Non-myopic Matching and Rebalancing in Large-Scale On-Demand Ride-Pooling Systems Using Simulation-Informed Reinforcement Learning

Farnoosh Namdarpour, Joseph Y. J. Chow*

C2SMARTER University Transportation Center, New York University Tandon School of Engineering, Brooklyn, NY, USA

farnoosh@nyu.edu, joseph.chow@nyu.edu

* Corresponding author

Extended abstract submitted for presentation at the 12th Triennial Symposium on Transportation Analysis conference (TRISTAN XII) June 22-27, 2025, Island of Okinawa

March 25, 2025

Keywords: ride-pooling, ride-sharing, reinforcement learning, fleet dispatch, vehicle rebalancing.

1 INTRODUCTION

Ride-pooling or ride-sharing refers to a system where passengers with different ride requests share one vehicle, offering greater efficiency for service providers and more affordable rides for customers compared to ride-hailing where only one passenger is onboard the vehicle with the driver. Ride-pooling can help reduce traffic congestion and environmental impacts. However, matching vehicles and riders efficiently and rebalancing the idle vehicles remains a challenging task. Operators value higher ridership and reduced vehicle distance traveled, while riders prioritize shorter waiting and in-vehicle times. Additionally, in a ride-pooling system, considering passengers already onboard the vehicle adds another layer of complexity when finding efficient matches for future requests. The system's efficiency depends on an optimization algorithm capable of balancing the priorities of both riders and operators.

In a dynamic ride-pooling system, ride requests are submitted over time while the system attempts to find a vehicle for each request once submitted. Similarly, it attempts to reposition vehicles at different time steps. The vehicle-request matching problem and vehicle rebalancing problem are both sequential decision problems. Decisions made at a certain time can impact the system in the future highlighting the importance of considering the long-term impact of decisions. Among the non-myopic optimization methods for sequential decision making, reinforcement learning (RL) has shown promising results.

While RL has been widely applied to ride-hailing systems, it is less explored in ride-pooling systems due to the more complex nature of the problem. Most existing RL-based papers on ride-pooling treat each vehicle as an individual agent, applying the trained model independently to each one. However, Didi's statistics have shown that centralized fleet dispatch, where the platform assigns vehicles to riders instead of vehicles being the decision-makers, can significantly improve system efficiency (Xu *et al.*, 2018).

While not directly related to ride-pooling, applications of physics-informed RL have gained traction in recent years in which RL is combined with model-based approaches to leverage the structure of the problem captured by the model. In this study, we build upon the work of Xu *et al.* (2018), which uses RL for matching in ride-hailing systems, but extend it to ride-pooling systems and add vehicle rebalancing operations using simulation models to help inform on the structure of the decision for the learning mechanism, i.e. simulation-informed. This is one of

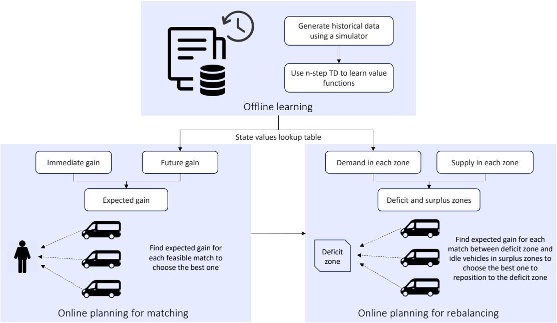
Generate historical data using a simulator
Use n-step TD to learn value functions
Offline learning
State values lookup table
Immediate gain
Future gain
Expected gain
Find expected gain for each feasible match to choose the best one
Online planning for matching
Demand in each zone
Supply in each zone
Deficit and surplus zones
Deficit zone
Find expected gain for each match between deficit zone and idle vehicles in surplus zones to choose the best one to reposition to the deficit zone
Online planning for rebalancing

Figure 1 – *Proposed framework.*

the first studies to propose a non-myopic RL approach for real-time matching and rebalancing in large-scale ride-pooling systems with a central dispatch unit. We propose an offline approach to learn the spatiotemporal patterns of supply and demand from episodes of experience generated in a simulated ride-pooling environment with historical demand data and propose online non-myopic policies to use the learned value functions from this simulation-informed process for making real-time matching decisions and rebalancing the idle vehicles.

2 METHODOLOGY

Vehicle dispatching in a ride-pooling system is a sequential decision problem that can be modeled as a Markov Decision Process (MDP). The main components of the MDP are defined below.

Agent: Although from a global perspective, there is only one centralized agent making dispatch decisions, the system is defined from a local perspective to simplify the model components, treating each vehicle as an agent, while individual vehicles are not differentiated.

State: The service operating time is divided into several time periods, e.g. five-minute intervals, and the service area is divided into multiple zones. The state of each agent is defined by the spatiotemporal status of the vehicle.

Action: For the matching problem, each agent has two defined actions: serving a new request or continuing its previous schedule without changes. Similarly, for the rebalancing problem, each idle agent has two actions: repositioning to another zone or remaining idle.

Reward: An agent's reward at time index t is defined as the number of new requests that have been assigned to the agent in that time index. By this definition, vehicles learn to be in the zones at times they are needed. This definition reflects the optimization goal of the system, which is maximizing the number of served requests while minimizing passengers' travel times.

The proposed framework is a learning and planning approach with three main components shown in Figure 1: offline learning, online planning for matching, and online planning for rebalancing, which are explained in the following subsections.

2.1 Offline simulation-informed learning

A sample-based approach is employed for offline learning, where historical demand data is fed into a ride-pooling simulator with a fleet size hyperparameter to generate simulated experiences, i.e. "simulation-informed". The simulator fleet size is set to a large value to ensure no request rejections occur in the system, thereby capturing all spatiotemporal demand-supply patterns. This can only be effectively achieved in a simulated environment, where control over variables allows for a thorough exploration of different possible scenarios. The simulator follows a fixed policy, e.g. a myopic policy, for assigning vehicles to passengers. The n-step temporal difference (TD) method, which combines the TD and Monte Carlo methods by considering the observed rewards of n steps ahead, is used as the model-free method for policy evaluation to learn value functions from simulated-informed samples. The discounted return $G^v_{t:t+n}$ using n-step TD for vehicle v at state S_t is found in Eq. 1.

$$G^v_{t:t+n} = r^v_{t+1} + \gamma r^v_{t+2} + ... + \gamma^{n-1} r^v_{t+n} + \gamma^n V(S_{t+n}) \tag{1}$$

where n is the number of future steps considered, r^v_{t+i} is the immediate reward at time $t+i$ for vehicle v, γ is the discount factor, and $V(S_{t+n})$ is the value of state S_{t+n}.

2.2 Online planning for matching

The online planning step for matching uses the learned value functions in the offline learning step to make real-time matching decisions. For each submitted request, the simulator finds a set of vehicles that meet the feasibility criteria (e.g. vehicle capacity, maximum wait time for passengers) to serve the request. The central dispatch unit uses the online planning algorithm to find the best vehicle among the feasible ones. The system's objective is to optimize total immediate and future gain, which is formulated as the objective function in Eq. 2 for each matching decision.

$$arg \max_v \left(R_v + \gamma^{\Delta t_{S'_v}} V(S'_v) - \gamma^{\Delta t_{S_v}} V(S_v) \right) \tag{2}$$

The value in the parentheses represents the expected gain for serving the request by vehicle v. The expected gain contains an immediate gain component (R_v) and a future gain component which is represented by the discounted difference of vehicle state values before and after assigning the new request represented by $V(S_v)$ and $V(S'_v)$, respectively. At any given time in the system, the vehicle can have multiple stops on its scheduled route. We define the vehicle's state by its final scheduled stop, i.e. the last drop-off location and time. The immediate gain component (R_v) is found using Eq. 3.

$$R_v = \lambda \left(c(v,\xi) - c(v,\xi') \right) \tag{3}$$

where $c(v,\xi)$ is the current cost for (vehicle v, route ξ) and $c(v,\xi')$ is the cost after assigning the new request to vehicle v. Parameter λ is used to scale the cost values to match the magnitude of the state values.

2.3 Online planning for rebalancing

The learned value functions in the offline learning step are used to represent the relative demand in each zone by dividing the zone's state value at a given time by the sum of all zones' state values at that time. The relative supply in each zone at each time step is defined as the number of vehicles in that zone divided by the total number of vehicles in the system. The difference between the relative supply and demand is used to identify surplus and deficit zones. At each time interval, the best idle vehicle in a surplus zone is found to reposition to a deficit zone, following the same matching policy outlined in the previous section.

3 COMPUTATIONAL EXPERIMENTS

We modified the NOMAD-RPS simulator (Namdarpour *et al.*, 2024) to implement our proposed method in a simulation setting using the NYC taxi data (Taxi & Commission, 2024). To evaluate the matching performance, the results were compared with two other matching algorithms across various fleet sizes, without rebalancing idle vehicles. Subsequently, the rebalancing operation was assessed against an alternative method. The algorithms tested are described below.

- **Myopic**: The myopic policy without rebalancing explained in Namdarpour *et al.* (2024) used as the baseline.

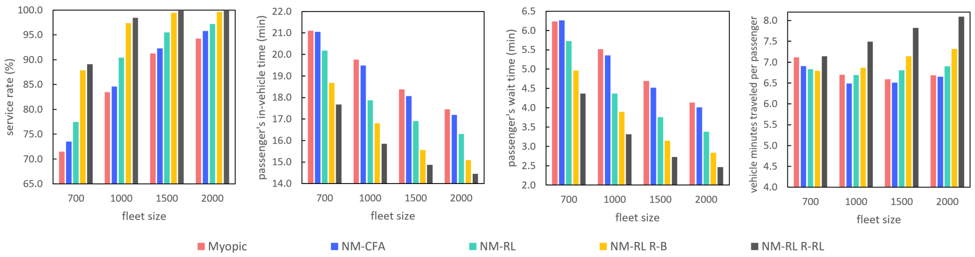

Figure 2 – *Comparison of test results for different policies across different fleet sizes.*

- **NM-CFA**: The CFA policy without rebalancing proposed by Hyytiä *et al.* (2012) which has a tunable parameter for lookahead approximation.
- **NM-RL**: The proposed learning and planning framework in this study using RL to optimize real-time matching decisions over a long-term horizon without rebalancing.
- **NM-RL R-B**: The basic rebalancing operation proposed by Alonso-Mora *et al.* (2017) added to the NM-RL matching.
- **NM-RL R-RL** The RL-based rebalancing operation proposed in this study added to the NM-RL matching.

The results in Figure 2 evaluate two measures from passengers' perspective: in-vehicle time and wait time, and two from the operator's perspective: service rate (percentage of accepted requests), and vehicle minutes traveled per passenger (VMT). Comparing the three matching policies without rebalancing shows that NM-RL achieves the best results, increasing service rate by up to 8.4% compared to a myopic policy and reducing passenger wait time and in-vehicle time with a slight increase in VMT. The two vehicle-rebalancing policies further improve service rate while the RL approach provides a more substantial reduction in passengers' in-vehicle (up to 12% reduction compared to NM-RL) and wait times (up to 27% reduction). However, this improvement comes at the cost of up to 15% increase in VMT. This trade-off can be adjusted by extending the rebalancing interval in the proposed approach.

Both matching and rebalancing results demonstrate that the proposed non-myopic approach effectively captures long-term consequences of dispatch decisions, improving service from both operators' and passengers' perspective. Further analysis of the results, including visualization of learned state values for different hours of day and the impact of training set size on the algorithm performance will be presented at the conference.

References

Alonso-Mora, Javier, Samaranayake, Samitha, Wallar, Alex, Frazzoli, Emilio, & Rus, Daniela. 2017. On-demand high-capacity ride-sharing via dynamic trip-vehicle assignment. *Proceedings of the National Academy of Sciences*, **114**(3), 462–467.

Hyytiä, Esa, Penttinen, Aleksi, & Sulonen, Reijo. 2012. Non-myopic vehicle and route selection in dynamic DARP with travel time and workload objectives. *Computers & Operations Research*, **39**(12), 3021–3030.

Namdarpour, Farnoosh, Liu, Bingqing, Kuehnel, Nico, Zwick, Felix, & Chow, Joseph YJ. 2024. On non-myopic internal transfers in large-scale ride-pooling systems. *Transportation Research Part C: Emerging Technologies*, **162**, 104597.

Taxi, NYC, & Commission, Limousine. 2024. *TLC Trip Record Data*. `https://www.nyc.gov/site/tlc/about/tlc-trip-record-data.page` [Accessed on June, 2024].

Xu, Zhe, Li, Zhixin, Guan, Qingwen, Zhang, Dingshui, Li, Qiang, Nan, Junxiao, Liu, Chunyang, Bian, Wei, & Ye, Jieping. 2018. Large-scale order dispatch in on-demand ride-hailing platforms: A learning and planning approach. *Pages 905–913 of: Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining.*