Learning-Based Heterogeneous Autonomous Vehicles Scheduling for On-Demand Last-Mile Transportation

Yongwu Liu, Binglei Xie, Yuying Long, Jiawei Chen, and Gangyan Xu, *Member, IEEE*

***Abstract*—Last-mile transportation, a critical component of public transit within integrated urban mobility systems, plays a pivotal role in promoting sustainability and requires immediate attention. However, due to the high real-time variability and uncertainties of last-mile travel demand, high passenger concurrency, and dispersed destinations, existing last-mile transportation systems often encounter significant challenges. These challenges include resource shortages and congestion during peak hours, as well as high operational costs and excessive passenger waiting times during off-peak periods. With the rapid development and widespread adoption of autonomous vehicles, which are characterized by centralized control and flexible scheduling, this study proposes leveraging heterogeneous autonomous vehicles to address these challenges. Specifically, a mixed-integer programming model is developed to maximize the service provider's profit by considering fare profit, passenger waiting time penalties, and operating costs. To enable real-time decision-making, then an attention-based deep reinforcement learning algorithm is introduced. This algorithm incorporates two decoder mechanisms for vehicle selection and passenger allocation in the scheduling of heterogeneous autonomous vehicles. This involves dynamically selecting vehicles from a heterogeneous fleet based on passenger demand using an attention mechanism, optimizing efficiency in serving last-mile travelers. Extensive numerical experiments and a real-world case study across various datasets demonstrate that the proposed service model and algorithm effectively solve the scheduling problem while meeting the demands of on-demand last-mile transportation. Furthermore, these innovations contribute to reducing fleet carbon emissions and advancing sustainable urban transportation.**

Received 28 February 2025; revised 23 July 2025; accepted 14 September 2025. Date of publication 18 September 2025; date of current version 24 September 2025. This article was recommended for publication by Associate Editor J. Liu and Editor J. Li upon evaluation of the reviewers' comments. This work was supported in part by the National Natural Science Foundation of China under Grant 72174042 and Grant 71974043, in part by the Natural Science Foundation of Guangdong Province under Grant 2023A1515011402, and in part by the Natural Science Foundation of Shenzhen Municipality under Grant JCYJ20230807140406013. *(Corresponding authors: Binglei Xie; Gangyan Xu.)*

Yongwu Liu is with the School of Architecture, Harbin Institute of Technology, Shenzhen 518055, China, and also with the Department of Aeronautical and Aviation Engineering, The Hong Kong Polytechnic University, Hong Kong (e-mail: yongwu.liu@connect.polyu.hk).

Binglei Xie is with the School of Architecture, Harbin Institute of Technology, Shenzhen 518055, China (e-mail: xiebingleihit@163.com).

Yuying Long and Jiawei Chen are with the Department of Aeronautical and Aviation Engineering, The Hong Kong Polytechnic University, Hong Kong (e-mail: yuying.long@connect.polyu.hk; superlaser-jw.chen@connect.polyu.hk).

Gangyan Xu is with the Department of Aeronautical and Aviation Engineering, The Hong Kong Polytechnic University, Hong Kong, and also with Hong Kong Polytechnic University Shenzhen Research Institute, Shenzhen 518057, China (e-mail: gagexgy@gmail.com).

Digital Object Identifier 10.1109/TASE.2025.3611676

***Note to Practitioners*—Last-mile transportation systems face critical challenges: overcrowded vehicles during peak hours, high costs due to underused resources in off-peak periods, and passenger dissatisfaction from long wait times. These issues hinder sustainable urban mobility and strain transit operators' budgets. This work addresses these problems by deploying heterogeneous autonomous vehicle fleets with varying sizes and capabilities for last-mile transportation. The proposed system dynamically assigns vehicles to passengers based on real-time demand, optimizing efficiency while reducing operational costs. For transit operators, this approach balances fare revenue with penalties for passenger delays, ensuring profitability even during fluctuating demand. Cities adopting shared autonomous vehicle services can benefit from reduced congestion and lower carbon emissions, aligning with sustainability goals. Our experiments validate that the method improves service reliability and fleet utilization across demand scenarios. Implementing this solution requires integrating demand prediction tools with a centralized autonomous vehicle dispatch platform, compatible with existing mobility apps. Future extensions could adapt the system to mixed fleets (combining autonomous vehicles with traditional vehicles) or expand to multi-city networks. By addressing both economic and environmental needs, this framework offers a scalable path toward smarter, greener urban transportation.**

***Index Terms*—Last-mile transportation, deep reinforcement learning, heterogeneous autonomous vehicles scheduling, on-demand service.**

I. Introduction

LAST-MILE transportation system plays a crucial role and encompasses the transportation services and designs that allow passengers to reach their final destinations from public transportation hubs. It represents a challenging segment of Mobility as a Service (MaaS), which aims to provide seamless connectivity between various modes of transportation [1]. The last-mile system can attract more people to use public transportation, promoting sustainable and integrated transportation, while also serving special populations such as the elderly, the sick, the disabled, and schoolchildren, especially as global trends like population aging become more pronounced [2]. To address the last-mile transportation problem, many efforts have been made on different modes, including building convenient pedestrian pathways, promoting bike and e-bike sharing, taxi and ride-sharing, and providing shuttle buses. While each mode has its advantages, it also presents certain limitations. For instance, walking is inefficient when time is critical. Bike and e-bike sharing depend on the availability of bike lanes and may be shortages during peak hours. Taxi and ride-sharing services can be relatively costly. Similarly,

1558-3783 © 2025 IEEE. All rights reserved, including rights for text and data mining, and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: Hong Kong Polytechnic University. Downloaded on February 04,2026 at 10:07:57 UTC from IEEE Xplore. Restrictions apply.

shuttle buses often lack the flexibility required for door-to-door transportation [3].

The growing limitations of traditional last-mile transportation modes highlight an urgent need to explore innovative technologies and solutions for building sustainable and efficient systems. Autonomous vehicles (AV), with their distinct advantages, present a promising breakthrough in addressing these challenges [4], [5], [6]. First, unlike human-driven vehicles, AV are dispatched by a control center, which is not constrained by driver workload [7], [8]. Second, in real-time scheduling of AV, a key advantage of AV is their capacity to safely and swiftly accept and implement plan changes [9], [10]. Third, in fully autonomous mode, the AV can work around the clock without breaks except for the time spent on charging [11]. Service providers can receive real-time information on the entire fleet's status and dynamically adjust their routes and plans, without concerns about drivers' rest, salary, or emotional state [12], [13]. Fourth, in on-demand transportation, flexible route planning and frequent passenger pickups and drop-offs may run out of the drivers' familiar areas and even challenge their patience [14]. Furthermore, the requirement for passengers to signal the driver or press a button to get off can be difficult for individuals with social anxiety, as seen with the minibus in Hong Kong [15]. However, when using an AV, passengers can pre-book the drop-off point in advance, and they can also use their phones to remind the AV to stop in real-time. Motivated by these advantages, this work proposes to use Heterogeneous Autonomous Vehicles (HAV) for on-demand last-mile transportation.

Meanwhile, many companies and researchers have already proposed or realized autonomous driving applications in reality. For example, Tesla initiated its Robotaxi project in 2024, leveraging recent advancements in Autonomous vehicles and featuring two electric HAV with different capacities: the Cybercab and the Robovan. Meanwhile, Apollo Go has commenced full autonomous driving services and tested in Beijing, Wuhan, Chongqing, Shenzhen, and Shanghai, China, to facilitate more convenient mobility services and sustainable transportation. AV-based transportation has emerged in various fields, such as urban mobility and robotaxis [16], [17], logistics and freight transportation [18], [19], agriculture and mining [20], and emergency material transportation [21], [22], demonstrating its wide and efficient application. Meanwhile, numerous studies have focused on improving transportation efficiency and quality, particularly in scheduling and route planning. These studies aim to minimize operational costs [7], [23], maximize stable dispatching policies [24], and integrate shared AV with public transport [25], [26].

However, fully harnessing these benefits and achieving widespread practical adoption of AV in last-mile transportation requires overcoming several complex real-world challenges that are crucial for the research background. These challenges encompass not only technological hurdles (e.g., diverse urban environments, unpredictable events) [27], [28] and the development of robust regulatory frameworks [29], [30], but also the optimization of operational strategies to handle dynamic demands and diverse vehicle capabilities [31], [32], [33]. Furthermore, the current readiness for widespread adoption also hinges on factors like achieving broad public acceptance and trust [34], [35], as well as the substantial need for supporting infrastructure (e.g., charging stations, parking areas) [36], [37]. Our study addresses a crucial aspect within this broader landscape: the HAV scheduling for last-mile transportation, which still presents several critical issues that must be effectively resolved to advance AV deployment.

Firstly, last-mile transportation exhibits significant tidal effects, with distinct peak and off-peak periods. During peak hours, vehicle resources are often insufficient to meet the high volume of real-time demands, while during off-peak hours, vehicles tend to remain idle, leading to resource waste, prolonged waiting times, and high operational costs. This imbalance in demand poses a major challenge for transportation systems, requiring real-time decision-making to balance vehicle utilization and operational costs while ensuring service quality. Additionally, the randomness of demand and the uncertainty of destinations further complicate the issue, necessitating dynamic adjustments to vehicle dispatching strategies, such as deciding whether to wait for new orders or how to optimize vehicle routing based on current demands.

Secondly, passengers can book on-demand last-mile services via smartphones in advance or request the service upon arrival at the station. Such co-existence of both request modes poses a great challenge to last-mile transportation service design. In practice, a common approach is to sequentially board passengers into one vehicle. As the minibus in Hong Kong, once the minibus is full, it departs, and the remaining passengers will board the next minibus. However, such a method may lead to long in-vehicle waiting times during off-peak hours and long queues in peak hours. Besides, since it does not consider the destination distribution of passengers, the route may not be efficient enough. Thus, a new service mode is needed to further improve efficiency and decrease the operation cost and energy consumption.

Thirdly, considering the real-time and uncertain environments, joint decisions, and HAV involved, it is challenging to realize real-time scheduling of last-mile transportation services. When a large batch of passengers arrives simultaneously at different destinations, efficient vehicle scheduling becomes difficult. Decisions should be made efficiently to dynamically select suitable vehicles based on actual passenger demand, allocate corresponding orders, and simultaneously plan routes for different vehicles. Here, the Capacitated Vehicle Routing Problem (CVRP) is inherently complex [38], and existing research on last-mile transportation primarily focused on using homogeneous vehicles. Efforts are still needed for real-time CVRP with heterogeneous vehicles.

To address the above issues, this work aims to improve the efficiency of on-demand last-mile transportation by integrating a new service mode with a deep reinforcement learning (DRL) based HAV scheduling method. The main contributions of this work lie in the following four aspects:

- A dynamic passenger allocation mode with HAV is proposed along with a mixed integer programming model for maximizing the provider's profit.

Authorized licensed use limited to: Hong Kong Polytechnic University. Downloaded on February 04,2026 at 10:07:57 UTC from IEEE Xplore. Restrictions apply.

- An attention-based DRL model is developed to realize real-time HAV scheduling in dynamic and uncertain last-mile transportation scenarios.
- Two tailored decoding mechanisms are designed that could well handle heterogeneous vehicle selection and passenger allocation, thus improving the adaptability of the proposed DRL method with HAV.
- Extensive numerical experiments and a real-world case study are conducted, not only demonstrating the superiority of the proposed service mode and DRL method but also providing a benchmark for future studies.

The remainder of this paper is organized as follows: Section II reviews related works. Section III presents the problem and develops the mathematical model. Section IV elaborates on the DRL framework and Section V gives computational experiments and analysis. Finally, Section VI concludes the paper and discusses future work.

II. Related Work

A. Last-Mile Transportation Access Modes

The last-mile transportation, which constitutes the final segment of urban transportation networks, significantly impacts accessibility and transportation efficiency, thus further affecting the sharing rate of public transportation. Urban planners and researchers have proposed various access modes to address this issue, including walking, electric scooter sharing, bike and e-bike sharing, taxis and ride-sharing, and shuttle buses [3]. Despite walking is a flexible and door-to-door mode of transportation, it is not convenient due to the distances, weather conditions, and safety concerns [39]. Electric scooter sharing offers a faster alternative. However, in many regions, legal restrictions prevent them from being used on roads [40]. Bike and e-bike sharing are popular for their flexibility and accessibility [41], but they face limitations such as weather conditions, availability of bike lanes, and shortage during peak hours [42]. In contrast to these sustainable modes, taxis and ride-sharing provide flexible, door-to-door last-mile services [43]. While many studies [44], [45] focus on using ride-sharing to connect with public transportation, these services remain relatively expensive [46]. Shuttle buses [47] are popular for their large capacity and low cost, but they operate on fixed routes and do not offer door-to-door service.

With the rapid advancement of AV technologies [48], [49], significant attention [50], [51] have been directed toward their potential to revolutionize last-mile transportation, capitalizing on their inherent advantages such as enhanced operational efficiency, precise driving patterns, and adaptive routing capabilities [52], [53]. Nevertheless, most existing studies remain confined to the paradigm of replacing conventional human-driven homogeneous fleets with autonomous counterparts, thereby neglecting the transformative potential of HAV fleet. This oversight, which represents a significant research gap, is particularly critical. On the one hand, it fails to account for the synergistic effects of strategically deploying a spectrum of vehicle capacities, ranging from small-capacity vehicles (e.g., 5-15 passenger minibus) to large-capacity vehicles (e.g., 25-30 passenger shuttles), tailored to specific operational contexts. On the other hand, by leveraging advanced fleet management algorithms, HAV systems can dynamically allocate vehicle capacities based on real-time demand fluctuations, spatial constraints, and energy efficiency considerations. Such an approach minimizes energy consumption by matching vehicle size to demand and optimizes road space utilization, thereby reducing traffic congestion and associated emissions. Consequently, the integration of HAV systems offers a promising pathway to achieving substantial improvements in operational efficiency and a measurable reduction in the carbon footprint of last-mile transportation networks.

B. Heterogeneous Autonomous Vehicles Scheduling

The access modes determine the vehicle types for last-mile transportation. Equally important are the scheduling and routing of vehicles, which significantly impact the efficiency and service quality of this transportation segment. Thus, in this section, we review the relevant literature on the scheduling and routing of HAV. The scheduling and routing problem for HAV in last-mile transportation, as studied in this paper, is fundamentally a Heterogeneous Capacitated Vehicle Routing Problem (HCVRP), which is a variant of CVRP [54]. Exact algorithms can yield optimal solutions for small-scale CVRP but often require excessive time for large-scale cases. For instance, [55] presented a Branch-Cut-and-Price algorithm for Heterogeneous Vehicle Routing Problem (HVRP) variants that generalize the CVRP, utilizing features like extended capacity cuts and vehicle-type dependent memory. However, similar to other exact algorithms, it faced the issue of long computation time for large-scale problems. Similarly, [56] introduced an exact algorithm for the heterogeneous green pickup and delivery problem, leveraging a set partitioning model to minimize carbon emissions and providing insights into the effective use of heterogeneous vehicles for emission reduction. Additionally, [57] investigated a heterogeneous multi-depot collaborative vehicle routing problem, proposing a Benders-based branch-and-cut algorithm that effectively selects transfer points for product transshipment. Reference [58] proposed a two-stage approach for shared bus dynamic route planning in the last mile scene, and designed a dynamic programming algorithm to generate optimal routes, and experiments on real data show its superiority over other methods.

Heuristic algorithms are also widely adopted for HCVRP, which can provide approximate optimal solutions within a reasonable time. For example, [59] addressed a time-constrained heterogeneous vehicle routing problem on a multigraph with parallel arcs, formulating it as a mixed-integer programming model and developing a tabu search heuristic. Reference [60] proposed a method for predicting route sequences in last-mile parcel delivery by leveraging historical route data and employing a two-stage greedy randomized adaptive search procedure for effective route planning. Reference [61] studied the precedence-constrained task assignment problem for a heterogeneous vehicle team (a truck and a micro drone) delivering packages to dispersed customers, and proposed heuristic task assignment algorithms integrating topological sorting. And

Authorized licensed use limited to: Hong Kong Polytechnic University. Downloaded on February 04,2026 at 10:07:57 UTC from IEEE Xplore. Restrictions apply.

[44] developed a column-generation metaheuristic approach for the first-mile ride-sharing problem involving public and private AV.

However, both exact and heuristic methods suffer from long computation time for large-scale problems. With the development of learning-based algorithms, deep learning, and reinforcement learning are used to solve the Vehicle Routing Problem (VRP) and its variants [62], [63]. Meanwhile, transformers based on encoder and decoder frameworks are widely embedded in reinforcement learning for VRPs. For instance, [64] designed a DRL method that ensembles a vehicle-selected decoder and a node-selected decoder for solving the HCVRP. Reference [65] presented a reinforcement learning-based hyper-heuristic for heterogeneous vehicle routing that minimizes maximum travel time. Reference [66] investigated the problem of collecting information from multiple UAVs in a disaster area and developed a DRL framework to solve it. Reference [67] designed a heterogeneous attention mechanism for solving the pickup and delivery problem. And [68] introduced the parking location and traveling salesman problem with homogeneous drones and proposed a two-phase learning-based approach to minimize operational costs in last-mile delivery. Reference [69] proposed a two-stage SA-AM method (combining simulated annealing algorithm and attention mechanism) to solve the location-routing problem for a green last-mile delivery system with shared pick-up stations.

Existing methodologies for HAV scheduling exhibit notable limitations. While exact algorithms (e.g., Branch-Cut-and-Price, Benders-based branch-and-cut) guarantee optimality, their computational complexity renders them impractical for large-scale or real-time scenarios. Heuristic approaches (e.g., tabu search, metaheuristics) balance solution quality and efficiency but often lack adaptability to dynamic on-demand environments and heterogeneous fleet constraints. Recent learning-based methods (e.g., DRL with transformer architectures, SA-AM hybrids) demonstrate promise in handling VRPs, yet critical gaps persist: (1) Most studies decouple demand allocation from route optimization, neglecting their interdependencies in dynamic decision systems; (2) Existing heterogeneous vehicle models inadequately address real-time constraints such as vehicle capacity heterogeneity, dynamic passenger allocation, and time-sensitive service requirements; (3) Current DRL frameworks for HCVRP lack mechanisms to efficiently coordinate multi-decoder structures (e.g., vehicle-passenger allocation) with real-time decision granularity.

To bridge these gaps, this work proposes an attention-based DRL framework that jointly optimizes demand allocation and routing for HAV in on-demand last-mile contexts. Our approach designs two decoder mechanisms with vehicle-specific decoders for different service modes, enabling simultaneous optimization of vehicle-passenger matching and route sequencing under capacity-time constraints. By incorporating real-time and adaptive demand allocation, the model addresses the transient nature of last-mile transportation while ensuring computational tractability for large-scale deployments.

III. Last-Mile Transportation With HAV

This section first discusses the modes of last-mile transportation and then presents the mathematical formulation of the HAV scheduling problem in last-mile transportation.

A. Service Modes of Last-Mile Transportation

Consider the last-mile transportation service around one mass transportation hub (e.g., subway station). Passengers can book the last-mile service via a smartphone application either before or upon arrival at the hub, providing their demand with destination details and expected boarding time. A fleet of AV is adopted to serve these demands. An integrated platform allocates passengers to these AV, makes routing decisions for all AV, and informs passengers about their assigned AV and the departure time. In practice, the last-mile transportation service can differ in two aspects: the types of vehicles adopted and the allocation mode of passengers.

1) Vehicle Type: Two categories exist: homogeneous vehicles [70], [71], [72] and heterogeneous vehicles [73], [74], [75], [76]. **Homogeneous vehicles** possess the same attributes in terms of speed, capacity, etc. There is no need to differentiate these vehicles when conducting passenger allocation and route planning, thus is relatively easier to make decisions. However, such settings cannot well meet variability in demand, would suffer from low service levels and high operation costs in off-peak hours, or are insufficient to cope with the high volume of demand in peak hours. Conversely, **Heterogeneous vehicles** include a variety of vehicles with different attributes, such as capacities and speeds. This diversity enables the effective allocation of the most suitable vehicles based on passenger demand.

Consider a small case with 5 last-mile transport demands. If the fleet contains only 15-pax vehicles (homogeneous vehicles), there will be a waste of capacity for the vehicle, or longer in-vehicle waiting time to fill the capacity. While a fleet contains vehicles with different capacities, it could select the one with a capacity close to or equal to 5, allowing for more convenient and efficient transportation. It would further offer opportunities to reduce both travel distance and energy consumption, thereby optimizing resource utilization and reducing carbon emissions.

2) Passenger Allocation Mode: The passenger allocation mode significantly affects fleet utilization efficiency and passenger travel time. **Sequential mode** refers to assigning a batch of passengers one by one to the same vehicle until it is full. Then, the remaining passengers are allocated to the next vehicle. In contrast, **Dynamic mode** allocates arriving passengers dynamically to all available vehicles. The passenger-vehicle matching is conducted based on the current states of all available vehicles and the distribution of demands. This dynamic allocation mode could effectively capture the overall states of the fleet, enable optimal passenger distribution, improve transportation efficiency, and reduce both the waiting and travel time. With the above two dimensions, four distinct service modes are available for last-mile transportation, as shown in Fig. 1.

Authorized licensed use limited to: Hong Kong Polytechnic University. Downloaded on February 04,2026 at 10:07:57 UTC from IEEE Xplore. Restrictions apply.

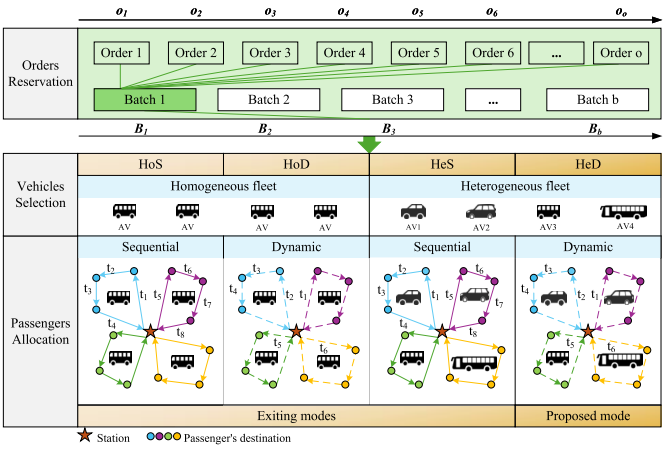

Fig. 1. In the orders reservation phrase, passengers submit their orders at different times (e.g., o_1, o_2, o_3) either before arriving at the transportation hub or upon arrival. The service platform collects real-time order information and categorizes the orders into distinct demand batches based on their arrival times (e.g., B_1, B_2, B_3). These demand batches are then fed into the vehicle selection and passenger allocation phases. Different service modes correspond to specific vehicle types (Homogeneous or Heterogeneous) and distinct passenger allocation strategies (Sequential or Dynamic), ensuring adaptability to varying operational requirements and demand patterns.

3) Homogeneous + Sequential (HoS): This mode employs a fleet with homogeneous vehicles and the scheduling process begins with the random selection of an AV from the fleet. A batch of arriving passengers is then sequentially allocated to this selected AV at each decision step. This allocation continues until the AV is full. Then, the remaining passengers will be assigned to the next available AV.

4) Homogeneous + Dynamic (HoD): This mode employs a fleet with homogeneous vehicles and the process begins with the random selection of an AV from the fleet. At each decision step, the selection of the AV is based on the current state of all AV. A batch of arriving passengers is then allocated to the selected AV one by one. This allocation process continues until all passengers have been assigned to AV.

5) Heterogeneous + Sequential (HeS): This mode employs a fleet with HAV and the process begins by sorting the AV in ascending order based on their capacity. A batch of arriving passengers is then allocated sequentially to the selected AV until it is full. Any remaining passengers will be subsequently assigned to the next available AV.

6) Heterogeneous + Dynamic (HeD): This mode employs a fleet with HAV and the process involves assessing the current and potential future profits of all AV in the fleet. At each decision step, an AV is selected based on the current state of all AV and their expected profitability. The batch of arriving passengers is then allocated to the selected AV one by one until all passengers have been assigned.

B. Mathematical Model of HAV Scheduling

Consider a last-mile transportation service provider that has a set of HAV with different capacities, represented as $H = \{h_1, h_2, h_3, \ldots, h_j\}$. For a given AV h, we use Q_h, u_{ih}, and r_h to denote its capacity, current passenger load, and operation cost per unit time, respectively. Meanwhile, we assume the fares for different types of AV are the same, represented as c. Given a set of last-mile transportation demands from a mass transportation hub to locations around it, the set of their destinations is represented as N. Passengers with the same destination are grouped into a batch and are matched as the same service node [77]. d_i denotes the number of passengers whose destination is node i.

TABLE I
NOTATION TABLE

Sets	
N	The set of destinations, $N = \{n_1, n_2, n_3, ..., n_i\}$
H	The set of HAV, $H = \{h_1, h_2, h_3, ..., h_j\}$
B	The set of passenger batch, $B = \{B_1, B_2, B_3, ..., B_b\}$
Parameters	
Q_h	The capacity of the AV h
u_{ih}	The current load of AV h at node i
d_i	The number of passengers whose destination is node i
c	The fixed fare for a passenger using AV
c_{ij}^h	The variable cost of AV h from node i to node j
r_h	The cost per time unit of using AV h, cost coefficient
λ_i^h	The time when AV h arrives at node $i, \forall i \in N$
T_i	The departure time to node $i, \forall i \in N$
T_0	The time when passengers arrive at station
t_{ij}^h	The travel time from node i to node j by the AV h
δ	The penalty for waiting
w_i	The waiting time of passengers with destination i
Decisions	
y_{ih}	$y_{ih} = 1$, if HAV h visits node i
x_{ij}^h	$x_{ij}^h = 1$, if HAV h travels directly from node i to node j

Unlike cargo transportation, wait time and travel time greatly affect the passenger experience. Therefore, the waiting time of passengers and the operation cost based on travel time are considered in the objective function. Each destination node is subject to a waiting time constraint, for which a total penalty δw_i will be given, where δ is a penalty coefficient and w_i is the total waiting time of passengers who have the same destination node i. The operation cost of an AV h depends on the time t_{ij} it travels on arc (i, j), that is $c_{ij}^h = t_{ij}^h \times r_h$. A mixed-integer programming model for this problem is formulated from (1) to (12) and the notations used are shown in Table I.

$$\max \sum_{i\in N}\sum_{h\in H}(cd_i - \delta w_i)y_{ih} - \sum_{i\in N}\sum_{j\in N}\sum_{h\in H} c_{ij}^h x_{ij}^h \tag{1}$$

$$s.t.\ w_i = (T_i - T_0)d_i, \quad \forall i \in N \tag{2}$$

$$\sum_{h\in H} y_{0h} \le |H| \tag{3}$$

$$\sum_{j\in N} x_{ij}^h = y_{ih}, \quad \forall i \in \{N \cup 0\},\ h \in H \tag{4}$$

$$\sum_{j\in N} x_{ji}^h = y_{ih}, \quad \forall i \in \{N \cup n+1\},\ h \in H \tag{5}$$

$$\sum_{h\in H} y_{ih} = 1, \forall i \in N \tag{6}$$

$$\sum_{h\in H}\sum_{i=0, i\neq j}^{n} x_{ij}^h \le 1, \quad \forall j \in N \tag{7}$$

$$T_i + t_{0i}^h = \lambda_i^h, \quad \forall i \in N,\ h \in H \tag{8}$$

$$d_i \le u_{ih} y_{ih} \le Q_h, \quad \forall i \in N,\ h \in H \tag{9}$$

$$\lambda_i^h + t_{ij}^h \le \lambda_j^h + L(1 - x_{ij}^h), \quad \forall i, j \in N,\ h \in H \tag{10}$$

Authorized licensed use limited to: Hong Kong Polytechnic University. Downloaded on February 04,2026 at 10:07:57 UTC from IEEE Xplore. Restrictions apply.

$$x_{ij}^{h} \in \{0,1\}, \quad \forall i,j \in N,\ h \in H \tag{11}$$

$$y_{ih} \in \{0,1\}, \quad \forall i \in N,\ h \in H \tag{12}$$

The objective function (1) maximizes the total profit of last-mile transportation by all AV, which depends on the fares, travel time, and penalties for waiting. Specifically, this function is composed of two primary terms, each capturing distinct aspects of profitability and cost, which explain their differing dependencies on the indices i and j. The first term, $\sum_{i\in N}\sum_{h\in H}(cd_i - \delta w_i)y_{ih}$, represents the total profit generated from serving passengers, adjusted by penalties incurred due to passenger waiting times. Here, cd_i denotes the fare profit collected from all passengers whose destination is node i, and δw_i is the penalty associated with the total waiting time for passengers destined for node i. The decision variable y_{ih} indicates if AV h serves node i. This term's dependency is solely on node i (the destination where service value is realized) and the assigned vehicle h, as it quantifies the benefit derived from serving a demand at a specific location, irrespective of the subsequent path segment. Conversely, the second term, $-\sum_{i\in N}\sum_{j\in N}\sum_{h\in H} c_{ij}^{h}x_{ij}^{h}$, quantifies the total operational costs incurred by all AV while traversing their routes. The variable c_{ij}^{h} represents the cost for AV h to travel directly from node i to node j, which is dependent on the travel time t_{ij}^{h} and the vehicle's cost per unit time r_h. The decision variable x_{ij}^{h} indicates if AV h travels directly from node i to node j. This term explicitly accounts for expenditures associated with physical movement along specific travel segments (arcs $i \to j$), hence its direct dependency on both indices i and j.

Equality (2) denotes the calculation of the total waiting time of node i. Constraint (3) limits the maximum number of AV depart from the station. Constraints (4) and (5) mean the out-degree and in-degree constraints, which guarantee flow conservation of the problem. Constraints (6) and (7) ensure that each node will be visited at most once. Equality (8) denotes the departure and arrival time of node i. Constraint (9) denotes the capacity constraint for each AV. The visiting sequence within each route is specified in constraint (10), where L is a large positive constant to linearize the inequality. Constraints (11) and (12) are the range of decision variables.

IV. Methodology

In this section, a DRL model with an attention mechanism is proposed to address HAV scheduling for last-mile transportation. By pre-training the DRL model, the method enables efficient decision-making in practical scenarios.

A. DRL-Based Framework

To solve the HAV scheduling problem using the DRL-based method, the Markov Decision Process (MDP) model for the problem is built first. Here, the problem is modeled as a finite MDP $M = (S, A, P, R)$.

1) State: The state, denoted as $s_t = (H_t, N_t) \in S$, represents a comprehensive snapshot of the system at each decision step t. S is a finite set of all possible states. Specifically, H_t encapsulates the real-time information of all HAV in the fleet, including their current passenger load, accumulated profit from previous assignments, and their last served node. N_t comprises the details of all unserved passenger demands, such as their respective destinations ($N = \{n_1, n_2, \ldots, n_i\}$) and the current number of passengers (d_i) waiting for each destination. This granular and complete state representation ensures that all necessary information for sequential decision-making is readily available.

2) Action: The action, denoted as $a_t = (h_t, n_t) \in A$, is taken at each decision step t. A represents a finite set of all possible actions. This action signifies the selection of an available heterogeneous autonomous vehicle $h_t \in H$ to serve a specific unvisited passenger demand node $n_t \in N$. The selection process considers the current state s_t to determine the most profitable vehicle-node assignment. This discrete action space facilitates real-time decision-making within the DRL framework.

3) Transition Function: The transition function P denotes the probability of transiting from the current state s_t to the next state s_{t+1}, based on the executed action a_t. Specifically, the transition function $P(s_{t+1}|s_t, a_t)$ describes how the system state evolves when action $a_t = (h_t, n_t)$ is taken in state s_t. Upon selecting AV h_t to serve a node n_t, the number of unserved passengers at node n_t is updated (or node n_t is marked as visited), and its demand d_{n_t} decreases. Concurrently, the current passenger load u_{h_t} and accumulated profit of AV h_t are deterministically adjusted based on the service completion. The new state $s_{t+1} = (H_{t+1}, N_{t+1})$ is a direct consequence of these updates, reflecting the altered status of vehicles and remaining demands.

4) Reward: R is the reward function $max_{h\in H}\sum_{t=0}^{T} f_t$, where f_t is the profit of fleet at step t, which is determined by the fares, travel time, and penalties for waiting according to the formulation 1. The immediate reward f_t at each decision step t is carefully formulated to represent the profit generated from the service. This profit is calculated as the fare revenue from assigned passengers, minus any penalties incurred due to passenger waiting times, and the operational cost associated with the vehicle's travel for that step. Specifically, f_t reflects the incremental profit contributed by the current action (h_t, n_t), and is derived from the structure of equation 1:

$$f_t = (c \cdot d_{n_t} - \delta w_{n_t}) - c_{current\_path}^{h_t} \tag{13}$$

where d_{n_t} is the number of passengers at node n_t being served at step t, w_{n_t} is the waiting time of n_t, and $c_{current\_path}^{h_t}$ represents the operational cost for the travel segment at step t associated with serving n_t. This design ensures that the DRL agent is incentivized to maximize the overall economic efficiency and service quality of the HAV fleet.

Note that the comprehensive state definition ensures that the problem inherently satisfies the Markov property. This critical property dictates that the future evolution of the system, including the probability distribution of the next state (s_{t+1}) and the expected immediate reward (f_t), depends solely on the current state (s_t) and the action taken (a_t), and is independent of any past states or actions leading to s_t. The detailed information encapsulated within H_t (real-time vehicle status) and N_t (unserved demand status) is sufficient to compute the transition function $P(s_{t+1}|s_t, a_t)$ and the reward function

Authorized licensed use limited to: Hong Kong Polytechnic University. Downloaded on February 04,2026 at 10:07:57 UTC from IEEE Xplore. Restrictions apply.

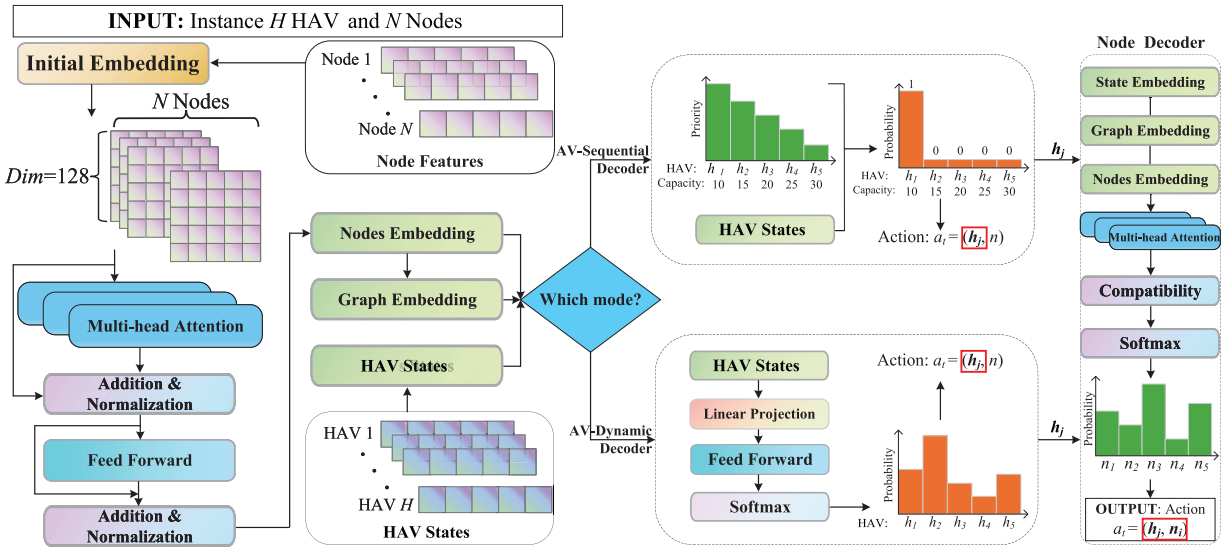

Fig. 2. Policy network framework.

$R(s_t, a_t)$. Therefore, no additional historical data is needed to predict future system states or rewards, which rigorously validates the use of an MDP for modeling this problem.

In the practical last-mile problem studied in this study, the optimization objective is to maximize the overall profit of the HAV fleet, specifically the sum of the cumulative profits of all the HAV. To reduce variance and improve training efficiency, the REINFORCE algorithm with a greedy rollout baseline is employed, which generates sequences representing the AV' travel routes and can directly return the overall rewards. The goal of REINFORCE is to maximize the expected value of the rewards, and the probability of generating a sequence τ during the interaction between the agent AV and the environment with policy π is as follows:

$$P(\tau \mid \pi) = \prod_{t=0}^{T-1} P\left(s_{t+1} \mid s_t, a_t\right) \pi\left(a_t \mid s_t\right) \tag{14}$$

where $P\left(s_{t+1} \mid s_t, a_t\right)$ is the state transition probability and π is the policy, denoting the probability of taking action a_t in state s_t. Then the objective function of REINFORCE can be denoted as:

$$J\left(\pi_\theta\right) = \int_\tau P\left(\tau \mid \pi_\theta\right) R(\tau) = \underset{\tau \sim \pi_\theta}{E}\left[R(\tau)\right] \tag{15}$$

The overall profit of the HAV fleet can be denoted as $F(a)$, where a denotes a complete set of actions. Then the algorithm's goal is to update the parameter θ to affect the distribution of policy π_θ, and thus optimize the $J\left(\pi_\theta\right)$. Based on the Monte Carlo sampling method to sample and average multiple sequences, the baseline b_s is added to reduce the variance and accelerate the algorithm's convergence, which does not affect the gradient of the equation 15. Then the policy gradient with baseline can be formulated as:

$$\nabla J\left(\pi_\theta\right) = E\pi_\theta[F(a) - b(s)\nabla \log \pi_\theta\left(a_t \mid s_t\right)] \tag{16}$$

By using the greedy rollout as the baseline $b(s)$, the function $F(a) - b(s)$ becomes negative when the sampled solution is superior to the greedy rollout. This reinforcement strengthens the selected actions, and vice versa. This training approach enables the model to continuously improve upon its own (greedy) performance [63]. Then the encoder and decoder framework will be described in detail.

B. Policy Network Framework

The policy network is based on the Transformer model [78] and consists of an encoder and a decoder. The encoder extracts features from the input problem instances using a multi-head attention mechanism. The decoder then outputs the selected action, indicating which AV will serve which node until all nodes have been visited. It is important to note that the encoder is applied only once to embed all the information of the nodes, after which the final sequence of input nodes is generated through multiple decoding steps. As shown in Fig. 2.

1) Encoder: The raw information for an instance of real last-mile transportation includes a set of destination nodes, each with different passenger demands. Similar to the structure of Transformer in [63] and [78], the encoder first embeds the information of each node n_i in the problem instance into a high-dimensional vector space N_i, with $dim_N = 128$ (i.e., initial embedding) using a learnable linear projection with parameters W and φ.

$$N_i = Wn_i + \varphi \tag{17}$$

Unlike conventional optimization algorithms such as exact methods or heuristic approaches [59], [75], [79], which often struggle with the scalability and intricate interdependencies in large-scale graphs, the attention-based encoder focuses on efficiently capturing the complex relationship information between nodes. This is crucial because the embedding of a single node cannot adequately reflect the global node information of the whole graph in such complex scenarios, leading to sub-optimal solutions with traditional methods. The node embeddings are then processed by 3 attention layers

Authorized licensed use limited to: Hong Kong Polytechnic University. Downloaded on February 04,2026 at 10:07:57 UTC from IEEE Xplore. Restrictions apply.

for better feature representation, each consisting of a multi-head attention (MHA) and a feed-forward layer (FF) [78]. The trainable parameters query, key, and value of each node i are denoted as W_i^q, W_i^k, and W_i^v to compute the attention weight by the dot-product and softmax. The dimension of k_i is $dim_k = dim_N/Z$, where $Z = 8$ is the head number of MHA. Then the vectors N_i' can be calculated by the q_i of node i, the k_j and v_j of node j.

$$q_i = N_i W_i^q, \quad k_i = N_i W_i^k, \ v_i = N_i W_i^v \tag{18}$$

$$N_i' = \sum_j \text{softmax}\left(\frac{q_i^T k_j}{\sqrt{\text{dim}_k}}\right) v_j \tag{19}$$

To further find different feature relationships in the graph, vectors N_i' are processed through MHA and divided into different subspaces $\left(N_{i,1}', N_{i,2}', \ldots, N_{i,Z}'\right)$ to compute the relationships by a head. Then concatenates all these heads and projects them into a new feature space with the same dimensions as the input N_i'. Represented as follows:

$$\text{MHA}\,(q_i, k_i, v_i) = W_i^o \,\text{Concat}\left(N_{i,1}', N_{i,2}', \ldots, N_{i,Z}'\right) \tag{20}$$

The output of the MHA sublayer is fed to the FF sublayer with the ReLU activation function [80] to obtain the next embeddings. Here, both MHA and FF sublayers use a skip-connection [81] and a batch normalization layer [82]. Finally, the encoder computes the graph embedding from the mean of the node embeddings, both of which are the input for the subsequent decoder.

2) Decoder: The decoder calculates the selection probability p_t of the next action $a_t = (h_j, n_i)$ at each time t, which consists of selecting an unvisited node n_i (note that nodes that do not satisfy the constraints will be masked), and an available AV h_j to visit the node. This decoder framework represents a significant practical advancement over existing DRL approaches [64], [65], [83], which often suffer from single-framework limitations. Our study specifically investigates four distinct service modes of AV selection, namely HoS, HoD, HeS, and HeD, demonstrating the framework's adaptability to diverse real-world last-mile scenarios. The distinction between Ho (Homogeneous) and He (Heterogeneous) refers to vehicles with the same or different attributes. During AV selection, modifying the corresponding AV attributes enables the implementation of these modes. Crucially, AV selection relies on the output of a probability distribution for choosing a specific vehicle to execute an action. To provide unparalleled flexibility and effectiveness in these varied modes, our framework designs two tailored AV selection methods: Sequential (S) and Dynamic (D). These primarily differ in their utilization of existing information, and are described as follows.

a) AV-Sequential decoder (AV-S): AV-S refers to sorting all AV in the fleet in ascending order of their capacities and then sequentially selecting AV. Once an AV is fully loaded, the next AV will be selected. During AV selection, only the information of the previously selected node is considered, without the current information of other AV. For instance, at time step 1, ... until step t, the selected actions $a_1 = (h_1, n_i), \ldots, a_t = (h_1, n_j)$ consistently involve AV h_1, and only at time $t+1$ is AV h_2 considered for selection. To capture each AV state at each time t, the AV state embedding is defined at each time step. For each AV selection, only the state of the current AV h_j is concerned, including the accumulated profit $\sum_t f_j^t$, the allocated nodes n_j^t and the last selected node n_{t-1}, i.e., the selection probability of the current vehicle is determined at the current step t.

$$V_j^t = \left\{\sum_t f_j^t, n_j^t, n_{t-1}\right\} \tag{21}$$

$$p_j^t = 1, \quad p_m^t = 0, \ m \in |H| \neq j \tag{22}$$

b) AV-Dynamic decoder (AV-D): Unlike the AV-S, the AV-D involves dynamic selection of AV across the entire fleet, requiring current information about all AV for each selection. Specifically, the probabilities associated with AV selection are uncertain and are calculated based on the current state of each AV. For each selection, all relevant feature information about the current AV is considered, including their accumulated profits $\sum_t f_j^t$, the allocated nodes n_j^t and the last selected node n_{t-1}. This comprehensive information allows the policy network to intrinsically learn from the sequence of visited nodes, rather than relying on simple masking techniques common in prior researches [62], [63] or repetitive individual route embedding leading to computational overhead [64], [67]. The AV state context V_t is defined at each time step to capture the AV features better.

$$V_t = \left\{\left[\sum_t f_1^t, n_1^t, n_{t-1}\right], \ldots, \left[\sum_t f_j^t, n_j^t, n_{t-1}\right]\right\} \tag{23}$$

Following this, the AV state context V_t is linearly projected using trainable parameters W_j and φ_j. Subsequently, it is passed through a 512-dimensional feedforward layer (FF) with the ReLU activation function to produce the AV state embedding V_t' at time step t.

$$V_t' = FF(W_j V_t + \varphi_j) \tag{24}$$

Subsequently, the selection probability for the current AV is obtained using softmax function, and the maximum value is selected as the current selection h_j based on a greedy strategy.

$$p_j^t = softmax(V_t') \tag{25}$$

The decoder's action selection consists of two parts: AV selection h_j and node allocation n_i, thus forming the current action $a_t = (h_j, n_i)$. Then a context embedding $N_c = [\bar{N}_i, V_t^{j'}, N_0^t]$ is first defined, including graph embedding $\bar{N}_i$, state embedding $V_t^{j'}$, and first node embedding N_0^t. The graph embedding is obtained from the encoder, the state embedding contains the current state of the selected AV h_j, and the first node embedding contains the starting node of the AV. Then a new embedding N_c' is computed by MHA based on context embedding N_c and node embeddings N_i' as a glimpse. Different from the encoder, here the query q_c comes from the context embedding, key k_i, and value v_i from the node embeddings.

$$q_c = N_c W^q, \quad k_i = N_i' W^k, \ v_i = N_i' W^v \tag{26}$$

$$N_c' = \text{MHA}\,(q_c, k_i, v_i), \quad q_j = N_c' W^j \tag{27}$$

Authorized licensed use limited to: Hong Kong Polytechnic University. Downloaded on February 04,2026 at 10:07:57 UTC from IEEE Xplore. Restrictions apply.

Algorithm 1 REINFORCE With Rollout Baseline

Require: Number of epochs K; batch size B; significance β
for each epoch = 1, 2,..., K **do**
  for each batch = 1, 2,..., T **do**
    Generate B instances randomly;
    for i = 1, 2,..., B **do**
      Select actions through current policy;
      Observe reward and next state;
    end for
    $F(a)$ = Max-Sum Profit
    Greedy Rollout with baseline θ_b, compute reward $b(s)$
    $\nabla\theta \leftarrow \frac{1}{B}\sum_{i=1}^{B}(F(a)-b(s))\nabla\log\pi_\theta(a_t \mid s_t)$
    $\theta \leftarrow \text{Adam}(\theta, \nabla\theta)$
  end for
  if OneSidedPairedTTest $(\pi_\theta, \pi_{\theta_b}) < \beta$ **then**
    $\theta_b \leftarrow \theta$
  end if
end for

Then the attention weight g_{ji} and selection probability p_{ji} of node n_i visited by AV h_j are computed as follows:

$$g_{ji} = C \cdot \tanh\left(\frac{q_j^T k_i}{\sqrt{\dim_k}}\right) \tag{28}$$

$$p_{ji} = softmax(g_{ji}) \tag{29}$$

Finally, at each time step, an action $a_t = (h_j, n_i)$ is determined by the probability p_j^t and p_{ji}. After obtaining a new action, the node will be allocated to its corresponding AV, and the cumulative profit of the AV will be computed. The environmental information is updated, and new states are observed. This iteration continues until all nodes are allocated to their respective AV, and the overall profit of each AV will be calculated using the formula 1. The policy is updated using the REINFORCE algorithm. The REINFORCE with rollout baseline is depicted in Algorithm 1.

c) Computational Complexity Analysis

The computational complexity of our proposed DRL algorithm (Algorithm 1) is primarily determined by the operations within its attention-based encoder-decoder architecture. We estimate the complexity based on the problem instance size, denoted by the number of passenger destination nodes N and the number of HAV H.

1) Encoder Complexity: The encoder utilizes MHA layers to process node embeddings. For an MHA layer with Z heads and key dimension d_k, the complexity of computing attention weights across N nodes is $O(Z \cdot N^2 \cdot d_k)$. The subsequent FF layers contribute an additional $O(N \cdot d_{hidden}^2)$ complexity, where d_{hidden} is the hidden embedding dimension. Thus, the encoder's complexity is dominated by the attention mechanism, approximated as $O(N^2 \cdot Z \cdot d_k)$.

2) Decoder Complexity: At each decoding step, the decoder selects an AV and an unvisited node. This involves attention computations between a context embedding and the remaining node embeddings. For selecting a node, the operation is approximately $O(N \cdot d_{hidden})$. Given that the decoding process continues for approximately N steps (until all nodes are visited), the total complexity of the decoder per instance is roughly $O(N \cdot N \cdot d_{hidden})$, or $O(N^2 \cdot d_{hidden})$.

3) Overall Training Complexity: During training, Algorithm 1 generates a batch of B instances. For each instance, a forward pass through the encoder-decoder network is performed. Additionally, the REINFORCE algorithm employs a greedy rollout baseline, which involves an extra forward pass for each sampled trajectory. Therefore, the dominant computational complexity per epoch for a batch of size B can be estimated as:

$$O(B \cdot (\text{Encoder Complexity} + N \cdot \text{Decoder Complexity})) \tag{30}$$

Substituting the dominant terms, the overall complexity becomes approximately:

$$O(B \cdot (N^2 \cdot Z \cdot d_k + N^2 \cdot d_{hidden})) \tag{31}$$

$$O(B \cdot N^2 \cdot (Z \cdot d_k + d_{hidden})) \tag{32}$$

This quadratic dependency on the number of nodes (N) is inherent to attention-based models, allowing them to capture global dependencies crucial for solving complex routing problems. While it imposes limitations on extremely large N, for typical last-mile scenarios (e.g., up to $N = 80$ nodes as in our experiments), this complexity is manageable and enables the model to learn high-quality solutions in practical timeframes.

V. Computational Experiments

This section evaluates the proposed last-mile transportation service mode and the DRL-based method through extensive experimental analysis. The evaluation begins with the comparison of the proposed service mode against existing models, followed by experiments demonstrating the proposed DRL algorithm for HAV scheduling in last-mile transportation.

A. Experiment Setting

The methods for generating data, parameter settings, and hyperparameter configurations used in this paper are based on existing research [63], [64] and have been adapted to improve the on-demand last-mile transportation studied in this work. In the experiments, a square area of $[0, 2] \times [0, 2]$ is designated to simulate the practical scope of last-mile services. Within this area, the passenger destinations are randomly generated, and the passenger demand at each destination follows a uniform distribution within the range of [1, 10]. The arrival time of each passenger at the transportation hub is randomly generated within [0, 120] before each batch's time point. The transportation hub is located at coordinates (1, 1). In terms of pricing, it is assumed that the fare for each AV is uniform. This assumption, represented by a single value c, is intentionally adopted because this study focuses on the last-mile problem as an extension of public transportation hubs. In such public-oriented services, a unified pricing approach is broadly adopted for greater public acceptance and ease of use. Furthermore, maintaining a uniform fare allows our research to isolate and effectively analyze the performance improvements derived

Authorized licensed use limited to: Hong Kong Polytechnic University. Downloaded on February 04,2026 at 10:07:57 UTC from IEEE Xplore. Restrictions apply.

TABLE II
PARAMETER SETTINGS

He/Ho	h_1	h_2	h_3	h_4	h_5
Capacity	10/20	15/20	20/20	25/20	30/20
Fare	3	3	3	3	3
Speed	0.3/0.2	0.25/0.2	0.2/0.2	0.15/0.2	0.1/0.2
Cost	0.45/0.55	0.5/0.55	0.55/0.55	0.6/0.55	0.65/0.55
Penalty	0.1	0.1	0.1	0.1	0.1

solely from the intelligent scheduling and optimized resource utilization of heterogeneous autonomous vehicles, without introducing the complexities of variable pricing strategies. The objective function includes the cost of travel time and penalties for waiting time, driving the optimization goal toward maximizing total profit and minimizing time expenditure, where the penalty coefficient for waiting time is also the same across all scenarios.

This study investigates four service modes, denoted as HoS, HeS, HoD, and HeD. To demonstrate the superiority of the proposed HeD mode under the corresponding DRL algorithm compared to the other three modes, the experimental section incorporates five AV with varying capacities, speeds, and operational costs. Heterogeneous AV have different capacities, speeds, and costs, while homogeneous AV share identical attributes. Detailed parameter settings are provided in the Table II. These parameter values (e.g., specific capacities, speeds, and costs for each AV type) are meticulously selected to represent a diverse and realistic fleet of HAV, allowing for a comprehensive evaluation of our model's ability to optimally leverage vehicle heterogeneity. The consistent fare and penalty coefficients across different AV types and scenarios are specifically chosen to simulate a unified public transportation service, focusing the optimization on resource utilization and efficiency rather than price discrimination.

While validating the performance of the proposed mode, the study also seeks to assess the performance of the proposed DRL algorithm across various data scales. The last-mile service nodes are configured to range from 10 to 80, with average demand levels spanning from 50 to 400 (each destination node has an average passenger demand of 5, derived from the uniform distribution mentioned earlier). The number of AV is set between 2 and 5 to serve these demands. Subsequently, each instance is uniquely labeled, as 'K5-N80' denotes 5 AV servicing 80 destination nodes. The hyperparameters utilized during the training process of DRL are outlined in Table III. These hyperparameters are carefully selected and adapted from [63] to ensure effective training and robust convergence of the model. **A detailed analysis of their impact and justification for their selection is provided below:**

- **Number of Instances (1,280,000) and Batch Size (512):** The selection of 1,280,000 instances is crucial to provide a sufficiently large and diverse dataset, enabling the DRL agent to learn generalizable policies that perform robustly across a wide range of last-mile scenarios and prevent overfitting. A batch size of 512 is chosen to optimally balance computational efficiency during training with the need for stable gradient updates, which is essential for consistent model improvement.
- **Dimension of Input and Hidden Embeddings (128):** The embedding dimension is set to 128. This specific value is determined to be sufficient for the model to capture rich and intricate feature representations from the complex input data (i.e., node attributes and HAV states), while simultaneously ensuring computational tractability and efficient memory utilization during the training process. A higher dimension might offer marginal gains in representation but would significantly increase computational overhead.
- **Learning Rate (10^{-5}) and Learning Rate Decay (0.996 per epoch):** The initial learning rate of 10^{-5} is carefully chosen to facilitate smooth and controlled updates to the model parameters, effectively preventing training instability, such as oscillations or divergence. The application of a learning rate decay of 0.996 per epoch is a standard and effective strategy in deep learning. It gradually reduces the step size of parameter updates as training progresses, allowing the model to finely tune its weights, converge more effectively to near-optimal solutions, and mitigate the risk of overshooting the true minimum. This approach ensures a stable and refined learning trajectory.

Our chosen settings are the result of extensive preliminary experimentation and adherence to best practices in DRL for combinatorial optimization problems. These selections contribute significantly to the model's observed performance, stability, and generalizability, as evidenced in subsequent experimental results. All experiments are conducted using Python on a Dell server equipped with four RTX 4090 GPUs.

B. Benchmark Methods

To comprehensively evaluate the performance of the proposed algorithm, we compare it with several benchmark methods, including an exact solver, a heuristic algorithm, a state-of-the-art DRL-based algorithm, and three alternative service modes introduced in this paper. All methods share the same experimental parameters (e.g., vehicle types, speeds, costs) to ensure a fair comparison, differing only in algorithm design and service mode configurations.

- **Exact Algorithm (Gurobi Solver)**: The Gurobi solver is a widely used tool for solving mixed integer programming models, known for its efficiency in finding optimal solutions in complex scenarios. It serves as a theoretical benchmark to assess whether the proposed DRL algorithm can produce near-optimal solutions.
- **Heuristic Algorithm (Genetic Algorithm)**: The Genetic Algorithm is a well-established heuristic inspired by natural selection, effective for combinatorial optimization problems like vehicle routing [59], [60]. We use it as a benchmark to evaluate the performance gap between heuristic solutions and those generated by the proposed DRL algorithm, highlighting DRL's advantages in large-scale and complex scenarios.
- **DRL-based Algorithm**: We compare our proposed algorithm with a state-of-the-art DRL method [64] that addresses the static Heterogeneous Capacitated Vehicle Routing Problem (HCVRP) using an attention-based

Authorized licensed use limited to: Hong Kong Polytechnic University. Downloaded on February 04,2026 at 10:07:57 UTC from IEEE Xplore. Restrictions apply.

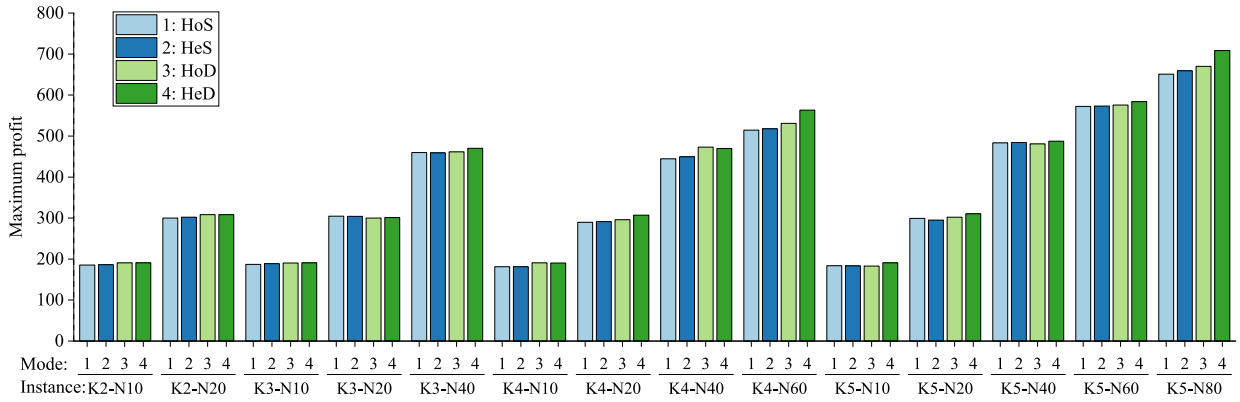

Fig. 3. Maximum profit of four service modes.

TABLE III
HYPERPARAMETER SETTINGS

Hyperparameter	Value
Number of instances	1280000
Batch size	512
Dimension of input embedding	128
Dimension of hidden embedding	128
Learning rate	10^{-5}
Learning rate decay per epoch	0.996

mechanism. This comparison allows us to evaluate the performance improvements achieved by our framework and real-time embedding strategies.

- **Other Service Modes**: We introduce three alternative service modes (HoS, HeS, HoD) to demonstrate the advantages of the proposed HeD service mode. Each mode has a tailored algorithm structure, enabling direct performance comparisons and highlighting the superiority of the HeD service mode and the corresponding DRL algorithm in efficiency and adaptability.

C. Comparison Results of Different Service Modes

This section aims to verify the advantages of the proposed service mode in maximizing the optimization objective. Each instance involves testing the trained model with a thousand randomly generated datasets, and the results are averaged to reduce the influence of outlier data on the experimental results. Table IV presents the average and maximum profit for the four service modes HoS, HeS, HoD, and HeD across different instances. In the case of small-scale data (specifically for instances K2-N and K3-N), the average of a thousand tests for the proposed HeD mode outperforms the other three modes.

In larger-scale data instances (K4-N and K5-N), the HeD mode outperforms the other three modes, dominating both average and maximum profit. The average gap in profit between HeD and the other modes is 5.07%. Specific data and gaps are detailed in rows 7-14 of Table IV. The HeD mode outperforms the other three modes in terms of both average and maximum profit, exceeding the HoS mode by 5.31%, the HeS mode by 4.84%, and the HoD mode by 1.75% in average profit. In terms of maximum profit, it surpasses the HoS mode by 3.98%, the HeS mode by 3.78%, and the HoD mode by 3.04%. These results indicate that in instances involving larger data scales, the HeD mode exhibits superior performance. The relatively minor gap values observed here are attributed to the small-scale parameters set in Table II, which do not impact the experimental results.

From the overall comparative experiments, the HeD mode outperforms the other three modes in terms of both average and maximum profit. In average profit, it surpasses the HoS mode by 4.95%, the HeS mode by 4.49%, and the HoD mode by 2.41%. Regarding maximum profit, it exceeds the HoS mode by 4.16%, the HeS mode by 3.81%, and the HoD mode by 2.41%. Fig. 3 illustrates the experimental results for the four service modes at different instances.

The results presented in Table IV demonstrate a clear trade-off between the complexity of the service mode and the resulting profit and computation time. While all modes operate within the second-level range of computation times (average 1.77s for HoS, 2.00s for HeS, 1.94s for HoD, and 2.25s for HeD), the Proposed HeD mode consistently achieves the highest average and maximum profit across diverse instances. This slightly longer computation time for HeD is a direct consequence of its inherent structural complexity and the sophisticated optimization enabled by its AV-Dynamic decoder for heterogeneous fleets, which allows it to more effectively leverage diverse vehicle capacities and dynamically allocate vehicles to maximize profit. This demonstrates that the marginal increase in computational effort for HeD is well justified by its superior operational performance.

D. Carbon Emissions of Different Service Modes

Heterogeneous fleets provide advantages over homogeneous fleets by enabling the allocation of vehicles with appropriate capacities based on actual passenger demand, thereby reducing unnecessary transportation resource waste and fleet carbon emissions. This section aims to verify the superiority of the proposed service mode in reducing carbon emissions. Table V illustrates the comparison of carbon emissions across four service modes (HoS, HeS, HoD, and HeD) at different data scales, indicating the varying environmental impacts of different modes and their respective contributions to the advancement of sustainable transportation. In Table V, ratio

Authorized licensed use limited to: Hong Kong Polytechnic University. Downloaded on February 04,2026 at 10:07:57 UTC from IEEE Xplore. Restrictions apply.

TABLE IV
COMPUTATIONAL RESULTS OF DIFFERENT INSTANCES

Instance	Demand	HoS			HeS			HoD			**Proposed HeD**	
		Profit/Max	Gap	Time	Profit/Max	Gap	Time	Profit/Max	Gap	Time	Profit/Max	Time
K2-N10	50	119.26/185.32	-0.48%	1.26	119.64/186.40	-0.16%	1.28	119.77/190.59	-0.05%	1.90	**119.83/190.69**	1.42
K2-N20	100	214.14/300.09	-1.92%	1.39	216.02/302.31	-1.06%	1.58	217.57/308.33	-0.35%	1.49	**218.33**/308.20	2.94
K3-N10	50	116.84/186.89	-3.11%	1.18	118.20/188.83	-1.98%	1.43	117.79/190.40	-2.32%	1.01	**120.59/190.59**	1.23
K3-N20	100	212.38/304.58	-2.78%	1.98	212.55/304.27	-2.70%	1.92	215.36/300.20	-1.41%	1.36	**218.45**/301.25	1.33
K3-N40	200	342.14/459.87	-2.99%	2.50	344.00/459.06	-2.47%	2.38	351.64/461.56	-0.30%	2.12	**352.70/470.26**	1.62
K4-N10	50	110.71/181.09	-7.09%	1.23	110.97/181.13	-6.87%	2.03	111.80/190.81	-6.18%	1.31	**119.16**/190.53	1.50
K4-N20	100	204.12/289.80	-6.54%	1.54	204.24/291.24	-6.49%	1.57	206.21/295.92	-5.59%	1.65	**218.41/307.05**	2.09
K4-N40	200	330.35/444.41	-6.48%	1.89	331.35/449.38	-6.19%	2.36	335.75/469.01	-4.95%	2.20	**353.23/473.82**	2.81
K4-N60	300	370.71/514.60	-6.99%	2.17	373.41/517.73	-6.31%	2.88	380.90/530.84	-4.43%	2.50	**398.56/563.10**	3.61
K5-N10	50	110.90/183.70	-6.80%	1.58	111.10/183.29	-6.63%	1.27	116.42/182.96	-2.16%	1.34	**118.99/190.76**	1.48
K5-N20	100	208.65/299.40	-7.97%	1.24	208.44/294.96	-8.07%	1.90	224.89/302.03	-0.81%	1.80	**226.73/310.35**	1.94
K5-N40	200	352.59/483.56	-4.08%	1.73	355.13/484.28	-3.39%	2.06	363.59/481.06	-1.09%	2.37	**367.59/487.47**	3.11
K5-N60	300	423.00/572.34	-2.76%	2.01	424.81/573.12	-2.34%	2.34	430.42/575.70	-1.06%	2.18	**435.01/583.76**	3.59
K5-N80	400	557.11/651.18	-6.63%	2.06	561.04/659.23	-5.97%	3.01	579.11/669.93	-2.94%	2.81	**596.65/708.68**	3.83

TABLE V
CARBON EMISSIONS OF DIFFERENT SERVICE MODES

Instance	D/K	HoS		HeS		HoD		**HeD**
		CO_2	Gap	CO_2	Gap	CO_2	Gap	CO_2
K2-N10	25.00	0.80	14.76%	0.79	12.83%	0.78	12.19%	**0.70**
K2-N20	50.00	1.49	16.62%	1.47	14.91%	1.46	14.74%	**1.28**
K3-N10	16.67	0.87	23.93%	0.74	5.79%	0.72	2.90%	**0.70**
K3-N20	33.33	1.34	22.14%	1.28	16.01%	1.27	15.60%	**1.10**
K3-N40	66.67	2.50	29.59%	2.38	23.34%	2.33	20.85%	**1.93**
K4-N10	12.50	0.94	38.51%	0.90	33.85%	0.72	6.22%	**0.68**
K4-N20	25.00	1.66	40.88%	1.65	39.92%	1.55	31.72%	**1.18**
K4-N40	50.00	4.97	83.83%	3.11	15.01%	2.85	5.49%	**2.70**
K4-N60	75.00	5.60	55.88%	4.56	27.03%	3.71	3.13%	**3.59**
K5-N10	10.00	1.07	70.56%	0.80	27.29%	0.70	11.08%	**0.63**
K5-N20	20.00	1.92	79.37%	1.43	33.64%	1.28	19.09%	**1.07**
K5-N40	40.00	3.78	56.29%	3.28	35.68%	2.75	13.67%	**2.42**
K5-N60	60.00	5.50	67.20%	4.50	37.00%	3.45	4.84%	**3.29**
K5-N80	80.00	6.25	32.48%	5.68	20.46%	5.05	7.00%	**4.72**

D/K means the average number of passenger demand allocated to each AV in different instances. Specifically, the larger this value, the more passenger demand each AV is allocated. Noted that the formula for calculating carbon emissions is as follows:

$$\begin{aligned}&\text{Carbon Emissions}\\&\quad=\text{Average Energy Consumption}\\&\qquad\times\text{Distance Traveled}\times\text{Carbon Emission Factor}\end{aligned}\tag{33}$$

Where the average energy consumption is calculated based on the average energy consumption of electric taxis in Shenzhen, which is 0.15.[1] Additionally, based on this, the average energy consumption values for four other AV with different capacities denoted as h_2, h_3, h_4, and h_5, are set to 0.2, 0.25, 0.3, and 0.35, respectively. The driving distance is the average total distance from 1,000 samples for each instance. The carbon emission factor used is based on the average for Shenzhen electricity grid over the last five years, set at 0.5.

The comparative results in Table V indicate that the proposed HeD mode achieves lower carbon emissions than the other three modes across all instances. Specifically, HeD's average emissions are 45.15% lower than HoS, 24.48% lower than HeS, and 12.04% lower than HoD. This improvement supports sustainable transportation and environmental protection. Fig. 4 illustrates the carbon emissions comparison among the four service modes across different instances.

[1]https://www.sz.gov.cn/attachment/0/927/927134/9442917.pdf

E. Comparison Analysis of Different Algorithms

After verifying the performance of the service modes, comparative experiments are conducted to assess the proposed algorithm across various instances. Common approaches for scheduling and routing problems include exact solutions, heuristic algorithms, and learning-based algorithms. This section compares the proposed algorithm with the exact solver Gurobi, the well-known Genetic Algorithm, and a state-of-the-art DRL-based algorithm [64] across different data scales.

As shown in Table VI, for smaller-scale instances, the Gurobi solver based on exact solutions can quickly determine the optimal solution. However, as the node scale increases to 40, the solution time exceeds half an hour, failing to meet the real-time computational needs of practical last-mile services. With the increase in data scale, the computation time of the Genetic Algorithm (1000 iterations) also gradually increases, and the solution quality becomes less stable. In comparison, the proposed DRL algorithm can promptly deliver good solutions in real-time. When Gurobi reaches the optimal solution, the Genetic Algorithm produces an average profit gap of -4.37%, with an average solving time that is only 37.45% of Gurobi's. The compared DRL algorithm has an average profit gap of -6.70%, with an average solving time that is only 0.27% of Gurobi's. The DRL algorithm proposed in this paper demonstrates an average profit gap of -3.39%, with an average solving time that is only 0.27% of Gurobi's. The experimental results indicate that the DRL algorithm proposed in this study outperforms the Genetic Algorithm in both average profit and average solving time, with an average profit superiority of 1.41% and a solving time that is only 0.73% of the latter. Furthermore, in terms of average profit, it also surpasses the compared DRL algorithm by 2.16%. The experimental results suggest that Genetic Algorithms exhibit certain advantages when dealing with smaller data scales. However, as the data

Authorized licensed use limited to: Hong Kong Polytechnic University. Downloaded on February 04,2026 at 10:07:57 UTC from IEEE Xplore. Restrictions apply.

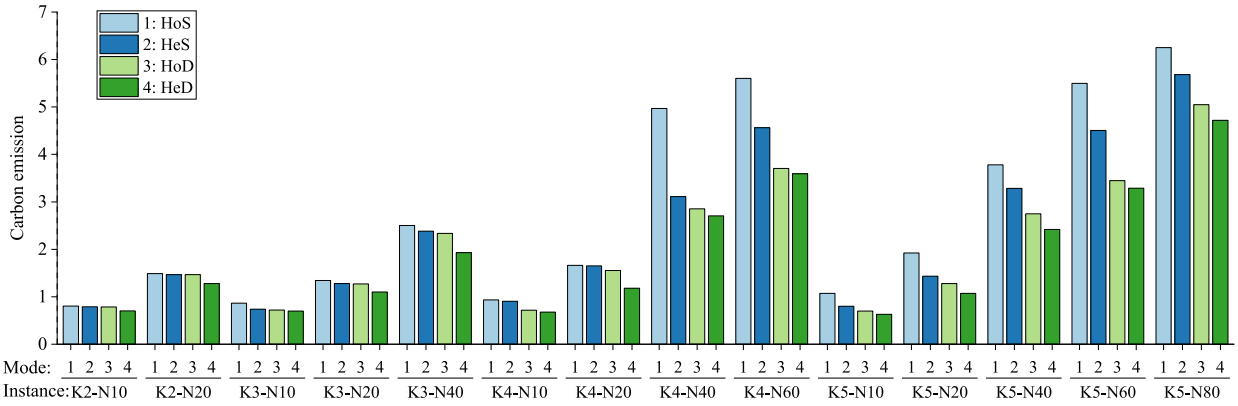

Fig. 4. Comparison of Carbon emissions.

TABLE VI
COMPARISON RESULTS OF DIFFERENT ALGORITHMS

Instance	Demand	Gurobi		Genetic Algorithm			Compared DRL			**Proposed DRL**		
		Profit	Time	Profit	Gap	Time	Profit	Gap	Time	Profit	Gap	Time
K2-N10	54	**130.57**	4.80s	123.94	-5.08%	41.82s	121.64	-6.84%	1.32s	127.53	-2.33%	1.48s
K2-N20	104	**248.63**	75.42s	240.70	-3.19%	48.90s	234.69	-5.61%	1.38s	236.38	-4.93%	1.57s
K3-N10	54	**134.06**	8.45s	129.70	-3.25%	82.36s	294.12	-9.73%	26.34s	122.93	-8.30%	1.34s
K3-N20	104	**250.02**	95.43s	244.21	-2.32%	216.88s	234.85	-6.07%	1.56s	238.91	-4.44%	1.50s
K3-N40	214	**432.88**	1721.00s	412.50	-2.63%	203.32s	401.45	-7.26%	2.04s	407.68	-5.82%	2.16s
K4-N10	54	**130.75**	37.61s	129.77	-0.75%	237.59s	123.11	-5.84%	1.30s	129.49	-0.96%	1.39s
K4-N20	104	**256.13**	175.23s	240.87	-5.96%	280.82s	238.94	-6.71%	1.93s	241.30	-5.79%	1.84s
K4-N40	214	**440.40**	2107.20s	408.93	7.15%	316.37s	410.19	-6.86%	2.54s	422.63	-4.03%	2.50s
K4-N60	332	-	-	441.44	-5.54%	384.51s	453.26	-3.01%	3.17s	**467.35**	0%	2.93s
K5-N10	54	**131.66**	54.76s	128.23	-0.33%	100.25s	125.15	-4.94%	1.22s	130.43	-0.93%	1.16s
K5-N20	104	**253.43**	387.47s	235.94	-6.90%	301.40s	232.57	-8.23%	1.48s	238.59	-5.86%	1.55s
K5-N40	214	**448.17**	3445.29s	420.77	-6.11%	416.59s	423.18	-5.58%	2.26s	429.74	-4.11%	2.21s
K5-N60	332	-	-	466.90	-2.22%	504.58s	469.33	-1.71%	3.11s	**477.50**	0%	3.00s
K5-N80	429	-	-	635.31	-2.21%	731.42s	637.64	-1.86%	3.72s	**649.70**	0%	3.63s

scale increases, the DRL algorithm proposed outperforms both the Genetic Algorithm and the compared DRL algorithm.

The comparative analysis in Table VI highlights the crucial trade-offs among exact, heuristic, and DRL-based algorithms concerning solution quality and computation time. Exact solvers like Gurobi, while guaranteeing optimal solutions for smaller instances, quickly become computationally intractable as problem scale increases, failing to meet real-time demands. Heuristic algorithms such as the Genetic Algorithm offer approximate solutions within reasonable time frames, but their computation time is generally longer than DRL methods, and solution quality can be less stable for larger instances. In contrast, DRL-based algorithms, including both the Compared DRL and our Proposed DRL, demonstrate the ability to provide high-quality solutions with real-time computation speeds, typically within a few seconds. Specifically, our Proposed DRL algorithm consistently shows a superior average profit compared to both the Genetic Algorithm and the Compared DRL algorithm, while maintaining a highly competitive and comparable solving time. This positions our approach as a highly practical solution for real-time HAV scheduling, balancing excellent solution quality with the necessary computational efficiency for dynamic last-mile transportation.

F. Case Study

To further validate the robustness and practical applicability of our proposed algorithm, and to directly address concerns regarding its performance under more realistic demand patterns, we conducted an extensive case study using real-world taxi trajectory and order data. This study focuses on the last-mile transportation service around Shenzhen North Station, a major transportation hub in Shenzhen, China, encompassing both high-speed rail and subway stations. The Fig. 5 shows the Last-mile transportation service area around Shenzhen North Station.

1) Data Collection and Scenario Emulation: We utilized a taxi trajectory dataset from March 16, 2023, a typical weekday, covering 19,690 electric taxis in Shenzhen. To emulate generalized last-mile transportation services, we filtered taxi orders originating from Shenzhen North Station and terminating within a 3km and 5km radius centered at Shenzhen North Station during both evening peak hours (18:00-18:20 selected) and off-peak hours (10:00-10:20 selected). Orders with destinations located within 150m of each other were aggregated as a single destination. This method was employed to simulate the scenario where a single destination has multiple demands.

Authorized licensed use limited to: Hong Kong Polytechnic University. Downloaded on February 04,2026 at 10:07:57 UTC from IEEE Xplore. Restrictions apply.

TABLE VII
COMPARISON RESULTS OF REAL-WORLD CASES AROUND SHENZHEN NORTH STATION

Range	Period	Instance	Demand	Gurobi		Genetic Algorithm			Compared DRL			**Proposed DRL**		
				Profit	Time	Profit	Gap	Time	Profit	Gap	Time	Profit	Gap	Time
3km	Peak	K5-N48	105	**294.50**	4235.66s	264.50	-10.20%	496.20s	266.10	-9.64%	1.32s	275.88	**-6.32%**	1.14s
	Off-peak	K5-N37	54	**103.40**	1260.57s	94.40	-8.70%	365.11s	92.16	-10.87%	1.73s	97.38	**-5.82%**	1.91s
5km	Peak	K5-N102	196	-	-	335.30	-7.99%	1580.42s	345.12	-5.31%	3.30s	**364.48**	0%	3.26s
	Off-peak	K5-N41	61	**121.20**	3805.89s	106.60	-12.04%	446.28s	109.47	-9.68%	2.3s	112.51	**-7.17%**	2.09s

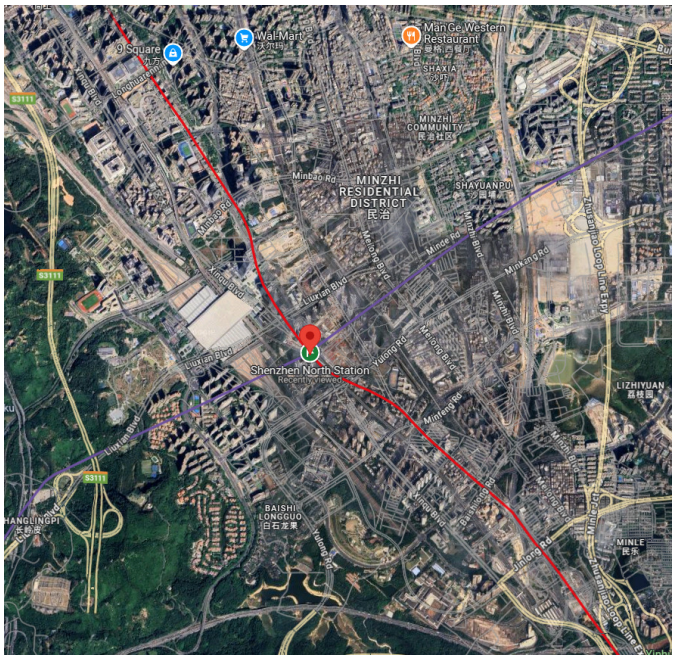

Fig. 5. The Last-mile Transportation Service Area around Shenzhen North Station.

2) Case Instances and Parameter Adjustments: Based on the real-world data, four distinct case instances were designed, differing in range and period, as summarized below:

- **3km Range, Peak Period (K5-N48):** This instance involved 105 orders distributed across 48 destinations, with an average demand of 2.19 passengers per destination.
- **3km Range, Off-peak Period (K5-N37):** This instance comprised 54 orders across 37 destinations, with an average demand of 1.46 passengers per destination.
- **5km Range, Peak Period (K5-N102):** A larger scenario with 196 orders distributed over 102 destinations, averaging 1.92 passengers per destination.
- **5km Range, Off-peak Period (K5-N41):** This instance included 61 orders across 41 destinations, with an average demand of 1.49 passengers per destination.

All distances and coordinates used in these case studies were derived from Google Maps. To reflect real-world conditions, model parameters were appropriately adjusted: the fare (3) and penalty coefficient (0.1) remained unchanged. The HAV fleet consisted of five AV with capacities [10, 15, 20, 25, 30] and cost coefficients [0.15, 0.18, 0.2, 0.23, 0.25] (consistent with Table II). Speeds were specifically calibrated for peak [25km/h, 22km/h, 20km/h, 18km/h, 15km/h] and off-peak [35km/h, 32km/h, 30km/h, 28km/h, 25km/h] periods, reflecting traffic conditions.

The detailed computational results for these real-world case instances are presented in Table VII. The superior performance of our proposed DRL algorithm in these diverse and realistic settings demonstrates its robustness and practical applicability. Even with the complexities introduced by real-world, non-uniform demand patterns, our method consistently provides high-quality solutions, reinforcing its potential for effective deployment in actual last-mile transportation. This validation is critical for showcasing the method's readiness to address practical challenges beyond simplified theoretical scenarios.

VI. CONCLUSION AND FUTURE WORK

This study tackles the challenge of on-demand last-mile transportation by proposing a learning-based algorithm for scheduling HAV. This topic is vital for enhancing public transportation, reducing environmental impact, and improving urban mobility. By integrating advanced optimization techniques and DRL, this research bridges the gap between theory and practical applications in sustainable transportation.

The study contributes in three main ways. First, it introduces a mixed-integer programming model aimed at maximizing operational profit while minimizing passenger travel and waiting times, addressing both efficiency and user satisfaction. Second, it analyzes four service modes: HoS, HoD, HeS, and HeD, with the HeD mode showing superior performance. The HeD mode uses a dynamic passenger allocation approach supported by a novel DRL algorithm featuring the *AV-Sequential decoder* and *AV-Dynamic decoder*, enhancing scheduling flexibility. Third, experimental results validate the proposed approach, indicating that the HeD mode maximizes optimization objectives while significantly reducing fleet carbon emissions. Furthermore, our extensive case study utilizing real-world data demonstrates the practical applicability and effectiveness of the proposed algorithm in handling realistic and complex demand patterns, reinforcing its potential for real-world deployment in last-mile transportation The findings suggest that improving last-mile scheduling can attract more passengers to public transportation, reducing reliance on private vehicles and alleviating urban congestion. Additionally, flexible HAV scheduling contributes to lower carbon emissions, aligning with global sustainability goals.

While this study lays a strong foundation for HAV scheduling, several future research avenues remain. These include investigating the impact of external factors (e.g., traffic, weather) on algorithm performance, exploring optimal charging station placement for electric AV, addressing

Authorized licensed use limited to: Hong Kong Polytechnic University. Downloaded on February 04,2026 at 10:07:57 UTC from IEEE Xplore. Restrictions apply.

real-world implementation challenges (e.g., communication delays), and integrating the algorithm with smart city infrastructure for enhanced practicality and effectiveness.

REFERENCES

[1] S.-H. Cho and D. Shin, "Estimation of route choice behaviors of bike-sharing users as first{-} and last-mile trips for introduction of mobility-as-a-service (MaaS)," *KSCE J. Civil Eng.*, vol. 26, no. 7, pp. 3102–3113, Jul. 2022.

[2] K. S. Shehadeh, H. Wang, and P. Zhang, "Fleet sizing and allocation for on-demand last-mile transportation systems," *Transp. Res. C, Emerg. Technol.*, vol. 132, Nov. 2021, Art. no. 103387.

[3] P. He, J. G. Jin, F. Schulte, and M. Trépanier, "Optimizing first-mile ridesharing services to intercity transit hubs," *Transp. Res. C, Emerg. Technol.*, vol. 150, May 2023, Art. no. 104082.

[4] M. M. Rahman and J.-C. Thill, "Impacts of connected and autonomous vehicles on urban transportation and environment: A comprehensive review," *Sustain. Cities Soc.*, vol. 96, Sep. 2023, Art. no. 104649.

[5] Y. Zhang and M. Kamargianni, "A review on the factors influencing the adoption of new mobility technologies and services: Autonomous vehicle, drone, micromobility and mobility as a service," *Transp. Rev.*, vol. 43, no. 3, pp. 407–429, May 2023.

[6] X. Zhao, Y. Fang, H. Min, X. Wu, W. Wang, and R. Teixeira, "Potential sources of sensor data anomalies for autonomous vehicles: An overview from road vehicle safety perspective," *Expert Syst. Appl.*, vol. 236, Feb. 2024, Art. no. 121358.

[7] S. Chen, H. Wang, and Q. Meng, "Solving the first-mile ridesharing problem using autonomous vehicles," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 35, no. 1, pp. 45–60, Jan. 2020.

[8] J. Wu, H. Yang, L. Yang, Y. Huang, X. He, and C. Lv, "Human-guided deep reinforcement learning for optimal decision making of autonomous vehicles," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 54, no. 11, pp. 6595–6609, Nov. 2024.

[9] Z. Wang, J. Ke, and S. Li, "Planning and operation of ride-hailing networks with a mixture of level-4 autonomous vehicles and for-hire human drivers," *Transp. Res. C, Emerg. Technol.*, vol. 160, Mar. 2024, Art. no. 104541.

[10] X. Li, X. Gong, Y.-H. Chen, J. Huang, and Z. Zhong, "Integrated path planning-control design for autonomous vehicles in intelligent transportation systems: A neural-activation approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 25, no. 7, pp. 7602–7618, Jul. 2024.

[11] M. Greifenstein, H. Güthner, F. Kuhnert, and A. Herrmann, "From test sites to public roads—A look at the global status of shared autonomous vehicles," *ATZ Worldwide*, vol. 126, no. 2, pp. 60–64, 2024.

[12] A. Y. Bin-Nun, P. Derler, N. Mehdipour, and R. D. Tebbens, "How should autonomous vehicles drive? Policy, methodological, and social considerations for designing a driver," *Humanities Social Sci. Commun.*, vol. 9, no. 1, pp. 1–13, Aug. 2022.

[13] D. Petrović, R. M. Mijailović, and D. Pešić, "Persons with physical disabilities and autonomous vehicles: The perspective of the driving status," *Transp. Res. A, Policy Pract.*, vol. 164, pp. 98–110, Oct. 2022.

[14] J. Zhao et al., "Autonomous driving system: A comprehensive survey," *Expert Syst. Appl.*, vol. 242, May 2024, Art. no. 122836.

[15] S. B. Sun, X. Zhao, and G. Zhang, "Shouting 'chin min yau lok' (stop at the front) in a minibus: Transportation assimilation among immigrants in Hong Kong," *Int. J. Population Stud.*, vol. 9, no. 1, pp. 30–50, 2023.

[16] D. Coelho, M. Oliveira, and V. Santos, "RLAD: Reinforcement learning from pixels for autonomous driving in urban environments," *IEEE Trans. Autom. Sci. Eng.*, vol. 21, no. 4, pp. 7427–7435, Oct. 2024.

[17] M. Xu, Y. Di, Z. Zhu, H. Yang, and X. Chen, "Design and analysis of ride-sourcing services with auxiliary autonomous vehicles for transportation hubs in multi-modal transportation systems," *Transportmetrica B: Transp. Dyn.*, vol. 12, no. 1, Dec. 2024, Art. no. 2333869.

[18] J. De Guia et al., "Advancing safety and robustness: Perception-planning system of an autonomous vehicle last mile delivery," in *Proc. IEEE Conf. Artif. Intell. (CAI)*, Jun. 2024, pp. 113–118.

[19] X. Zhou, C. Zhao, and F. Xie, "Reshaping urban logistics: Route planning strategies for collaborative delivery using autonomous vehicles and passenger-freight combined buses," in *Proc. 12th Int. Conf. Traffic Logistic Eng. (ICTLE)*, Aug. 2024, pp. 238–242.

[20] L. Chen, Y. Li, W. Silamu, Q. Li, S. Ge, and F.-Y. Wang, "Smart mining with autonomous driving in industry 5.0: Architectures, platforms, operating systems, foundation models, and applications," *IEEE Trans. Intell. Vehicles*, vol. 9, no. 3, pp. 4383–4393, Mar. 2024.

[21] Y. Long, G. Xu, J. Zhao, B. Xie, and M. Fang, "Dynamic truck–UAV collaboration and integrated route planning for resilient urban emergency response," *IEEE Trans. Eng. Manage.*, vol. 71, pp. 9826–9838, 2023.

[22] H. D. Nguyen, M. Choi, and K. Han, "Risk-informed decision-making and control strategies for autonomous vehicles in emergency situations," *Accident Anal. Prevention*, vol. 193, Dec. 2023, Art. no. 107305.

[23] R. Guo, W. Guan, M. Vallati, and W. Zhang, "Modular autonomous electric vehicle scheduling for customized on-demand bus services," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 9, pp. 10055–10066, Sep. 2023.

[24] J. Robbennolt and M. W. Levin, "Maximum throughput dispatch for shared autonomous vehicles including vehicle rebalancing," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 9, pp. 9871–9885, Sep. 2023.

[25] Z. Wang and S. Li, "Competition between autonomous and traditional ride-hailing platforms: Market equilibrium and technology transfer," *Transp. Res. C, Emerg. Technol.*, vol. 165, Aug. 2024, Art. no. 104728.

[26] S. Yang, J. Wu, H. Sun, and Y. Qu, "Trip planning for a mobility-as-a-service system: Integrating metros and shared autonomous vehicles," *Transp. Res. E, Logistics Transp. Rev.*, vol. 176, Aug. 2023, Art. no. 103217.

[27] K. Wang, G. Zhao, and J. Lu, "A deep analysis of visual SLAM methods for highly automated and autonomous vehicles in complex urban environment," *IEEE Trans. Intell. Transp. Syst.*, vol. 25, no. 9, pp. 10524–10541, Sep. 2024.

[28] M. M. Kabir, J. R. Jim, and Z. Istenes, "Terrain detection and segmentation for autonomous vehicle navigation: A state-of-the-art systematic review," *Inf. Fusion*, vol. 113, Jan. 2025, Art. no. 102644.

[29] D. Suhariyanto et al., "Autonomous vehicles: From technology to law and regulation," *Eng. Sci. Lett.*, vol. 3, no. 2, pp. 62–67, May 2024.

[30] T. Sever and G. Contissa, "Automated driving regulations—Where are we now?," *Transp. Res. Interdiscipl. Perspect.*, vol. 24, Mar. 2024, Art. no. 101033.

[31] P. Dakic et al., "Intrusion detection using metaheuristic optimization within IoT/IIoT systems and software of autonomous vehicles," *Sci. Rep.*, vol. 14, no. 1, p. 22884, Oct. 2024.

[32] Y. Chen, Y. Liu, Y. Bai, and B. Mao, "Real-time dispatch management of shared autonomous vehicles with on-demand and pre-booked requests," *Transp. Res. A, Policy Pract.*, vol. 181, Mar. 2024, Art. no. 104021.

[33] Z. Li, M. Lokhandwala, A. O. Al-Abbasi, V. Aggarwal, and H. Cai, "Integrating reinforcement-learning-based vehicle dispatch algorithm into agent-based modeling of autonomous taxis," *Transportation*, vol. 52, no. 2, pp. 641–667, Apr. 2025.

[34] C. Luo, M. He, and C. Xing, "Public acceptance of autonomous vehicles in China," *Int. J. Hum.–Comput. Interact.*, vol. 40, no. 2, pp. 315–326, Jan. 2024.

[35] M. Naiseh et al., "Trust, risk perception, and intention to use autonomous vehicles: An interdisciplinary bibliometric review," *AI Soc.*, vol. 40, no. 2, pp. 1091–1111, Feb. 2025.

[36] Y. Ji et al., "Toward autonomous vehicles: A survey on cooperative vehicle-infrastructure system," *iScience*, vol. 27, no. 5, May 2024, Art. no. 109751.

[37] M. Yazgan et al., "Shuttle2X-overcoming operational borders of autonomous shuttles by infrastructure support," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2024, pp. 390–396.

[38] R. Asín-Achá, A. Espinoza, O. Goldschmidt, D. S. Hochbaum, and I. I. Huerta, "Selecting fast algorithms for the capacitated vehicle routing problem with machine learning techniques," *Networks*, vol. 84, no. 4, pp. 465–480, 2024.

[39] L. M. Braun, J. M. Barajas, B. Lee, and R. Martin, "Walking the last mile: Barriers and solutions to suburban transit access," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 2676, no. 12, pp. 456–467, Dec. 2022.

[40] E. Huang, Z. Yin, A. Broaddus, and X. Yan, "Shared e-scooters as a last-mile transit solution? Travel behavior insights from Los Angeles and Washington DC," *Travel Behav. Soc.*, vol. 34, Jan. 2024, Art. no. 100663.

[41] T. Zuo, H. Wei, N. Chen, and C. Zhang, "First-and-last mile solution via bicycling to improving transit accessibility and advancing transportation equity," *Cities*, vol. 99, Apr. 2020, Art. no. 102614.

[42] J. Zhao, C. Guo, R. Zhang, D. Guo, and M. Palmer, "Impacts of weather on cycling and walking on twin trails in Seattle," *Transp. Res. D, Transp. Environ.*, vol. 77, pp. 573–588, Dec. 2019.

[43] R. Wang, F. Chen, X. Liu, X. Liu, Z. Li, and Y. Zhu, "A matching model for door-to-door multimodal transit by integrating taxi-sharing and subways," *ISPRS Int. J. Geo-Inf.*, vol. 10, no. 7, p. 469, Jul. 2021.

Authorized licensed use limited to: Hong Kong Polytechnic University. Downloaded on February 04,2026 at 10:07:57 UTC from IEEE Xplore. Restrictions apply.

[44] P. He, J. G. Jin, and F. Schulte, "The flexible airport bus and last-mile ride-sharing problem: Math-heuristic and metaheuristic approaches," *Transp. Res. E, Logistics Transp. Rev.*, vol. 184, Apr. 2024, Art. no. 103489.

[45] J. Husemann, S. Kunz, and K. Berns, "On demand ride sharing: Scheduling of an autonomous bus fleet for last mile travel," *Robot. Auto. Syst.*, vol. 170, Dec. 2023, Art. no. 104559.

[46] X. Ren, Z. Chen, C. Liu, T. Dan, J. Wu, and F. Wang, "Are vehicle on-demand and shared services a favorable solution for the first and last-mile mobility: Evidence from China," *Travel Behav. Soc.*, vol. 31, pp. 386–398, Apr. 2023.

[47] P. Shu, Y. Sun, B. Xie, S. X. Xu, and G. Xu, "Data-driven shuttle service design for sustainable last mile transportation," *Adv. Eng. Informat.*, vol. 49, Aug. 2021, Art. no. 101344.

[48] S. Hakak et al., "Autonomous vehicles in 5G and beyond: A survey," *Veh. Commun.*, vol. 39, Feb. 2023, Art. no. 100551.

[49] H. Taghavifar, C. Hu, C. Wei, A. Mohammadzadeh, and C. Zhang, "Behaviorally-aware multi-agent RL with dynamic optimization for autonomous driving," *IEEE Trans. Autom. Sci. Eng.*, vol. 22, pp. 10672–10683, 2025.

[50] C. Klinkhardt, K. Kandler, N. Kostorz, M. Heilig, M. Kagerbauer, and P. Vortisch, "Integrating autonomous busses as door-to-door and first-/last-mile service into public transport: Findings from a stated choice experiment," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 2678, no. 2, pp. 605–619, Feb. 2024.

[51] Y. Huang, K. M. Kockelman, and V. Garikapati, "Shared automated vehicle fleet operations for first-mile last-mile transit connections with dynamic pooling," *Comput., Environ. Urban Syst.*, vol. 92, Mar. 2022, Art. no. 101730.

[52] M. Thorhauge, A. Fjendbo Jensen, and J. Rich, "Effects of autonomous first{-} and last mile transport in the transport chain," *Transp. Res. Interdiscipl. Perspect.*, vol. 15, Sep. 2022, Art. no. 100623.

[53] A. Fidanoglu, I. Gokasar, and M. Deveci, "Integrating shared autonomous vehicles in last-mile public transportation," *Sustain. Energy Technol. Assessments*, vol. 57, Jun. 2023, Art. no. 103214.

[54] B. Golden, A. Assad, L. Levy, and F. Gheysens, "The fleet size and mix vehicle routing problem," *Comput. Oper. Res.*, vol. 11, no. 1, pp. 49–66, Jan. 1984.

[55] A. Pessoa, R. Sadykov, and E. Uchoa, "Enhanced branch-cut-and-price algorithm for heterogeneous fleet vehicle routing problems," *Eur. J. Oper. Res.*, vol. 270, no. 2, pp. 530–543, Oct. 2018.

[56] W. Sun, Y. Yu, and J. Wang, "Heterogeneous vehicle pickup and delivery problems: Formulation and exact solution," *Transp. Res. E, Logistics Transp. Rev.*, vol. 125, pp. 181–202, May 2019.

[57] Q. Zhang, Z. Wang, M. Huang, Y. Yu, and S.-C. Fang, "Heterogeneous multi-depot collaborative vehicle routing problem," *Transp. Res. B, Methodol.*, vol. 160, pp. 1–20, Jun. 2022.

[58] X. Kong, M. Li, T. Tang, K. Tian, L. Moreira-Matias, and F. Xia, "Shared subway shuttle bus route planning based on transport data analytics," *IEEE Trans. Autom. Sci. Eng.*, vol. 15, no. 4, pp. 1507–1520, Oct. 2018.

[59] D. S. Lai, O. C. Demirag, and J. M. Leung, "A Tabu search heuristic for the heterogeneous vehicle routing problem on a multigraph," *Transp. Res. E, Logistics Transp. Rev.*, vol. 86, pp. 32–52, Feb. 2016.

[60] J. P. Mesa, A. Montoya, R. Ramos-Pollán, and M. Toro, "A two-stage data-driven metaheuristic to predict last-mile delivery route sequences," *Eng. Appl. Artif. Intell.*, vol. 125, Oct. 2023, Art. no. 106653.

[61] X. Bai, M. Cao, W. Yan, and S. S. Ge, "Efficient routing for precedence-constrained package delivery for heterogeneous vehicles," *IEEE Trans. Autom. Sci. Eng.*, vol. 17, no. 1, pp. 248–260, Jan. 2020.

[62] I. Bello, H. Pham, Q. V. Le, M. Norouzi, and S. Bengio, "Neural combinatorial optimization with reinforcement learning," 2017, *arXiv:1611.09940*.

[63] W. Kool, H. van Hoof, and M. Welling, "Attention, learn to solve routing problems!," 2019, *arXiv:1803.08475*.

[64] J. Li et al., "Deep reinforcement learning for solving the heterogeneous capacitated vehicle routing problem," *IEEE Trans. Cybern.*, vol. 52, no. 12, pp. 13572–13585, Dec. 2022.

[65] W. Qin, Z. Zhuang, Z. Huang, and H. Huang, "A novel reinforcement learning-based hyper-heuristic for heterogeneous vehicle routing problem," *Comput. & Ind. Eng.*, vol. 156, Mar. 2021, Art. no. 107252.

[66] P. Wan, G. Xu, J. Chen, and Y. Zhou, "Deep reinforcement learning enabled multi-UAV scheduling for disaster data collection with time-varying value," *IEEE Trans. Intell. Transp. Syst.*, vol. 25, no. 7, pp. 6691–6702, Jul. 2024.

[67] J. Li, L. Xin, Z. Cao, A. Lim, W. Song, and J. Zhang, "Heterogeneous attentions for solving pickup and delivery problem via deep reinforcement learning," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 3, pp. 2306–2315, Mar. 2022.

[68] A. Arishi, K. Krishnan, and M. Arishi, "Machine learning approach for truck-drones based last-mile delivery in the era of industry 4.0," *Eng. Appl. Artif. Intell.*, vol. 116, Nov. 2022, Art. no. 105439.

[69] Z. Zhao, C. K. Lee, Y. P. Tsang, and X. Xu, "A heuristic-attention method for location-routing problems with shared pick-up stations in green last-mile delivery," *Transp. Res. C, Emerg. Technol.*, vol. 172, Mar. 2025, Art. no. 105031.

[70] J. Escobar, J. Duque, and R. García-Cáceres, "A granular Tabu search for the refrigerated vehicle routing problem with homogeneous fleet," *Int. J. Ind. Eng. Comput.*, vol. 13, no. 1, pp. 135–150, 2022.

[71] Y. Wang et al., "A homogeneous multi-vehicle cooperative group decision-making method in complicated mixed traffic scenarios," *Transp. Res. C, Emerg. Technol.*, vol. 167, Oct. 2024, Art. no. 104833.

[72] N. M. Khallaf, O. Abdel-Raouf, and M. Hadhoud, "Reinforcement learning-driven enhancement of medical waste collection within capacity-homogeneous vehicle routing," *Int. J. Comput. Inf.*, vol. 11, no. 2, pp. 79–94, 2024.

[73] Ç. Koç, T. Bektaş, O. Jabali, and G. Laporte, "Thirty years of heterogeneous vehicle routing," *Eur. J. Oper. Res.*, vol. 249, no. 1, pp. 1–21, 2016.

[74] V. Leggieri and M. Haouari, "Lifted polynomial size formulations for the homogeneous and heterogeneous vehicle routing problems," *Eur. J. Oper. Res.*, vol. 263, no. 3, pp. 755–767, Dec. 2017.

[75] V. G. Pereira, O. C. Alves-Junior, and F. Baldo, "An approach to solve the heterogeneous fixed fleet vehicle routing problem with time window based on adaptive large neighborhood search meta-heuristic," *IEEE Trans. Intell. Transp. Syst.*, vol. 25, no. 7, pp. 8148–8157, Jul. 2024.

[76] D. Fitzek, T. Ghandriz, L. Laine, M. Granath, and A. F. Kockum, "Applying quantum approximate optimization to the heterogeneous vehicle routing problem," *Sci. Rep.*, vol. 14, no. 1, p. 25415, Oct. 2024.

[77] H. Wang, "Routing and scheduling for a last-mile transportation system," *Transp. Sci.*, vol. 53, no. 1, pp. 131–147, Feb. 2019.

[78] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–11.

[79] F. Y. Vincent, P. T. Anh, A. Gunawan, and H. Han, "A simulated annealing with variable neighborhood descent approach for the heterogeneous fleet vehicle routing problem with multiple forward/reverse cross-docks," *Expert Syst. Appl.*, vol. 237, Mar. 2024, Art. no. 121631.

[80] A. Fred Agarap, "Deep learning using rectified linear units (ReLU)," 2018, *arXiv:1803.08375*.

[81] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[82] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 448–456.

[83] J. M. Vera and A. G. Abad, "Deep reinforcement learning for routing a heterogeneous fleet of vehicles," in *Proc. IEEE Latin Amer. Conf. Comput. Intell. (LA-CCI)*, Nov. 2019, pp. 1–6.

Yongwu Liu received the B.S. degree in network engineering and the M.S. degree in software engineering from the School of Software, Liaoning Technical University, Liaoning, China, in 2019 and 2021, respectively. He is currently pursuing the dual Ph.D. degree with Harbin Institute of Technology, Shenzhen, China, and The Hong Kong Polytechnic University, Hong Kong.

His research interests include deep reinforcement learning, large language models, data-driven optimization, and intelligent transportation.

Authorized licensed use limited to: Hong Kong Polytechnic University. Downloaded on February 04,2026 at 10:07:57 UTC from IEEE Xplore. Restrictions apply.

Binglei Xie received the B.S. degree in applied mathematics and the M.S. and Ph.D. degrees in management science and engineering from Southwest Jiaotong University, Chengdu, China, in 1996, 1999, and 2003, respectively.

He is currently a Professor with Harbin Institute of Technology, Shenzhen (HITSZ), China, where he also serves as the Director of the Research Center for Urban Emergency Management and Transportation Safety. He joined HITSZ as a Post-Doctoral Fellow in 2005, became an Associate Professor in 2007, and has been a Full Professor since 2015. His research interests include urban emergency management, transportation safety, logistics engineering, and operations research.

Prof. Xie is a Standing Committee Member of the Emergency Management Society of Chinese Society of Optimization, Overall Planning and Economic Mathematics, and an Editorial Board Member of *Journal of Transportation Systems Engineering and Information Technology*.

Yuying Long received the B.S. degree in traffic and transportation from Southwest Jiaotong University, Chengdu, China, and the M.E. degree in transportation engineering from Harbin Institute of Technology, Shenzhen, China, in 2020 and 2022, respectively. She is currently pursuing the Ph.D. degree in aviation engineering from The Hong Kong Polytechnic University, Hong Kong. Her research interests include UAV-assisted systems, intelligent transportation systems, and emergency management.

Jiawei Chen received the B.S. degree in automation from Harbin Institute of Technology, Shenzhen, China, in 2022. She is currently pursuing the Ph.D. degree with the Department of Aeronautical and Aviation Engineering, The Hong Kong Polytechnic University, Hong Kong.

Her research interests include reinforcement learning for combinatorial optimization, data-driven optimization and control, healthcare and emergency decision-making, and cooperative AI.

Gangyan Xu (Member, IEEE) received the B.S. degree in automation and the M.E. degree in systems engineering from the Huazhong University of Science and Technology, Wuhan, China, in 2009 and 2012, respectively, and the Ph.D. degree in systems engineering from The University of Hong Kong, Hong Kong, in 2016.

He was an Assistant Professor with Harbin Institute of Technology, Shenzhen, China; a Research Fellow with Nanyang Technological University, Singapore; and a Research Assistant with the City University of Hong Kong, Hong Kong. He is currently an Assistant Professor with The Hong Kong Polytechnic University, Hong Kong. His research interests include data-driven optimization, intelligent transportation systems, dynamic and cooperative decision-making, and emergency management.

Dr. Xu is an Editorial Board Member of *Advanced Engineering Informatics* and a Special Corresponding Expert of *Frontiers of Engineering Management*.

Authorized licensed use limited to: Hong Kong Polytechnic University. Downloaded on February 04,2026 at 10:07:57 UTC from IEEE Xplore. Restrictions apply.