Approximating a ride-sourcing system with block matching

Siyuan Feng[1], Jintao Ke[*2], Feng Xiao[3], and Hai Yang[4, 5]

[1]*Department of Civil and Environmental Engineering, The Hong Kong University of Science and Technology, Hong Kong, China*
[2]*Department of Civil Engineering, The University of Hong Kong, Hong Kong, China*
[3]*Faculty of Business Administration, Southwestern University of Finance and Economics, Chengdu, China*
[4]*Department of Civil and Environmental Engineering, The Hong Kong University of Science and Technology, Hong Kong, China*
[5]*Intelligent Transportation Thrust, The Hong Kong University of Science and Technology (Guangzhou), Guangzhou, China*

August 16, 2022

Abstract

On-demand matching between waiting passengers and idle drivers is one of the most important components in a ride-sourcing system. A variety of matching mechanisms have been developed to meet different needs of ride-sourcing platforms, e.g. mitigating supply-demand imbalance, maximizing platform revenue. In this paper, we focus on a block matching system, a special type of matching mechanism, where the region of interest is partitioned into blocks, and on-demand matching is separately and simultaneously conducted in each block. Block matching can bring many benefits, such as limiting order assignment with long pick-up distance, simplifying the process of deployment, etc. However, it still remains a challenging yet interesting issue to determine the block size for the matching system, which is a key decision variable governing passengers' waiting time. To solve the problem, we model the ride-sourcing system with block matching via a M/M/c queue, in which the service rate is endogenous and partially determined by passengers' average pick-up time. Based on the model, we find that the average queueing time of passengers decreases with block size increasing, while the average pick-up time may increase instead. In addition, the average total waiting time (sum of average queueing and pick-up time) become nearly invariant to the change of block size when the block size is large, which we call plateau phenomenon. In the plateau, ride-sourcing platforms can choose the block size based on other standards while the average total waiting time is always maintained at the nearly lowest value. The findings are verified via an agent-based simulation study, demonstrating that the proposed model can be an effective tool to approximate block matching system.

Keywords: Ride-sourcing service, matching mechanism, queueing theory

*Corresponding author. E-mail address: kejintao@hku.hk (J. Ke).

1 Introduction

Recent years have witnessed a fast popularization of ride-sourcing service. Transportation network companies (TNCs), such as Uber, Lyft and Didi, are using smart-phone APPs to offer on-demand mobility services to passengers around the world, via the broad application of modern mobile communication and Global Position System (GPS). Uber, for instance, is now offering a variety of services in more than 700 metropolitan areas in 65 countries (Wang and Yang, 2019). Didi, the largest ride-sharing company in China, is generating millions of daily ride-hailing demand in a single city, Beijing (Tong et al., 2017). In New York City, Lyft and Uber cars are even estimated to outnumber conventional taxis 4 to 1 (Jiang et al., 2018).

The rapid development of ride-sourcing services has raised many operational issues, such as estimated time of arrival (ETA), on-demand matching, ride-pooling operations, empty vehicle re-positioning, information sharing and disclosure, and rating mechanism (Wang and Yang, 2019). Among these issues, on-demand matching is the main footstone for real-time operations, and thus has intrigued much attention from researchers. In general, a matching algorithm is implemented by ride-sourcing platforms to assign waiting passengers to idle vehicles. There are multiple objectives for a matching operation, such as maximization of platform revenue, maximization of the number of matched orders, or minimization of passengers' average waiting time.

The on-demand matching approaches developed in the literature can be majorly grouped into two streams: bipartite matching (Xu et al., 2018; Chen et al., 2019; Shah et al., 2020), and queue-based matching (Liao, 2003; Lee et al., 2004; Zhang and Pavone, 2016; Xu et al., 2020b; Feng et al., 2020; Besbes et al., 2021). For bipartite matching, waiting passengers and idle drivers are respectively collected and grouped in a batch way within a certain time window, and pairs are formed between each pair of waiting passenger and idle driver. The matching problem is then transformed to finding the best matching on the bipartite graph structure, which can be solved by combinatorial optimization algorithms. Bipartite matching may generate some extra waiting time for passengers since the platform does not assign vehicles to passengers during the time window for order and vehicle accumulation. In comparison, queue-based matching may mitigate this problem, where the arriving vehicles are always assigned to waiting passengers instantly. When there are multiple waiting passengers and no arriving vehicles, the passengers will be formed into a queue, and wait to be served in some pre-defined order, which is often described via queueing models. Specially, when passenger queue is served via First-Come-First-Serve (FCFS) rule, the fairness is protected for longer-waiting passengers, since they possess higher priority for the next matching. In comparison, the bipartite matching usually does not differentiate the waiting time of passengers when making matching decision, and thus some passengers may keep unmatched for a long period.

In this paper, we focus on block matching, a special type of queue-based matching that has been utilized by some ride-sourcing platforms (Xu et al., 2020b) or bike-sharing platforms (He et al., 2021). The core idea of block matching is to partition the whole region (e.g. one city) into various small blocks, and the on-demand queue-based matching is separately and simultaneously implemented in each block. This special matching mechanism has several advantages over regular queue-based matching without matching blocks: 1) The blocks can help to avoid distant matching, since the matching processes are limited within each block. Therefore, the time and operational cost led by distant pick-up can be reduced. 2) The setting of blocks potentially makes it more flexible and convenient for other operations, such as pricing and idle vehicle repositioning. For example, He et al. (2021) study a bike-sharing problem where pricing and queue-based

matching are implemented within blocks with the same partition. Under some scenarios, the price of service within a block is mainly based on the matching process of that individual block, and the pricing problem for the whole region can thus be divided into sub-problem for each single block, which simplifies the optimal solution seeking process. 3) The length of queue in each block is reduced, and thus the computation speed to obtain matching result is improved. Take matching under First-Dispatch rule as an example, with First-Dispatch rule, when there are fewer passengers than vehicles, the arriving passengers is always dispatched to the nearest idle vehicle. When there are more passengers than vehicles, the next idle vehicle is dispatched to the longest waiting passenger, which actually follows FCFS rule. The interest of longer waiting passengers will be considered, as mentioned before. Under First-Dispatch rule, the time complexity for the block matching is found to be negatively related to the number of blocks, as shown in Appendix A. This means the increase of the number of blocks can help to reduce computational complexity given the same size of inputs in some cases. This is validated from the comparison of running time for simulation under different block sizes in Fig. 16.

To properly design a block matching system, a key problem is to determine the area of each block (block size). On one hand, a larger block size may lead to a larger pick-up distance because passengers may be matched to some faraway drivers in the block. The system service rate is thus decreased and the waiting time for passengers may increase. In addition, a larger block size can increase the number of passengers in each block, which will further increase passenger's waiting time. On the other hand, the increasing number of vehicles within a larger block may help to reduce waiting time of passengers. These mixed effects make it challenging to decide the proper block size. In addition, the proper block size also depends on the specific market-related parameters (number of vehicles, average trip time, unit arrival rate). To solve this challenging issue, we model the ride-sourcing system with block matching with a M/M/c queue for each block, in which the service rate is endogenously interacted with the average pick-up time of the system. We also develop an algorithm to find the steady state solution for the system. For system performance metrics, we focus on average queueing time, average pickup time, and average total waiting time for passengers, which are important measurements of system efficiency and passengers' satisfaction. Based on the developed model, patterns of the metrics in terms of block size are depicted and analyzed under different number of vehicles, average trip time, passengers' arrival rate. Significant insights on block size determination are summarized for platform managers. Specially, we find a plateau phenomenon of total waiting time, which means this metrics almost keeps the same low value within an interval of large block size. The platform manager can thus select block size within this interval without worrying the influence on average total waiting time of passengers. Via extensive simulation studies, the phenomenon and other insights of the modelling analysis are validated under different supply-demand scenarios. In summary, this paper makes the following contributions:

- We develop a theoretical model to delineate a ride-sourcing market under a block matching mechanism, which is a practical mechanism for ride-sourcing companies but was rarely investigated in the literature. Most importantly, we spell out the endogenous relationship between the system's service rate and average pick-up time, by leveraging a M/M/c queuing model with endogenous pick-up time. Namely, the average pick-up time accounts for vehicles' service time and thus affects the service rate, while the service rate governs idle vehicles' density and in turn influences the average pick-up time.
- Based on the proposed model, we explore and study the trends of average queuing time, average pickup time, and average total waiting time for passengers with respect to block

size under different supply-demand scenarios. An interesting phenomenon is that the metrics becomes invariant to the block size when it already becomes large. The useful insights can be utilized by platform managers for the determination of proper block size when adopting block matching mechanism.

- Simulation studies are conducted to demonstrate that our proposed model can well approximate the simulated outcome, which is regarded as a proxy for the reality. In the future, the proposed model can be also explored in the analysis for other scenarios that potentially use block matching, such as food order delivery and freight order matching.

The remainder of this paper is organized as follows. Section 2 provides a literature review on past studies for matching operation and application of queueing theory in ride-sourcing service. Section 3 details the model framework for block matching. Section 4 provides the method to find the steady-state solution of the model, conduct a graphical analysis and explain the observed special phenomenon for block matching. Numerical experiments and discussions are provided in Section 5, followed by conclusions in Section 6.

2 Literature Review

2.1 Matching operation for ride-sourcing service

One important task for ride-sourcing service is the matching between waiting passengers and idle vehicles. As mentioned above, there are a variety of objectives for the matching operation, including minimization of passengers' average waiting time and other kinds of delay (Wong and Bell, 2006; Seow et al., 2009; Alonso-Mora et al., 2017), minimization of the required number of vehicles (Vazifeh et al., 2018), maximization of matching quantity (Özkan and Ward, 2020), and maximization of drivers' revenue over a time period (Xu et al., 2018; Tang et al., 2019; Yu et al., 2019).

As mentioned above, there are two types of matching approaches, including bipartite matching and queue-based matching. For bipartite matching, a current trend of researches is to consider the effects of current decision on the future state of the system, and integrate reinforcement learning technology with bipartite matching to achieve long-term objectives. For example, Xu et al. (2018) propose a reinforcement learning method to obtain long-term rewards, which are added with the immediate reward in the online bipartite matching model. The long-term reward represents the expected value of the next order after he/she completes the current one, while the immediate reward reflects a driver's expected revenue from serving the current order. Chen et al. (2019) first shows the intrinsic relationship between matching and pricing, and then optimizes the two operations simultaneously. The pricing strategies are learned via a contextual bandit algorithm and the matching strategies are optimized with the help of temporal difference. Inspired by the fact that extending the matching time interval may significantly reduce the average pickup time (Yang et al., 2020), Ke et al. (2020) adopts deep reinforcement learning methods to delay the bipartite matching of some orders for a potential better matching outcome (with a short pick-up time) in the incoming time intervals. Shi et al. (2019) develop a reinforcement learning based algorithm to operate a community owned electric vehicle fleet, which provides ride-hailing services to local residents. The goals are to minimize passengers' waiting time, electricity cost, and operational costs of the vehicle, and multiple operations are implemented together via bipartite matching.

Queue-based matching also attracts a variety of interests from researchers and ride-sourcing companies. Wang et al. (2019a) analyze the dynamics of passengers and drivers in a queueing model where the platform can control the matching process by setting a threshold on the expected pick-up time. Applying fluid approximations, they explore the impacts of the threshold on the number of vehicles with different states (idle/pick-up/occupied), based on which a policy to adjust the threshold is designed for time-varying demand. Feng et al. (2020) conduct extensive numerical experiments in two cases with circular road and grid network under queue-based matching rules, in order to explore the relationship between system performance metrics and the utilization level, which represents the traffic density of the system. The relationship is found not monotone, and the phenomenon is further analyzed via a theoretic queuing model for the system.

Still, most previous researches focus more on queue-based matching without region partition, while there are only two papers examining queuing systems with matching blocks, i.e. Xu et al. (2020b) for ride-hailing systems and He et al. (2021) for bike-sharing systems. Xu et al. (2020b) study the supply curve of ride-hailing systems under different market conditions based on a double-ended queueing model. The supply curve with finite matching radius is found always backward bending, but weaker bend can be gained via adjustment of the radius. In comparison, we focus more on the impact of block size on the system performance metrics, such as passengers' queuing time and pick-up time. We also examine the impacts of a few important parameters, such as the number of vehicles and the length of average trip time, on the selection of matching block size. In addition, the endogenous relationship between the average pick-up time and the service rate is well characterized in our model, and the solution finding procedure for the steady state of the system is developed. Moreover, we implement an extensive simulation study on a realistic simulator to validate the model and analytical results. Meanwhile, while He et al. (2021) try to address the joint design of incentives (via "crowdsourcing") and spatial capacity allocations (enabled by "geo-fencing") based on strategic queues for bike-sharing platforms, the attention in this paper is paid to the determination of block size under the block matching mechanism.

2.2 Application of queueing theory for ride-sourcing service

In addition to matching operation, queueing theoretic models have been adopted for other operation issues in ride-souring systems. For idle vehicle repositioning (rebalancing), the vehicles is guided by the designed algorithm to cruise to some area, where they can get matched under a certain queue-based matching rule, in order to balance the supply and demand. There has been a rich stream of research on this important issue (Zhang et al., 2018; Yahia et al., 2021; Ma et al., 2019; Calafiore et al., 2017; Braverman et al., 2019; Wollenstein-Betech et al., 2020; Zhang et al., 2016; Sayarshad and Chow, 2017; Spieser et al., 2016a; Li et al., 2021; Bazan et al., 2018; Spieser et al., 2016b). For example, Zhang et al. (2018) model the mobility-on-demand (MoD) systems as two coupled closed Jackson networks with passenger loss. They show that the system can be approximately balanced by solving two decoupled linear programs and exactly balanced through nonlinear optimization, based on which a real-time closed-loop rebalancing policy is designed and tested. Ma et al. (2019) focus on the combination of ride-sourcing system and existing transit system. Queueing-theoretic algorithms are developed to make joint decision of idle vehicle relocation and ride sharing. Braverman et al. (2019) focus on empty-car routing based on a closed queueing network model of ridesharing systems. They establish both process-level and steady-state convergence of the queueing network to a fluid limit in a large market regime where

demand for rides and supply of cars tend to infinity, and use this limit to study a fluid-based optimization problem.

In addition to idle vehicle repositioning, another important operation for the application of queueing theory is pricing (Bai et al., 2019; Castillo et al., 2017; Yan et al., 2020; Courcoubetis and Dimakis, 2018; Taylor, 2018; Ruch et al., 2019; Waserhole and Jost, 2016; Li et al., 2019; Banerjee et al., 2015; Xu et al., 2020a). Among the researches, Bai et al. (2019) consider an on-demand service platform using earning-sensitive independent providers with heterogeneous reservation price (for work participation) to serve its time and price-sensitive passengers with heterogeneous valuation of the service. They include the steady-state waiting time performance based on a queueing model in the passenger utility function to characterize the optimal price and wage rates that maximize the profit of the platform, and discuss the determination of price and payout ratio under different market situation. Castillo et al. (2017) discuss the wild goose chase (WGC) phenomenon in ride-sourcing market, where vehicles are dispatched to pick up distant passengers, wasting drivers' time and reducing earnings. Based on queueing models for the matching process, they suggest to utilize dynamic surge pricing to control the WGC under changing market conditions.

Moreover, queueing models are also frequently utilized in the pooling/sharing operations for ride-sourcing platforms (Yan et al., 2020; Zhang et al., 2018; Ma et al., 2019; Özkan and Ward, 2020; Braverman et al., 2019; Wang and Honnappa, 2017; Waserhole and Jost, 2016; Jacob and Roet-Green, 2021; Banerjee et al., 2015). For instance, Jacob and Roet-Green (2021) develop a queueing model to find the ride-sharing platform's optimal revenue in equilibrium when passengers are strategic and drivers are independent agents, with both solo and pooling service available. They find that offering both solo and pooled rides is optimal when the distribution of passenger-type is not skewed and congestion is not high. Counter intuitively, when congestion is high, the platform benefits from offering only one ride choice. Other interesting topics raised by ride-sourcing operations with queueing model applied include fleet sizing and capacity planning (Besbes et al., 2021; Bazan et al., 2018; Li et al., 2019), service reservation (Yahia et al., 2021), curbside stopping (Qiu et al., 2020), system coordination (Ruch et al., 2019). Still, less attention is paid to block matching system and the resulting problem of block size determination, which we focus on in this study.

3 Model

In this section, we first make several simple assumptions about the studied market, and provide the nomenclature table as preliminary. The matching process in one block of the studied region is then modelled via a M/M/c queue, based on which the steady-state probability of the queue length and the corresponding metrics are obtained. Moreover, we also construct a formula to consider the impact of average pick-up time on the service rate, which completes the mathematical description of the system.

3.1 Preliminary

To simplify the process of model construction and analysis, we make several simple and common assumptions of the ride-sourcing market for the studied region. The drivers' and passengers' spatial distribution are assumed homogeneous, and matching blocks are of the equal size. The inter-arrival time for passengers and drivers is assumed to obey exponential distribution, which

Symbol	Description
A_{total}	Area for the studied region.
M	The number of matching blocks.
A	Area for one block. $A = A_{total}/M$.
K	Vehicle fleet size for the studied region.
λ_{unit}	Arrival rate of passengers for unit area.
λ	Arrival rate of passengers for area of one block. $\lambda = \lambda_{unit}A$.
μ	Service rate of vehicles.
c	Average number of vehicles in one block. $c = K/M$.
n	The number of passengers in the system for one block.
t	Average trip time for passengers in the studied region.
v	Average vehicle speed.
w_0	Maximal tolerable expected waiting time for passengers when joining the queue.
$d(i)$	Function of a passenger's average distance to the closest idle vehicle in a unit-size block with i idle vehicle available for dispatching.
L_q	Average queue length of passengers in the steady state of the system.
W_q	Average queueing time for passengers in the steady state of the system.
W_p	Average pick-up time in the steady state of the system.
W_{tw}	Average total waiting time (including queueing and pick-up time) for passengers in the steady state of the system.

Table 1: List of main symbols

is a regular assumption for queueing theoretic studies (Feng et al., 2020; Xu et al., 2020b; Besbes et al., 2021). In addition, the balking behavior can also be considered in the model, where passengers may reject to join the waiting queue of a block if the expected waiting time is longer than a threshold. This is practical in reality, since the platforms like Didi show the current queue length and expected waiting time before passenger choose to join. For the rule of assignment between passenger queues and arriving vehicles, we focus on the First-Dispatch (FD) rule as mentioned before, which is extensively utilized and studied in previous researches (Xu et al., 2020b; Besbes et al., 2021). Under FD rule, when there are fewer waiting passengers than idle vehicles, the arriving passenger is always dispatched to the nearest idle vehicle. When there are more passengers than idle vehicles, the next idle vehicle is dispatched to the longest waiting passenger.

Under the assumptions and rules made above, we can efficiently model the matching process in an individual block of the region via a M/M/c queueing model specified in the next section. The major symbols for the model construction are listed in Table 1.

3.2 M/M/c model

M/M/C model is a classic modelling method in queueing theory. The first and second *M* represent that the interarrival time of customers and service time by the system are assumed to be exponentially distributed, while *C* means that the number of servers (e.g. vehicles in our study) is larger than one. In a M/M/C model, customers gradually arrive in the system, forming as a queue, and get served by the servers in the system, and the equilibrium state of this process can be theoretically depicted by the model. The detail of M/M/C queue utilization in this study is provided as follows. Suppose the platform have a fleet of K vehicles, and the area of the studied

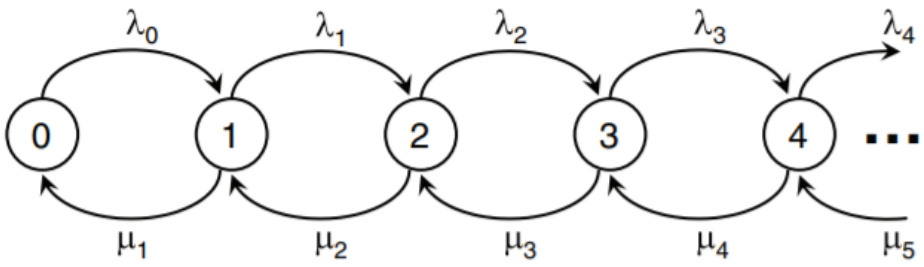


Figure 1: Birth and death process in each single block

region is A_{total}. The platform partitions the space into M equal-size matching blocks. Thus, the average number of vehicles in each block is $c = K/M$, the area of one block is $A = A_{total}/M$, and the arrival rate of passengers for one block is $\lambda = \lambda_{unit}A$. Since we mainly consider the stationary equilibrium state of the market and focus on the extraction of general insights, we only aggregately consider the average number of vehicles in each block in this study. The impact of the changing demand and supply can be explored in the future study. The average service rate of an individual vehicle is defined as μ. Considering the similarity in supply and demand situation for each block as assumed in the last section, we can focus on the matching process within each individual block, which is modelled as a M/M/c queue and described by the birth-death process in Fig. 1. The state represents the number of passengers in the block, and the "birth" rate and "death" rate are respectively λ_n and μ_n. For rate of completions (or "deaths"), it depends on the number of passengers in the block. If there are c or more passengers, then all c idle vehicles (as previously mentioned, there are c idle vehicles in one block on average) must be matched and become busy. Otherwise, when there are fewer than c passengers in the system, $n < c$, only n of the c idle vehicles will be matched and occupied. This leads to the following state-dependent service rate:

$$\mu_n = \begin{cases} n\mu, & 1 \leq n < c \\ c\mu, & n \geq c \end{cases} \tag{1}$$

For arrival rate of passengers ("birth" rate), passengers' potential abandonment of joining the queue can be described by a function b_n, and thus $\lambda_n = b_n\lambda$. When there are fewer passengers than the average number of idle vehicles c, it can be expected that the arriving passenger can get served instantly, resulting in no abandonment behavior for the passenger, that is, $b_n = 1$. Otherwise, the system of one block is fully busy with system service rate $c\mu$, and the waiting time can be expected as $\frac{n}{c\mu}$ for the arriving passenger n. When $\frac{n}{c\mu} < w_0$, the passengers are still willing to join, according to the assumed balking behavior mentioned in the last section. When $\frac{n}{c\mu} \geq w_0$ ($n \geq w_0 c\mu = N_p$), the abandonment emerges and $b_n = 0$. The resulting arrival rate is summarized in the equation 2:

$$\lambda_n = b_n\lambda = \begin{cases} \lambda, & n < N_p = c\mu w_0 \\ 0, & n \geq N_p \end{cases} \tag{2}$$

To find the steady-state probability p_n, we first list flow balance equations below:

$$\begin{cases} p_n = p_0 \prod_{i=1}^{n} \frac{\lambda_{i-1}}{\mu_i} \\ \sum_{i=0}^{\infty} p_i = 1 \end{cases} \tag{3}$$

The upper one in 3 depicts the relationship between p_0 and the probability of any given state, while the lower one limits the summation of the probabilities of all the queueing states to be one. Combining Eq. 1 to 3, the steady-state probability can be obtained as follows, where $r = \frac{\lambda}{\mu}$ and $\rho = \frac{\lambda}{c\mu} = \frac{r}{c}$. For clarity, the detailed derivation process is provided in Appendix B.

$$p_n = \begin{cases} p_0 \frac{\lambda^n}{n! \mu^n}, & 0 \le n < c \\ p_0 \frac{\lambda^n}{c^{n-c} c! \mu^n}, & c \le n \le N_p \\ 0, & n > N_p \end{cases} \tag{4}$$

$$p_0 = \begin{cases} (\sum_{n=0}^{c-1} \frac{r^n}{n!} + \frac{r^c}{c!} \cdot \frac{1-\rho^{N_p-c+1}}{1-\rho})^{-1}, & \rho \neq 1 \\ [\sum_{n=0}^{c-1} \frac{r^n}{n!} + \frac{r^c}{c!}(N_p - c + 1)]^{-1}, & \rho = 1 \end{cases} \tag{5}$$

3.3 Metrics

In this part, we first introduce several key system performance metrics, including average queueing time, average pick-up time and average total waiting time, based on the model presented in the last section. Afterwards, we utilize an equation to capture the intrinsic relationship between system service rate and average pick-up time.

1) Average queueing time W_q

$$W_q = \frac{L_q}{\lambda(1 - p_{N_p})} \tag{6}$$

$$= \frac{0 + \sum_{n=c+1}^{N_p} (n-c) p_n}{\lambda(1 - p_{N_p})} \tag{7}$$

$$= \frac{p_0}{\lambda(1 - p_{N_p})} \cdot \frac{r^c \rho}{c!} \cdot \frac{\rho^{N_p - c}[(N_p - c)(\rho - 1) - 1] + 1}{(\rho - 1)^2} \tag{8}$$

2) Average pick-up time W_p

$$W_p = \frac{1}{v}[\sum_{n=0}^{c-1} (p_n d(c-n)\sqrt{A}) + \sum_{n=c}^{N_p} p_n d(1)\sqrt{A}] \tag{9}$$

$$= \frac{\sqrt{A}p_0}{v}[\sum_{n=0}^{c-1}\frac{\lambda^n}{n!\mu^n}d(c-n) + d(1)\frac{r^c}{c!}\cdot\frac{1-\rho^{N_p-c+1}}{1-\rho}] \tag{10}$$

3) Average total waiting time W_{tw}

$$W_{tw} = W_q + W_p \tag{11}$$

The derivation process for the metrics are summarized in Appendix B. Here, W_q represents the average time spent in queue for a passenger in the given system, which has significant impacts on different shake-holders. For passengers, it can influence the final arrival time for the trips, and thus highly correlates to the social welfare for the whole passenger group; for ride-sourcing platform, the average queueing time is an important factor for passengers' choices between the platform and other group of competitors, which can further influence the long-term revenue of the platform. W_p is another significant system performance metrics. It represents the time for the idle vehicle (or server) to reach its assigned passenger, which captures one major difference of the ride-sourcing system from classic counter service system, where passengers are always assumed to get served immediately once they finish queueing. For both platform managers and passengers, the total waiting time before passengers get picked may be a main concern. To this end, we utilize W_{tw} to consider the integrated delay of service, including both queueing time and pick-up time.

The system performance metrics can also influence some system variables, such as the service rate μ for vehicles. As discussed in Feng et al. (2020) and Besbes et al. (2021), the pick-up process can also be treated as part of the service procedure, and we carefully consider this point via the equation below:

$$\mu = \frac{1}{t + W_p} \tag{12}$$

where t is the average trip time, W_p is the average pick-up time, and the combination of the two involves the whole process for the service. The service rate now becomes an endogenous variable determined by Eq. 9 and Eq. 12, which increases the complexity of finding steady-state probabilities. In next section, we develop an efficient solution-finding approach to solve the problem.

4 Modelling analysis

In this section, we use the developed model to make analysis from both intuitive and theoretic perspective. We first introduce some parameters setting, and develop an approach to find the endogenous service rate and the steady-state probability. Afterwards, we draw the plots of metrics in terms of block size, in order to uncover the impacts of block size on the metrics under different traffic related parameters. Managerial insights are then provided based on the patterns shown in the plots. In addition, an in-depth analysis is made for a special phenomenon (which we call plateau phenomenon as mentioned before) both theoretically and intuitively.

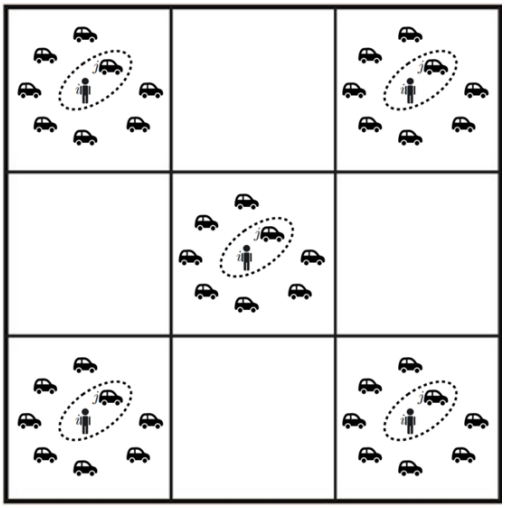


Figure 2: Studied region and partition

4.1 Case setting and solution finding

We focus on a square region as shown in Fig. 2, which is partitioned into equal-size blocks. The square grid is also often adopted for region partition in other researches related to demand management, such as Wang et al. (2019b), Yoshida et al. (2020) and Pan et al. (2019). The side length of the region, $a = \sqrt{A}$, is 20 km, and the area of the region is 400 km^2, about half of the New York City. To consider the impacts of different block sizes, we set the block area ranging from 1 km^2 to 25 km^2. Referring to Vignon et al. (2021), the benchmark unit arrival rate λ_{unit} is adjusted to 0.133 per minute per km^2, and the vehicle speed is 10 m/s. Similar to Feng et al. (2020) and Besbes et al. (2021), we mainly consider the case with $\rho < 1$, and thus the passengers are assumed always willing to join the queue when arriving, that is, $w_0 = +\infty$. The resulting modification to Eq. 4 to Eq. 9 is shown below.

$$p_n = \begin{cases} p_0 \dfrac{\lambda^n}{n!\mu^n}, & 0 \le n < c \\ p_0 \dfrac{\lambda^n}{c^{n-c} c!\mu^n}, & c \le n \end{cases} \tag{13}$$

$$p_0 = \left(\sum_{n=0}^{c-1} \frac{r^n}{n!} + \frac{r^c}{c!(1-\rho)}\right)^{-1}, \quad \rho < 1 \tag{14}$$

$$W_q = p_0 \cdot \frac{r^c}{c!(c\mu)(1-\rho)^2} \tag{15}$$

$$W_p = \frac{\sqrt{A} p_0}{v} \left[\sum_{n=0}^{c-1} \frac{\lambda^n}{n!\mu^n} d(c-n) + d(1) \frac{r^c}{c!(1-\rho)}\right] \tag{16}$$

To guarantee $\rho < 1$, the benchmark vehicle fleet size is set to 1500. With a common assumption that the origin and destination of a trip is randomly and equally distributed in the studied square region, and considering the detour behavior of drivers via a coefficient 1.27 (Yang et al., 2018), the benchmark average trip time t can be calculated by $1.27 \times \frac{0.521a}{v} = 1323.3s$, where 0.521 represents the expected distance between two random points in a unit-size square (Moltchanov, 2012). Similarly, $d(1)$ is set to 0.521, and $d(i)$ is equal to $\frac{d(1)}{\sqrt{i}}$, following the result by Besbes et al. (2021).

Based on the benchmark parameters above, we further consider different fleet size, unit arrival rate and average trip time, in order to depict the impact of block size under different supply-demand scenarios. The specific settings are listed as follows:

- **Fleet size**: 1500 to 3000 for vehicle fleet size. All the other parameters are set to the benchmark ones.
- **Unit arrival rate**: 0.4 to 1.0 times benchmark unit arrival rate. All the other parameters are set to the benchmark ones.
- **Average trip time**: 300 s to 1323.3 s for average trip time. All the other parameters are set to the benchmark ones.

With the definitions above, the following steps are 1) to find the endogenous service rate; 2) to combine it with other exogenous parameters to obtain system performance metrics as shown in Eq. 11, 14, and 15. The second step is straightforward, and we only need to focus on the first step. We first combine Eq. 11 and 15, and obtain a new function F about service rate μ below.

$$F(\mu) = \sum_{n=0}^{c-1} \frac{\lambda^n}{n!\mu^n} \cdot [\frac{1}{\mu} - t - \frac{\sqrt{A}}{v} d(c-n)] + [\frac{1}{\mu} - t - \frac{\sqrt{A}}{v} d(1)] \cdot \frac{\lambda^c}{c!\mu^c} \cdot \frac{1}{1 - \frac{\lambda}{c\mu}}, \quad \frac{\lambda}{c} < \mu < \frac{1}{t} \tag{17}$$

The determination of endogenous service rate under Eq. 11 and 15 now becomes the root finding for Eq. 16 within the given interval. Based on the parameters defined before, we find that $F(\frac{\lambda}{c}) \cdot F(\frac{1}{t}) < 0$ is always true for the studied cases. Considering $F(\mu)$ is a continuous function within the interval, we adopt a bisection method to find the root of F (that is, the endogenous service rate), which is then utilized to generate the system performance metrics. The logic of bisection method is shown in Algorithm 1.

4.2 Graphical illustration and insights

Considering the complexity of the system with the endogenous service rate, we graphically illustrate the pattern of metrics under varying block size and supply-demand scenarios, and summarize insights for platform managers. In Fig 3, we depict curves of average queueing time and pick-up time with respect to block size under different fleet size, while similar plots are generated for different unit arrival rate and average trip time respectively in Fig 4 and Fig. 5.

From Fig. 3a, we observe that the average queueing time gradually decreases with block size. The reason is that the expansion of block increases the average number of vehicles within the block, which improves the total service speed and overwhelms the negative influence of more arriving passengers and larger pick-up distance on the system efficiency. The effect of increasing

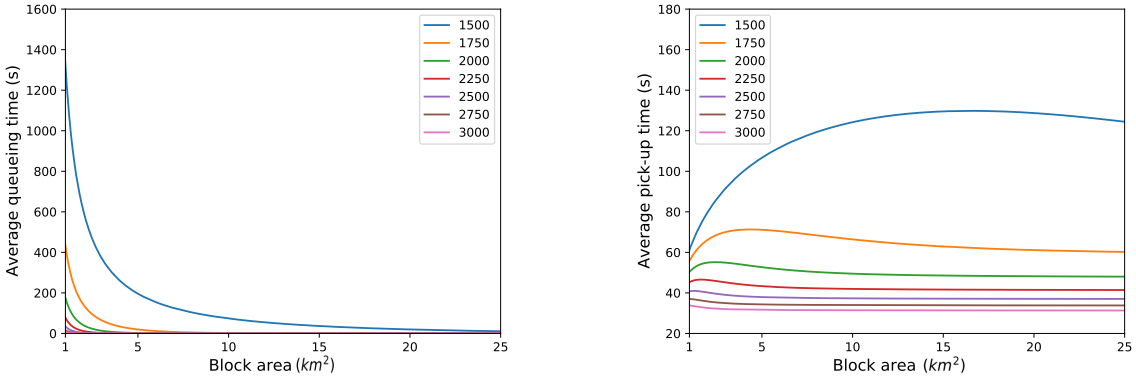
Algorithm 1 Bisection method to determine the service rate

Input: Function F**, lower bound** $\frac{\lambda}{c}$**, upper bound** $\frac{1}{t}$**, tolerable error** e**, a small value** ϵ
$x_0 = \frac{\lambda}{c} + \epsilon$
$x_1 = \frac{1}{t}$
while $x_1 - x_0 \geq e$ **do**
    $x_2 = \frac{x_0 + x_1}{2}$
    if $F(x_0) \cdot F(x_2) < 0$ **then**
        $x_1 = x_2$
    else
        $x_0 = x_2$
return x_0

block size on reducing queueing time is found more significant for smaller block size. As the original block size increases, the descending slope of queuing time with respect to block size becomes smoother. In addition, we also find that the queueing time becomes less sensitive to the variation of block size under larger vehicle fleet size. This is reasonable because the relative variation of the number of vehicles within a block become slower when it originally possesses many vehicles, resulting in a smaller response of the metrics. Moreover, when the block size is relatively small, the increasing fleet size is found able to reduce queueing time more significantly, similarly caused by the original degree of vehicle sufficiency. For average pick-up time in Fig. 3b, an observation is that the pick-up time generally increases as the block size extends, when the original block size is relatively small. In this case, the maximal pick-up distance and the number of people in the system both increase with block size, making the new arriving passenger more difficult to find a close vehicle to match. When the vehicle fleet size is large, the phenomenon becomes less obvious. Under a large block size, the pick-up time goes into a plateau with a nearly fixed value regardless of the change in block size. The potential reason is analyzed later via an approximation of pick-up time. In comparison with queueing time, the pickup time can be more effectively reduced by enlarging vehicle fleet size under a large block size, instead of a small one.

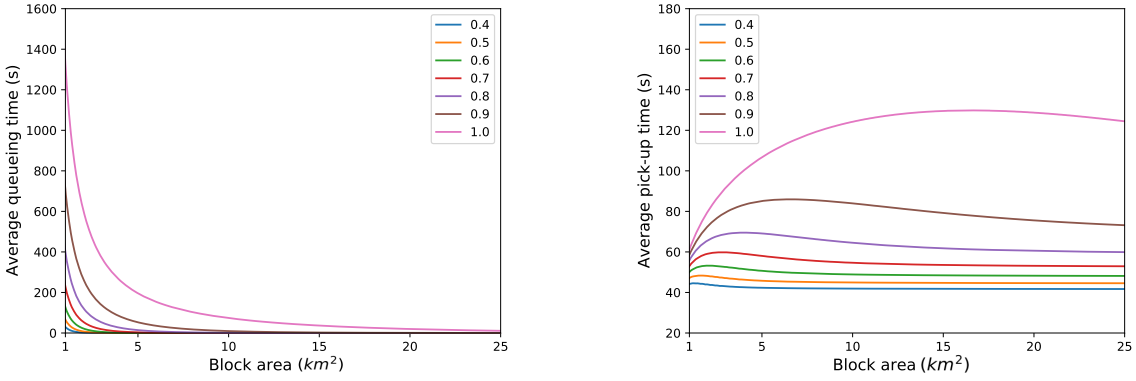
Similarly, Fig. 4 shows the impact of different unit arrival rate on the metrics. Generally, the average queueing time still decreases with the increase of block size, but the trend becomes less obvious as unit arrival rate declines, due to the more sufficient relative vehicle supply in the block. In addition, a smaller block size is better for the reduction of queueing time via limiting unit arrival rate. For pick-up time, its variation with respect to the block size is more significant under larger arrival rate. When the block size is large, the limitation of arrival rate can more effectively reduce average pick-up time. From Fig. 5, it is straightforward to find that the general patterns are highly similar to those in Fig. 4. The increase of average trip time lowers the service rate and reduces the system efficiency from the supply side, while the similarly negative impact results from the increase of unit arrival rate from the demand side.

In addition to the average queueing time and pick-up time, we also focus on the average total waiting time, a comprehensive metrics for the trip delay. The patterns are summarized in Fig. 6, considering the influence of different fleet size, unit arrival rate and average trip time. In general, the average total waiting time is decreasing with the extension of block size. When the block size is small, the curves are close to those of average queueing time. In comparison, under the range of larger block size, the trends are more similar to those of average pick-up time, where



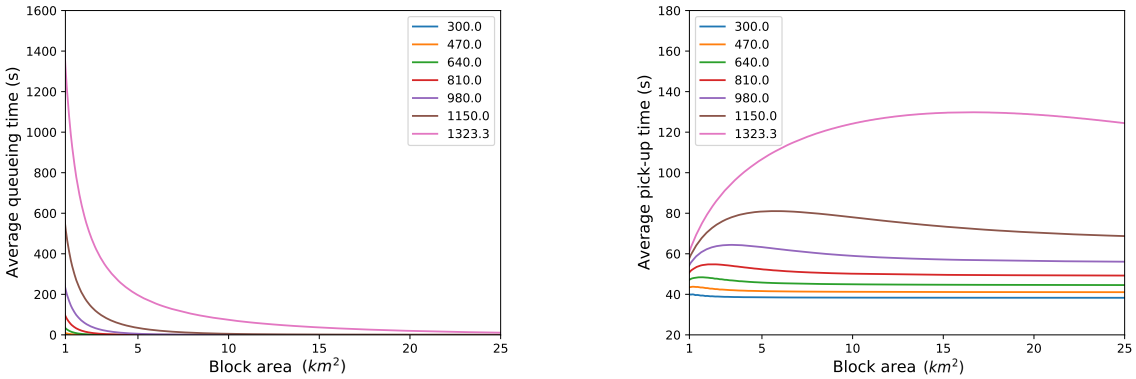
(a) (b)

Figure 3: Average queueing time and pick-up time under different block sizes and vehicle fleet size.



(a) (b)

Figure 4: Average queueing time and pick-up time under different block sizes and unit arrival rate. The value in the label represents the ratio of the studied arrival rate to the benchmark unit arrival rate.



(a) (b)

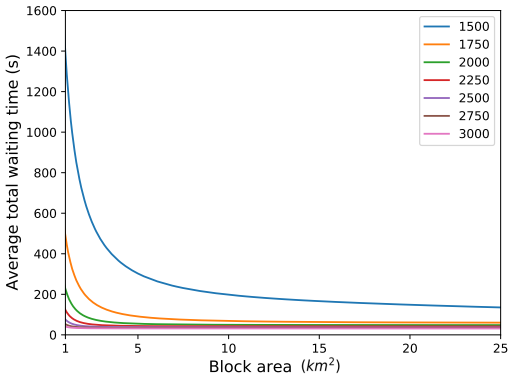
Figure 5: Average queueing time and pick-up time under different block sizes and average trip time (unit: s).

a plateau emerges for the metrics. The overall insights from these observations are summarized below, and an in-depth analysis is made in the next section to explain the phenomenon of plateau from both theoretic and intuitive perspective.

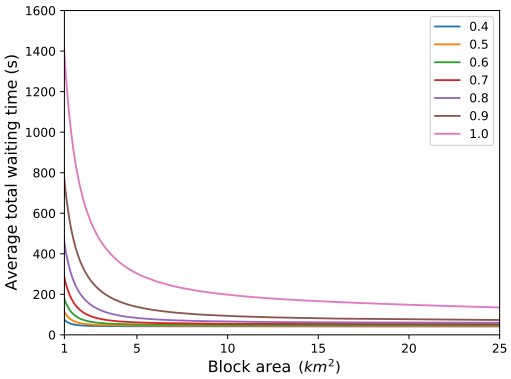
- An increased block size can reduce the average queueing time, and the effect is more significant under smaller block size, smaller vehicle fleet size, larger unit arrival rate or larger average trip time.
- An increased block size can increase the average pick-up time and the effect is more significant under smaller block size, smaller vehicle fleet size, larger unit arrival rate or larger average trip time.
- When the block size is fixed, the increase of vehicle fleet size or the limitation of passengers' arrival rate (via pricing operation, for example) can favor the reduction of average queueing time and pick-up time. The effect will be more obvious with smaller fixed block size for average queueing time, and larger block size for average pick-up time.
- For managers who aim to reduce average total waiting time for passengers in the block matching system, an useful method is to increase block size, especially when the original block size is relatively small. This metrics may become close to a fixed low value within the range of large block size, where the platform managers can choose their target block size based on other goals, such as computation speed or the coordination with other operations, without the need to worry its impact on average total waiting time.

4.3 Analysis for the plateau phenomenon

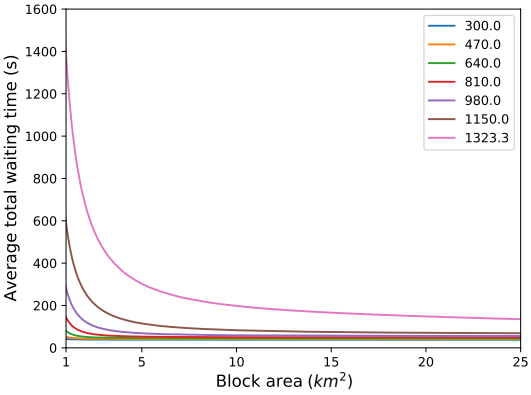
In this section we analyze the plateau of average total waiting time when block size is relatively large, where the metrics are basically fixed regardless of the variation of block size. Our analysis can be divided into two steps: 1) we construct an approximation formula for the average total waiting time W_{tw} under large block size, based on which the metrics is shown nearly a constant;



(a)



(b)



(c)

Figure 6: Average total waiting time under different block sizes and vehicle fleet size (a), unit arrival rate (b), average trip time (unit: *s*) (c). The value in the label of (b) represents the ratio of the studied arrival rate to the benchmark unit arrival rate.

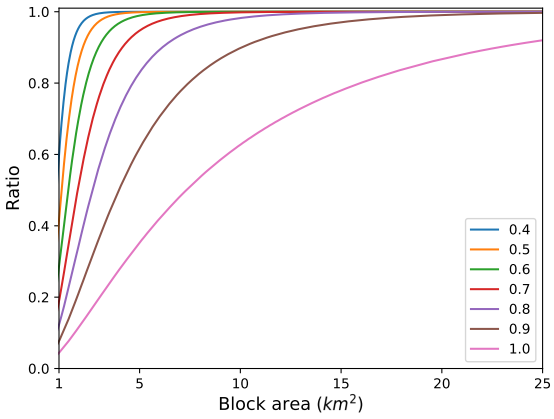


Figure 7: Ratio of average pickup time to average total waiting time. The value in the label represents the ratio of the studied arrival rate to the benchmark unit arrival rate.

2) the approximation is further analyzed from an intuitive perspective, in order to explain the reason for the emergence of the plateau.

From Fig. 3a, 4a and 5a, we find that the average queueing time becomes very small when the block size is larger than a certain threshold, such as 5 km^2. This indicates that the pick-up time is the major part of total waiting time within this interval of block size (say I_L for simplicity). To confirm the point, we depict the ratio of average pickup time to average total waiting time under different unit arrival rate in Fig. 7. The results show that the average pick-up time dominates queueing time in I_L, and thus leads to the following approximation:

$$W_{tw} \approx W_p = \frac{1}{v}[\sum_{n=0}^{c-1}(p_n d(c-n)\sqrt{A}) + \sum_{n=c}^{N_p} p_n d(1)\sqrt{A}], \quad A \in I_L \tag{18}$$

The small value of average queueing time in I_L indicates that the probability for passengers to wait in queue is low. The inference is confirmed by the curve in Fig. 8, where we select a certain unit arrival rate and observe the total probability for passengers to wait, $\sum_{n=c}^{N_p} p_n$, under block size in I_L. The probability is found very small for most of the block size, and thus we can only focus on the steady states without queueing, that is, $n = 0.....c-1$. The resulting approximation is made below:

$$W_{tw} \approx W_p \approx \frac{1}{v}[\sum_{n=0}^{c-1}(p_n d(c-n)\sqrt{A}) + 0 \cdot d(1)\sqrt{A}] \tag{19}$$

$$= \frac{\sqrt{A}}{v}\sum_{n=0}^{c-1} p_n d(c-n) \tag{20}$$

$$= \frac{\sqrt{A}}{v}\sum_{n=0}^{c-1} p_n \frac{d(1)}{\sqrt{\frac{K}{A_{total}}A - n}} \tag{21}$$

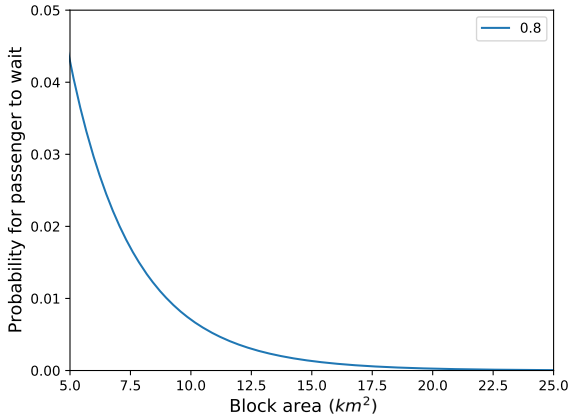


Figure 8: Probability for passenger to wait in queue under different average block sizes. The value in the label represents the ratio of the studied arrival rate to the benchmark unit arrival rate.

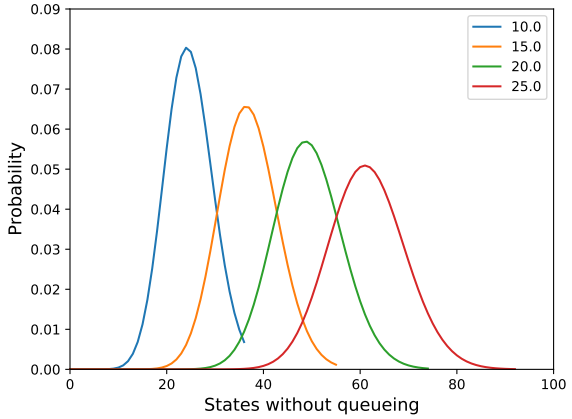


Figure 9: Probability distributions of steady states without queueing under certain values of block size and 0.8 times benchmark unit arrival rate. The value in the label represents block size (unit: km^2).

$$= \frac{d(1)}{v} \sum_{n=0}^{c-1} p_n \frac{1}{\sqrt{\frac{K}{A_{total}} - \frac{n}{A}}} \tag{22}$$

let $G(n)$ denote $\frac{1}{\sqrt{\frac{K}{A_{total}} - \frac{n}{A}}}$. For simplicity, we pick several value of block size in I_L for the previously selected service rate, and draw their probability distribution of steady states without queueing in Fig. 9. From the figure we discover two properties of the distribution: 1) The distribution is highly symmetric around certain point, say s; 2) The majority of the non-zero probability concentrates within a certain interval, say $[s - l, s + l]$. The two characteristics and the resulting transformation of W_{tw} can be described in the equations below:

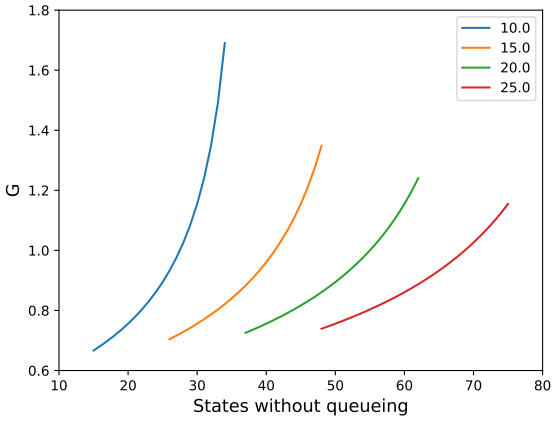


Figure 10: G(n) under certain values of block size and 0.8 times benchmark unit arrival rate. The value in the label represents block size (unit: km^2).

$$p_{n=s+m} \approx p_{n=s-m}, \quad n \in [s-l, s+l] \tag{23}$$

$$\sum_{n=s-l}^{s+l} p_n \approx 1 \tag{24}$$

$$W_{tw} \approx W_p \approx \frac{d(1)}{v} \sum_{n=s-l}^{s+l} p_n G(n) \tag{25}$$

Afterwards, we take a closer look at the function $G(n)$. The plots of $G(n)$ are also generated under different block sizes in Fig. 10. The curves show high linearity, indicating we can approximate $G(n)$ by some linear approximation $en + h$ (such as first-order Taylor Expansion around s), with certain coefficients e and h. Combining the linearity of $G(n)$ and the properties in Eq. 22 and 23, we can further transform W_{tw} as follows:

$$W_{tw} \approx W_p \approx \frac{d(1)}{v} \sum_{n=s-l}^{s+l} p_n(en+h) \tag{26}$$

$$\approx \frac{d(1)}{v}\{p_s(es+h) + \sum_{i=1}^{l}[p_{s+i}(e(s+i)+h) + p_{s-i}(e(s-i)+h)]\} \tag{27}$$

$$= \frac{d(1)}{v}[p_s(es+h) + 2\sum_{i=1}^{l} p_{s+i}(es+h)] \tag{28}$$

$$= \frac{d(1)}{v} \sum_{n=s-l}^{s+l} p_n(es+h) \tag{29}$$

$$\approx \frac{d(1)}{v}(es+h) \tag{30}$$

$$\approx \frac{d(1)}{v}G(s) \tag{31}$$

$$= \frac{d(1)}{v}\cdot\frac{1}{\sqrt{\frac{K}{A_{total}}-\frac{s}{A}}} \tag{32}$$

In Eq. 31, the non-constant variables are s and A, while $\frac{s}{A}$ is found always around 2.4 regardless of the variation of block size in I_L. Therefore, W_{tw} nearly becomes a constant in I_L, resulting in the plateau phenomenon. Based on the approximation, we can also interpret the reason for plateau in a more intuitive way. We first equivalently transform Eq. 31 into another form, $\frac{1}{v}\cdot\frac{d(1)\sqrt{A}}{\sqrt{\frac{K}{A_{total}}\cdot A-\frac{s}{A}\cdot A}}$. In this formula, three parts are directly related to the block size, including $d(1)\sqrt{A}$, $\frac{K}{A_{total}}\cdot A$ and $\frac{s}{A}\cdot A$. Considering the parameters $d(1)$, $\frac{K}{A_{total}}$ and $\frac{s}{A}$ are constants or near to constants, all the three parts are monotonically increasing with respect to the block size A, which represent different meanings. The first part represents the physical maximum pick-up distance within a block, and its extension naturally leads to the increase of average pick-up time. In comparison, the second part represents the average number of vehicles in the block, whose growth favors the reduction of average pick-up time under First-Dispatch rule. The reason is that the platform always assigns the closest idle vehicle to an arriving passenger when there are more vehicles than passengers in the block. With more vehicles available, it is always possible to find a closer vehicle for the passenger. For the third part, it is equal to the symmetric point s for steady state distribution, representing the most possible number of passengers in the block. When this value increases, more idle vehicles are occupied and the new arriving passenger has to choose the closest vehicle from a smaller pool of candidate vehicles, which naturally results in the increase of pick-up time. The effects of the three parts compensate with each other, leading to a fixed average total waiting time (average pick-up time) when the block size is large.

5 Model validation

5.1 Experiment design

To verify the insights and phenomena in the modelling part, we design a simulation study based on a realistic agent-based simulator. The simulation settings are similar to the benchmark setting utilized in section 4.1. The studied region is a square, which is further partitioned into smaller square blocks. Block size ranges from 1 to 25 km^2. The benchmark unit arrival rate is similarly set to 0.133 per minute per km^2. Passengers will not balk before joining the queue. The vehicle speed is 10 m/s. For trip generation, in the modelling part, we assume the origin and destination of a trip request by a passenger are both randomly and equally distributed in the studied square region, with a detour ratio 1.27. The estimation of the average passenger trip time is then 1323.3 s. In the experiment part, we adopt similar settings as in the modelling part. The origin and destination for a new generated trip are also randomly and independently selected in the studied region. Thus, the expectation of the passenger trip distance keeps the same. The matching rule is still First-Dispatch. For faster computation speed, we shrink the side length of the studied region into 10 km. The total vehicle fleet size is accordingly reduced to 375. Under the setting above, it is easy to find that the supply and demand related parameters for an individual block

keep the same as in the modelling section. It is reasonable to utilize the setting above to conduct simulation study for the validation of modelling results.

Considering the large number of iterations required to reach steady states in the simulation, we only focus on two scenarios: one with 0.8 times benchmark unit arrival rate, and another with 0.4 times benchmark unit arrival rate. The previous one represents the market with relatively high demand while the other corresponds to low demand. Under both the scenarios and different block sizes, the steady-state system performance metrics are documented and compared with the modelling results. In addition, we also test and collect the standard deviation of passenger queueing time and pickup time with respect to block size, in order to more deeply consider the experience of passengers under block matching. Based on the large demand scenario, we further record and compare the cumulative computation time for matching operation over a long period (100000 steps of simulation in the steady state of the system) under different block sizes, in order to show the effect of block size on the computation time.

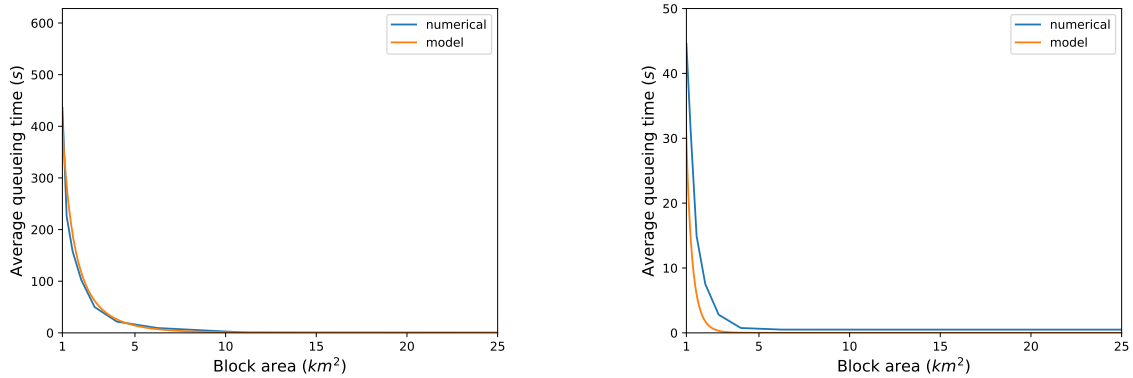
To conduct simulation, we develop an agent-based simulator with block matching rule, which is shown in Algorithm 2.

Algorithm 2 Simulator for a ride-sourcing market

1: Initialize states for platform and drivers.
2: **for** matching time interval $t = 0$ to T **do**
3: **Block matching:** Conduct matching between passenger and idle vehicle queues in each block respectively, following the First-Dispatch rule.
4: **Update matching outcomes:** The status of matched vehicles become occupied. The matched orders and vehicles are removed from the waiting queues of their current block.
5: **Request generation:** New orders are generated with origin and destination randomly distributed in the studied region. The new orders are added to the waiting queue of the block where their origins are located.
6: **Update states for next time interval:** Update states of drivers and orders in the system under next time interval. The drivers who finish their trips will join the waiting queues of idle vehicles of the current blocks.

5.2 Results and analysis

We repeat the simulations until convergence of metrics is reached for each experiment. The comparison between simulation results and modelling results are shown in Fig. 11 to 13. The curves generated by the proposed model matches well with the simulation results for all the three metrics under different arrival rates. The slight differences between the modelling and simulation results are common in M/M/c model, as discussed in Feng et al. (2020). The reason is that the drivers in the simulation may not obey the assumption of uniform spatial distribution, and the service time may not always obey exponential distribution, leading to larger variance of the whole system and thus the differences in performance metrics. Still, such differences are less than 10 seconds for most of the block size, which is acceptable for ride-sourcing platforms. The simulation results show that the average queueing time and total waiting time decreases with the increase of block size, while the average pick-up time may increase as block size extends in some interval. The variations of metrics with respect to block size are more obvious under smaller block size. When the block size is large, the metrics becomes less sensitive, and the plateau phenomenon emerges in the simulation results, as expected in the modelling analysis. To



(a) 0.8 ratio

(b) 0.4 ratio

Figure 11: Modelling and experiment results for average queueing time under different unit arrival rate. The ratio represents $\frac{studied\ unit\ arrival\ rate}{benchmark\ unit\ arrival\ rate}$.

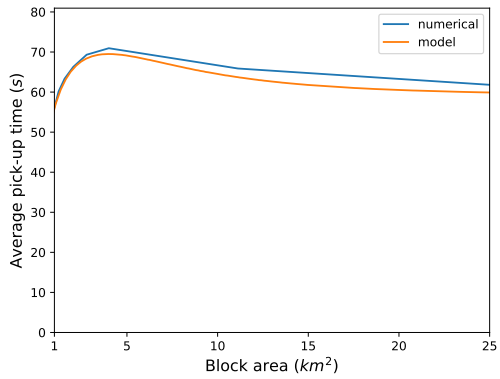
summarize, the phenomena and insights in the modelling phase are validated by the simulation results, demonstrating that the proposed model is an useful and reliable tool for analysis of the ride-sorucing market with block matching system.

Besides, the results for the standard deviation (std) of passenger queueing time and pickup time are shown in Fig. 14 and Fig. 15. Std of queueing time generally deceases with the increase of block size. The metrics is obviously larger under higher arrival rate when the block size is relatively small. When the block size is large, the metrics always becomes close to zero, since the passenger is always immediately matched with some driver. Compared to queueing time, the std of pickup time is much smaller and less sensitive to the change of block size. To summarize, ride-sourcing platforms with block matching system may avoid excessively small block size, to prevent the emergence of extremely long waiting time for some passengers, which may result in bad travelling experience.

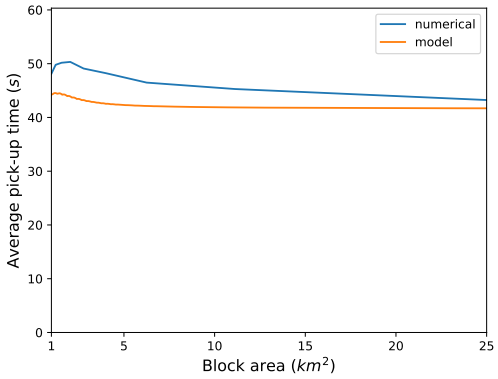
In addition, the comparison of computation time is shown in Fig. 16. The figure demonstrates that the computation time for matching operation is increasing with block size extending (that is, increasing with the number of blocks decreasing), matched with the discussion in the introduction section. Under 25 km^2 block area, the computation time for matching increases over 40 %, compared to that under 1 km^2. From the perspective of waiting time and computation time, a proper block area in this case can be 5 km^2, where the average total waiting time is in the plateau (as shown in Fig 13a), and the computation time is lowest compared to the other larger block size of the plateau. The determination is supported by both theoretical and simulation results.

6 Conclusion

This paper presents a model to approximate a ride-sourscing system with matching blocks. The model can be used to determine the proper block size, and investigate the impacts of block size on three key system performance metrics, including passengers' average queueing time, average pick-up time and average total waiting time. The model utilizes a M/M/c queue to depict the

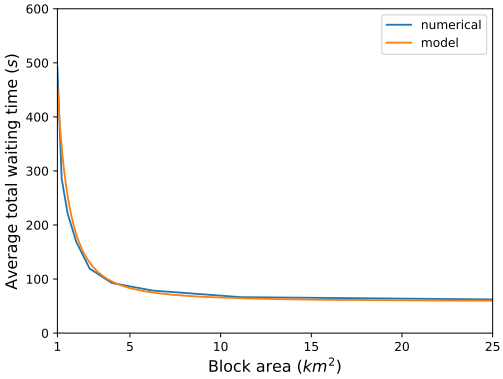


(a) 0.8 ratio

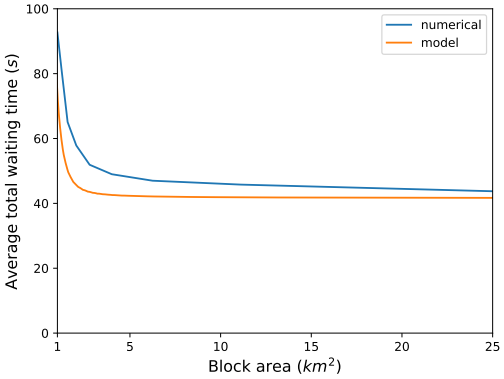


(b) 0.4 ratio

Figure 12: Modelling and experiment results for average pick-up time under different unit arrival rate. The ratio represents $\frac{studied\ unit\ arrival\ rate}{benchmark\ unit\ arrival\ rate}$.

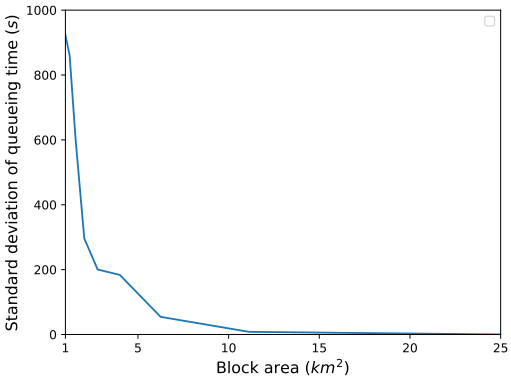


(a) 0.8 ratio

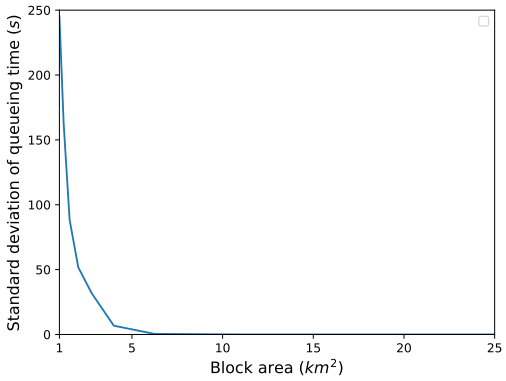


(b) 0.4 ratio

Figure 13: Modelling and experiment results for average total waiting time under different unit arrival rate. The ratio represents $\frac{studied\ unit\ arrival\ rate}{benchmark\ unit\ arrival\ rate}$.

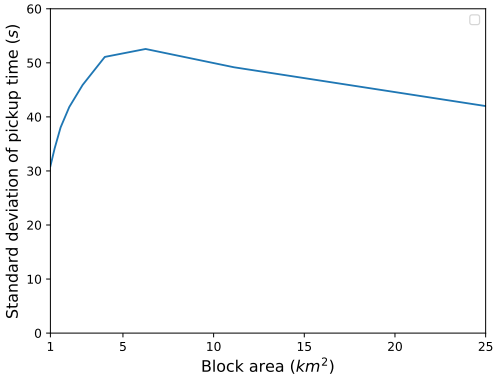


(a) 0.8 ratio

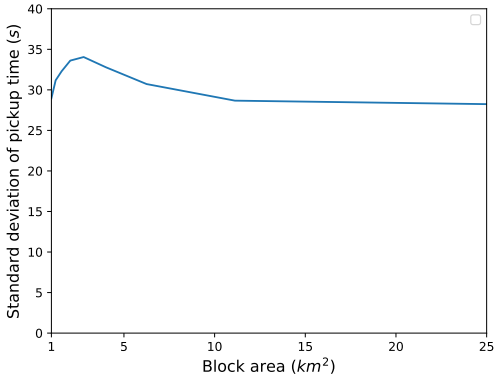


(b) 0.4 ratio

Figure 14: Experiment results for standard deviation of queueing time under different unit arrival rate. The ratio represents $\frac{studied\ unit\ arrival\ rate}{benchmark\ unit\ arrival\ rate}$.



(a) 0.8 ratio



(b) 0.4 ratio

Figure 15: Experiment results for standard deviation of pickup time under different unit arrival rate. The ratio represents $\frac{studied\ unit\ arrival\ rate}{benchmark\ unit\ arrival\ rate}$.

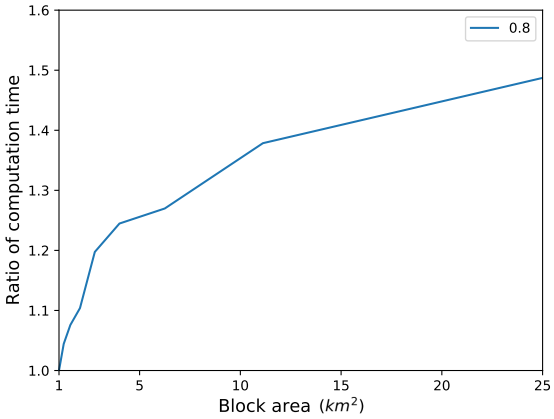


Figure 16: Comparison of cumulative computation time under different block sizes. The vertical axis represents the ratio of the cumulative computation time for matching operations under a certain block area to that under the benchmark block area (1 km^2). The value in the label represents the ratio of the studied arrival rate to the benchmark unit arrival rate.

matching process in each block, where the service rate is treated as an endogenous variable depending on the average pick-up time. A solution-finding approach is developed to solve for the endogenous service rate and the corresponding steady-state probabilities. The trends of the key metrics with respect to the block size are then portrayed and analyzed under different supply-demand scenarios. It is found that passengers' total waiting time first decreases and then keeps unchanged (reaching a plateau) as the block size increases. This indicates that the platform can almost select any block size in the plateau to guarantee passengers' total waiting time is minimized. Furthermore, an in-depth analysis is made for the plateau phenomenon of total waiting time when block size is large, based on a theoretic approximation and an intuitive interpretation. By conducting a large-scale simulation study, we validate that the proposed model can well approximate ride-sourcing systems and verify the observations and insights obtained so far.

As for future research, the block matching problem in ride-sourcing systems can be further explored from the following aspects: 1) Bipartite matching can be also implemented in a block-wise way, requiring new theoretic models to approximate the corresponding system performance metrics. 2) It is interesting to investigate the joint-decision with other operations (such as pricing or rewarding), e.g. multiple objectives can be jointly considered for optimizing overall system performance. 3) The model can be further adjusted to consider the market with ride-splitting service. Under spatial heterogeneous scenario (the distributions of demand and supply patterns are different across the whole area), a simple yet effective potential solution is to partition the whole area into smaller sub-areas. The historical supply and demand patterns for each block within the same sub-area should be close to each other. Proper and different block sizes can be set based on our model for each sub-area, respectively. Still, the detailed adjustment of the current model under such scenario is worth further exploration. 4) Proper models under changing demand and supply patterns can be further considered. Under mild changing pattern, one can rely on the steady-state models. If the pattern changes highly dynamically and shows high heterogeneity, some dynamic model (like Markov Decision Process) may be used. But such dynamic models are generally time-costing to find a solution and generate decisions.

Acknowledgements

The work described in this paper was supported by grants from Hong Kong Research Grants Council under projects HKUST16210520 and HKU15209121, and a grant from NSFC/RGC Joint Research Scheme under project N HKUST627/18 (NSFC-RGC 71861167001).

Appendix A

A1. Discussion of time complexity for block matching

Consider a region with N passenger arriving in order, and a total of K idle vehicles within a certain time period. Assume $N < K$ because $\rho < 1$ is considered in this study, which means supply can generally satisfy demand. Consider there are k available vehicles when a passenger arrives. Under First-Dispatch rule, the platform need to find the nearest idle vehicle to the arriving passenger. The core process can be simplified into finding the minimum from an array with length k, whose item represents the distance between the passenger and an idle vehicle. The time complexity for this process is $\mathcal{O}(k)$. For simplicity, consider a most simple yet efficient way to find the minimum: compare item in the array in order. Under the worst case where the minimum item is always at the end of array, we need $k-1$ comparison to find the minimum for an individual passenger. Thus, the total amount of comparison required for all the arriving passengers within this time period is $\sum_{k=K-N+1}^{K} k-1 = \frac{(2K-N+1)N}{2} - N$, representing $\mathcal{O}((K-N)N)$ time complexity. Consider now we partition the region into M blocks. On average, each block has $\frac{N}{M}$ arriving passengers and $\frac{K}{M}$ vehicles, resulting in $\mathcal{O}(\frac{(K-N)N}{M^2})$ time complexity for each block, and $\mathcal{O}(\frac{(K-N)N}{M})$ for the whole region. This indicates that the increase of the number of blocks favors the computation speed for the matching system.

Appendix B

B1. Derivation of steady-state probability

From classic queueing theory (Shortle et al., 2018), we know that once we obtain the formula of p_0, the other probabilities for other states can be easily derived based on Eq. 3. For simplicity, we only show the derivation of p_0 for $\rho \neq 1$. By integrating the first row of Eq. 3 into the second row, we can have:

$$p_0 = \left(\sum_{n=0}^{c-1} \frac{\lambda^n}{n!\mu^n} + \sum_{n=c}^{N_p} \frac{\lambda^n}{c^{n-c}c!\mu^n}\right)^{-1} \tag{33}$$

$$= \left(\sum_{n=0}^{c-1} \frac{r^n}{n!} + \sum_{n=c}^{N_p} \frac{r^n}{c^{n-c}c!}\right)^{-1} \tag{34}$$

$$= \left(\sum_{n=0}^{c-1} \frac{r^n}{n!} + \frac{r^c}{c!} \cdot \sum_{n=c}^{N_p} \left(\frac{r}{c}\right)^{n-c}\right)^{-1} \tag{35}$$

$$= \left(\sum_{n=0}^{c-1} \frac{r^n}{n!} + \frac{r^c}{c!} \cdot \sum_{m=0}^{N_p-c} \left(\frac{r}{c}\right)^m\right)^{-1} \tag{36}$$

$$= \left(\sum_{n=0}^{c-1} \frac{r^n}{n!} + \frac{r^c}{c!} \cdot \sum_{m=0}^{N_p-c} \rho^m\right)^{-1} \tag{37}$$

$$= \left(\sum_{n=0}^{c-1} \frac{r^n}{n!} + \frac{r^c}{c!} \cdot \frac{1-\rho^{N_p-c+1}}{1-\rho}\right)^{-1} \tag{38}$$

B2. Derivation of metrics

For simplicity, we still mainly consider the case with $\rho \neq 1$. To obtain W_q, we first derive average queue length by definitions and some simple simplifications:

$$L_q = 0 + \sum_{n=c+1}^{N_p} (n-c)p_n \tag{39}$$

$$= \sum_{n=c+1}^{N_p} (n-c)\frac{r^n}{c^{n-c}c!}p_0 \tag{40}$$

$$= \frac{r^c p_0}{c!} \sum_{n=c+1}^{N_p} (n-c)\rho^{n-c} \tag{41}$$

$$= \frac{r^c p_0}{c!} \sum_{m=1}^{N_p-c} m\rho^m \tag{42}$$

$$= \frac{r^c \rho p_0}{c!} \frac{\mathrm{d}\sum_{m=1}^{N_p-c} \rho^m}{\mathrm{d}\rho} \tag{43}$$

$$= \frac{r^c \rho p_0}{c!} \frac{\mathrm{d}}{\mathrm{d}\rho}\left[\frac{\rho(1-\rho^{N_p-c})}{1-\rho}\right] \tag{44}$$

$$= \frac{r^c \rho p_0}{c!} \cdot \frac{\rho^{N_p-c}[(N_p-c)(\rho-1)-1]+1}{(\rho-1)^2} \tag{45}$$

Following Little's Law (Little, 1961), we can further obtain:

$$W_q = \frac{L_q}{\lambda(1-p_{N_p})} \tag{46}$$

$$= \frac{p_0}{\lambda(1-p_{N_p})} \cdot \frac{r^c \rho}{c!} \cdot \frac{\rho^{N_p-c}[(N_p-c)(\rho-1)-1]+1}{(\rho-1)^2} \tag{47}$$

Here, we modify the arrival rate from λ to $\lambda(1-p_{N_p})$. The reason is that in our scenario, the passengers may possibly abandon joining the queue if the length of the queue is out of their tolerance. The probability for this situation is p_{N_p}, where N_p represents the longest queue length the passengers can accept. To this end, the actual arrival rate becomes $\lambda(1-p_{N_p})$. For average pick-up time, the derivation process is also not complex:

$$W_p = \frac{1}{v}[\sum_{n=0}^{c-1}(p_n d(c-n)\sqrt{A}) + \sum_{n=c}^{N_p} p_n d(1)\sqrt{A}] \tag{48}$$

$$= \frac{\sqrt{A}}{v}[\sum_{n=0}^{c-1} p_0 \frac{\lambda^n}{n!\mu^n} d(c-n) + d(1)\sum_{n=c}^{N_p} p_0 \frac{\lambda^n}{c^{n-c}c!\mu^n}] \tag{49}$$

$$= \frac{\sqrt{A}p_0}{v}[\sum_{n=0}^{c-1} \frac{\lambda^n}{n!\mu^n} d(c-n) + d(1)\frac{r^c}{c!}\cdot\frac{1-\rho^{N_p-c+1}}{1-\rho}] \tag{50}$$

Here, $\frac{1}{v}\sum_{n=0}^{c-1}(p_n d(c-n)\sqrt{A})$ represents the average pickup time when the queue length is smaller than c, that is, there are still idle drivers within the block to serve passengers immediately. In comparison, $\frac{1}{v}\sum_{n=c}^{N_p} p_n d(1)\sqrt{A}$ represents the average pickup time when the queue length is larger than c, and passengers have to wait in queue for matching.

References

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

Bai, J., So, K. C., Tang, C. S., Chen, X., and Wang, H. (2019). Coordinating supply and demand on an on-demand service platform with impatient customers. *Manufacturing & Service Operations Management*, 21(3):556–570.

Banerjee, S., Riquelme, C., and Johari, R. (2015). Pricing in ride-share platforms: A queueing-theoretic approach. *Available at SSRN 2568258*.

Bazan, P., Djanatliev, A., Pruckner, M., German, R., and Lauer, C. (2018). Rebalancing and fleet sizing of mobility-on-demand networks with combined simulation, optimization and queueing network analysis. In *2018 Winter Simulation Conference (WSC)*, pages 1527–1538. IEEE.

Besbes, O., Castro, F., and Lobel, I. (2021). Spatial capacity planning. *Operations Research*.

Braverman, A., Dai, J. G., Liu, X., and Ying, L. (2019). Empty-car routing in ridesharing systems. *Operations Research*, 67(5):1437–1452.

Calafiore, G. C., Novara, C., Portigliotti, F., and Rizzo, A. (2017). A flow optimization approach for the rebalancing of mobility on demand systems. In *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, pages 5684–5689. IEEE.

Castillo, J. C., Knoepfle, D., and Weyl, G. (2017). Surge pricing solves the wild goose chase. In *Proceedings of the 2017 ACM Conference on Economics and Computation*, pages 241–242.

Chen, H., Jiao, Y., Qin, Z., Tang, X., Li, H., An, B., Zhu, H., and Ye, J. (2019). Inbede: Integrating contextual bandit with td learning for joint pricing and dispatch of ride-hailing platforms. In *2019 IEEE International Conference on Data Mining (ICDM)*, pages 61–70. IEEE.

Courcoubetis, C. and Dimakis, A. (2018). A surge-type pricing in ridesharing systems is stability optimal. *Available at SSRN 3145663*.

Feng, G., Kong, G., and Wang, Z. (2020). We are on the way: Analysis of on-demand ride-hailing systems. *Manufacturing & Service Operations Management*.

He, Q.-C., Nie, T., Yang, Y., and Shen, Z.-J. (2021). Beyond repositioning: Crowd-sourcing and geo-fencing for shared-mobility systems. *Production and Operations Management*.

Jacob, J. and Roet-Green, R. (2021). Ride solo or pool: Designing price-service menus for a ride-sharing platform. *European Journal of Operational Research*.

Jiang, S., Chen, L., Mislove, A., and Wilson, C. (2018). On ridesharing competition and accessibility: Evidence from uber, lyft, and taxi. In *Proceedings of the 2018 World Wide Web Conference*, pages 863–872.

Ke, J., Xiao, F., Yang, H., and Ye, J. (2020). Learning to delay in ride-sourcing systems: a multi-agent deep reinforcement learning framework. *IEEE Transactions on Knowledge and Data Engineering*.

Lee, D.-H., Wang, H., Cheu, R. L., and Teo, S. H. (2004). Taxi dispatch system based on current demands and real-time traffic conditions. *Transportation Research Record*, 1882(1):193–200.

Li, S., Luo, Q., and Hampshire, R. C. (2021). Optimizing large on-demand transportation systems through stochastic conic programming. *European Journal of Operational Research*.

Li, S., Tavafoghi, H., Poolla, K., and Varaiya, P. (2019). Regulating tncs: Should uber and lyft set their own rules? *Transportation Research Part B: Methodological*, 129:193–225.

Liao, Z. (2003). Real-time taxi dispatching using global positioning systems. *Communications of the ACM*, 46(5):81–83.

Little, J. D. (1961). A proof for the queuing formula: L= λ w. *Operations research*, 9(3):383–387.

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

Moltchanov, D. (2012). Distance distributions in random networks. *Ad Hoc Networks*, 10(6):1146–1166.

Özkan, E. and Ward, A. R. (2020). Dynamic matching for real-time ride sharing. *Stochastic Systems*, 10(1):29–70.

Pan, Z., Wang, Z., Wang, W., Yu, Y., Zhang, J., and Zheng, Y. (2019). Matrix factorization for spatio-temporal neural networks with applications to urban flow prediction. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pages 2683–2691.

Qiu, H., Dai, X., and Chen, J. (2020). A macroscopic analysis of curbside stopping activities of on-demand mobility service. In *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*, pages 1–6. IEEE.

Ruch, C., Richards, S. M., and Frazzoli, E. (2019). The value of coordination in one-way mobility-on-demand systems. *IEEE Transactions on Network Science and Engineering*, 7(3):1170–1181.

Sayarshad, H. R. and Chow, J. Y. (2017). Non-myopic relocation of idle mobility-on-demand vehicles as a dynamic location-allocation-queueing problem. *Transportation Research Part E: Logistics and Transportation Review*, 106:60–77.

Seow, K. T., Dang, N. H., and Lee, D.-H. (2009). A collaborative multiagent taxi-dispatch system. *IEEE Transactions on Automation science and engineering*, 7(3):607–616.

Shah, S., Lowalekar, M., and Varakantham, P. (2020). Neural approximate dynamic programming for on-demand ride-pooling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 507–515.

Shi, J., Gao, Y., Wang, W., Yu, N., and Ioannou, P. A. (2019). Operating electric vehicle fleet for ride-hailing services with reinforcement learning. *IEEE Transactions on Intelligent Transportation Systems*.

Shortle, J. F., Thompson, J. M., Gross, D., and Harris, C. M. (2018). *Fundamentals of queueing theory*, volume 399. John Wiley & Sons.

Spieser, K., Samaranayake, S., and Frazzoli, E. (2016a). Vehicle routing for shared-mobility systems with time-varying demand. In *2016 American Control Conference (ACC)*, pages 796–802. IEEE.

Spieser, K., Samaranayake, S., Gruel, W., and Frazzoli, E. (2016b). Shared-vehicle mobility-on-demand systems: A fleet operator's guide to rebalancing empty vehicles. In *Transportation Research Board 95th Annual Meeting*, number 16-5987. Transportation Research Board.

Tang, X., Qin, Z., Zhang, F., Wang, Z., Xu, Z., Ma, Y., Zhu, H., and Ye, J. (2019). A deep value-network based approach for multi-driver order dispatching. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 1780–1790.

Taylor, T. A. (2018). On-demand service platforms. *Manufacturing & Service Operations Management*, 20(4):704–720.

Tong, Y., Chen, Y., Zhou, Z., Chen, L., Wang, J., Yang, Q., Ye, J., and Lv, W. (2017). The simpler the better: a unified approach to predicting original taxi demands based on large-scale online platforms. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1653–1662.

Vazifeh, M. M., Santi, P., Resta, G., Strogatz, S. H., and Ratti, C. (2018). Addressing the minimum fleet problem in on-demand urban mobility. *Nature*, 557(7706):534–538.

Vignon, D. A., Yin, Y., and Ke, J. (2021). Regulating ridesourcing services with product differentiation and congestion externality. *Transportation Research Part C: Emerging Technologies*, 127:103088.

Wang, G., Zhang, H., and Zhang, J. (2019a). On-demand ride-matching in a spatial model with abandonment and cancellation. *Available at SSRN 3414716*.

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

Wang, R. and Honnappa, H. (2017). The "concert queueing game" with feedback routing. Technical report, Working Paper, Purdue University.

Wang, Y., Yin, H., Chen, H., Wo, T., Xu, J., and Zheng, K. (2019b). Origin-destination matrix prediction via graph convolution: a new perspective of passenger demand modeling. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1227–1235.

Waserhole, A. and Jost, V. (2016). Pricing in vehicle sharing systems: Optimization in queuing networks with product forms. *EURO Journal on Transportation and Logistics*, 5(3):293–320.

Wollenstein-Betech, S., Paschalidis, I. C., and Cassandras, C. G. (2020). Joint pricing and rebalancing of autonomous mobility-on-demand systems. In *2020 59th IEEE Conference on Decision and Control (CDC)*, pages 2573–2578. IEEE.

Wong, K.-I. and Bell, M. G. (2006). The optimal dispatching of taxis under congestion: A rolling horizon approach. *Journal of advanced transportation*, 40(2):203–220.

Xu, Z., Li, Z., Guan, Q., Zhang, D., Li, Q., Nan, J., Liu, C., Bian, W., and Ye, J. (2018). Large-scale order dispatch in on-demand ride-hailing platforms: A learning and planning approach. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 905–913.

Xu, Z., Yin, Y., Chao, X., Zhu, H., and Ye, J. (2020a). A generalized fluid model of ride-hailing systems. *Available at SSRN 3743112*.

Xu, Z., Yin, Y., and Ye, J. (2020b). On the supply curve of ride-hailing systems. *Transportation Research Part B: Methodological*, 132:29–43.

Yahia, C. N., de Veciana, G., Boyles, S. D., Abou Rahal, J., and Stecklein, M. (2021). Book-ahead & supply management for ridesourcing platforms. *Transportation Research Part C: Emerging Technologies*, 130:103266.

Yan, C., Zhu, H., Korolko, N., and Woodard, D. (2020). Dynamic pricing and matching in ride-hailing platforms. *Naval Research Logistics (NRL)*, 67(8):705–724.

Yang, H., Ke, J., and Ye, J. (2018). A universal distribution law of network detour ratios. *Transportation Research Part C: Emerging Technologies*, 96:22–37.

Yang, H., Qin, X., Ke, J., and Ye, J. (2020). Optimizing matching time interval and matching radius in on-demand ride-sourcing markets. *Transportation Research Part B: Methodological*, 131:84–105.

Yoshida, N., Noda, I., and Sugawara, T. (2020). Multi-agent service area adaptation for ride-sharing using deep reinforcement learning. In *International Conference on Practical Applications of Agents and Multi-Agent Systems*, pages 363–375. Springer.

Yu, X., Gao, S., Hu, X., and Park, H. (2019). A markov decision process approach to vacant taxi routing with e-hailing. *Transportation Research Part B: Methodological*, 121:114–134.

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

Zhang, R., Rossi, F., and Pavone, M. (2016). Model predictive control of autonomous mobility-on-demand systems. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1382–1389. IEEE.

Zhang, R., Rossi, F., and Pavone, M. (2018). Analysis, control, and evaluation of mobility-on-demand systems: A queueing-theoretical approach. *IEEE Transactions on Control of Network Systems*, 6(1):115–126.