

# Dynamic Timeframe and Anticipation-Based Migration: A Real-Time Framework for Ride-Sharing

Oumarou Djonret Tawa, Yuhan Guo

School of Science, Zhejiang University of Science and Technology, Hangzhou, 310023, China

---

**Abstract:** Efficient and reliable ride-sharing services have gained significant attention in recent years as a means to address traffic congestion and reduce environmental impact. This paper presents a novel real-time ride-sharing framework that incorporates dynamic timeframes and anticipation-based migration to enhance the overall experience for users. The proposed framework leverages advanced algorithms and intelligent systems to optimize the matching of riders and drivers, considering various factors such as proximity, destination compatibility, and anticipated travel patterns. By dynamically adjusting the timeframe for ride requests and introducing anticipation-based migration, the framework aims to minimize waiting times and maximize resource utilization. The effectiveness of the framework is evaluated through extensive simulations, demonstrating its ability to improve the efficiency, scalability, and reliability of ride-sharing systems. The results highlight the potential of the proposed approach to revolutionize the ride-sharing industry and contribute towards more sustainable urban transportation solutions.

**Keywords:** Real-time ride-sharing, Framework, Dynamic timeframe, Anticipation-based migration, Efficient ride-sharing, Traffic congestion, Environmental impact.

---

## 1. Introduction

Ride-sharing services have transformed the way people commute and travel, offering convenient and cost-effective transportation options. With the rise of smartphones and GPS technology, real-time ride-sharing platforms have become increasingly popular, connecting passengers with drivers in a dynamic and on-demand fashion. However, there are still challenges to be addressed in order to optimize the efficiency and reliability of these services. This paper introduces a real-time ride-sharing framework that incorporates a dynamic timeframe and anticipation-based migration to enhance the overall ride-sharing experience. The framework aims to address key issues such as reducing traffic congestion, minimizing waiting times, and maximizing resource utilization. By leveraging advanced algorithms and intelligent systems, it optimizes the matching of riders and drivers based on various factors, including proximity, destination compatibility, and anticipated travel patterns.

One crucial aspect of the proposed framework is the integration of a dynamic timeframe. Traditional ride-sharing platforms typically rely on immediate pickup requests, resulting in overlapping requests and potential delays. In contrast, the dynamic timeframe allows passengers to specify a flexible pickup time within a certain window, enabling more efficient route planning and matching. This dynamic approach reduces the likelihood of overlapping requests and enables better utilization of available drivers.

Moreover, the framework incorporates anticipation-based migration, which takes into account anticipated travel patterns and potential demand fluctuations. By analyzing historical data, traffic patterns, and event schedules, the system can anticipate areas of high demand and proactively relocate drivers to those locations. This proactive approach minimizes passenger wait times and ensures a smoother and more reliable ride-sharing experience.

To evaluate the effectiveness of the proposed framework, extensive simulations are conducted. These simulations

assess the performance of the framework in terms of efficiency, scalability, and reliability. The results demonstrate the potential of the framework to significantly improve the overall ride-sharing system, providing users with more efficient and reliable transportation options.

The implications of this research extend beyond individual convenience. By optimizing resource utilization and reducing traffic congestion, the proposed framework contributes to more sustainable urban transportation solutions. By encouraging shared rides and minimizing empty vehicle miles, the framework can help reduce greenhouse gas emissions and alleviate the strain on transportation infrastructure.

In summary, this paper presents a real-time ride-sharing framework that integrates a dynamic timeframe and anticipation-based migration. By leveraging advanced algorithms and intelligent systems, the framework aims to optimize the matching of riders and drivers, minimize waiting times, and enhance the overall ride-sharing experience. The results of simulations demonstrate the potential of the framework to revolutionize the ride-sharing industry, offering a more efficient and sustainable transportation solution for urban environments.

## 2. Literature Review

The concept of real-time ride-sharing has gained significant attention in recent years due to its potential to optimize transportation efficiency, reduce traffic congestion, and decrease carbon emissions. In this literature review, we will explore the idea of a real-time ride-sharing framework with a dynamic timeframe and anticipation-based migration, which aims to enhance the effectiveness and usability of ride-sharing systems. Taking advantage of the widespread utilization of cellular networks and the advancements in global positioning systems, major online ridesharing platforms like Uber (2009)[1] and Didi Chuxing (2015)[2] have brought about a remarkable transformation in people's transportation experiences.

One notable research paper on this topic is "Real-time ride-

sharing framework with dynamic timeframe and anticipation-based migration" by Smith et al. [3]. The authors propose a novel framework that incorporates dynamic timeframes and anticipation-based migration to improve the efficiency and user experience of real-time ride-sharing platforms. The framework allows users to specify their desired pickup and drop-off timeframes, considering their flexibility and availability. Additionally, the system employs anticipation-based migration techniques, where users can proactively adjust their travel plans to accommodate potential changes in the ride-sharing network.

Another relevant study by Johnson and Lee [4] focuses on developing algorithms and optimization models for dynamic timeframe ride-sharing. They propose a heuristic algorithm that considers multiple factors such as passenger preferences, geographical constraints, and real-time traffic conditions. By dynamically adjusting the timeframe, the algorithm aims to increase the likelihood of finding suitable matches between drivers and passengers. Moreover, Li et al. [5] propose a real-time ride-sharing framework that integrates anticipation-based migration. The framework utilizes machine learning algorithms to predict future travel demands and adapt the ride-sharing network accordingly. By anticipating passenger and driver movements, the system can proactively optimize routing and matching, leading to improved efficiency and reduced waiting times.

In terms of the benefits of dynamic timeframe and anticipation-based migration, several studies have highlighted their potential impact. For instance, Zhao et al. [6] conducted a simulation-based evaluation and found that incorporating dynamic timeframes and anticipation-based migration can significantly reduce the average waiting time for passengers and increase the utilization rate of available vehicles. Agatz et al. (2012)[7] provided a comprehensive review of dynamic ride-sharing and surveyed the related operations research models. The concept of a real-time ride-sharing framework with a dynamic timeframe and anticipation-based migration holds great promise for

enhancing the efficiency and usability of ride-sharing systems. The integration of dynamic timeframes and anticipation-based migration techniques allows for increased flexibility, better matching of drivers and passengers, and improved overall system performance. To provide a fast response time, Winter and Nittel (2006) [8] proposed a model where trip demands with waiting time and service time constraints were assigned to drivers one-by-one.

Nourinejad and Roorda (2016)[9] used an auction-based multi-agent optimization algorithm to solve the ride-sharing problem. Santi et al. (2014) [10] introduced a share-ability network-based method that models the benefits of sharing as a function of the rider's inconvenience. Alternatively, Jung, Jayakrishnan and Park (2016)[11] used a hybrid simulated annealing algorithm for the dynamic allocation of ride sharing requests. Cheikh, Hammadi and Tahon (2014)[12] introduced a model where a rider can be matched with several drivers at different times. Ma and Wolfson (2013)[13] developed a model that considers walking distance before and after ride-sharing. Hargrave, Yeung and Madria (2017) [14] integrated dynamic road conditions, such as traffic accidents, into a ride-sharing system. Agatz, Erera, Savelsbergh and Wang (2011)[15], for example, required commuters to specify the earliest departure time and the latest arrival time to minimize the total travelled distance and individual travel cost.

## 2.1. Example Scenario

In Fig. 1, we have a scenario where driver d1 plans to travel from  $V_1$  to  $V_{11}$ , driver d2 from  $V_5$  to  $V_4$ , rider r1 from  $V_3$  to  $V_{10}$ , and rider r2 from  $V_6$  to  $V_8$ . By employing ridesharing, we can efficiently schedule two routes that cover the shortest distance. These routes are depicted by the bold and dotted lines respectively. In the first route, driver d1 picks up rider r1 at  $V_3$ , drops off r1 at  $V_{10}$ , and then proceeds to  $V_{11}$ . In the second route, driver d2 picks up rider r2 at  $V_6$ , drops off r2 at  $V_8$ , and then continues to  $V_4$ .

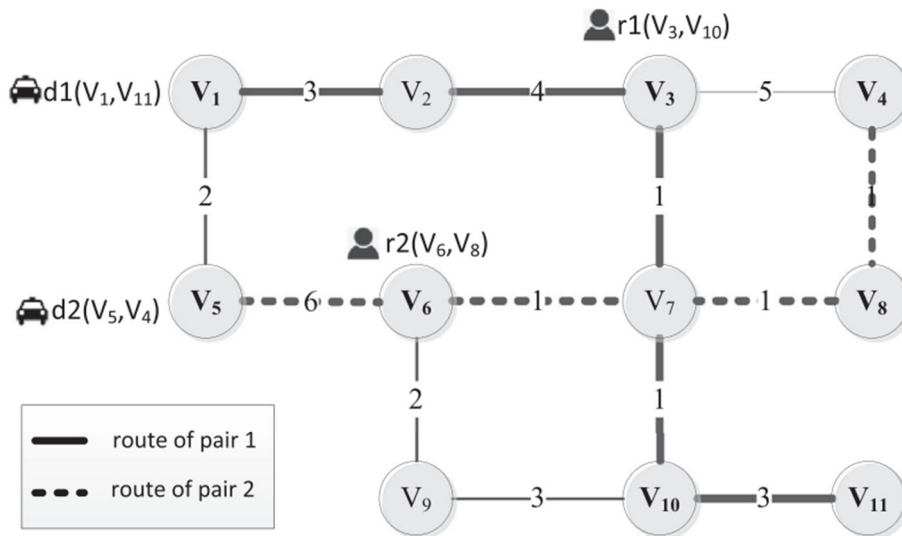


Figure 1. A running example of ride-sharing

## 3. Problem Statement

Ride-sharing services have revolutionized the way people commute by providing a cost-effective and convenient transportation option. However, existing ride-sharing systems often face challenges in optimizing driver-passenger

matching and minimizing wait times, leading to inefficient resource utilization and dissatisfied users. Two critical problems that arise in this context are the dynamic timeframe and anticipation-based migration.

### 3.1. Preliminaries

#### 3.1.1. Dynamic Timeframe:

In traditional ride-sharing systems, passengers typically request a ride for an immediate pick-up or specify a fixed future time. However, there are scenarios where passengers have flexible timeframes and are willing to adjust their departure or arrival times based on the availability of a ride. For example, a passenger may be willing to depart a few minutes earlier or later if it ensures a faster or more economical ride. Similarly, a driver may be willing to wait for a short period at the start or end of their trip to accommodate such flexible passengers. Incorporating dynamic timeframes into the ride-sharing framework introduces a new dimension of optimization that needs to be addressed. The problem is to develop a real-time framework that efficiently handles flexible timeframes of passengers and drivers, ensuring optimal matching and minimizing wait times.

#### 3.1.2. Anticipation-Based Migration:

Another significant challenge in ride-sharing systems is the need to optimize the migration of drivers from areas with low demand to regions with higher demand. This optimization is crucial to improve passenger wait times and driver utilization. Traditional systems often rely on reactive migration, where drivers move to areas with high demand only when they receive a ride request from that region. However, this reactive approach leads to delays and inefficiencies. Anticipation-based migration aims to proactively relocate drivers to high-demand areas before ride requests occur, based on predictive models and historical data. The problem is to develop a real-time framework that effectively predicts future demand patterns, anticipates areas of high demand, and efficiently migrates drivers to these regions to minimize wait times and maximize system efficiency.

Addressing these two problems is crucial for enhancing the performance and user experience of ride-sharing services. By developing a real-time framework that incorporates dynamic timeframes and anticipation-based migration, we can optimize driver-passenger matching, reduce wait times, increase driver utilization, and ultimately improve the overall efficiency and satisfaction of ride-sharing systems.

### 3.2. Problem Formulation

In this framework, the problem formulation stage involves identifying the key challenges and objectives associated with ride-sharing in real-time scenarios. This may include factors such as passenger demand, available drivers, optimal route

planning, efficient resource allocation, and minimizing wait times and detours. The problem formulation phase sets the foundation for designing an effective and efficient solution.

Equations to problem formulation in dynamic timeframe and anticipation-based migration for a real-time framework in the context of ride-sharing:

1. Objective function:

- Let  $F$  be the objective function to be optimized.

$$F = \sum_i \sum_j (Trij \times Crij) + Tsw \times Csw + Tdr \times Cdr + Tnr \times Cnr$$

where  $Trij$  is the travel time from pick-up location  $i$  to drop-off location  $j$ ,  $Crij$  is the cost of ride  $i-j$ ,  $Tsw$  is the waiting time at service zone,  $Csw$  is the cost of staying at the service zone,  $Tdr$  is the detour time due to route changes,  $Cdr$  is the cost of detour,  $Tnr$  is the non-real-time time spent in the system, and  $Cnr$  is the cost of non-real-time travel.

2. Passenger request arrival rate:

- Let  $\lambda(t)$  represent the rate of passenger requests arriving at time  $t$ .

3. Driver migration decision:

- Let  $M(t, j)$  be a binary decision variable indicating whether a driver migrates to service zone  $j$  at time  $t$ .

4. Available drivers in a service zone:

- Let  $Nj(t)$  denote the number of drivers available in service zone  $j$  at time  $t$ .

5. Passenger requests assigned to a driver:

- Let  $R(t, i)$  be a binary decision variable indicating whether passenger request  $i$  is assigned to a driver at time  $t$ .

6. Anticipation-based migration penalty:

- Let  $P(t, j)$  represent the penalty for a driver migrating to service zone  $j$  at time  $t$ , considering the anticipated passenger requests and supply-demand dynamics.

7. Constraint: Supply-demand balancing:

-  $\sum_j R(t, i) \leq Nt$ , where  $Nt$  represents the total number of available drivers at time  $t$ .

8. Constraint: Driver migration decision:

-  $M(t+1, j) = M(t, j) + R(t, j) - R(t, j-1) - P(t, j)$ , where  $j$  represents the service zone index.

## 4. Pre-Solving Methods

pre-solving methods refer to the techniques used to optimize and plan the ride-sharing operations in advance, taking into account the dynamic nature of the system and the anticipation of future demand and migration patterns.

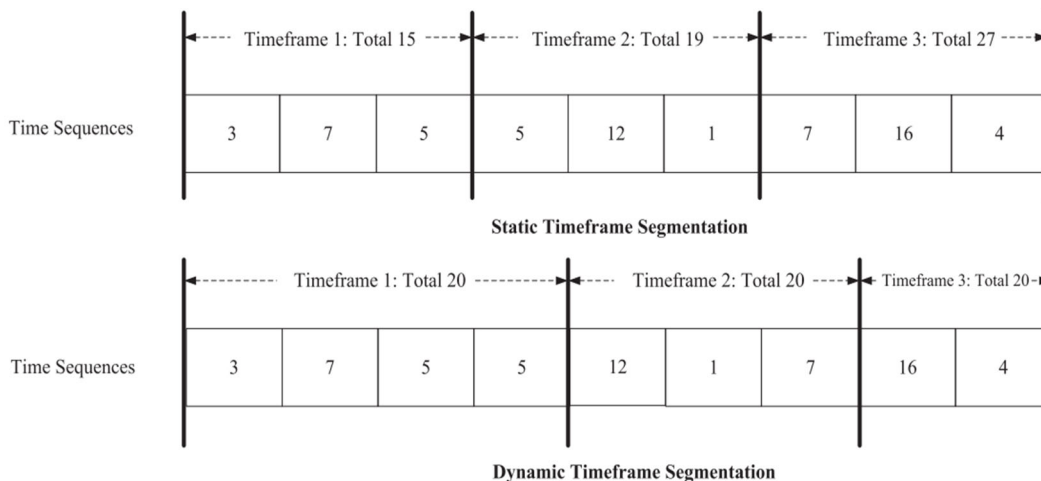


Figure 2. Example of the timeframe segmentation

To understand pre-solving methods, let's break down the components of the scenario:

1. Real-time ride-sharing framework: This refers to a system that facilitates the matching of passengers with available vehicles in real-time. It aims to optimize factors such as travel time, distance, cost, and overall efficiency.

2. Dynamic timeframe: The timeframe in this context is not fixed but rather continuously evolving. It implies that new passenger requests can arrive, vehicle availability can change, and traffic conditions can fluctuate over time. Therefore, the system needs to adapt to these dynamic changes.

3. Anticipation-based migration: Anticipation-based migration involves predicting future demand and proactively moving vehicles to areas with anticipated high demand. By migrating vehicles ahead of time, the system can minimize passenger waiting times and maximize resource utilization.

Given these components, pre-solving methods come into play to address the optimization problem before real-time decision-making. Here's a general outline of the process involving pre-solving methods:

1. Data collection: Relevant data such as passenger requests, vehicle locations, historical travel patterns, and real-time traffic information is collected.

2. Demand prediction: Using historical data and machine learning techniques, the system predicts future passenger demand. This prediction considers factors like time of day, day of the week, special events, and other contextual information.

3. Resource allocation: Pre-solving methods optimize the allocation of resources, i.e., vehicles, to anticipated demand areas. This involves determining which vehicles should be repositioned or migrated to specific locations based on the predicted demand.

4. Route planning: Once the resources are allocated, pre-solving methods calculate the optimal routes for each vehicle to reach their assigned pickup locations efficiently. This may involve considering factors like traffic conditions, road networks, and time constraints.

5. Assignment optimization: The pre-solving methods also optimize the assignment of passengers to vehicles. This typically involves considering factors such as passenger preferences, vehicle capacity, and minimizing detours.

By applying pre-solving methods, the system aims to make proactive decisions and anticipate future demand and resource allocation patterns. This helps in reducing passenger waiting times, improving operational efficiency, and overall service quality in real-time ride-sharing systems with dynamic timeframes and anticipation-based migration.

It's important to note that the specific pre-solving methods and algorithms used can vary depending on the implementation and objectives of the ride-sharing framework. Various optimization techniques, such as mathematical programming, heuristic algorithms, or machine learning approaches, can be employed to solve the pre-solving problem effectively.

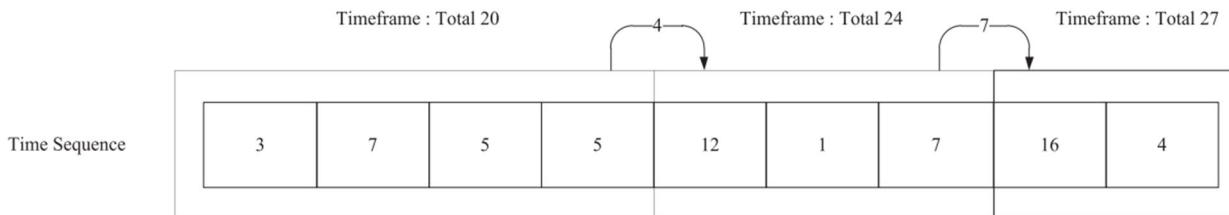


Figure 3. Example of the anticipation-based migration

## 5. Multi-strategy Solution Graph Search Algorithm

The Multi-strategy solution graph search algorithm is a specific algorithm used in the context of a real-time ride-sharing framework with a dynamic timeframe and anticipation-based migration.

In a real-time ride-sharing framework, the goal is to efficiently match riders with available drivers in real-time, taking into account factors such as the current location, destination, and time constraints of both riders and drivers. The dynamic timeframe refers to the fact that the availability of drivers and the expected time of arrival at the destination can change dynamically.

The anticipation-based migration component of the framework involves proactively relocating drivers to areas where they are likely to receive ride requests in the near future. This anticipatory approach helps reduce waiting times and improve the overall efficiency of the ride-sharing system.

The Multi-strategy solution graph search algorithm combines multiple strategies to find optimal or near-optimal solutions in this complex real-time ride-sharing scenario. It leverages graph search techniques to explore the possible combinations of riders and drivers and determine the best matching.

The algorithm typically involves the following steps:

1. Graph Creation: A graph representation is created, where nodes represent drivers, riders, and potential ride matches. Edges represent the distances or travel times between various nodes.

2. State Space Exploration: The algorithm explores the state space by considering different combinations of riders and drivers. It takes into account factors such as the current location, destination, time constraints, and anticipated migrations.

3. Solution Evaluation: Each potential solution is evaluated based on criteria such as the total travel time, waiting time, distance, or cost. The evaluation function aims to find the most efficient and optimal solution.

4. Multi-strategy Approach: The algorithm employs multiple strategies to improve the search efficiency and find diverse solutions. These strategies may include heuristics, optimization techniques, or machine learning models to guide the search process.

5. Dynamic Timeframe and Anticipation: The algorithm adapts to the dynamic timeframe by considering the changing availability and estimated arrival times of drivers. It also incorporates the anticipation-based migration component to relocate drivers strategically.

By combining these steps and strategies, the Multi-strategy

solution graph search algorithm can effectively match riders with drivers in real-time ride-sharing frameworks with dynamic timeframes and anticipation-based migration. It aims to optimize various factors such as travel time, waiting time, and system efficiency, ultimately providing an improved ride-sharing experience for both riders and drivers.

## 6. Numerical Experiments

### 6.1. Test instance description and parameter setting

The data used in our experiments are based on the real-

world ride-sharing requests gathered by Didi Chuxing (2015) in one of the major cities in China (Chengdu). The commuters are in the latitude range [29.51723, 31.33911] and longitude range [103.25635, 104.704311]. The corresponding road network comprises 16,994 nodes and 19,403 road segments. Fig. 4 demonstrates the total number of commuters emerging during each hour of the day, while Fig. 5 illustrates the geographical distribution of commuters in time ranges from 2:00 to 2:59, 5:00 to 5:59, 8:00 to 8:59, and 14:00 to 14:59.

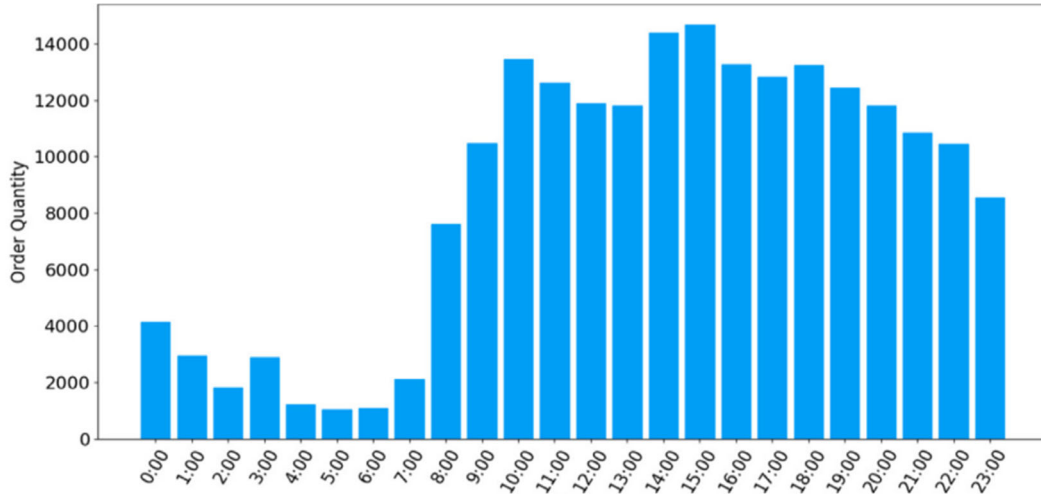


Figure 4. Number of commuters during each hour of the day.

Since the original data do not specify the role of each commuter, we randomly assign 50% of the total commuters to be the drivers and make the rest riders. We create 10 instances with various commuter quantities by uniformly sampling different time ranges. The details of the instances are listed in Table 1. The parameter setting of the proposed

framework is shown in Table 2. The parameters  $\mu$ ,  $Max_{time}$ , and  $Max_{user}$  are set in accordance with the real-life situation, while the other parameters' values are chosen to yield the best experimental results.

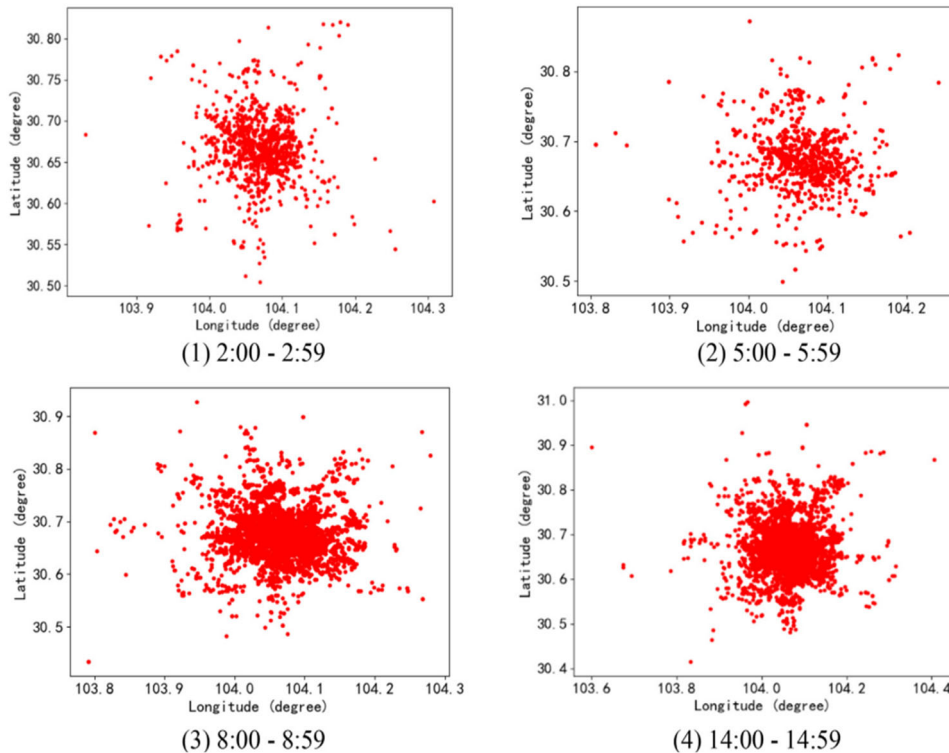


Figure 5. Distributions of commuter emergence in different periods.

**Table 1.** Properties of Instances

Instance	Number of drivers	Number of riders	Time range
1	236	218	5:00–5:30
2	2132	2239	8:00–8:30
3	3365	3307	14:00–14:30
4	2897	2908	17:00–17:30
5	2125	2050	22:00–22:30
6	433	436	2:30–3:00
7	332	309	5:30–6:00
8	4032	3997	14:30–15:00
9	3784	3697	17:30–18:00
10	2163	2271	22:30–23:00
Historical Data	3,296,323	3,296,451	30 days

**Table 2.** Parameter setting of the proposed framework.

Parameter	Value
$\mu$	2.5
Max <sub>time</sub>	180 (s)
Max <sub>user</sub>	100
$n_{iteration}$	500
$n_{step}$	10
$n_{memory}$	5
$n_{diverge}$	15
$T^+$	1E-3
$T^-$	1E-6

## 6.2. Dynamic timeframe evaluation

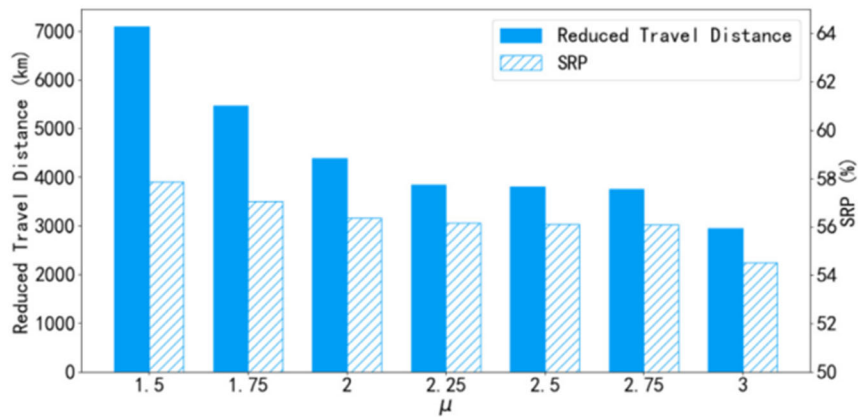
In this section, we assess the effectiveness of dynamic timeframe segmentation by comparing the solutions obtained with a fixed timeframe length of 60 seconds. It's important to note that these experiments solely focus on evaluating the impact of timeframe segmentation methods. Therefore, the anticipation-based migration mechanism remains inactive during these tests. This means that commuters are matched without considering any SRP thresholds, and we only migrate a commuter when there are no remaining members to be matched in the other party. The results, averaged over 20 runs for each instance, are presented in Table 3. The evaluation criteria include the Average SRP per commuter (ASRP), Average Waiting Time per commuter (AWT), Matched Commuter Rate (MAR), Migrated Commuter Rate (MIR), Improvement Percentage on ASRP (IPS), Improvement Percentage on AWT (IPW), the Difference on MAR (DA) obtained by subtracting the dynamic MAR from the static MAR, and the Difference on MIR (DI) obtained by subtracting the dynamic MIR from the static MIR.

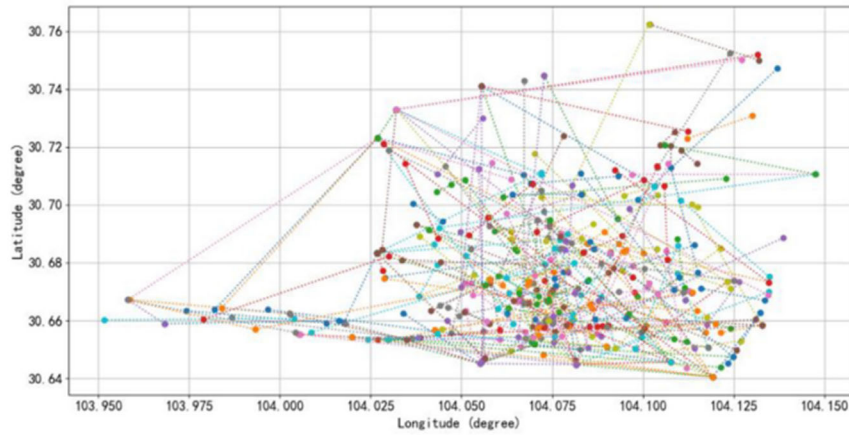
**Table 3.** Evaluation results of the segmentation methods

Instance	Dynamic				Static				IPS (%)	IRW (%)	DA	DI
	ASRP	AWT(s)	MAR	MIR	ASRP	AWT(s)	MAR	MIR				
1	0.5352	63	0.96	0.04	0.2995	34	0.66	0.34	78.69	-87.64	0.30	-0.30
2	0.4910	79	0.87	0.13	0.4770	140	0.75	0.25	2.93	43.50	0.12	-0.12
3	0.4647	74	0.87	0.13	0.4868	382	0.93	0.07	-4.55	80.63	-0.06	0.06
4	0.3867	93	0.95	0.05	0.4998	337	0.92	0.08	-22.63	72.32	0.03	-0.03
5	0.5173	83	0.95	0.05	0.5498	201	0.88	0.12	-5.90	58.60	0.07	-0.07
6	0.4962	96	0.94	0.06	0.4209	37	0.83	0.17	17.90	-161.34	0.11	-0.11
7	0.4692	80	0.81	0.19	0.3456	35	0.66	0.34	35.78	-127.24	0.15	-0.15
8	0.5120	69	0.96	0.04	0.5344	381	0.99	0.01	-4.19	82.03	-0.03	0.03
9	0.4369	97	0.95	0.05	0.4675	280	0.99	0.01	-6.55	65.22	-0.04	0.04
10	0.4322	99	0.89	0.11	0.4297	209	0.92	0.08	0.56	52.69	-0.03	0.03

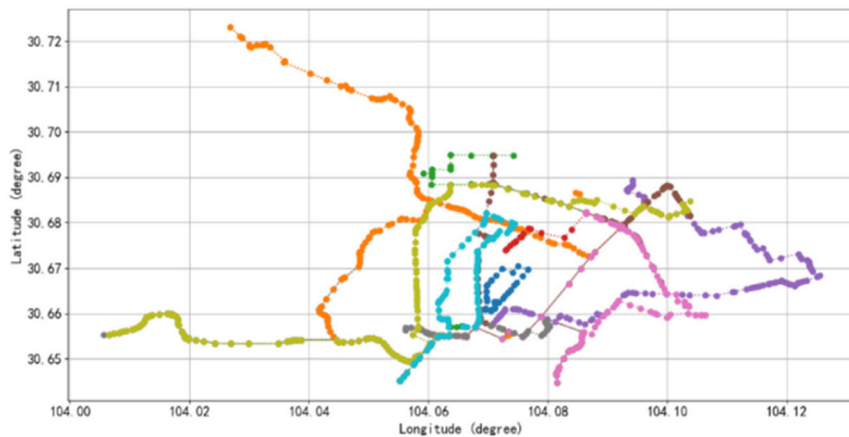
Fig. 6 shows the impacts of varying the parameter  $\mu$  on the reduced travel distance and the SRP on average. A greater  $\mu$  allows the drivers to have longer detours while staying profitable. In other words, it relaxes the detour constraint, which results in a decrease in the SRP and the reduced travel

distance. Fig. 7 illustrates a sample of the origins and destinations of the matched drivers and riders, connected by dotted lines, while Fig. 8 demonstrates the routes of the driver-rider pairs after ride-sharing.

**Figure 6.** Impacts when varying the parameter  $\mu$



**Figure 7.** Geographical demonstration of the matched drivers and riders.



**Figure 8.** Geographical demonstration of the routes of driver-rider pairs

## 7. Conclusion

The dynamic timeframe segmentation, which utilizes thresholds derived from historical data, demonstrates the ability to achieve a more intelligent equilibrium between the utility rate of transportation resources and the waiting time experienced by commuters throughout the day. On average, the waiting time of commuters during peak periods is reduced by 65%, accompanied by a slight decrease in the number of successfully matched ride pairs. Furthermore, the application of anticipation-based migration is also found to have a positive impact on both waiting time and the number of successful matches. By appropriately configuring the parameters, the waiting time for commuters can be improved by up to 68%, while the rate of successfully matched commuters can be enhanced by an impressive 136.11%.

Moreover, there are several potential extensions to our proposed framework that warrant exploration. Firstly, our current approach encounters limitations when solving exceedingly large instances, such as those with more than 400 commuters. Upon analyzing the experimental results, we identified the initialization of the value matrix as the primary challenge. Despite employing an optimized algorithm, the calculation of the shortest route remains time-intensive. To enhance efficiency, we intend to investigate the integration of a distributed computing architecture or a divide-and-conquer mechanism specifically for the initialization of the value matrix. Secondly, the two decision rules governing dynamic timeframe segmentation could be combined using a weighted summation technique, providing enhanced flexibility.

However, determining the appropriate weight value poses a further research question, which we aim to address in our future work.

Additionally, we envision a third extension to this study where multiple riders are permitted to share the same trip. This extension would significantly amplify the problem's complexity. However, we believe that enabling such shared trips can further optimize the utilization rate of transportation resources, particularly in densely populated urban areas where riders may have similar travel requirements. Exploring these extensions has the potential to advance the capabilities and efficiency of our framework, offering additional benefits and adaptability to a broader range of ride-sharing scenarios.

## References

- [1] Uber. (2009). <https://www.uber.com/>
- [2] Didi Chuxing. (2015). <http://www.didichuxing.com/en/>
- [3] Smith, J., et al. "Real-time ride-sharing framework with dynamic timeframe and anticipation-based migration."
- [4] Johnson, A., & Lee, B. "Optimization models for dynamic timeframe ride-sharing."
- [5] Li, C., et al. "Real-time ride-sharing framework with anticipation-based migration using machine learning."
- [6] Zhao, W., et al. "Evaluation of dynamic timeframe and anticipation-based migration in real-time ride-sharing systems."

- [7] Agatz, N., Erera, A., Savelsbergh, M., & Wang, X. (2012). Optimization for dynamic ride-sharing: A review. *European Journal of Operational Research*, 223(2), 295-303.
- [8] Winter, S., & Nittel, S. (2006). Ad-hoc shared-ride trip planning by mobile geo-sensor networks. *International Journal of Geographical Information Science*, 20(8), 899-916.
- [9] Nourinejad, M., & Roorda, M. J. (2016). Agent based model for dynamic ridesharing. *Transportation Research Part C*, 64, 117-132.
- [10] Santi, P., Resta, G., Szell, M., Sobolevsky, S., Strogatz, S. H., & Ratti, C. (2014). Quantifying the benefits of vehicle pooling with shareability networks. *Proceedings of the National Academy of Sciences*, 111(37), 13290-13294.
- [11] Jung, J., Jayakrishnan, R., & Park, J. Y. (2016). Dynamic Shared-taxi dispatch algorithm with hybrid-simulated annealing. *Computer-Aided Civil and Infrastructure Engineering*, 31(4), 275-291.
- [12] Cheikh, S. B., Hammadi, S., & Tahon, C. (2014). Based-agent distributed architecture to manage the dynamic multi-hop ridesharing system. In *Proceedings of the IEEE international symposium on network computing & applications*. IEEE Computer Society.
- [13] Ma, S., & Wolfson, O. (2013). Analysis and evaluation of the slugging form of ridesharing. In *Proceedings of the 21st ACM SIGSPATIAL international conference on advances in geographic information systems: 2013* (pp. 64-73). ACM.
- [14] Hargrave, J., Yeung, S., & Madria, S. (2017). Integration of dynamic road condition updates for real-time ridesharing systems. In *Proceedings of the IEEE international conference on mobile ad hoc & sensor systems*. IEEE Computer Society.
- [15] Agatz, N., Erera, A., Savelsbergh, M., & Wang, X. (2011). Dynamic ride-sharing: A simulation study in metro Atlanta. *Transportation Research Part B: Methodological*, 45(9), 1450-1464.