A Game-Theoretic Spatio-Temporal Reinforcement Learning Framework for Collaborative Public Resource Allocation

Songxin Lei*
The Hong Kong University of Science and Technology (Guangzhou)
Guangzhou, China
slei924@connect.hkust-gz.edu.cn

Qiongyan Wang*
The Hong Kong University of Science and Technology (Guangzhou)
Guangzhou, China
qwang650@connect.hkust-gz.edu.cn

Yanchen Zhu
The Hong Kong University of Science and Technology (Guangzhou)
Guangzhou, China
yzhu156@connect.hkust-gz.edu.cn

Hanyu Yao
Peking University
Beijing, China
hanyuyao25@stu.pku.edu.cn

Sijie Ruan
Beijing Institute of Technology
Beijing, China
sjruan@bit.edu.cn

Weilin Ruan
The Hong Kong University of Science and Technology (Guangzhou)
Guangzhou, China
wruan792@connect.hkust-gz.edu.cn

Yuyu Luo
The Hong Kong University of Science and Technology (Guangzhou)
Guangzhou, China
yuyuluo@hkust-gz.edu.cn

Huaming Wu
Tianjin University
Tianjin, China
whming@tju.edu.cn

Yuxuan Liang†
The Hong Kong University of Science and Technology (Guangzhou)
Guangzhou, China
yuxliang@outlook.com

Abstract

Public resource allocation involves distributing resources, including urban infrastructure, energy, and transportation, which are typically limited in capacity, to meet social demands. In real-world scenarios, resources are typically limited in capacity, which makes coordination among multiple resources essential. However, existing methods often optimize resource movements in an isolated manner and do not explicitly account for capacity-aware collaboration under spatio-temporal dynamics. To address this limitation, we introduce the Collaborative Public Resource Allocation (CPRA) problem, and propose a Game-Theoretic Spatio- Temporal Reinforcement Learning (GSTRL) framework to solve it. Our contributions are twofold: 1) We formulate CPRA as a potential game and construct the potential function based on the objective function of CPRA, laying a theoretical foundation for approximating the Nash equilibrium of this NP-hard problem; and 2) Our GSTRL framework effectively captures the spatio-temporal dynamics of the overall system. We evaluate GSTRL on two real-world datasets, where experiments show its superior performance. Our source codes are available at https://github.com/thunderlrr/GSTRL.

CCS Concepts

• **Computing methodologies** → **Planning and scheduling**; • **Applied computing** → **Decision analysis**.

*Both authors contributed equally to this research.
†Corresponding author.



This work is licensed under a Creative Commons Attribution 4.0 International License.
CIKM '26, Rome, Italy
© 2026 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-2539-5/2026/11
https://doi.org/10.1145/3799682.3840588

Keywords

Public Resource Allocation, Reinforcement Learning, Potential Game

ACM Reference Format:
Songxin Lei, Qiongyan Wang, Yanchen Zhu, Hanyu Yao, Sijie Ruan, Weilin Ruan, Yuyu Luo, Huaming Wu, and Yuxuan Liang. 2026. A Game-Theoretic Spatio-Temporal Reinforcement Learning Framework for Collaborative Public Resource Allocation. In *Proceedings of the 35th ACM International Conference on Information and Knowledge Management (CIKM '26), November 07–11, 2026, Rome, Italy.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3799682.3840588

1 Introduction

Effective public resource allocation is a cornerstone of equitable and resilient urban development. From emergency response [6] and disaster relief [31] to mobile healthcare [17] and traffic control [10], the ability to dynamically deploy limited resources directly impacts the safety, well-being, and quality of life of millions of residents. In particular, underserved communities, high-density zones, and vulnerable populations are disproportionately affected by inefficiencies in service delivery [39]. Because traditional static systems often rely on historical averages and pre-defined fixed locations, they are inherently ill-equipped to handle sudden demand surges or unexpected spatial shifts in crowd flow. Consequently, as cities grow increasingly complex and real-time demands fluctuate rapidly across space and time, these static approaches inevitably fall short in meeting such critical needs.

To ensure responsive and inclusive service delivery, *dynamic public resource allocation* is emerging as a vital AI-enabled mechanism for optimizing operational efficiency under complex uncertainties, such as sudden traffic disruptions or spontaneous crowd gatherings [21]. Yet, as shown in **Fig. 1**, real-world deployment remains highly challenging. Because individual resources are typically limited in their service capacity, and public demands are inherently

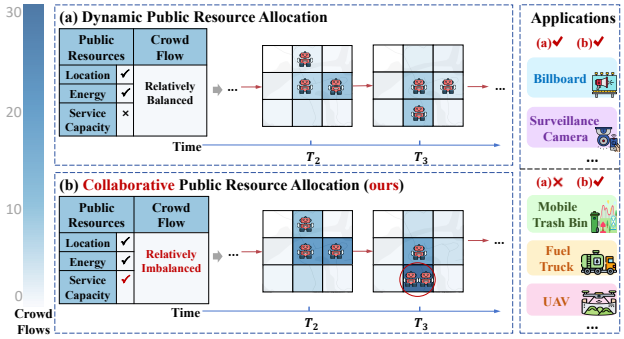

Figure 1: Dynamic public resource allocation vs. Collaborative public resource allocation (CPRA).

volatile and spatially clustered, relying on independent dispatches is often inadequate, so that explicit coordination among multiple resources to collectively serve a high-demand region becomes strictly necessary to prevent localized system overload [22]. These factors underscore the urgency of developing principled and scalable approaches that can reason about multi-agent cooperation, adapt to rapidly changing environmental conditions, and improve social outcomes in a wider range of applications.

Unfortunately, most existing AI-based approaches to dynamic allocation treat each resource independently, neglecting the vital alignment between individual actions and global system objectives [27]. This oversimplification fails to capture the collaborative nature of real-world urban systems [36], where multiple limited-capacity resources must jointly and strategically serve emerging high-demand areas [28]. For example, during peak traffic hours or large-scale emergencies, a single ambulance or mobile testing unit is rarely sufficient to handle the overwhelming influx of local demands, inevitably leading to prolonged response latencies and unmet public needs [11]. This critical gap reveals the fundamental requirement for effective dynamic allocation: the absolute necessity for *constraint-aware cooperative decision-making* under highly volatile *spatio-temporal dynamics*.

To bridge this gap, we introduce the Collaborative Public Resource Allocation (CPRA) problem, as shown in **Fig. 1**. CPRA explicitly models system-level coordination under practical constraints (e.g., limited service capacity) and spatio-temporally evolving demands. By integrating collaborative capacity awareness directly into the formulation, CPRA fundamentally shifts the paradigm from independent planning to cooperative coalition formation. This ensures that emerging high-demand areas are served by an optimal group of resources, effectively preventing both localized service overload and redundant deployments. However, compared to dynamic public resource allocation, CPRA involves a fundamentally more complex state and action space, presenting two core challenges:

1) Combinatorial complexity in collaborative scheduling. Enabling effective collaborative scheduling among all resources is highly challenging due to the explosion of the joint action and state spaces as the number of public resources scales. Traditional Multi-Agent Reinforcement Learning (MARL) methods often struggle in this highly constrained setting [14]. Specifically, when multiple resources collaborate to serve a shared high-demand region, it becomes notoriously difficult to accurately evaluate each individual resource's contribution to the global utility, which is known as the credit assignment problem [36]. Consequently, existing methods easily fall into suboptimal solutions, either failing to converge efficiently during training or yielding uncoordinated allocation strategies that inadvertently violate capacity constraints.

2)Extreme dynamics in spatio-temporal demands. In CPRA, public demands exhibit highly volatile, non-stationary, and complex spatio-temporal patterns [37]. Existing heuristic algorithms are inherently myopic, they passively react to immediate localized demands and fail to optimize long-term system utility. On the other hand, purely predictive spatio-temporal models, while capable of forecasting future trends, typically lack the mechanism to integrate these data-driven priors into the reinforcement learning environment [30] [7]. Therefore, they struggle to translate the extracted spatio-temporal dependencies into proactive, high-coverage decision-making. Bridging the gap between extracting complex spatio-temporal dynamics and directly utilizing them to guide collaborative dispatching remains a significant hurdle.

To address these challenges, we propose a new framework called Game-Theoretic Spatio-Temporal Reinforcement Learning (GSTRL) for solving CPRA. Targeting the first challenge, we prove that the CPRA problem can be formulated as a *Potential Game*, and then guide the reward learning rule for the system based on the potential function, providing theoretical guarantees for approximating the Nash equilibrium of this NP-hard problem. To tackle the second challenge, GSTRL integrates spatio-temporal feature extraction [19] and adaptive policy learning for multiple public resources and demands, enabling effective dispatch strategies under real-world constraints. Our major contributions can be summarized as follows:

- *A novel perspective from the game theory*: To the best of our knowledge, we provide the first game-theoretic formulation for CPRA, considering *agent collaboration* under service capacity constraints. By formulating it as a potential game with a constructed potential function to capture collaboration dynamics, we provide theoretical guarantees for accelerating convergence towards the Nash equilibrium.
- *A new framework for CPRA*: We propose GSTRL, a reinforcement learning (RL) framework to solve the CPRA problem. By extracting the spatio-temporal features of crowd flow distribution, adaptively fusing features from public resources and crowd flows, and incorporating masking mechanisms, GSTRL effectively makes decisions based on the system's *spatio-temporal dynamics* while strictly adhering to environmental constraints.
- *Comprehensive empirical evidence*: Extensive experiments on two real-world datasets show that GSTRL outperforms state-of-the-art baselines by up to 40%, demonstrating its superior performance on the CPRA problem across various parameter settings. Ablation and spatio-temporal dynamics experiments further highlight the role of its key components in enhancing learning effectiveness.

2 Overview

In this section, we give some preliminaries, then we formally define CPRA and prove its NP-hardness for the first time.

2.1 Preliminaries

Definition 1 (Location). *We uniformly partition the area of interest into N grid locations, denoted by $\mathcal{G} = \{g_i\}$.*

Definition 2 (Public Resource). *At any time interval T_t, the resource $\mathbf{m}_k$ is represented as a triple $\mathbf{m}_t = \left(l_k^t, e_k^t, p_k^t\right)$, where $l_k^t \in \mathcal{G}$ denotes its current location, e_k^t represents its remaining energy, and p_k^t indicates its service capacity.*

Definition 3 (Energy Cost). *When a resource moves from g_i to g_j, the energy cost is denoted by $c_{i,j}$. We treat all resources of the same type, making cost solely dependent on Euclidean distance [27].*

Definition 4 (Service Capacity). *The service capacity of a resource k at time t, denoted by p_k^t, measures the crowd flow that the resource can serve at time t. We assume that the service capacity is identical for all resources and is updated at each time step, such that $p_k^{t+1} = p_k^t = p_{k+1}^t$.*

Definition 5 (Resource Depot). *A resource depot $g_\omega \in \mathcal{G}$ is static. All public resources must start and end their journey at this resource depot during the scheduling process.*

Definition 6 (Crowd Coverage). *The crowd coverage during the time interval T_t is defined as the total number of people successfully served by all public resources across the entire area. Let $u_{i,k}^t$ indicate the presence of public resource m_k in grid cell i at time t, and let λ_i^t represent the remaining crowd flow in grid i at time t. Consequently, the crowd coverage during the time interval T_t can be expressed as:*

$$C_t = \sum_{k\in\mathcal{K}} \sum_{g_i\in G} \left(u_{i,k}^t \cdot \min\left\{\lambda_i^t, p_k^t\right\}\right). \tag{1}$$

2.2 Problem Statement

Given a set of resources $\mathcal{K} = \{k = 1, \cdots, k_{\max}\}$, a set of time intervals $\mathcal{T} = \{T_t \mid t = 1, \cdots, t_{\max}\}$, an initial energy E, and the location of the resource depot g_ω, our objective is to find a scheduling strategy that maximizes long-term coverage while adhering to energy limits and ensuring that all resources start and end their journeys at the resource depot. Since most resources operate in fixed positions, decision-making time dominates travel time, which is neglected.

Based on flow distributions, this optimization problem is formulated as an Integer Linear Programming (ILP) problem with respect to the decision variable $u_{i,k}^t$. To facilitate the solution, we introduce an auxiliary variable $x_{i,j,k}^t$, which indicates whether resource m_k travels from location g_i to location g_j at the beginning of T_t:

$$\sum_{g_h\in G} x_{h,i,k}^t = \sum_{g_j\in G} x_{i,j,k}^{t+1} = u_{i,k}^t, \forall g_i \in \mathcal{G}, T_t \in \mathcal{T}, k \in \mathcal{K}. \tag{2}$$

The formulation of CPRA is expressed as follows:

$$\mathcal{P}: \max_{\left\{u_{i,k}^t\right\}} \sum_{t\in\mathcal{T}} \sum_{k\in\mathcal{K}} \sum_{g_i\in\mathcal{G}} \left(u_{i,k}^t \cdot \min\left\{\lambda_i^t, p_k^t\right\}\right) \tag{3}$$

$$s.t.\ C_1: \sum_{k\in\mathcal{K}} \sum_{g_i\in\mathcal{G}\setminus\{g_\omega\}, g_j\in\mathcal{G}} x_{i,j,k}^t = 0, \tag{3a}$$

$$C_2: \sum_{k\in\mathcal{K}} \sum_{g_j\in\mathcal{G}} x_{\omega,j,k}^1 = \sum_{k\in\mathcal{K}} \sum_{g_i\in\mathcal{G}} x_{i,\omega,k}^{n+1} = k_{\max}, \tag{3b}$$

$$C_3: \sum_{k\in\mathcal{K}} u_{i,k}^t \le \frac{\sum_{g_i\in\mathcal{G}} \lambda_i^t}{p_k^t}, \forall g_i \in \mathcal{G}, T_t \in \mathcal{T}, \tag{3c}$$

$$C_4: \sum_{t\in\mathcal{T}\cup\{t_{n+1}\}} \sum_{g_i, g_j\in\mathcal{G}} c_{i,j} \cdot x_{i,j,k}^t \le E, \forall k \in \mathcal{K}, \tag{3d}$$

where constraint C_1 ensures all routes start from the resource depot g_ω. C_2 enforces that each resource starts and ends at g_ω, occupying a location at both T_1 and $T_{t_{\max}}$. C_3 limits the number of resources in g_i at time T_t, preventing trivial stationary solutions. C_4 ensures each resource has sufficient energy E to return to the depot.

Theorem 1. *The CPRA problem $\mathcal{P}$ is NP-hard.*

Proof. To prove that $\mathcal{P}$ is NP-hard, we reduce a NP-hard problem, the Multiple Knapsack Problem (MKP), to a case of $\mathcal{P}$.

Consider a special case of problem $\mathcal{P}$ where: a) There is a single time interval, i.e., $|T| = 1$. b) Each grid cell $g_i \in G$ contains exactly one unit of crowd flow λ_i^t. c) Each public resource $m_k \in K$ corresponds to a knapsack. d) The service capacity p_k^t of each public resource corresponds to the capacity of each knapsack. e) The decision variable $u_{i,k}^t$, which indicates whether the resource m_k serves the grid cell g_i, corresponds to placing a task in the knapsack.

Under these conditions, problem $\mathcal{P}$ reduces to selecting a subset of tasks to maximize the objective $\sum_{i\in G} \min\left\{\lambda_i^t, p_k^t\right\}$ while satisfying the capacity constraint $\sum_{i\in G} u_{i,k}^t \le p_k^t$. This mirrors the structure of the Multiple Knapsack Problem, where the goal is to maximize the value of selected items under capacity constraints.

Since the Multiple Knapsack Problem is NP-hard, and the special case of $\mathcal{P}$ is equivalent to MKP, it follows that $\mathcal{P}$ is at least as hard as the Multiple Knapsack Problem. Hence, $\mathcal{P}$ is NP-hard. □

We demonstrate that CPRA $\mathcal{P}$ is computationally intractable, underscoring the need for advanced algorithmic approaches.

3 A Game-Theoretic View on CPRA

To design an effective algorithm for the NP-hard CPRA problem, we analyze the relationship between individual agents and the overall system based on Potential game theory.

3.1 Potential Game Model

We treat each public resource as a player and the combination of resources and crowd flow as a system. The CPRA problem is modeled as a game $G = \{\mathcal{K}, \mathcal{S}, \{U_k\}_{k\in\mathcal{K}}\}$:

- $\mathcal{K}$ is the set of players (public resources).
- $\mathcal{S} = S_1 \times S_2 \times \cdots \times S_{k_{max}}$ represents the joint strategy space of the game, with S_k being the set of all possible strategies for public resource m_k.
- U_k is the utility function of public resource m_k .

Each strategy profile $\zeta \in \mathcal{S}$ is defined as $\zeta = (\zeta_1, \zeta_2, \ldots, \zeta_{k_{max}})$, which can also be expressed as $\zeta = (\zeta_k, \zeta_{-k})$, where ζ_{-k} represents the strategies of all resources except m_k. The goal of each resource is to maximize its utility $U_k(\zeta)$ based on the allocation strategy.

Definition 7 (Utility function of public resource m_k). *The utility of public resource m_k under strategy ζ is defined as the sum of the crowd coverage achieved at time t:*

$$U_k(\zeta) = \sum_{g_i\in\mathcal{G}} \left(u_{i,k}^t \cdot \min\left\{\lambda_i^t, p_k^t\right\}\right). \tag{4}$$

Definition 8 (Potential Game). *A game $G = \{\mathcal{K}, \mathcal{S}, \{U_k\}_{k\in\mathcal{K}}\}$ is a **potential game** if there exists a potential function $\Phi : \mathcal{S} \to \mathbb{R}$ such that for all $k \in \mathcal{K}$, all $\zeta_{-k} \in S_{-k}$, and all $\zeta_k, \zeta_k' \in S_k$,*

$$U_k(\zeta_k, \zeta_{-k}) - U_k(\zeta_k', \zeta_{-k}) = \Phi(\zeta_k, \zeta_{-k}) - \Phi(\zeta_k', \zeta_{-k}).$$

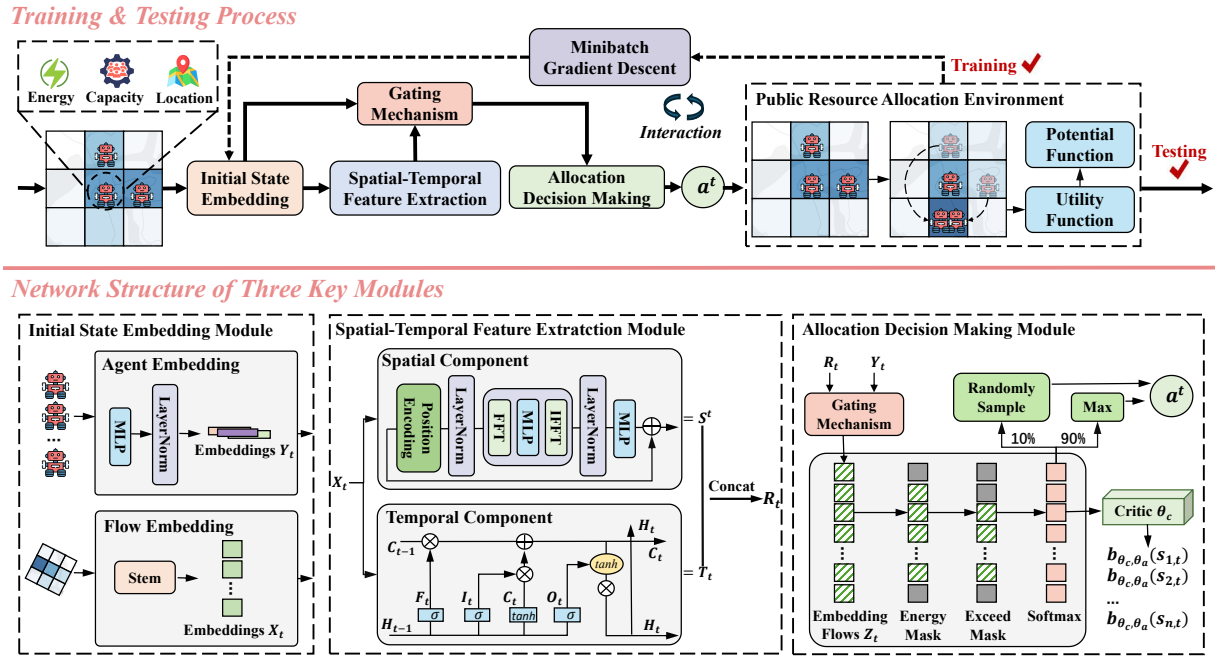

Figure 2: Overall architecture of the proposed GSTRL framework.

Theorem 2. *The game $G = \{\mathcal{K}, \mathcal{S}, \{U_k\}_{k\in\mathcal{K}}\}$ is a potential game with the potential function:*

$$\Phi(\zeta) = \sum_{k\in\mathcal{K}} \sum_{g_i\in\mathcal{G}} \left(u_{i,k}^t \cdot \min\left\{\lambda_i^t, p_k^t\right\}\right). \tag{5}$$

Proof. The potential function $\Phi(\zeta)$ is defined as the total crowd coverage achieved by all public resources:

$$\Phi(\zeta) = \sum_{k\in\mathcal{K}} U_k(\zeta) = \sum_{k\in\mathcal{K}} \sum_{g_i\in\mathcal{G}} \left(u_{i,k}^t \cdot \min\left\{\lambda_i^t, p_k^t\right\}\right).$$

Now, consider the change in the strategy of a single resource m_k from ζ_k to ζ_k', while all other strategies ζ_{-k} remain fixed. The resulting change in the potential function is:

$$\Phi(\zeta_k, \zeta_{-k}) - \Phi(\zeta_k', \zeta_{-k}) = U_k(\zeta_k, \zeta_{-k}) - U_k(\zeta_k', \zeta_{-k}).$$

This confirms that the difference in the potential function directly reflects the change in the utility of the public resource m_k, satisfying the conditions for a potential game. □

Theorem 3. *Every finite potential game has at least one pure-strategy Nash equilibrium.*

Proof. Since G is a finite potential game and the potential function $\Phi(\zeta)$ is bounded over the finite strategy space $\mathcal{S}$, there exists a strategy profile ζ^* that maximizes Φ:

$$\zeta^* = \arg\max_{\zeta\in\mathcal{S}} \Phi(\zeta).$$

At ζ^*, no resource m_k can change its strategy to increase its utility, as such a change would increase Φ, contradicting the minimality of $\Phi(\zeta^*)$. Therefore, ζ^* is a pure-strategy Nash equilibrium. □

3.2 Further Discussion

Guided by **Theorem 3**, we employ an RL algorithm to address CPRA. We adopt the potential function as the reward function, establishing theoretical guarantees for the algorithm's convergence and approximating the near-optimal solution.

For the CPRA problem, we further demonstrate that there exists no gap between the objective and the Nash equilibrium [4]. From a system-level perspective, this ensures that maximizing each player's utility inherently maximizes overall system performance [29], leading to no resource redundancy. From a decision-making perspective, this alignment implies that information sharing does not introduce bias [5]. Motivated by these insights, we adopt a centralized decision-making framework in which the entire system learns an optimal strategy under a unified reward structure.

4 Methodology

4.1 Framework Overview

Given a distribution of crowd flows and resources, our goal is to maximize long-term crowd coverage by placing resources in the optimal locations at each time step. **Fig. 2** illustrates the framework, which mainly consists of three modules:

- **Initial State Embedding**: This module is divided into two components: crowd flow embedding and agent embedding. First, we flatten the predicted flows and use an MLP layer to process the initial state of public resources [33]. Then, a Stem and LayerNorm component is applied to extract high-level features (denoted as $\mathbf{X_t}$ and $\mathbf{Y_t}$), where $t \in \{t_1, t_2, \ldots, t_{\max}\}$.
- **Spatio-Temporal Feature Extraction**: Starting with the crowd flow embedding output X_t, this module uses three Bidirectional LSTMs and a Fourier Neural Operator to capture the spatial and

temporal dynamics of X_t. The spatial feature S^t and temporal feature T_t are then concatenated as R_t for further processing [3].

- **Allocation Decision Making**: We first apply a gating mechanism to integrate the spatio-temporal embedding of flows R_t with the resources embedding Y_t, followed by a mask criterion to ensure the strategies comply with practical constraints. A softmax layer is then used to calculate the score for each action Q_k, which is executed accordingly.

During training, agents interact with the environment and get experience. We sample mini-batches from the experience and refine the strategy using mini-batch gradient descent until convergence [25].

4.2 CPRA Modeled as Markov Decision Process

To capture long-term effects and avoid myopic decisions, we formulate the resource allocation process as a Markov Decision Process (MDP) [34]. It consists of four components: states, actions, transitions, and rewards.

a) State. We denote the state at time t by $s_t = (\mathbf{d}_t, \mathcal{M}_t)$, where $\mathbf{d}_t = \{\lambda_i^t\}_{g_i \in \mathcal{G}}$ is the set of predicted flows in each grid, and $\mathcal{M}_t = \{(l_k^t, e_k^t, p_k^t)\}_{k=1}^{k_{\max}}$ is the set of resources.

b) Action. At time step t, the action $a_t = \left(g_i^t, \mathbf{m}_k^t\right)$ means that we allocate resource $\mathbf{m}_k$ to grid g_i. We adopt a sequential assignment strategy within each time step. Specifically, once a resource is allocated, the remaining crowd flow of the target grid is immediately updated to reflect the fulfilled demand before executing the allocation for the subsequent resources.

c) Transitions. Once an action a_t is taken, the system deterministically transitions to s_{t+1} based on s_t and a_t.

d) Reward. Based on potential game theory, we adopt the potential function $\Phi(\zeta)$ as the reward function, which is $r_t = \sum_{k \in \mathcal{K}} \sum_{g_i \in \mathcal{G}} \left(u_{i,k}^t \cdot \min\{\lambda_i^t, p_k^t\}\right)$ at time t. By using a global optimization function instead of individual utility functions for each resource, GSTRL can capture the agent collaboration dynamics more effectively so as to accelerate convergence toward a near-Nash equilibrium.

4.3 Initial State Embedding

We design an Initial State Embedding Module to convert predicted crowd flows and resource states into high-level features for subsequent learning. This involves two parts: Crowd Flow Embedding and Agent Embedding.

4.3.1 Flow Embedding. Input data reflects the dynamics of crowd flows [20]. After flattening, we feed the data into a stem module with 1×1 convolutions to extract a feature map $\mathbf{X_t}$.

4.3.2 Agent Embedding. Each resource is denoted by (l_k^t, e_k^t, p_k^t). We project the 3-dimensional vector into a higher-dimensional space via a fully connected layer, followed by LayerNorm. We denote the embedding resulting from each public resource by $\mathbf{Y_t}$.

4.4 Spatio-Temporal Feature Extraction

Our module refines the features from the Crowd Flow Embedding via two components: a Temporal and a Spatial Component.

4.4.1 Temporal Component. We adopt a BiLSTM to capture long-term temporal patterns in crowd flow. Unlike unidirectional models [18], BiLSTM processes each time step in both forward and backward directions. This process yields hidden states $\overrightarrow{\mathbf{H}}_t$ and $\overleftarrow{\mathbf{H}}_t$. Concatenating them produces a comprehensive temporal embedding $\mathbf{T}_t$, which helps avoid suboptimal dispatching [8, 9].

4.4.2 Spatial Component. We adopt FNO [15] to capture global correlations in the frequency domain with reduced complexity. Let $\mathbf{X}_t \in \mathbb{R}^{N \times E}$ denote the spatial feature map at time t. After position encoding and LayerNorm, we transform $\mathbf{X}_t$ via FFT, multiply by a learnable weight matrix $\hat{\mathbf{W}}$, and apply the inverse FFT:

$$\mathbf{s}_i^t = \mathrm{IFFT}\big(\hat{\mathbf{W}} \cdot \mathrm{FFT}(\mathbf{X}_t)\big)_i, \tag{6}$$

where i indexes the grids, and the structure of $\hat{W}$ reduces parameter count. This yields spatial embeddings $\mathbf{S}^t = \{\mathbf{s}_1^t, \ldots, \mathbf{s}_N^t\}$ in $O(NE \log N)$ time, where N is the number of grids and E the embedding dimension.

4.4.3 Output. We concatenate $\mathbf{T}_t$ and $\mathbf{S}^t$ to form the spatio-temporal embedding $\mathbf{R}_t = \mathrm{Concat}(\mathbf{T}_t, \mathbf{S}^t)$, which captures both time-series trends and cross-grid interactions. $\mathbf{R}_t$ is then passed to the Allocation Decision Making Module, enabling resource allocation under capacity constraints and fluctuating demands [16] [35].

4.5 Allocation Decision Making

This module integrates the spatio-temporal embedding R_t and the resource embedding Y_t via a gating mechanism:

$$\mathbf{J}_t = \sigma\big(\mathbf{W}_j \cdot [\,\mathbf{R}_t,\, \mathbf{Y}_t\,] + \mathbf{b}_j\big), \mathbf{Z}_t = \mathbf{J}_t \cdot \mathbf{R}_t + \big(1 - \mathbf{J}_t\big) \cdot \mathbf{Y}_t, \tag{7}$$

where $\mathbf{W}_j$ and $\mathbf{b}_j$ are learnable parameters, and $\sigma(\cdot)$ is the sigmoid function. The resulting $\mathbf{Z}_t$ represents the overall feature for the system for subsequent decision-making.

When making decisions, the four constraints C_1–C_4 must be satisfied. Although C_1 and C_2 are effective by design, C_3 and C_4 require restricting locations. We thus apply masking mechanisms to $\mathbf{Z}_t$, yielding a masked representation $\overline{\mathbf{Z}_t}$.

Finally, We decode $\overline{\mathbf{Z}_t}$ through a softmax layer $\mathbf{Q}_k = \mathrm{Softmax}\big(\overline{\mathbf{Z}_t}\big)$. During training, the agent chooses the highest-probability action 90% of the time and randomly explores 10% of the time. The processed state is also passed to the Critic $b_{\theta_c,\theta_a}(s_{i,t})$ for estimation.

4.6 Training Process

To learn an effective allocation strategy, we adopt an Actor-Critic training framework [23]. The procedural flow of the training process is depicted in **Algorithm 1**.

4.6.1 Actor-Critic Overview. Our Actor-Critic method comprises two main components:

- **Actor (Policy Network):** The policy network, denoted as π_{θ_a}, governs resource allocation decisions of GSTRL. The Initial State Embedding Module, Spatio-Temporal Feature Extraction Module, and Allocation Decision Module are essential components of the policy network.
- **Critic (Value Network):** The value network approximates the value function $V_{\pi_{\theta_a}}(s_t)$, which evaluates state quality and guides the Actor towards higher-value actions. Implemented as a feed-forward network with a regression layer and parameterized by θ_c, the value network is trained alongside the Actor network but is not used after training.

Algorithm 1 GSTRL Algorithm

Input: Input sets of each public resource's state M_t, and sets of predicted crowd flow's amount G_t. The actor network with parameters θ_a, and the critic network with parameters θ_c.
Output: Trained actor network parameter θ_a^*.

```
Randomly initialize θ_a and θ_c.
Initialize replay buffer B.
for episodes k = 0, 1, 2, ..., EPI_max do
    Generate the action a_t according to π_{θ_α}.
    Execute the action a_t and obtain s_t and r_t.
    Store (s_t, a_t, s_{t+1}, r_t) into replay buffer B.
    Randomly sample M samples from the replay buffer of length N.
    for t = 0, 1, 2, ..., t_max do
        for k = 0, 1, 2, ..., k_max do
            Calculate the expected reward.
            Compute the Q-value Q_{π_{θ_a}}(s_t, a_t) based on the samples.
            Compute Advantage L(ω) based on Eq. (11).
        end for
    end for
    Compute the loss of critic L_{θ_c} based on Eq. (12).
    Compute the loss of policy L_{θ_a} based on Eq. (13).
    Calculate the joint loss L_{θ_ac} based on Eq. (14).
    Update the parameters of the actor-critic model.
end for
return θ_a^* = θ_a.
```

By integrating these two components, GSTRL dynamically optimizes allocation strategies using mini-batch gradient descent.

4.6.2 Value Functions and Advantage. Given a policy π_{θ_a}, we define the state-action function $Q_{\pi_{\theta_a}}(s_t, a_t)$ and the state-value function $V_{\pi_{\theta_a}}(s_t)$ as:

$$Q_{\pi_{\theta_a}}(s_t, a_t) = \mathbb{E}_{\pi_{\theta_a}}\left[r_t \mid s = s_t, a = a_t\right], \tag{8}$$

$$V_{\pi_{\theta_a}}(s_t) = \mathbb{E}_{a_t \sim \pi_{\theta_a}(s_t)}\left[Q_{\pi_{\theta_a}}(s_t, a_t)\right]. \tag{9}$$

The estimated value function is subsequently used to compute the advantage function $A_{\pi_{\theta_a}}(s_t, a_t)$:

$$A_{\pi_{\theta_a}}(s_t, a_t) = Q_{\pi_{\theta_a}}(s_t, a_t) - V_{\pi_{\theta_a}}(s_t). \tag{10}$$

The advantage function reflects how much better (or worse) taking action a_t in state s_t is compared to the average action. To expedite computation, we approximate it as follows [12]:

$$A_{\pi_{\theta_a}}(s_t, a_t) \approx r_t + \gamma V_{\pi_{\theta_a}}(s_{t+1}) - V_{\pi_{\theta_a}}(s_t). \tag{11}$$

4.6.3 Loss Functions. The Critic is optimized via:

$$\mathcal{L}_{\theta_c} = \frac{1}{N}\sum_{i=1}^{N}\sum_{t\in\mathcal{T}} \text{smoothL1}\big(b_{\theta_c,\theta_a}(s_{i,t}) - r_{i,t}\big), \tag{12}$$

where the predicted value $b_{\theta_c,\theta_a}(s_{i,t})$ is defined as a baseline to evaluate the value function. The Actor is trained to maximize the expected advantage:

$$\mathcal{L}_{\theta_a} = \frac{1}{N}\sum_{i=1}^{N}\sum_{t\in\mathcal{T}} \log \pi_{\theta_a}(a_{i,t} \mid s_{i,t})\, A_{\pi_{\theta_a}}(s_{i,t}, a_{i,t}), \tag{13}$$

where N denotes the number of sampled trajectories, π_{θ_a} is the policy, and $A_{\pi_{\theta_a}}(\cdot)$ is the advantage function. We train the Actor and Critic jointly by combining their losses:

$$\mathcal{L}_{\theta_{ac}} = \alpha\,\mathcal{L}_{\theta_a} + \beta\,\mathcal{L}_{\theta_c}, \tag{14}$$

where α and β are hyperparameters controlling the trade-off between policy optimization and value estimation.

5 Experiments

In this section, we conduct experiments to address the following research questions:

- **RQ1:** How does GSTRL perform compared to heuristic algorithms in terms of maximizing crowd coverage?
- **RQ2:** How do the key components of GSTRL contribute to learning effective resource allocation strategies?
- **RQ3:** How do we choose the component for extracting spatio-temporal dynamics from crowd flow?
- **RQ4:** How do the different settings of each hyperparameter impact the overall performance of GSTRL?
- **RQ5:**How does the trained GSTRL framework devise practical scheduling strategies in real-world scenarios?

5.1 Experimental Settings

5.1.1 Datasets. We utilize two datasets, Happy Valley and TaxiBJ, for our capacitated resource scheduling task.

- **Happy Valley**: This dataset, obtained from Ruan et al.[27], provides hourly gridded crowd flow observations for Beijing Happy Valley, a theme park covering $1.25 \times 10^5 m$. The data spans from January 1, 2018, to October 31, 2018. We divide the park into a 51×108 grid for fine-grained analysis. The dataset is split into training (60%), validation (20%), and test (20%) sets.
- **TaxiBJ**: This dataset, published by Zhang et al.[43], records taxi flow data across Beijing from July 1, 2013, to March 31, 2016. We represent the city as a 32×32 grid to model the spatial distribution of taxi movements. The dataset is partitioned into 70% for training, 20% for validation, and 10% for testing.

5.1.2 Baselines & Evaluation Metrics. To evaluate the CPRA problem, we implement several heuristic algorithms as baselines. Besides, as we are the first to study CPRA, we adapt state-of-the-art RL algorithms from related domains for a comprehensive comparison.
Heuristic Algorithms:

- **Static:** An approach that allocates resources to the location with the highest crowd flow within the resources' capacity at the first time step, keeping the positions fixed thereafter.
- **B&B [13]:** Utilizes a branch-and-bound approach with the PuLP ILP solver to determine the optimal solution for the ILP problem.
- **MYOPIC [27]:** An aggressive allocation strategy that makes scheduling decisions based solely on the predicted crowd flow for the next time step, using a greedy approach. Resources execute local optimal strategies within the initial energy.
- **EADS [27]:** A two-step algorithm that schedules resources to maximize crowd coverage within agents' service times, serving as the state-of-the-art baseline among heuristic algorithms for this task. The selection criterion is adjusted to maximize the minimum

	Method	Algorithms	#Param(K)	E=30		E=40		E=50		E=60	
				ADCC ↑	Δ ↑	ADCC ↑	Δ ↑	ADCC ↑	Δ ↑	ADCC ↑	Δ ↑
Happy Valley	Heuristic	Static	-	1,566	-14%	1,673	-20%	2,076	-17%	2,106	-13%
		BB	-	1,717	-6%	1,686	-20%	2,128	-15%	2,235	-8%
		MYOPIC	-	1,756	-4%	1,789	-15%	2,206	-12%	2,289	-6%
		EADS	-	1,831	-	2,096	-	2,493	-	2,432	-
	RL	REINFORCE	211	$1{,}936_{\pm 8}$	6%	$2{,}181_{\pm 3}$	4%	$2{,}239_{\pm 5}$	-10%	$2{,}103_{\pm 10}$	-14%
		PPO	220	$1{,}956_{\pm 5}$	7%	$2{,}203_{\pm 2}$	5%	$2{,}336_{\pm 3}$	-6%	$2{,}167_{\pm 7}$	-11%
		SAC	220	$1{,}963_{\pm 5}$	7%	$2{,}365_{\pm 3}$	13%	$2{,}396_{\pm 3}$	-4%	$2{,}196_{\pm 6}$	-14%
		HEM	20	$1{,}657_{\pm 5}$	-10%	$1{,}896_{\pm 6}$	-10%	$2{,}136_{\pm 10}$	-14%	$2{,}176_{\pm 6}$	-11%
	MARL	QMIX	3,976	$1{,}958_{\pm 12}$	7%	$2{,}209_{\pm 10}$	5%	$2{,}306_{\pm 5}$	-8%	$2{,}303_{\pm 8}$	-5%
		DAE	5,743	$1{,}982_{\pm 12}$	8%	$2{,}218_{\pm 8}$	6%	$2{,}325_{\pm 6}$	-7%	$2{,}316_{\pm 10}$	-5%
		GSTRL (ours)	**216**	**2,298**$_{\pm 16}$	**26%**	**2,530**$_{\pm 5}$	**20%**	**2,585**$_{\pm 8}$	**4%**	**2,517**$_{\pm 23}$	**3%**

	Method	Algorithms	#Param(K)	E=30		E=40		E=50		E=60	
				ADCC ↑	Δ ↑	ADCC ↑	Δ ↑	ADCC ↑	Δ ↑	ADCC ↑	Δ ↑
TaxiBJ	Heuristic	Static	-	19.3	-92%	19.3	-92%	19.3	-92%	19.3	-92%
		MYOPIC	-	19.3	-92%	19.3	-92%	19.3	-92%	19.3	-92%
		EADS	-	258	-	242	-	238	-	242	-
	RL	REINFORCE	669	$223_{\pm 6}$	-14%	$230_{\pm 2}$	-15%	$285_{\pm 5}$	20%	$296_{\pm 6}$	22%
		PPO	691	$229_{\pm 3}$	-11%	$241_{\pm 2}$	-0.4%	$293_{\pm 2}$	23%	$302_{\pm 6}$	25%
		SAC	697	$230_{\pm 5}$	-11%	$246_{\pm 2}$	2%	$295_{\pm 3}$	24%	$305_{\pm 6}$	26%
		HEM	60	$190_{\pm 3}$	-26%	$197_{\pm 2}$	-19%	$256_{\pm 6}$	8%	$273_{\pm 5}$	13%
	MARL	QMIX	8,675	$235_{\pm 8}$	-9%	$246_{\pm 5}$	2%	$297_{\pm 7}$	25%	$306_{\pm 10}$	26%
		DAE	12,256	$243_{\pm 9}$	-6%	$253_{\pm 6}$	5%	$305_{\pm 8}$	28%	$316_{\pm 12}$	31%
		GSTRL (ours)	**685**	**259**$_{\pm 2}$	**0.4%**	**260**$_{\pm 5}$	**7%**	**313**$_{\pm 2}$	**32%**	**338**$_{\pm 6}$	**40%**

Table 1: Model comparison on the Happy Valley and TaxiBJ datasets. The Δ represents the improvements in ADCC compared to the EADS approach. Bold and underlined digits are the best and EADS approach values.

of crowd flow and capacity, with the crowd flow updated after each agent's service.

RL Algorithms:

- **REINFORCE [41]:** By mapping the extracted spatio-temporal features of crowd demands and resource states into a probability distribution over the dispatching action space, it updates the policy parameters using the cumulative rewards, serving as a fundamental baseline for policy-based methods.
- **PPO [46]:** As an advanced on-policy Actor-Critic framework, this baseline utilizes a clipped surrogate objective to constrain the policy update step size. It estimates the long-term state values of public resources and adaptively updates scheduling policies to effectively cope with volatile, non-stationary crowd fluctuations.
- **SAC [45]:** This baseline implements an off-policy maximum entropy Actor-Critic algorithm, encourages robust exploration within the vast and complex joint action space of CPRA, effectively preventing the policy from trapping into local optima while maintaining high sample efficiency during training.
- **HEM [32]:** This baseline leverages a hierarchical reinforcement learning framework. It decomposes the allocation space into a macro-level manager for regional resource targeting and a micro-level worker for capacity-aware resource dispatching, striking a balance between global coordination and localized efficiency.

MARL Algorithms:

- **QMIX [26]:** By imposing a monotonicity constraint on the relationship between individual utilities and the total action-value function, QMIX allows each public resource to make localized dispatching decisions while guaranteeing that their joint actions maximize the global potential function, effectively addressing the multi-agent credit assignment problem.
- **DAE [14]:** By explicitly decomposing the joint action space and modeling the localized spatio-temporal dependencies among neighboring agents, DAE successfully handles the combinatorial explosion of the resources' action variables under strict service capacity constraints, thereby preventing resource conflicts or redundant clustering in high-demand zones.

Evaluation Metrics: We follow [27] and use the average daily crowd coverage (ADCC) to assess model performance. Each method is run 5 times, and we report the average ADCC value along with the variance. The notation Δ indicates the relative improvement of ADCC over the EADS approach under varying initial energy E. ADCC is calculated as follows:

$$ADCC = \frac{\sum_{d \in \mathcal{D}} \sum_{t \in \mathcal{T}} C_t}{|\mathcal{D}|}. \tag{15}$$

5.1.3 Implementation Details. The baselines are implemented in Python, with training using PyTorch 2.1.3 on a single NVIDIA GeForce RTX 3090 GPU. The Adam optimizer is used with a batch size of 16, an initial learning rate of 0.05, and a hidden size of 32.

Each agent's capacity is set to 10. The resource number is set to 10 and the energy is explored within the range of 30 to 60.

5.2 Model Comparison (RQ1)

In this section, we compare the performance of GSTRL with existing baselines in terms of ADCC, using two real-world datasets as shown in **Table 1**. Our model consistently outperforms all baselines under varying initial energy budgets E, achieving an improvement of 3% to 26% in ADCC on the Happy Valley dataset, and 0.4% to 40% on the TaxiBJ dataset. This significant margin demonstrates the effectiveness of our method in tackling the highly constrained CPRA problem across scenarios of different scales. Specifically, while traditional heuristic and RL baselines often struggle with the joint action space and constraint limits, GSTRL successfully overcomes these hurdles. By incorporating potential game theory, our framework explicitly aligns individual resource dispatching behaviors with the system coverage, thereby avoiding uncoordinated assignments and consistently approaching the near-optimum.

As the initial energy E increases, we observe a upward trend in ADCC for all algorithms across both datasets. A higher energy budget inherently expands the feasible exploration space for the public resources. With relaxed energy constraints, agents are no longer restricted to purely localized servicing, instead, they can execute longer-range cooperative dispatching to collectively serve distant high-demand clusters without the immediate risk of premature battery depletion. Consequently, this enlarged action space allows both heuristic and RL methods to discover more flexible allocation solutions that yield higher overall utility.

Additionally, for the Happy Valley dataset, we notice that when E reaches a certain threshold (e.g., $E = 50$), the increase in ADCC noticeably slows down, and the performance gap between GSTRL and other baselines begins to narrow. This phenomenon can be attributed to the limited spatial scope of the Happy Valley region. Because the geographical bounding box is relatively small, an energy budget of $E = 50$ is highly abundant, meaning most individual resources already possess sufficient mobility to traverse and cover almost the entire area. In such states, the spatial and energy constraints of CPRA are significantly weakened, allowing even suboptimal baselines to achieve satisfactory coverage, which inevitably leads to a performance saturation.

In contrast, the dynamics within the TaxiBJ dataset are markedly different. While most RL methods continue to show significant improvement as E increases, heuristic algorithms perform poorly, notably, MYOPIC completely degenerates into the Static baseline, and EADS's performance paradoxically declines with higher energy budgets. This divergence is fundamentally due to the larger spatial scale and greater spatio-temporal complexity of the TaxiBJ dataset. In such a vast and non-stationary environment, heuristic algorithms lack predictive foresight and easily fall into local optima, such as redundantly dispatching multiple resources to a single hotspot while neglecting future demands elsewhere. Conversely, RL methods, particularly GSTRL with its advanced spatio-temporal feature extraction module, can effectively anticipate complex future flow shifts. This allows GSTRL to proactively and strategically distribute resources across the complex grid, maintaining superior exploration strategies and maximizing long-term coverage.

Variant	Happy Valley			TaxiBJ		
	E=30	E=40	E=50	E=30	E=40	E=50
w/o AE	2,016	2,021	2,121	217	219	257
w/o S	1,272	1,372	1,385	139	141	167
w/o T	2,158	2,216	2,300	227	228	273
w/o PF	1,982	2,046	2,165	225	213	262
w/o EM	136	368	551	89	106	130
GSTRL	**2,298**	**2,530**	**2,585**	**259**	**260**	**313**

Table 2: Ablation Study Results.

5.3 Ablation Study (RQ2)

To comprehensively assess the impact of each component on the performance of GSTRL, we perform ablation studies on two real-world datasets. These ablations validate our claims regarding the necessity of modeling spatio-temporal dynamics, reward alignment, and strict capacity constraints.

5.3.1 Effects of Feature Learning. To evaluate the effectiveness of the modules responsible for representation learning, we conduct ablation studies by independently removing three key components: a) **Agent Embedding (w/o AE)**: the module that encodes agent characteristics (e.g., remaining energy, current location, and available service capacity) into the state representation. Without this, agents become entirely blind to their physical limitations and those of their peers, leading to uncoordinated assignments where resources with insufficient battery or depleted capacity are mistakenly dispatched to demanding regions. b) **Spatial Component (w/o S)**: the spatial feature extractor responsible for modeling the geographical correlations of crowd flows. Without this component, the model treats each grid independently, failing to capture the cross-regional influence of demands and resulting in spatially myopic deployments. c) **Temporal Component (w/o T)**: the module that captures the temporal dynamics of crowd flows. Its absence prevents the model from anticipating future demand trends, causing resources to passively react to dispersing crowds rather than proactively moving to emerging hotspots. As shown in **Table 2**, the absence of any single component leads to a severe decline in overall performance, confirming their collective necessity in learning robust spatio-temporal representations for both the environment and the resources.

5.3.2 Effects of Potential Function (w/o PF). We perform the ablation of the potential function by modifying the reward design, fundamentally changing it from our constructed potential-based reward to a naive maximum utility among all agents. As shown in **Table 2**, removing this component leads to a dramatic performance drop. This explicitly highlights the critical role of the potential function in solving the multi-agent credit assignment problem. By breaking the theoretical properties of the potential game, individual agent objectives are no longer perfectly aligned with the global system coverage. Consequently, the system degenerates into selfish competition, where resources greedily cluster around highly rewarding localized hotspots, while neglecting other regions, ultimately failing to approximate the Nash Equilibrium.

5.3.3 Effects of Exceed Mask (w/o EM). The exceed mask is specifically designed as a regularizer derived directly from the service capacity constraints of CPRA. It actively prevents GSTRL from converging to suboptimal solutions, where a overwhelmingly large

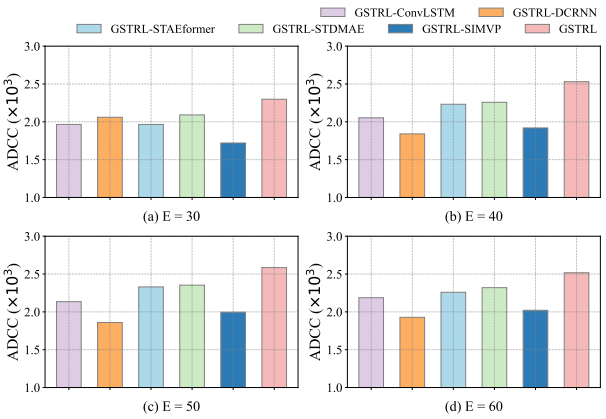

Figure 3: ADCC on different GSTRL variants.

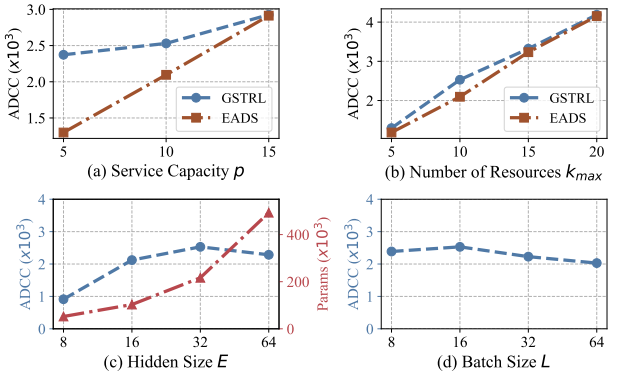

Figure 4: ADCC on different hidden sizes and batch sizes.

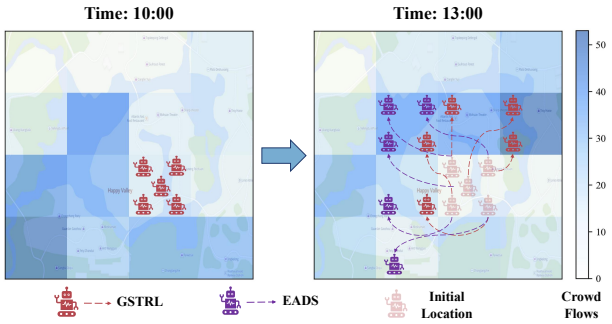

Figure 5: Strategies visualization of GSTRL compared with EADS in Happy Valley dataset.

number of agents needlessly crowd into a single grid merely due to its high crowd flow density. As shown in **Table 2**, removing the exceed mask leads to a marked decrease in performance and training stability. The mask effectively serves as an inductive bias, ensuring that the agents form capacity-aware coalitions without wasting service availability on over-supplied regions.

5.4 Spatio-Temporal Dynamics Analysis (RQ3)

To evaluate the ability of the spatial-temporal feature extraction module, we design five GSTRL variants by replacing this module. These include state-of-the-art models for spatio-temporal prediction, such as SIMVP [30], which excels on grid-based data, and STDMAE [7], which is designed for graph-structured data.

As shown in **Fig. 3**, GSTRL outperforms all variants across different energies. We attribute this to two factors: a) Data is grid-based, and adapting graph-based methods requires constructing an adjacency matrix, which can distort the characteristics of the grid. b) Compared to SIMVP, which extracts spatio-temporal features holistically, our module aggregates features at each time step. Given the structurally simple, yet time-varying flow, this aggregation proves to be more effective, despite relying on classical components.

5.5 Hyperparameters Study (RQ4)

In this subsection, we evaluate the effectiveness of GSTRL using the Happy Valley dataset under different hyperparameters, which consist of model parameters and network parameters. The initial energy is set to 40. EADS, the state-of-the-art baseline for this dataset, is used for comparison.

5.5.1 Service Capacity. **Fig. 4 (a)** illustrates how ADCC varies with different service capacities. As the service capacity increases, the performance gap has narrowed. However, when the service capacity is low, GSTRL significantly outperforms EADS, highlighting its advantage in multi-agent collaboration.

5.5.2 Number of Resources. **Fig. 4 (b)** presents the ADCC for different numbers of resources. As shown, GSTRL outperforms EADS across all configurations. Furthermore, increasing the number leads to greater ADCC, as more locations can be served.

5.5.3 Hidden Size and Batch Size. We investigate the impact of hidden size and batch size with default settings. **Fig. 4 (c)** reveals that a small hidden size reduces performance, and a hidden size of 64 achieves comparable performance but incurs higher cost.

In **Fig. 4 (d)**, we examine the impact of batch size. The trend shows that increasing the batch size from 8 to 16 leads to improved performance, while increasing from 16 to 32 causes a slight decline, which suggests that a batch size of 16 may be optimal. The curve exhibits a smooth trend, indicating that GSTRL is insensitive to variations in batch size, highlighting its strong robustness.

5.6 Case Study (RQ5)

We perform the allocation and visualize the process from 10:00 AM to 1:00 PM on September 13, 2018. To provide a more intuitive view, we merge the grid cells into a 4x4 grid with unequal grid dimensions and plot a heat map based on the maximum crowd flow within the merged cells. The study is conducted with 5 resources, a service capacity of 10, and an energy limit of 40. We compare GSTRL with the state-of-the-art EADS baseline on the Happy Valley dataset.

As shown in **Fig. 5**, the process begins at 10:00 AM with five resources from the resource depot. By 1:00 PM, resources allocated using GSTRL are strategically placed in areas of higher crowd density compared to those allocated with EADS. This superiority stems from GSTRL's proactive coordination mechanism: while EADS merely reacts to immediate localized demands and occasionally clusters redundantly in single hotspots, GSTRL leverages its spatio-temporal module to anticipate crowd movements. Governed by the potential game formulation, the resources collaboratively distribute

themselves across multiple high-demand centers, maximizing coverage under their capacity without exhausting their energy limits. In addition, we developed a CPRA system and visualized the deployment of GSTRL on the TaxiBJ dataset, as detailed in **Appendix**.

6 Related Work

Dynamic Resource Allocation. Dynamic resource allocation has been studied across various urban applications, such as ambulance deployment [11], electric bus scheduling [38], and disaster response [24]. Early studies primarily relied on exact optimization or heuristic algorithms [2, 44], which offer foundational solutions but excel in static environments. As urban dynamics intensify, these methods fail to adequately capture the rapid fluctuations of demands. Although recent approaches [27] incorporate real-time data to improve scheduling efficiency, they typically optimize resource movements in an isolated manner and lack foresight regarding future rewards. Crucially, real-world public resources have strictly limited service capacities. When crowd flows exhibit extreme spatio-temporal heterogeneity, isolated dispatches inevitably lead to localized service failures or redundant deployments [1]. To address this gap, our work introduces the CPRA problem, shifting the paradigm from independent planning to collaborative allocation.

Reinforcement Learning for Linear Programming. Resource allocation problems are traditionally formulated as Integer Linear Programming (ILP) or combinatorial optimization tasks. However, due to their NP-hard nature, classical LP solvers suffer from prohibitive computational costs in large-scale scenarios. Consequently, RL has emerged as a powerful alternative for approximating solutions to these programming problems by maximizing long-term discounted returns. Algorithms such as SAC [45] and hierarchical frameworks like HEM [32] have been widely applied to dynamic environments. Despite their promise, directly applying standard RL to the ILP of collaborative allocation faces severe scalability bottlenecks. The service capacity constraints and the necessity for multi-agent coordination create a highly complex state-action space, causing existing RL methods to frequently converge to suboptimal solutions. To overcome these challenges, our GSTRL framework bridges mathematical programming and RL by employing potential game theory. By aligning the system's objective with individual agent rewards through a potential function, we guarantee theoretical convergence towards a near-optimal Nash equilibrium.

Crowd Flow Prediction. Accurate crowd flow prediction is a fundamental prerequisite for proactive urban resource management, requiring sophisticated modeling of both complex spatial correlations and non-stationary temporal dependencies. While early methods struggled to capture cross-region influences, modern deep learning approaches [30, 40, 42] have significantly improved prediction accuracy. However, most existing works treats flow prediction as an isolated, standalone objective, rather than seamlessly integrating it with downstream decision-making tasks like dynamic resource allocation. In the context of CPRA, merely forecasting values is insufficient, the multi-agent system must dynamically react to these evolving distributions under capacity limits. To bridge this gap, our framework focuses on deep spatio-temporal feature extraction. By fusing the extracted spatio-temporal representations of crowd flows with the status of resources, GSTRL effectively translates predictive insights into constraint-aware allocation strategies.

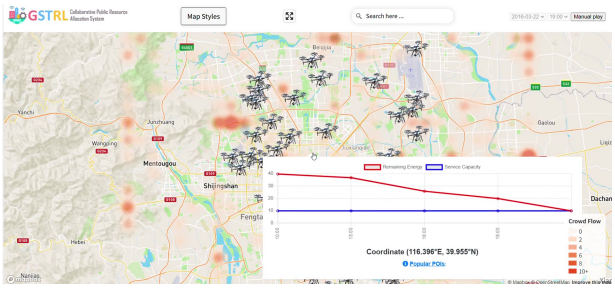

Figure 6: Web platform of CPRA system.

7 Conclusion and Future Work

In this paper, we introduce the CPRA problem and model it as a potential game with a potential function to align individual utilities with global outcomes. We propose a GSTRL framework to solve this problem, which incorporates initial state embedding, spatio-temporal feature extraction, and related modules to capture the dynamics of the system. Extensive experiments on two real-world datasets demonstrate the superiority of GSTRL over baselines across varying parameter settings and the positive impact of its key components. In future work, we aim to optimize energy consumption and explore more on-policy learning scenarios jointly.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (No. 62402414, No. 62306033). This work is supported by the Guangdong Basic and Applied Basic Research Foundation (No. 2025A1515011994), Guangdong Provincial Project 2025D03J0014, Guangzhou Municipal Science and Technology Project (No. 2023A03-J0011), the Guangzhou Industrial Information and Intelligent Key Laboratory Project (No. 2024A03J0628), and Guangdong Provincial Key Lab of Integrated Communication, Sensing and Computation for Ubiquitous Internet of Things (No. 2023B1212010007). This work is also supported by Tianjin Natural Science Foundation Project (No. 25JCYBJC01540).

A Real-World Deployment Visualization

To provide an intuitive view of the behavior of GSTRL, we perform a deployment visualization on TaxiBJ. We first aligned the flow and their geographic coordinates with a real map of Beijing. Public resources are modeled as UAVs unconstrained by the road network, and the scenario is framed as drone delivery units that can respond directly. Throughout this study, we fix the initial energy to 40, deploy 50 resources, and set their service capacity to 10.

Using the trained GSTRL, we replay the test period from March 22 to March 31, 2016 and record the location of each public resource at each time step. **Fig. 6** overlays their locations on the heatmap. As the demand distribution changes over time, most UAVs migrate visibly to the densest regions, highlighting the superior performance of GSTRL. We also plot the remaining energy and service capacity traces for each public resource, confirming that the policy satisfies two constraints of CPRA, every UAV retains sufficient energy to return to the resource depot at the end of each day and the service capacity remains constant over time.

B GenAI Disclosure

We used a large language model (LLM) to assist with grammar editing of this manuscript. The LLM was not used to generate scientific content, results, or conclusions. All technical content and claims were produced and verified by the authors.

References

[1] Jackie Baek and Will Ma. 2022. Technical Note—Bifurcating Constraints to Improve Approximation Ratios for Network Revenue Management with Reusable Resources. *Operations Research* 70, 4 (2022), 2226–2236. arXiv:https://doi.org/10.1287/opre.2022.2282 doi:10.1287/opre.2022.2282

[2] Dimitris Bertsimas et al. 2019. Emergency response optimization with robust optimization techniques. *INFORMS Journal on Optimization* 1, 1 (2019), 101–127.

[3] Wei Chen, Yuqian Wu, Yuanshao Zhu, Xixuan Hao, Shiyu Wang, and Yuxuan Liang. 2025. Select, then Balance: A Plug-and-Play Framework for Exogenous-Aware Spatio-Temporal Forecasting. arXiv:2509.05779 [cs.LG] https://arxiv.org/abs/2509.05779

[4] Lefeng Cheng, Pengrong Huang, Mengya Zhang, Ru Yang, and Yafei Wang. 2025. Optimizing Electricity Markets Through Game-Theoretical Methods: Strategic and Policy Implications for Power Purchasing and Generation Enterprises. *Mathematics* 13, 3 (2025). doi:10.3390/math13030373

[5] Bing Ding, Zening Li, Zhengmao Li, Yixun Xue, Xinyue Chang, Jia Su, and Hongbin Sun. 2025. Cooperative Operation for Multiagent Energy Systems Integrated With Wind, Hydrogen, and Buildings: An Asymmetric Nash Bargaining Approach. *IEEE Transactions on Industrial Informatics* 21, 8 (2025), 6410–6421. doi:10.1109/TII.2025.3563589

[6] Zhenhui Feng and Renbin Xiao. 2024. Three-dimensional task allocation for smart transportation in spatial crowdsourcing: An intelligent role division approach. *Advanced Engineering Informatics* 62 (2024), 102736. doi:10.1016/j.aei.2024.102736

[7] Haotian Gao, Renhe Jiang, Zheng Dong, Jinliang Deng, Yuxin Ma, and Xuan Song. 2024. Spatial-Temporal-Decoupled Masked Pre-training for Spatiotemporal Forecasting. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24*, Kate Larson (Ed.). International Joint Conferences on Artificial Intelligence Organization, 3998–4006. doi:10.24963/ijcai.2024/442 Main Track.

[8] Jiaxi Hu, Yuehong Hu, Wei Chen, Ming Jin, Shirui Pan, Qingsong Wen, and Yuxuan Liang. 2024. Attractor Memory for Long-Term Time Series Forecasting: A Chaos Perspective. arXiv:2402.11463 [cs.LG] https://arxiv.org/abs/2402.11463

[9] Jiaxi Hu, Disen Lan, Ziyu Zhou, Qingsong Wen, and Yuxuan Liang. 2024. Time-SSM: Simplifying and Unifying State Space Models for Time Series Forecasting. arXiv:2405.16312 [cs.LG] https://arxiv.org/abs/2405.16312

[10] Shenggong Ji, Yu Zheng, and Tianrui Li. 2016. Urban sensing based on human mobility. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing* (Heidelberg, Germany) *(UbiComp '16)*. Association for Computing Machinery, New York, NY, USA, 1040–1051. doi:10.1145/2971648.2971735

[11] Shenggong Ji, Yu Zheng, Zhaoyuan Wang, and Tianrui Li. 2019. A Deep Reinforcement Learning-Enabled Dynamic Redeployment System for Mobile Ambulances. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 3, 1, Article 15 (March 2019), 20 pages. doi:10.1145/3314402

[12] Yaser Keneshloo, Tian Shi, Naren Ramakrishnan, and Chandan K. Reddy. 2019. Deep Reinforcement Learning For Sequence to Sequence Models. arXiv:1805.09461 [cs.LG] https://arxiv.org/abs/1805.09461

[13] E. L. Lawler and D. E. Wood. 1966. Branch-and-Bound Methods: A Survey. *Oper. Res.* 14, 4 (Aug. 1966), 699–719. doi:10.1287/opre.14.4.699

[14] Yueheng Li, Guangming Xie, and Zongqing Lu. 2022. Difference Advantage Estimation for Multi-Agent Policy Gradients. In *International Conference on Machine Learning*. https://api.semanticscholar.org/CorpusID:250340910

[15] Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. 2021. Fourier Neural Operator for Parametric Partial Differential Equations. arXiv:2010.08895 [cs.LG] https://arxiv.org/abs/2010.08895

[16] Yuxuan Liang, Kun Ouyang, Hanshu Yan, Yiwei Wang, Zekun Tong, and Roger Zimmermann. 2021. Modeling Trajectories with Neural Ordinary Differential Equations. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, Zhi-Hua Zhou (Ed.). International Joint Conferences on Artificial Intelligence Organization, 1498–1504. doi:10.24963/ijcai.2021/207 Main Track.

[17] Jia Liu, Donghai Zhai, Wei Huang, Shenggong Ji, Junbo Zhang, and Tianrui Li. 2024. Daily Schedule Recommendation in Urban Life Based on Deep Reinforcement Learning. *IEEE Transactions on Neural Networks and Learning Systems* (2024), 1–10. doi:10.1109/TNNLS.2024.3353215

[18] Qingxiang Liu, Xu Liu, Chenghao Liu, Qingsong Wen, and Yuxuan Liang. 2024. Time-FFM: Towards LM-Empowered Federated Foundation Model for Time Series Forecasting. arXiv:2405.14252 [cs.LG] https://arxiv.org/abs/2405.14252

[19] Qingxiang Liu, Sheng Sun, Yuxuan Liang, Xiaolong Xu, Min Liu, Muhammad Bilal, Yuwei Wang, Xujing Li, and Yu Zheng. 2024. REFOL: Resource-Efficient Federated Online Learning for Traffic Flow Forecasting. arXiv:2411.14046 [cs.LG] https://arxiv.org/abs/2411.14046

[20] Qingxiang Liu, Sheng Sun, Yuxuan Liang, Jingjing Xue, and Min Liu. 2024. Personalized Federated Learning for Spatio-Temporal Forecasting: A Dual Semantic Alignment-Based Contrastive Approach. arXiv:2404.03702 [cs.LG] https://arxiv.org/abs/2404.03702

[21] Yang Liu, Fanyou Wu, Cheng Lyu, Shen Li, Jieping Ye, and Xiaobo Qu. 2022. Deep dispatching: A deep reinforcement learning approach for vehicle dispatching on online ride-hailing platform. *Transportation Research Part E: Logistics and Transportation Review* 161 (2022), 102694. doi:10.1016/j.tre.2022.102694

[22] Di Lv, Yuhao Wang, Liang Wang, Yang Fei, Kai Wang, and Xiaobo Qu. 2024. Modular flying vehicles: Scheduling modes, social benefits, and challenges. *Communications in Transportation Research* 4 (2024), 100144. doi:10.1016/j.commtr.2024.100144

[23] Xiaowei Mao, Haomin Wen, Hengrui Zhang, Huaiyu Wan, Lixia Wu, Jianbin Zheng, Haoyuan Hu, and Youfang Lin. 2023. DRL4Route: A Deep Reinforcement Learning Framework for Pick-up and Delivery Route Prediction. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (Long Beach, CA, USA) *(KDD '23)*. Association for Computing Machinery, New York, NY, USA, 4628–4637. doi:10.1145/3580305.3599811

[24] F. Mohammed et al. 2022. Dynamic public resource allocation for disaster response. *ACM Transactions on Spatial Algorithms and Systems* 8, 1 (2022), 1–25.

[25] Haobo Qi, Du Huang, Yingqiu Zhu, Danyang Huang, and Hansheng Wang. 2023. Mini-batch Gradient Descent with Buffer. arXiv:2312.08728 [stat.CO] https://arxiv.org/abs/2312.08728

[26] Tabish Rashid, Mikayel Samvelyan, Christian Schroeder de Witt, Gregory Farquhar, Jakob Foerster, and Shimon Whiteson. 2020. Monotonic Value Function Factorisation for Deep Multi-Agent Reinforcement Learning. *Journal of Machine Learning Research* 21, 178 (2020), 1–51. http://jmlr.org/papers/v21/20-081.html

[27] Sijie Ruan, Jie Bao, Yuxuan Liang, Ruiyuan Li, Tianfu He, Chuishi Meng, Yanhua Li, Yingcai Wu, and Yu Zheng. 2020. Dynamic Public Resource Allocation Based on Human Mobility Prediction. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 4, 1, Article 25 (March 2020), 22 pages. doi:10.1145/3380986

[28] Soheil Sadeghi Eshkevari, Xiaocheng Tang, Zhiwei Qin, Jinhan Mei, Cheng Zhang, Qianying Meng, and Jia Xu. 2022. Reinforcement Learning in the Wild: Scalable RL Dispatching Algorithm Deployed in Ridehailing Marketplace. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (Washington DC, USA) *(KDD '22)*. Association for Computing Machinery, New York, NY, USA, 3838–3848. doi:10.1145/3534678.3539095

[29] Simone Sagratella. 2019. On generalized Nash equilibrium problems with linear coupling constraints and mixed-integer variables. *Optimization* 68, 1 (2019), 197–226. arXiv:https://doi.org/10.1080/02331934.2018.1545125 doi:10.1080/02331934.2018.1545125

[30] Cheng Tan, Zhangyang Gao, Siyuan Li, and Stan Z. Li. 2024. SimVPv2: Towards Simple yet Powerful Spatiotemporal Predictive Learning. arXiv:2211.12509 [cs.LG] https://arxiv.org/abs/2211.12509

[31] Hao Wang, Chi Harold Liu, Zipeng Dai, Jian Tang, and Guoren Wang. 2021. Energy-Efficient 3D Vehicular Crowdsourcing for Disaster Response by Distributed Deep Reinforcement Learning. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining* (Virtual Event, Singapore) *(KDD '21)*. Association for Computing Machinery, New York, NY, USA, 3679–3687. doi:10.1145/3447548.3467070

[32] Jie Wang, Zhihai Wang, Xijun Li, Yufei Kuang, Zhihao Shi, Fangzhou Zhu, Mingxuan Yuan, Jia Zeng, Yongdong Zhang, and Feng Wu. 2024. Learning to Cut via Hierarchical Sequence/Set Model for Efficient Mixed-Integer Programming. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 46, 12 (2024), 9697–9713. doi:10.1109/TPAMI.2024.3432716

[33] Qiongyan Wang, Yutong Xia, Siru ZHong, Weichuang Li, Yuankai Wu, Shifen Cheng, Junbo Zhang, Yu Zheng, and Yuxuan Liang. 2025. AirRadar: Inferring Nationwide Air Quality in China with Deep Neural Networks. arXiv:2501.13141 [cs.LG] https://arxiv.org/abs/2501.13141

[34] Qiongyan Wang, Xingchen Zou, Yutian Jiang, Haomin Wen, Jiaheng Wei, Qingsong Wen, and Yuxuan Liang. 2025. Urban-R1: Reinforced MLLMs Mitigate Geospatial Biases for Urban General Intelligence. arXiv:2510.16555 [cs.AI] https://arxiv.org/abs/2510.16555

[35] Chenyu Wu, Zhipeng Ma, Junbo Zhang, Songyu Ke, and Yu Zheng. 2025. Spatio-Temporal Forecasting under Open-World Missingness with Adaptive Mixture-of-Experts. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management*. 3366–3375.

[36] Yuqian Wu, Yuhong Peng, Jiapeng Yu, and Raymond S. T. Lee. 2024. MAS4POI: a Multi-Agents Collaboration System for Next POI Recommendation. arXiv:2409.13700 [cs.IR] https://arxiv.org/abs/2409.13700

[37] Yuqian Wu, Yuhong Peng, Jiapeng Yu, Xiangyu Liu, Zeting Yan, Kang Lin, Weifeng Su, Bingqing Qu, Raymond Lee, and Dingqi Yang. 2025. Beyond Regularity: Modeling Chaotic Mobility Patterns for Next Location Prediction. arXiv:2509.11713 [cs.LG] https://arxiv.org/abs/2509.11713

[38] Yimo Yan, Haomin Wen, Yang Deng, Andy H.F. Chow, Qihao Wu, and Yong-Hong Kuo. 2024. A mixed-integer programming-based Q-learning approach for electric bus scheduling with multiple termini and service routes. *Transportation Research Part C: Emerging Technologies* 162 (2024), 104570. doi:10.1016/j.trc.2024.104570

[39] Zhaoxing Yang, Haiming Jin, Guiyun Fan, Min Lu, Yiran Liu, Xinlang Yue, Hao Pan, Zhe Xu, Guobin Wu, Qun Li, Xiaotong Wang, and Jiecheng Guo. 2024. Rethinking Order Dispatching in Online Ride-Hailing Platforms. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (Barcelona, Spain) *(KDD '24)*. Association for Computing Machinery, New York, NY, USA, 3863–3873. doi:10.1145/3637528.3672028

[40] Huaxiu Yao, Xianfeng Tang, Hua Wei, Guanjie Zheng, and Zhenhui Li. 2018. Revisiting Spatial-Temporal Similarity: A Deep Learning Framework for Traffic Prediction. arXiv:1803.01254 [cs.LG] https://arxiv.org/abs/1803.01254

[41] Junzi Zhang, Jongho Kim, Brendan O'Donoghue, and Stephen Boyd. 2020. Sample Efficient Reinforcement Learning with REINFORCE. arXiv:2010.11364 [cs.LG] https://arxiv.org/abs/2010.11364

[42] Jiani Zhang, Xingjian Shi, Junyuan Xie, Hao Ma, Irwin King, and Dit-Yan Yeung. 2018. GaAN: Gated Attention Networks for Learning on Large and Spatiotemporal Graphs. arXiv:1803.07294 [cs.LG] https://arxiv.org/abs/1803.07294

[43] Junbo Zhang, Yu Zheng, and Dekang Qi. 2017. Deep spatio-temporal residual networks for citywide crowd flows prediction. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 31.

[44] L. Zhao et al. 2019. Adaptive mobility service optimization using real-time traffic data. *Transportation Research Part C: Emerging Technologies* 102 (2019), 1–20.

[45] Jiaohao Zheng, Mehmet Necip Kurt, and Xiaodong Wang. 2024. Stochastic Integrated Actor–Critic for Deep Reinforcement Learning. *IEEE Transactions on Neural Networks and Learning Systems* 35, 5 (2024), 6654–6666. doi:10.1109/TNNLS.2022.3212273

[46] Qiangfeng Zhu, Jun Zheng, and Abbas Jamalipour. 2025. Deep Reinforcement Learning-Based UAV Base Station Deployment for Content Delivery in Cellular IoT Networks. *IEEE Internet of Things Journal* 12, 13 (2025), 23389–23401. doi:10.1109/JIOT.2025.3552123