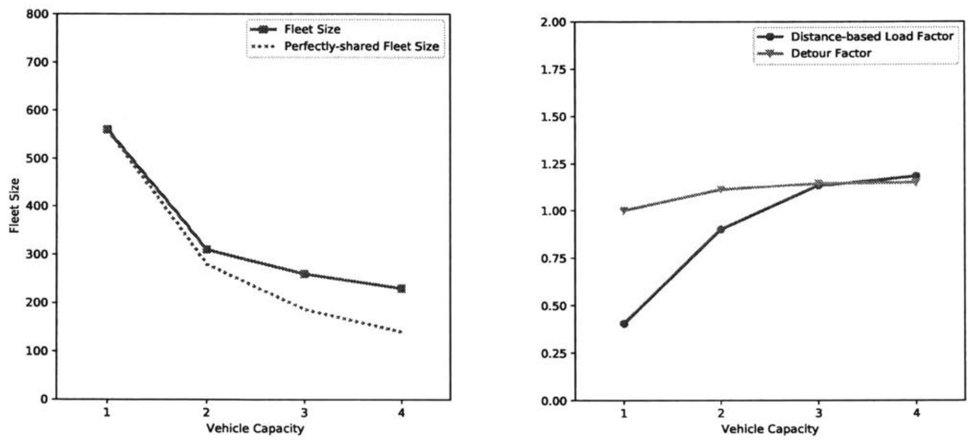
(d) Load decreases when fleet size increases; detour factor remains steady.

Figure 5-1: Impact of fleet sizes on travelers and the operator.

investment, a fleet size of 230 might be a good point of balance. At this fleet size, we will have around 1950 trips per hour. The rebalancing distance accounts for less than 10% of the total distance traveled.

5.2.2 Vehicle Capacity

A taxi-like vehicle usually accommodates 4 travelers. However, allowing trips to be shared between many travelers does have side effects on travel experience: each traveler has less space in the vehicle, has higher uncertainty in estimated arrival time as well as experiences more delay due to pickups and drop-offs of fellow passengers. The operator may limit the capacity of sharing to 2 or 3 in order to improve the travel experience; they can also choose to provide non-shared rides if capacity is set to be 1. We assume that the same discount for sharing still applies and test here the different capacities. Figure 5-2a shows that, the operator need around 560 vehicles to guarantee the 99% service rate if sharing is not available. If vehicles can be shared by at most 2, 3 and 4 travelers, the required fleet size would be 310, 260 and 230 respectively. This demonstrates the power of sharing. When sharing becomes the common practice, the number of vehicles on road can be reduced by more than half.



(a) The vehicle capacity-fleet size curve is decreasing and above the ideal curve.

(b) Both load factor and detour factor increases as vehicle capacity becomes larger.

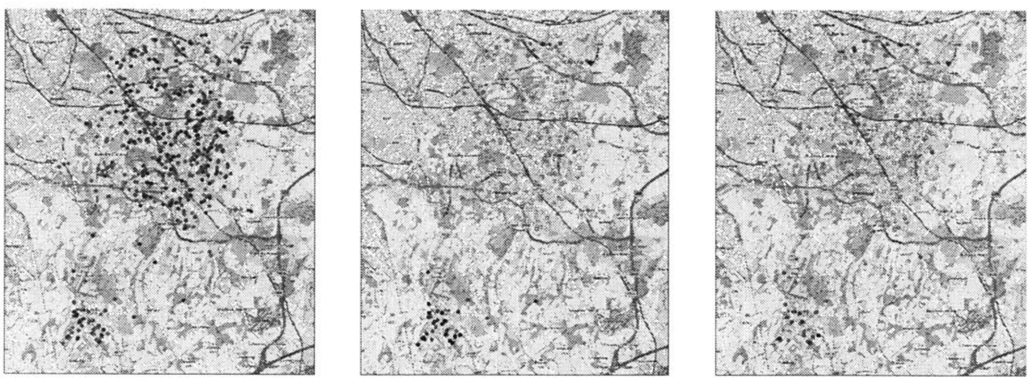
Figure 5-2: Impact of vehicle capacities on travelers and the operator.

We also compare the results with the ideal scenario where all trips can be shared perfectly. In that case, the necessary fleet size should be cut to 1/4 when capacity increases from 1 to 4 as shown by the dotted curve in Figure 5-2a. However, the perfectly-shared trips (with same origin, destination and departure time) can only be found very occasionally in real life, especially in the real-time systems. Therefore, the benefit of moving to larger vehicles becomes less significant when vehicle capacity is already sufficiently large. This conclusion neglects the existence of micro-transit services which do use vans/buses. Additionally, since most mobility-on-demand services use shared rides, the commonality of cars means that they are the predominant choice of shared vehicle. The travel experience, as mentioned in the beginning of this part, is indeed another consideration.

The power of sharing to reduce vehicles on road can also be explained by the increase in vehicle load factor. When sharing is not allowed, each vehicle has an average load of 0.40, which implies that during more than half of the operation time vehicles are traveling empty. The low efficiency of vehicle use is also a result of the first-mile service. Since most of the travelers use AV from home to rail station for downtown in the morning, the pickup trips (from station to another trip origin) are usually empty. If vehicles are shared, the load factor increases from 0.40 to up to 1.18, leading to a much higher efficiency and lower operational cost. We also notice that sharing also causes detour, with the factor ranging between 1.10 to 1.15. This indicates that travelers would experience around a 10% to 15% increase in their in-vehicle travel times if sharing is activated

5.2.3 Hailing Policy

Providing more vehicles and rebalancing idle vehicles can help to improve the availability of the service. However, due to the asymmetrical pattern of first-mile trips and the nature of on-demand dispatching, it's nearly impossible for operator to ensure 100% service rate.



(a) Scatter of the origins of the predicted trips. The satellite town, Biggin Hill, is the agglomeration in the southwest.

(b) Likelihood of walking away assuming on-demand requests only. Dark marker indicates high chance.

(c) Likelihood of walking away if in-advance requests are allowed in satellite town.

Figure 5-3: Impact of hailing policy on service availability.

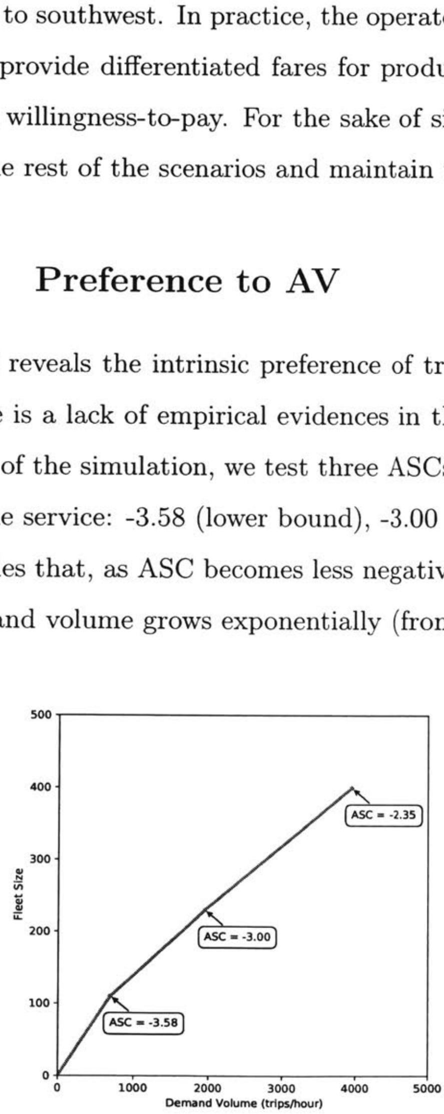
The origins of the predicted trips are geographically scattered in CSA this shown in Figure 5-3a. Given that there is a limited number of vehicles and only on-demand requests are allowed, it's likely that some travelers would walk away due to long wait times. In this case, the origin-specific likelihood of walking away is shown in Figure 5-3b. The darker the marker is, the less likely a traveler from there will get a vehicle. We notice that the satellite town, Biggin Hill, in the southwest of CSA is particularly underserved because it's distant from the main concentration of demand. Only 34.7% of trips from there are served there.

In contrast, the service rate of the rest of CSA is 99.2%.

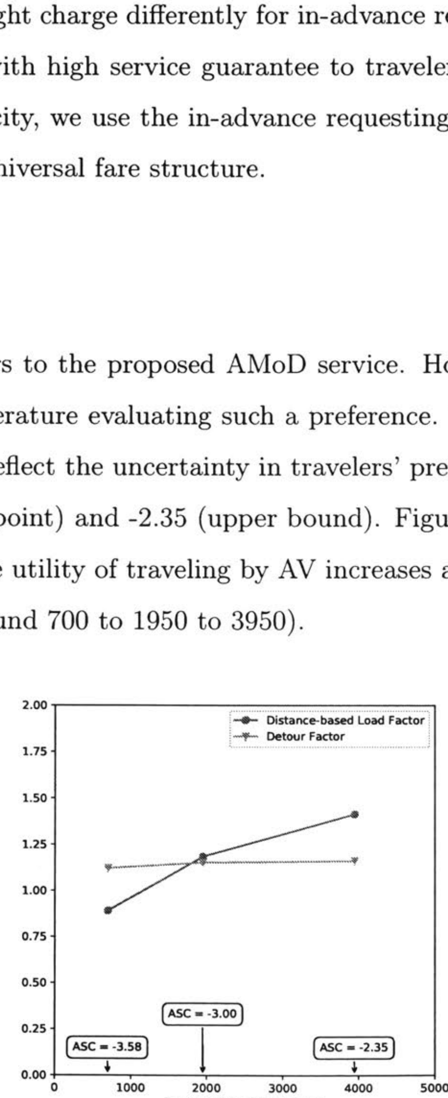
To respond to the low availability of service, we propose that in the rest of the scenarios travelers from Biggin Hill are able to request in advance and assume that half of them choose to do so. In this case, the service rate there is largely improved (85.5%) as shown in Figure 5-3, with the availability in the rest of CSA experiencing a relatively small decrease (98.5%). This could be a result of planning for the in-advance requests and moving vehicles from main area to southwest. In practice, the operator might charge differently for in-advance requests and provide differentiated fares for products with high service guarantee to travelers with high willingness-to-pay. For the sake of simplicity, we use the in-advance requesting policy in the rest of the scenarios and maintain the universal fare structure.

5.3 Preference to AV

ASC reveals the intrinsic preference of travelers to the proposed AMoD service. However, there is a lack of empirical evidences in the literature evaluating such a preference. In this part of the simulation, we test three ASCs to reflect the uncertainty in travelers' preference to the service: -3.58 (lower bound), -3.00 (midpoint) and -2.35 (upper bound). Figure 5-4a implies that, as ASC becomes less negative, the utility of traveling by AV increases and the demand volume grows exponentially (from around 700 to 1950 to 3950).



(a) The relation between demand and fleet size indicates the economies of scale.



(b) A higher volume lead to higher load. Detour factor remains steady.

Figure 5-4: Impact of preference (ASC) on system scale and performance.

In order to maintain the same level of service, the operator should deploy a much larger fleet if the service rate of 99% is still guaranteed. When ASC grows from -3.58 to -3.00 and -2.35, the fleet size increases from 110 to 230 and 400 respectively. Figure 5-4a also depicts the relation between demand volume and fleet size. The concave curve indicates the economies of scale of the system as the increased volume of demand results in decreased number of vehicles per traveler. The economies of scale can be explained by the fact that travelers are more likely to get paired together when more requests emerge at the same time. The increasing load factor in Figure 5-4b also supports this observation. In contrast, the detour factor remains at the level of around 1.15.

5.4 Before-and-After Mode Shares

Table 5.3 shows the total trips made by all modes before and after the launch of the AMoD service. "Before" and "After" are the number of trips and mode shares for each of the modes in the system. "Shift to AMoD" shows the number and percentage of trips shifted from car, bus, rail and P/K+R to the AMoD service. Trips to downtown, despite the mode, have much higher loss rate because AV as the first-mile connection to rail produces more attractive trip characteristics in terms of shorter travel time and higher availability. The total ridership of rail does not decrease in this case, since AV only serves the first-mile connection. The conversion from car trips to AMoD will bring even more travelers to rail.

Table 5.3: AMoD Service Ridership and Mode Share Analysis

Trip Type	Mode	Before		After		Shift to AMoD	
		Trips	Share	Trips	Share	Trips	% Shift
Downtown	Car	775	10%	441	5%	334	43%
	Rail	5968	74%	3661	46%	2307	39%
	P/K+R	1277	16%	723	9%	554	43%
	AMoD	0	0%	3195	40%	N/A	N/A
	Total	8020	100%	8020	100%	3195	40%
Intrazonal	Car	22715	86%	20373	77%	2342	10%
	Bus	3629	14%	3109	12%	520	14%
	AMoD	0	0%	2862	11%	N/A	N/A
	Total	26344	100%	26344	100%	2862	11%

* No observations for walk, bike and taxi.

** All of the downtown trips use AV for rail connection, while only 10% of the intrazonal trips involves bus or rail. The rest of them are AV-only.

As for intrazonal trips, conversion from car is the smallest. This is likely the result of the high car ownership in CSA. In the long term, the new service may change the attitude of the residents towards owning private vehicles. However, bus operator does see a decrease of 14% in ridership. This confirms that AV service would have negative impact on the ridership and revenue of PT operators.

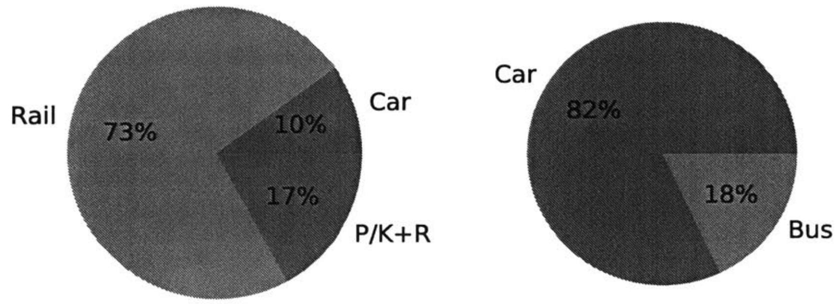


Figure 5-5: Percentage of AMoD trips from each existing mode (left: downtown trips; right: intrazonal trips).

Figure 5-5 shows the percentage of AMoD trips from each mode. Car trips were dominating the CSA, and if AMoD service is introduced, they contribute to around 45% of the AMoD market (10% for downtown trips, and 82% for intrazonal trips). Also, we notice that for downtown trips, more than 73% of the AMoD travelers were using rail for downtown. They will continue using rail but will take advantage of AV as their access mode. Bus and P/K+R travelers contribute equally to the rest of the AMoD trips.

THIS PAGE INTENTIONALLY LEFT BLANK

Chapter 6

Value of Information

6.1 AMoD Systems with Information

Information and communication technologies are critical factors in successful AMoD applications. A good information system allows smooth exchange of information between travelers and the operator, enabling both parties to be better informed and make more coordinated use of the resources. Current MoD applications often rely on online-enabled platforms for communication, and the conceptual autonomous systems are highly likely to follow the same technology choice. However, as the amount of data is multiplying and the desire for real-time solutions continues to grow, the effective management of information in such AMoD systems becomes critical and should require more investigation.

In this chapter, the discussion will be focused on the demand information in AMoD systems as well as its value for the operation. The demand information is defined to include all details about the requests at both aggregated level and individual level. Minimum information such as the origins, destinations and time constraints of the on-demand requests is the baseline of providing the AMoD service. By gaining more details of the demand information and knowing them ahead of time, the operator also has a chance to improve the dispatching decisions relating to both assignment and rebalancing. Potentially, it will help to achieve higher level of service and better financial profitability.

Travel information prepared for the travelers is also important for advising on their travel choices and guiding their behaviors. However, this topic is beyond the scope of the discussion in this chapter.

6.1.1 Individual Demand Information

The individual demand information includes the origins, destinations, and time constraints (typically, earliest pick-up time, latest pick-up time and latest drop-off time) of the upcoming requests. The dataflow of the individual information often involves the mutual communication between the travelers and the operator: travelers send on-demand or in-advance requests to the operator; based on the constraints of a request as well as the availability at the supply side, the operator responds by either assigning a vehicle for its service or rejecting the request.

In most of the cases, the MoD system only serves on-demand requests. As a result, the individual request information becomes known to the system in a fully dynamic way, and the operator responds to the requests in real time. For travelers, on-demand services seem to be particularly desirable because it provides absolute flexibility both in time and space. However, from the operator's perspective, having individual information a priori is beneficial since it enlarges the search space and makes better dispatching solutions possible. For this reason, the operator may want to nudge for in-advance requests from travelers.

One way of doing so is by incentivizing travelers through either monetary leverage or product differentiation. The latter approach consists of prioritizing in-advance requests in the dispatching algorithms. In fact, the in-advance requests are associated naturally with the service priority: by sending a request in advance, the traveler avoids competing with other on-demand requests traveling at the same time and has a higher probability of getting assigned a vehicle. Once assigned, the service is secured as we assume there is no turning down after the assignment has been made.

Under this assumption, in-advance requests reduce the uncertainty in the making of the trip, including the potential long wait, late arrival and even rejection of service. Therefore, it can be attractive to some travelers, despite the loss of flexibility in travel time. For the operator, the trade-off of in-advance requests is also evident: the operator benefits from the information a priori to improve the dispatching efficiency; however, it has to pay for it by promising service priority to these trips, although some trips may turn out to be not profitable.

The operator may also infer future trips based on the historical data at its disposition. Technically, this involves predicting the individual spatio-temporal behavior of frequent travelers based on Bayesian models or Markov chain-based models. A very relevant research work can be found in Zhao et al. [53]. However, predicting the exact starting time and location of a trip is still extremely challenging, and the state-of-the-art algorithms often give probability distributions for the values instead of point estimates. To incorporate the distributions into dispatching as input, stochastic assignment algorithms are preferred to their deterministic counterparts. The performance of such algorithms varies, depending on not only the form and quality of the prediction, but also the computational capacities, as stochastic methods often have high complexity. Therefore, the algorithms should be selected on a case-by-case basis.

6.1.2 Aggregated Demand Information

The aggregated demand information usually takes the form of OD-specific demand volumes or zonal demand volumes. Although aggregated information has little impact on the performance of deterministic assignment and routing algorithms, it does contribute to the system-level decisions such as determining the best fleet size and rebalancing the idles vehicles dynamically. Specifically, the optimal rebalancing algorithm discussed in Section 4.2 and used in the following simulations relies on the predicted demand volumes to estimate the potential benefit of rebalancing vehicles to each zone.

In contrast to the individual case, the models for predicting the aggregated demand information have been well established in the literature. To estimate the potential demand of a new service, mode choice models are powerful tools, which permit comparison between the importance of attributes of the new mode and that of the existing alternatives. If the service is currently running, the operator could take advantage of the existing data to predict the aggregated demand in short and long terms.

6.2 Value of Demand Information

6.2.1 Level of Information

Depend on the different system settings, operating modes, data collection mechanisms and prediction/inference models, an AMoD operator may have different levels of knowledge of demand information. When evaluating the level of information in dynamic systems, dynamism, accuracy and granularity are three important attributes.

- Dynamism measures the reaction time between the time a piece of information is available, and the time it expires. When it comes to individual demand information, the reaction time is defined as the period between the time a request is known to the dispatcher, and the time it is supposed be picked up [42]. A system which allows only on-demand requests is said to be fully dynamic; a system that has all information a priori is fully static.

- Accuracy represents the closeness of a piece of information to the true value using statistical and probabilistic tools. Requests directly communicated with the operator often accurately describe the actual trips the travelers will make. For this reason, when making dispatching decisions, large-scale MoD applications often ignore the uncertainty, such as potential delayed pick-up, changed destination and cancellation due to travelers. In contrast, demand information inferred from the data is much less accurate. Point estimate that gives a single value of estimate is unlikely to happen. Instead, we may use confidence interval to specify a range within which the value is estimated to lie as well as its probability to lie; some other forms of probability distribution including normal distribution are also widely used.

- Granularity in both spatial and temporal dimensions indicates the level of detail presented in a set of data. When conducting demand analysis, the operator often divides the service area into transport analysis zones and choose appropriate size of zones for desired level of detail. The period of study may also be separated into multiple time segments to deal with fluctuating demand. The confidence interval is another example of the granularity, as the interval itself could be treated as a level of fineness. Highly granular (finer) information is often desirable, yet it increases the scale of the problem in almost every aspect.

Given the different levels of information, the operator should plan the dispatching strategies accordingly to take the best advantage of the available information and fit the special operational needs. In general, the state-of-the-art algorithms have no special difficulty when moving from static systems to dynamic ones, and from low granularity to high granularity. The only challenge arises from the probability distributions, for which complex stochastic algorithms are required.

A compromised balance that stems from the information inference is the trade-off between accuracy and granularity. As in interval estimates, the confidence interval is tightly coupled with the confidence level: the narrower the range is (higher granularity), the less probable that the true value falls into the estimated range (lower accuracy). A larger sample size will normally lead to better estimate. However, in practice, the cost of collecting and processing large amount of data should be taken into account.

Another concern tied to the accuracy-granularity trade-off is the privacy. The travel information often contains sensitive personal information that may be used to locate a single person's home and workplace, to follow its activities, even to distinguish the individual identity. In this sense, travelers may find it unwanted to share highly granular data. In response to this concern, some MoD operators propose selected spots for picking up and dropping off as an alternative of the door-to-door service. This option also facilitates the operation, as multiple riders are more likely to be grouped together. By deliberately avoiding locations that require long detour and recommending selected spots instead, the routing part of the algorithms can also be improved.

Spatial Granularity for Aggregated Information

In this chapter, we will compare scenarios with different levels of spatial granularity with regard to the aggregated demand information. The base scenario treats the entire service area as a whole and the operator is only informed of the estimated total demand. This scenario has the coarsest granularity. Each of the remaining scenarios divide the area into a selected number of gridded zones. For each specific zone, the predicted demand volume is made available to the operator. Finer griding indicates higher spatial granularity of the aggregated demand information.

We also assume that the predicted volumes are known a priori (no dynamism), exact (prefect accuracy) and constant throughout the simulation period (zero temporal granularity). As long as the aggregated demand information is available at the zonal level, the operator rebalances the idle vehicles dynamically following the optimal rebalancing strategy presented in Section 4.2.

Degree of Dynamism for Individual Information

In the following simulation experiments, we assume that the individual demand information comes directly from the requests of travelers. We also suppose that a portion of the travelers are allowed to send requests in advance. The partially dynamic system then serves a mix of in-advance and on-demand requests in real time.

The information of a request r is described as a combination of origin o_r, destination d_r, earliest pick-up time $t_{e,r}$ and latest pick-up time $t_{l,r}$. We further assume that these values have prefect accuracy and granularity. Consequently, dynamism becomes the only determining factor of the level of information. Following Larsen et al. [42], the degree of dynamism is defined as the average reaction time of all requests, divided by the total time of study T_s.

$$dod = \frac{1}{R}\sum_{r\in\mathcal{R}}\left(1 - \frac{t_{e,r} - t_{s,r}}{T_s}\right) \tag{6.1}$$

$\mathcal{R}$ is the set of all requests including on-demand requests $\mathcal{R}_{ond}$ and in-advance requests $\mathcal{R}_{adv}$. The sizes of the sets are R, R_{ond} and R_{adv} respectively. $\mathcal{R} = \mathcal{R}_{ond} \cup \mathcal{R}_{adv}$. $R = R_{ond} + R_{adv}$. $t_{s,r}$ is the sending time of request r; $t_{e,r}$ is the earliest pick-up time. The difference between $t_{e,r}$ and $t_{s,r}$ is the reaction time for system to dispatch. For any on-demand request $r \in \mathcal{R}_{ond}$, the system should respond in real time and we have thus $t_{e,r} = t_{s,r}$. The reaction time for $r \in \mathcal{R}_{ond}$ is therefore zero.

It is easy to see that $dod = 100\%$ when system is totally dynamic and all requests are sent on demand. The degree of dynamism decreases as the proportion of in-advance requests increases. Longer reaction time also leads to less dynamic system.

Multiple scenarios have also been created to compare different levels of individual demand information. The base scenario assumes a fully dynamic system with on-demand requests only. This is compared with systems under two different hailing policies: (a) a portion of travelers from the underserved Biggin Hill send requests in advance; (b) a portion of travelers randomly selected from the entire area are in-advance users. The degree of dynamism reflects the percentage of in-advance requests in the system; however, the spatial distribution of the in-advance requests is omitted in this metric and requires further discussion together with the simulation results.

We also assume that the reaction time for all in-advance requests is 30 minutes (1800s), same as the value used in Chapter 5. The period of study T_s remains one hour (3600s), with both warm-up and cool-down periods before and after. Under these assumptions, the degree of dynamism of a system with perfect in-advance information is 50%.

6.2.2 Value of Information

Inspired by Mitrović-Minić et al. [39], the value of information in this chapter is defined as:

$$voi = \frac{F - F'}{F'} \tag{6.2}$$

The metric measures the objective gain in percentage. F is the value of the objective function when additional information is available; F' is the objective of the base scenario in which information is limited to the necessary minimum. Generally speaking, the value of information is positive since information enlarges the search space and has the potential of improving the performance of the system. However, it is still probable that in some scenarios information may lead to negative value of information.

The operation of the AMoD system may have different objectives. Profit-driven operators seek for maximum profit; travelers care about level of service and travel cost; and the government have more considerations in mind regarding system-level performance. The interests of multiple stakeholders may either correlate or contradict with each other. In this thesis, the shared AMoD system is run by a non-profit-driven government-owned operator so as to reflect the consensus among stakeholders. The ultimate objective of the service is therefore the common good at the system level. To this end, the dispatching algorithms have also embodied the considerations of service availability and equity and have the objective of minimizing the system-wide travel time, as illustrated in Chapter 4.

However, the system-wide travel time as an objective is not appropriate for the evaluation of the performance when system is running over capacity and all requests can not be served. In this chapter, we propose to use the average number of served requests instead. The average number of served requests, simple in its form, is representative of the interests of all stakeholders. From the system-level point of view, it reflects the size of the population that can benefit from the service. Travelers also prefer higher value since it indicates higher service availability. As for the operator, the average number of served requests implies revenue. As long as the AMoD service is in good profitability, a higher value is always preferred as it leads to higher profit.

Two more indicators are also used in the analysis. For travelers, the adjust wait time is a critical indicator for the attractiveness of the service. For the operator, it can not always afford running a service with very short wait time, since it requires very large fleet and reduces the vehicle occupancy. This to some extent contradicts with the ultimate interests of profitability, even if it's assumed to be not profit-driven. In this chapter, we use profit made during the period of study as an indicator for the AMoD operator. The profit can be derived from its fare revenue and operational cost.

The revenue is generated from the farebox. In Section 3.3.2, the price of a single trip is defined based on the fare structure in Equation 3.4. The total revenue of the operator will be the sum of fares collected from all trips served during T_s. In the following analysis, we keep using the assumed values in Section 3.3.2: the base fare c_{base} = \$0.83, per-unit-time fare c_{time} = \$0.11/min, per-unit-distance fare c_{dist} = \$0.55/km and the discount for sharing DC remains 25%.

The cost of operating a fleet can break down into fixed cost and variable cost. Wen et al. [15] propose that the current fare Zipcar adopts for renting cars to Uber drivers can be used. This makes a fixed cost of \$0.06/min and a variable cost of \$0.29/km. Based on the vehicle distance traveled and the operating hours of each vehicle, we can then get the total operating cost.

6.3 Value of Information Experiments

6.3.1 System Settings and Simulation Scenarios

In this section, the value of demand information is studied through agent-based simulation experiments. The values determined in Table 5.2 remain unchanged throughout the simulation. As for the system settings, we further assume that ASC is -3.58 for all scenarios; fleet size is 110; and vehicle capacity is 4.

For illustration purposes, only a few scenarios are selected and simulated, as shown in Table 6.1. This table is designed to demonstrate the impact of the level of information on the value of information using metrics such as spatial granularity for aggregated information and degree of dynamism for individual information. Due to the limitation of deterministic and heuristic dispatching algorithms, we do not seek to exhaust all possible scenarios, as many of them require major modification to the current algorithms.

Scenario 1 represents the base scenario of a mobility-on-demand system with minimum information, in which all requests are sent on demand and the operator has no knowledge of the demand beforehand. Scenarios 2a and 2b assume that demand volumes are known at the

Table 6.1: System Settings and Simulation Scenarios (Value of Information)

Simulation Scenario	Level of Information			
	Aggregated		Individual	
	Spatial Granularity	Other Attributes	Dynamism	Other Attributes
Scenario 1	total demand (coarsest)	as in 6.2.1	on-demand requests only (fully dynamic)	as in 6.2.1
Scenario 2a	zonal demand, 5×5 griding	as in 6.2.1	on-demand requests only (fully dynamic)	as in 6.2.1
Scenario 2b	zonal demand, 10×10 griding	as in 6.2.1	on-demand requests only (fully dynamic)	as in 6.2.1
Scenario 3a	zonal demand, 10×10 griding	as in 6.2.1	50% in-advance requests from Biggin Hill	as in 6.2.1
Scenario 3b	zonal demand, 10×10 griding	as in 6.2.1	5% in-advance requests from entire CSA	as in 6.2.1
Scenario 3b	zonal demand, 10×10 griding	as in 6.2.1	10% in-advance requests from entire CSA	as in 6.2.1

* The number of trips from Biggin Hill accounts for 10% of the total number of trips. In this case, 50% of the Biggin Hill trips being in-advance is equivalent to 5% of the total trips in terms of the degree of dynamism.

** In simulation, a request is in-advance request with the aforementioned probability. The choice of a request is independent to others.

zonal level, which is true for most of the current MoD operators. However, current systems barely rebalance idle vehicles due to the difficulty in negotiating with human drivers. The conceptual autonomous systems, in contrast, comply perfectly with the dispatching decisions and make the execution of optimal rebalancing algorithms almost effortless.

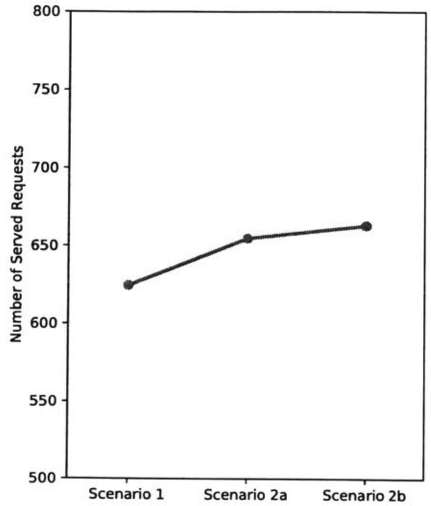
Scenarios 3a, 3b and 3c are not widely seen in real-world applications. However, as the share of the service continues to grow, we expect that differentiated products such as in-advance requests and prioritized services will emerge in the market. Huge amount of data collected during the service will also contribute to individual trip prediction. This in return pushes forward the intelligent operation and product differentiation. Travelers will also have the freedom to choose from a variety of products. In the sense of free choice, Scenario 3a appears to be more realistic, since travelers from Biggin Hill are more likely to take advantage of the in-advance requests in order to secure service. The service rate in the rest of the area is above 99% in general (see Section 5.2.3). With almost perfect service availability, travelers would prefer to travel as they like. In this case, on-demand request outperforms its in-advance counterpart for its flexibility in time and space.

6.3.2 Results and Analysis

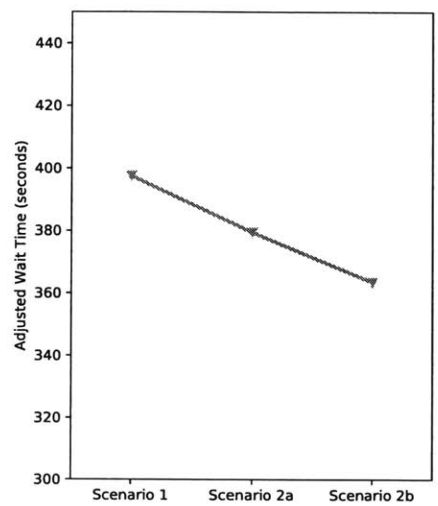
Aggregated Information

As illustrated in Figure 6-1a, rebalancing is effective for increasing the service availability. The more granular the aggregated demand information is, the better the performance of rebalancing will be. Higher granularity also leads to shorter adjusted wait times as shown in Figure 6-1b. This could be explained by two reasons: (a) rebalancing moves idle vehicles to be closer to the potential demand; (b) higher service availability imposes less penalty on the adjusted value.

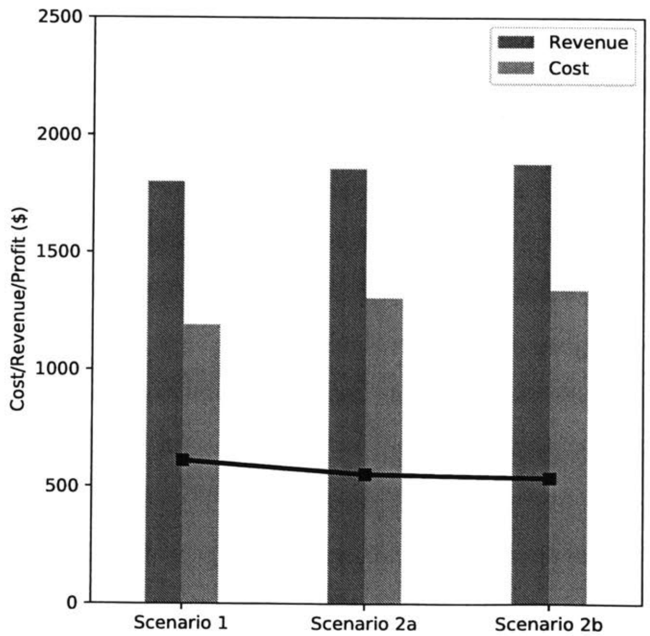
From the operator's perspective, applying rebalancing strategies has slightly negative impact on profitability. On one hand, when rebalancing is enabled, the revenue increases as more travelers are served during the period of study. On the other hand, rebalancing will inevitably increase the operational cost since it induces empty rebalancing distance. Unfortunately, the revenue could not compensate for all the induced cost. As shown in Figure 6-1c, the profit the operator makes each hour drops from \$610 to less than \$550.



(a) The number of served requests increases as granularity of aggregated information becomes higher.



(b) The adjusted wait time decreases as granularity of aggregated information becomes higher.



(c) Rebalancing increases both revenue and cost. Collectively it has slightly negative impact on profit.

Figure 6-1: Impact of aggregated demand information on system performance.

Individual Information

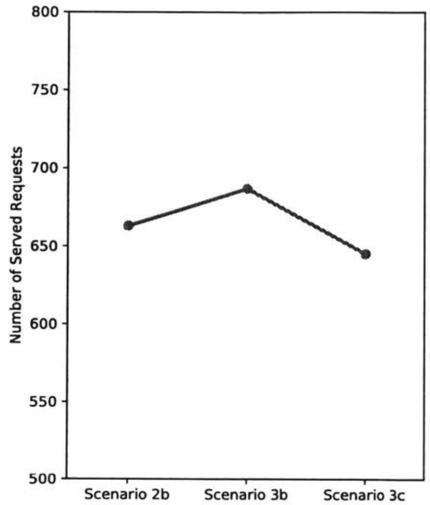
We also compare the different degrees of dynamism: Scenario 2b allows only on-demand requests and has 100% degree of dynamism; in Scenario 3b, 5% of the travelers are in-advance users and the degree of dynamism is 97.5%. Scenario 3c has 10% in-advance requests and therefore a degree of dynamism of 95.0%

Figure 6-2a and Figure 6-2b show the performance at the system level and from traveler's point of view. The number of served requests increases as the percentage of in-advance requests grows from 0% to 5%, which corresponds with our expectation that information brings value. However, as the percentage of in-advance requests continues to grow from 5% to 10%, its impact on the system becomes negative. The negative value of information can be explained by the associated service priority. In Section 6.1.1, the assumption has been made that a traveler can not be turned down (i.e. rejected service) after it has been assigned a vehicle. In this case, the in-advance requests become naturally prioritized and this imposes constraints on the dispatching. Heuristic assignment methods that reoptimize the pairing may alleviate the problem.

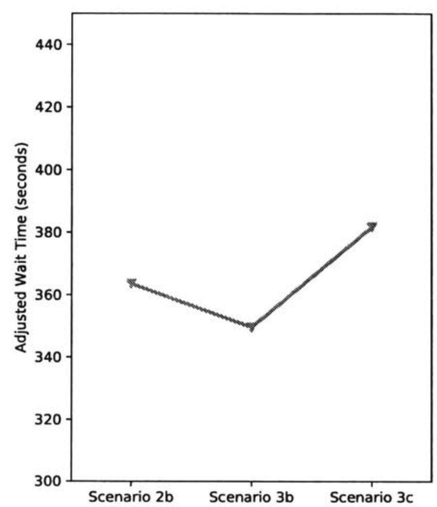
Surprisingly, the decrease in level of service does not lead to the reduction in profit, as shown in Figure 6-2c. In contrast, the profit keeps growing when more in-advance requests are coming in. This might be due to the fact that having demand information a priori makes ride-sharing more possible and enables better assignment solutions (for travelers, more sharing increases the detour factor from 1.11 to 1.13). This reduces the service distance traveled by the vehicles from 19.0km to 17.0km per vehicle per hour, thus decreases the operational cost. If the operator is profit-driven, the in-advance requests could be even more profitable. It could charge differently for products with prioritized service, since some of the travelers that are more sensitive to travel time have higher willingness-to-pay.

Lastly, we compare Scenario 3b with Scenario 3a. According to the demand prediction, the number of trips from Biggin Hill accounts for 10% of the total number of trips. Scenario 3a assumes that 50% of the travelers from Biggin Hill send requests in-advance, equivalent to 5% of the total trips. The degree of dynamism is therefore 97.5%, same as Scenario 3b. The only factor that makes difference is the spatial distribution of the in-advance requests.

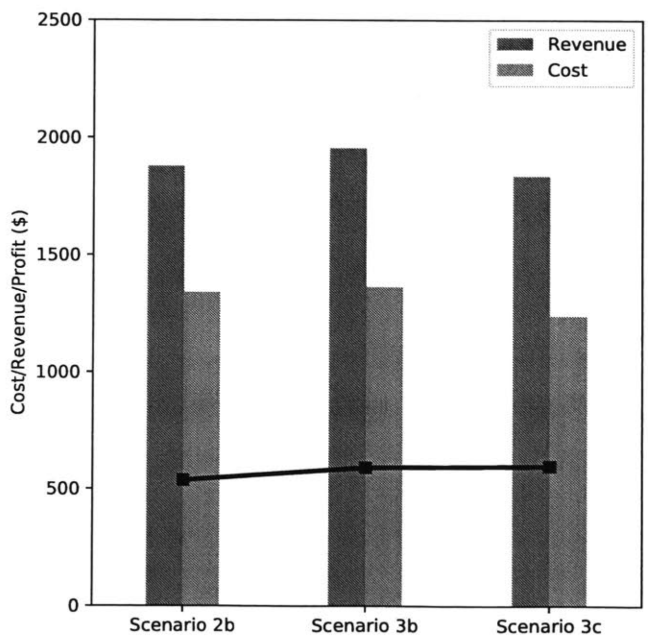
Simulation results show that, by having travelers from Biggin Hill send requests in advance, the number of served requests per hour increases significantly from 663 to 723. Ad-



(a) The number of served requests first increases then decreases as degree of dynamism of individual information decreases.

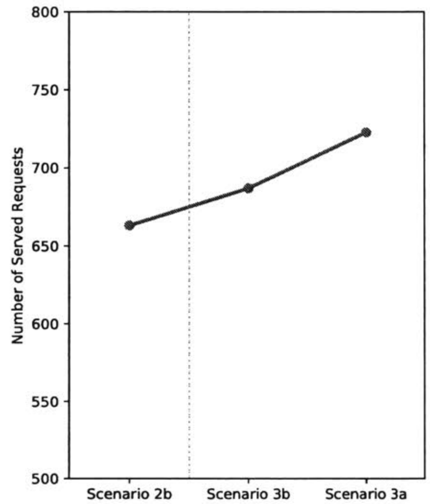


(b) The adjusted wait time first decreases then increases as degree of dynamism of individual information decreases.

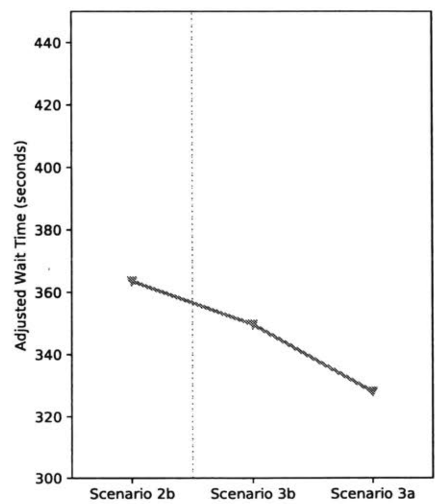


(c) The profit increases as degree of dynamism of individual information decreases.

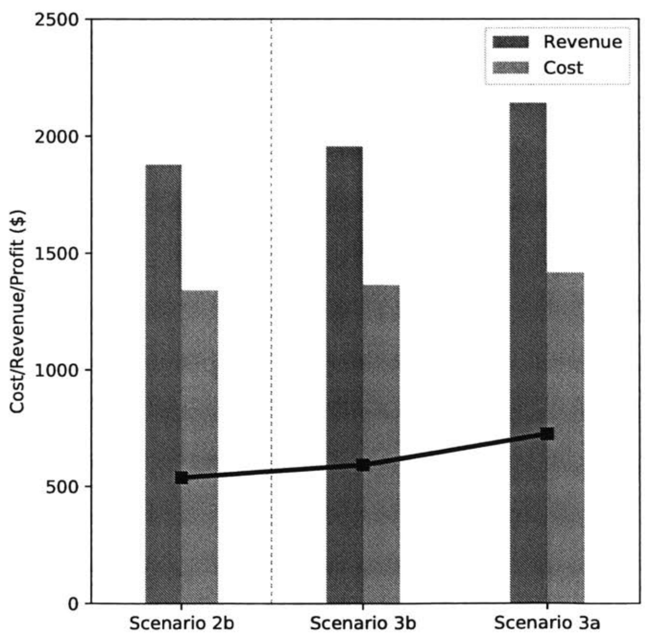
Figure 6-2: Impact of the degree of dynamism of individual demand information on system performance.



(a) The Biggin Hill scenario has much higher number of served requests.



(b) The Biggin Hill scenario has much shorter adjusted wait time.



(c) The Biggin Hill scenario makes much more profit.

Figure 6-3: Impact of the spatial distribution of individual demand information on system performance.

justed wait time also decreases from 364 seconds to 328 seconds. Using the number of served requests as the objective in Equation 6.2 and Scenario 2b as the base, the value that the in-advance requests in Biggin Hill brings to the system is more than 9%. As a comparison, the value of information in Scenario 3b is only 4%. As for the operator, in Scenario 3a, the profit it makes increases from $538 to $726 (+35%), while in Scenario 3b the profit is only $592 (+10%). We notice that, the travelers that the base scenario rejects are generally from Biggin Hill and other remote areas and have relatively longer travel distance. Once in-advance requests are enabled and they can served, these travelers contribute a much higher portion to the farebox. The growth of profit is therefore much more significant that the growth of number of served requests.

THIS PAGE INTENTIONALLY LEFT BLANK

Chapter 7

Conclusion

7.1 Future Research Needs

This thesis offers a systematic approach to the design, simulation and evaluation of AMoD systems and demonstrates specific AMoD service designs. The system design and modeling framework reflect the transit-oriented considerations that are important for high service availability, seamless connections, and equity. Using demand-supply interaction, the simulation platform represents the choices of both travelers and operators. System design experiments show the trade-off between improving level of service and traveler experience, and the cost of larger fleet size and low occupancy, providing the starting point of identifying the optimal balance. We also observe that encouraging ride-sharing and allowing in-advance requests are tools to enable efficient service and sustainable travel.

The thesis also identifies the critical role of demand information in successful AMoD operation. Multiple attributes, including dynamism, accuracy and granularity of information, collectively shape the performance of a system. The numerical results help rationalize the efforts operators should spend on data collection, information inference and advanced dispatching algorithms. If the demand information is managed effectively, it has significant impact on the value the system can bring to either traveler, operator, or society as a whole.

We point the future research needs in three broad areas:

The first is to explore a range of service design scenarios and value of information scenarios. Due to the time limit as well as the technological constraints, this thesis only examines part of the scenarios through simulation in an empirical manner. Introducing differentiated products, various fare structures, and different hailing policies will change both the travel behavior and the system design, and all these scenarios could be evaluated through simulation. Based on the interests of different stakeholders, the objective function of dispatching should also be carefully designed to reflect this need.

The second is to develop advanced dispatching strategies so as to accommodate probabilistic information. The probability distribution blurs the boundary of assigning vehicles to the known requests and rebalancing idle vehicles for the unknown. In this case, stochastic algorithms should come into play to replace the deterministic methods. Moreover, high percentage of in-advance requests enlarges the search space. The gap between the accuracy of current algorithms and that of optimum is therefore widen. This also requires the implementation of metaheuristic algorithms with higher accuracy; although the accuracy could be at the sacrifice of the computational speed.

The third is to examine how AMoD system will impact the transportation system, especially active modes and public transit. The convenience of the door-to-door service may deprive people of the opportunity to walk and cycle. Simulations, state preference surveys, and field studies could help empirically assess the impact of AVs on these modes. Similarly, it is also important to examine how bus service needs to be reoptimized. More broadly the simulation model should to be expanded to include bus service so that we can consider the AV fleet and bus fleet management jointly.

Other potential directions include:

- to understand the cost of operating and maintaining the AV fleet in AMoD systems;
- to evaluate performance of different system designs under uncertainty, as many factors, parameters and assumptions made in the thesis are uncertain and different sources of uncertainty have different impacts on system performance;
- to close the system design loop in Figure 3-1 by incorporating design variables into the interaction mechanism.

7.2 Rethinking `amod-abm`

System design is the process of conceptually defining infrastructure and elements of a system to satisfy the specified requirements and constraints. The design of the agent-based simulation platform, `amod-abm`, which has been intensively used in this research, also followed the system design guideline. However, during the past one year and half, the scale and scope of the project have grown continuously, much faster than expected. The research objective and requirements have also been redefined throughout the period to answer new questions. In this case, many of the original elements in `amod-abm` have become incompatible with present needs. Although the author has been adding new modules to replace old ones and create new features at a good pace, the development of `amod-abm` is still far from complete.

The following features and modifications have been planned and the development of them should be prioritized in the near future:

- to move from constant demand to time-variant demand so as to simulate the AMoD operation of an entire day;
- to shift from static travel time to stochastic time and upgrade routing modules.
- to implement stochastic algorithms that are compatible with the probabilistic demand information;
- to develop partially dynamic assignment algorithms to accommodate high percentage of in-advance requests and decouple with service priority.
- to extend the deep Q network to multi-agent models to correct overreacting and improve training performance;

It would also be very helpful to rethink the original system design. Impossible though, if the following points could have been thought of in the very beginning, the development would follow a clearer path and be more efficient.

- The first is the framework of the interaction mechanism. In fact, in this research, the concept of interaction was developed a posteriori, and the original simulation infrastructure did not explicitly include the mode choice model. The model was not incorporated into the system until we realized that the demand and supply in the system were interdependent. Even today, part of the mode choice model still relies on Excel files: ugly, inconsistent and slow.
- The second is the modularization and interfacing. Initially, the author alone was responsible for the development of the platform. However, with the expanding scope of research, the team has also grown rapidly and more researchers have been involved.

Consequently, the collaboration on coding becomes necessary and multiple new modules have been planned. However, the original system design underestimated the importance of extensibility.

- The third is the forever trade-off between the computational speed and the solution accuracy. Specifically, the design of the offline routing engine and static map database all targeted at fast computation for scalability. However, the current computational performance is still limited due to the capacity of single machine. As the number of decision variables increases and more sophisticated algorithms come in use, the problem of computational speed will become even more severe. Applying heuristics can alleviate this, rewriting Python code in C++ and redesigning the data structure also help, but the ultimate solution should be undoubtedly parallelization.

Bibliography

[1] Yu Shen, Hongmou Zhang, and Jinhua Zhao. Embedding autonomous vehicle sharing in public transit system: An example of last-mile problem. Technical report, 2017.

[2] Daniel J Fagnant and Kara M Kockelman. Dynamic ride-sharing and fleet sizing for a system of shared autonomous vehicles in austin, texas. *Transportation*, pages 1–16, 2016.

[3] Katarzyna Anna Marczuk, Harold Soh Soon Hong, Carlos Miguel Lima Azevedo, Muhammad Adnan, Scott Drew Pendleton, Emilio Frazzoli, et al. Autonomous mobility on demand in simmobility: Case study of the central business district in singapore. In *Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM), 2015 IEEE 7th International Conference on*, pages 167–172. IEEE, 2015.

[4] Luis M Martinez, Gonçalo HA Correia, and José M Viegas. An agent-based simulation model to assess the impacts of introducing a shared-taxi system: an application to lisbon (portugal). *Journal of Advanced Transportation*, 49(3):475–495, 2015.

[5] Luis M Martinez and José Manuel Viegas. 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, 2017.

[6] Jie Yang, Jing Dong, and Wei Wang. A simulation model for performance analysis of electric autonomous taxi systems. Technical report, 2017.

[7] Katarzyna A Marczuk, Harold SH Soh, Carlos ML Azevedo, Der-Horng Lee, and Emilio Frazzoli. Simulation framework for rebalancing of autonomous mobility on demand systems. In *MATEC Web of Conferences*, volume 81, page 01005. EDP Sciences, 2016.

[8] Jaeyoung Jung, R Jayakrishnan, and Ji Young Park. Dynamic shared-taxi dispatch algorithm with hybrid-simulated annealing. *Computer-Aided Civil and Infrastructure Engineering*, 31(4):275–291, 2016.

[9] Javier Alonso-Mora, Samitha Samaranayake, Alex Wallar, Emilio Frazzoli, and Daniela Rus. On-demand high-capacity ride-sharing via dynamic trip-vehicle assignment. *Proceedings of the National Academy of Sciences*, page 201611675, 2017.

[10] Jie Chen, Kian Hsiang Low, Yujian Yao, and Patrick Jaillet. Gaussian process decentralized data fusion and active sensing for spatiotemporal traffic modeling and prediction in mobility-on-demand systems. *IEEE Transactions on Automation Science and Engineering*, 12(3):901–921, 2015.

[11] Justin Miller, Andres Hasfura, Shih-Yuan Liu, and Jonathan P How. Dynamic arrival rate estimation for campus mobility on demand network graphs. In *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*, pages 2285–2292. IEEE, 2016.

[12] M Keith Chen and Michael Sheldon. Dynamic pricing in a labor market: Surge pricing and flexible work on the uber platform. In *EC*, page 455, 2016.

[13] Han Qiu, Ruimin Li, and Jinhua Zhao. Dynamic pricing in shared mobility on demand service and itssocial impacts. Elsevier, 2017.

[14] Jian Wen. amod-abm. `https://github.com/wenjian0202/amod-abm`, 2017.

[15] Jian Wen, Yu Xin Chen, Neema Nassir, and Jinhua Zhao. Transit-oriented autonomous vehicle operation with integrated demand-supply interaction framework. under review at Transportation Research Part C, 2017.

[16] Jian Wen, Jinhua Zhao, and Patrick Jaillet. Rebalancing shared mobility-on-demand systems: a reinforcement learning approach. IEEE, 2017.

[17] Kevin Spieser, Kyle Treleaven, Rick Zhang, Emilio Frazzoli, Daniel Morton, and Marco Pavone. Toward a systematic approach to the design and evaluation of automated mobility-on-demand systems: A case study in singapore. In *Road Vehicle Automation*, pages 229–245. Springer, 2014.

[18] Daniel J Fagnant and Kara M Kockelman. The travel and environmental implications of shared autonomous vehicles, using agent-based model scenarios. *Transportation Research Part C: Emerging Technologies*, 40:1–13, 2014.

[19] Daniel J Fagnant and Kara M Kockelman. Dynamic ride-sharing and optimal fleet sizing for a system of shared autonomous vehicles. In *Transportation Research Board 94th Annual Meeting*, number 15-1962, 2015.

[20] Michael W Levin and Stephen D Boyles. Effects of autonomous vehicle ownership on trip, mode, and route choice. *Transportation Research Record: Journal of the Transportation Research Board*, (2493):29–38, 2015.

[21] Suzanne Childress, Brice Nichols, Billy Charlton, and Stefan Coe. Using an activity-based model to explore the potential impacts of automated vehicles. *Transportation Research Record: Journal of the Transportation Research Board*, (2493):99–106, 2015.

[22] T Donna Chen and Kara M Kockelman. Management of a shared autonomous electric vehicle fleet: Implications of pricing schemes. *Transportation Research Record: Journal of the Transportation Research Board*, (2572):37–46, 2016.

[23] Han Qiu, Ruimin Li, and Jinhua Zhao. Dynamic pricing in shared mobility on demand service and its social impacts. Manuscript submitted for publication, 2017.

[24] Menno D Yap, Gonçalo Correia, and Bart Van Arem. Preferences of travellers for using automated vehicles as last mile public transport of multimodal train trips. *Transportation Research Part A: Policy and Practice*, 94:1–16, 2016.

[25] Burak Boyacı, Konstantinos G Zografos, and Nikolas Geroliminis. An optimization framework for the development of efficient one-way car-sharing systems. *European Journal of Operational Research*, 240(3):718–733, 2015.

[26] Ganjar Alfian, Jongtae Rhee, Muhammad Fazal Ijaz, Muhammad Syafrudin, and Norma Latif Fitriyani. Performance analysis of a forecasting relocation model for one-way carsharing. *Applied Sciences*, 7(6):598, 2017.

[27] Mauro Dell'Amico, Eleni Hadjicostantinou, Manuel Iori, and Stefano Novellani. The bike sharing rebalancing problem: Mathematical formulations and benchmark instances. *Omega*, 45:7–19, 2014.

[28] Supriyo Ghosh, Pradeep Varakantham, Yossiri Adulyasak, and Patrick Jaillet. Dynamic repositioning to reduce lost demand in bike sharing systems. *Journal of Artificial Intelligence Research*, 58:387–430, 2017.

[29] Marco Pavone, Stephen L Smith, Emilio Frazzoli, and Daniela Rus. Robotic load balancing for mobility-on-demand systems. *The International Journal of Robotics Research*, 31(7):839–854, 2012.

[30] Rick Zhang and Marco Pavone. Control of robotic mobility-on-demand systems: a queueing-theoretical perspective. *The International Journal of Robotics Research*, 35 (1-3):186–203, 2016.

[31] Kevin Spieser, Samitha Samaranayake, Wolfgang Gruel, and E Frazolli. Shared-vehicle mobility-on-demand systems: a fleet operatorâĂŹs guide to rebalancing empty vehicles. In *Transportation Research Board 96th Annual Meeting*, 2016.

[32] Barbara Lenz and Eva Fraedrich. New mobility concepts and autonomous driving: the potential for change. In *Autonomous Driving*, pages 173–191. Springer, 2016.

[33] Xiao Liang, Gonçalo Homem de Almeida Correia, and Bart Van Arem. Optimizing the service area and trip selection of an electric automated taxi system used for the last mile of train trips. *Transportation Research Part E: Logistics and Transportation Review*, 93:115–129, 2016.

[34] Akhil Vakayil, Wolfgang Gruel, and Samitha Samaranayake. Integrating shared-vehicle mobility-on-demand systems with public transit. Technical report, 2017.

[35] Yu Shen, Hongmou Zhang, and Jinhua Zhao. Integrating shared autonomous vehicle in public transit system: An example of the first-mile problem. Manuscript submitted for publication, 2017.

[36] Allan Borodin and Ran El-Yaniv. *Online computation and competitive analysis*. cambridge university press, 2005.

[37] Patrick Jaillet and Michael R Wagner. Generalized online routing: New competitive ratios, resource augmentation, and asymptotic analyses. *Operations research*, 56(3): 745–757, 2008.

[38] Victor Pillac, Michel Gendreau, Christelle Guéret, and Andrés L Medaglia. A review of dynamic vehicle routing problems. *European Journal of Operational Research*, 225 (1):1–11, 2013.

[39] Snežana Mitrović-Minić, Ramesh Krishnamurti, and Gilbert Laporte. Double-horizon based heuristics for the dynamic pickup and delivery problem with time windows. *Transportation Research Part B: Methodological*, 38(8):669–685, 2004.

[40] Michel Gendreau, Francois Guertin, Jean-Yves Potvin, and Eric Taillard. Parallel tabu search for real-time vehicle routing and dispatching. *Transportation science*, 33(4): 381–390, 1999.

[41] Mariam Tagmouti, Michel Gendreau, and Jean-Yves Potvin. A dynamic capacitated arc routing problem with time-dependent service costs. *Transportation Research Part C: Emerging Technologies*, 19(1):20–28, 2011.

[42] Allan Larsen, OBGD Madsen, and Marius Solomon. Partially dynamic vehicle routing-models and algorithms. *Journal of the operational research society*, pages 637–646, 2002.

[43] Marco Diana. The importance of information flows temporal attributes for the efficient scheduling of dynamic demand responsive transport services. *Journal of advanced Transportation*, 40(1):23–46, 2006.

[44] Yosef Sheffi and Warren B Powell. An algorithm for the equilibrium assignment problem with random link times. *Networks*, 12(2):191–207, 1982.

[45] Dennis Luxen and Christian Vetter. Real-time routing with openstreetmap data. In *Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, GIS '11, pages 513–516, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-1031-4. doi: 10.1145/2093973.2094062. URL `http://doi.acm.org/10.1145/2093973.2094062`.

[46] Gilbert Laporte. The vehicle routing problem: An overview of exact and approximate algorithms. *European journal of operational research*, 59(3):345–358, 1992.

[47] Niels Agatz, Alan Erera, Martin Savelsbergh, and Xing Wang. Optimization for dynamic ride-sharing: A review. *European Journal of Operational Research*, 223(2):295–303, 2012.

[48] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013.

[49] Hado Van Hasselt, Arthur Guez, and David Silver. Deep reinforcement learning with double q-learning. In *AAAI*, pages 2094–2100, 2016.

[50] Erwin Walraven, Matthijs TJ Spaan, and Bram Bakker. Traffic flow optimization: A reinforcement learning approach. *Engineering Applications of Artificial Intelligence*, 52: 203–212, 2016.

[51] Miyoung Han, Pierre Senellart, Stéphane Bressan, and Huayu Wu. Routing an autonomous taxi with reinforcement learning. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, pages 2421–2424. ACM, 2016.

[52] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[53] Zhan Zhao, Haris N Koutsopoulos, and Jinhua Zhao. Mobility as a language: Predicting individual mobility in public transportation using n-gram models. Technical report, 2017.