Routing and Charging Scheduling for EV Battery Swapping Systems: Hypergraph-based Heterogeneous Multiagent Deep Reinforcement Learning

Shuai Mao, Jiangliang Jin, and Yunjian Xu, *Member, IEEE,*

***Abstract*—This work studies the joint electric vehicle (EV) routing and battery charging scheduling problem in a transportation network with multiple battery swapping stations (BSSs) under random EV swapping demands, renewable generation, and electricity prices. The joint scheduling problem is formulated as a decentralized partially observable Markov decision process (Dec-POMDP) with an objective to minimize the expected sum of the battery charging cost and the travel/waiting cost of EV owners. The formulated Dec-POMDP is hard to solve, due the curse of dimensionality and the unknown system dynamics. To tackle the challenges, we propose a new heterogeneous multiagent hypergraph attention actor-critic (HMA-HGAAC) framework, which integrates hypergraph attention (HGAT) networks to multiagent deep reinforcement learning (MADRL) to enhance the learning efficiency with a hypergraph where multiple nodes can be connected by a single hyperedge. Numerical experiments based on real-world data and a 180-node transportation network show that the proposed approach can save the system cost achieved by state-of-the-art benchmarks, independent proximal policy optimization (IPPO), multiagent proximal policy optimization (MAPPO), and heterogeneous multiagent graph attention proximal policy optimization (HMA-GAPPO), by 23.5%, 18.9%, and 13.3%, respectively.**

***Index Terms*—Electric vehicles, battery swapping stations, decentralized partially observable Markov decision process, multiagent deep reinforcement learning, hypergraph neural networks**

NOMENCLATURE

Acronyms

BSS	Battery Swapping Station
CB	Charging Bay
DB	Depleted Battery
FB	Fully-charged Battery
DRL	Deep Reinforcement Learning
MADRL	MultiAgent Deep Reinforcement Learning
MAPPO	MultiAgent Proximal Policy Optimization
IPPO	Independent Proximal Policy Optimization
GNNs	Graph Neural Networks
HGAT	HyperGraph ATtention
Dec-POMDP	Decentralized Partial Observation Markov Decision Process
HMA-HGAAC	Heterogeneous MultiAgent HyperGraph Attention Actor-Critic (HMA-HGAAC)
HMA-HGAPPO	Heterogeneous MultiAgent HyperGraph Attention Proximal Policy Optimization
HMA-GAPPO	Heterogeneous MultiAgent Graph Attention Proximal Policy Optimization

Sets and Indices

$\mathcal{B}_{m,t}$	Set of all batteries in BSS m at timeslot t
$\mathcal{B}^k_{m,t}$	Set of type-k batteries in BSS m at timeslot t
$\mathcal{CB}^k_{m,t}$	Set of type-k batteries in the CBs of BSS m at timeslot t
$\mathcal{V}$	Set of nodes
$\mathcal{E}$	Set of hyperedges
$\mathcal{K}$	Set of types of batteries
$\mathcal{M}$	Set of BSSs
$\mathcal{N}$	Set of EVs
$\mathcal{N}^k$	Set of type-k EVs
$\mathcal{Q}^k_{m,t}$	Set of type-k EVs in the queue of BSS m at timeslot t

Parameters

c	EV owners' monetary value of time
d_n	The destination of EV n
E	The number of hyperedges
ℓ^{bss}_m	The location of BSS m
T	Total number of timeslots
α	The learning rate
ϵ	The clipping parameter
γ	The discount factor
ζ	The energy consumption of a driving EV per timeslot
λ	The exponential weight discount
ψ	FBs' target SoC

Variables

$a^{bss,k}_{m,t}$	The number of type-k batteries charged at

S. Mao is with the Department of Mechanical and Automation Engineering, the Chinese University of Hong Kong. Email: smao@mae.cuhk.edu.hk.

J. Jin is with the College of Information Science and Technology, Donghua University, Shanghai, China. Email: jinjiangliang@dhu.edu.cn.

Y. Xu is with the Department of Mechanical and Automation Engineering, the Chinese University of Hong Kong, Hong Kong SAR, China, and the CUHK Shenzhen Research Institute (SZRI), Shenzhen, China. Email: yjxu@mae.cuhk.edu.hk. (*Corresponding Author: Yunjian Xu*).

This research was supported in part by the General Research Fund (GRF) project 14200720 of the Hong Kong University Grants Committee, the National Natural Science Foundation of China (NSFC) Project 62073273, the "Fundamental Research Funds for the Central Universities" Project 23D110417, and the Shanghai Pujiang Program (Project 23PJ1400300).

	BSS m and timeslot t
$A^{bss}_{m,t}$	The charging action of BSS m at timeslot t
$a^{ev}_{n,t}$	The routing action of EV n at timeslot t
$\ell^{ev}_{n,t}$	The location of EV n at timeslot t
$\boldsymbol{o}^{bss}_{m,t}$	The observation of BSS m at timeslot t
$\boldsymbol{o}^{ev}_{n,t}$	The observation of EV n at timeslot t
p_t	The electricity price at timeslot t
$r^{bss}_{m,t}$	Stage cost of BSS m at t
$r^{ev}_{n,t}$	Stage cost of EV n at t
$x_{b,t}$	The energy level of battery b at BSS m and timeslot t
$\boldsymbol{x}_{m,t}$	The SoCs of all batteries at BSS m and timeslot t
$x_{n,t}$	The energy level of EV n at timeslot t
y_t	The renewable generation at timeslot t
$\tau_{n,m,t}$	The shortest travel time between EV n to BSS m at timeslot t
$\tau_{a^{ev}_{n,t},d_n}$	The shortest travel time between EV n to its destination at timeslot t
$\eta_{n,t}$	Binary variable ($\eta_{n,t}=1$ if the swapping demand of EV n is fulfilled, else 0)

I. Introduction

Achieving carbon neutrality by 2050 has been nominated by the United Nations as the most urgent mission in the world [1]. As an effective alternative to fuel vehicles, electric vehicles (EVs) have achieved remarkably growing market share worldwide [2]. To meet the rapidly increasing EV charging demand, the efficient operation of EV charging facilities including charging stations and battery swapping stations (BSSs) has received recent attention [3]. Compared with charging stations, battery swapping can save EV owners charging time and and help to extend battery life [4].

There is extensive literature that explores the individual EV routing and battery charging scheduling for BSSs, without taking into account the system uncertainty. Various approaches have been adopted to explore the optimal scheduling of battery charging at BSSs, including linear programming [5], mixed-integer programming [6], rolling-horizon optimization [7], heuristic optimization [8], [9] and bi-level optimization [10]. For the optimal EV routing to BSSs, the authors of [11] propose a bi-level programming model to solve the joint BSS location and electric truck routing problem.

Recent studies have explicitly modelled the system uncertainty in a BSS system, for EV routing [12], [13] and battery charging [14]–[17]. The authors of [18] formulate the EV routing problem (to multiple BSSs) as a two-stage optimization problem, considering the uncertainty in EV swapping demands. the authors of [19] formulate the battery charging scheduling problem as a Markov decision process (MDP) with random swapping demand and electricity prices, which is solved by a near-optimal Monte Carlo sampling-based approach.

The aforementioned literatures on EV routing and charging scheduling overlook the intrinsic interdependencies between the two problems, which may lead to suboptimal operation of the entire system. EV routing scheduling without considering charging scheduling at BSSs may increase the waiting time of EVs at BSSs and the charging cost at BSSs. Analogously, battery charging scheduling at BSSs without EV routing may lead to suboptimal routing and increase travel time for EVs. Recently, coordinated EV routing and charging has been implemented by third party companies including Uber, Waymo, and Cruise, for the purpose of efficient passenger transportation [20], [21] and food delivery [22], [23].

There are few studies on the joint scheduling of EV routing and battery charging for multiple BSSs under system uncertainties. A pioneering work [24] formulates the joint operation of BSSs and EV fleet as a mixed-integer linear programming model to maximize the revenue from battery swapping service under deterministic swapping demand and electricity prices. The authors of [25] explore the joint operation of BSSs and autonomous mobility-on-demand through a bilevel framework with accurate information of future electricity prices and customer demand. However, future customer demand and electricity prices can be random and accurate forecast can be difficult to obtain. In this work, we construct a joint EV routing and battery charging scheduling framework for multiple BSSs, with explicit incorporation of random EV swapping demand, electricity prices, and renewable generation.

In the recent decade, deep reinforcement learning (DRL) has demonstrated great success in tackling sequential decision-masking problems with unknown system dynamics such as Atari games [26]. As an active branch of DRL, multiagent deep reinforcement learning (MADRL) has shown great potential in solving coordination decision problems for multiagent systems (*e.g.*, the StarCraft [27]).

MADRL adopts DRL to train multiple agents to coordinate their actions in a distributed environment, so as to achieve specific goals and to learn adaptive strategies in dynamic and uncertain environments. Compared with conventional DRL, MADRL can bettern tackle the partial observability issue in distributed decision making, and accelerate the training process by alleviating the computational and communication burden.

Although MADRL has been extensively adopted on the routing and charging scheduling for battery charging stations [28]–[30], to our knowledge, it has not been applied to the coordinated operation of BSSs. In this work, we propose the first MADRL framework to coordinate the routing and battery charging decisions made by EVs and BSSs, which have partial observability of the environment with stochastic electricity prices, renewable generation, and swapping demand.

For the training and execution of MADRL, agents can adopt centralized training and decentralized execution (CTDE) based approaches (such as multiagent proximal policy optimization, MAPPO [31]), which require access to full system information during training but only their own observation during execution. Decentralized training and decentralized execution (DTDE) based approaches (such as independent proximal policy optimization, IPPO [32]) make decision based only on each agent's own observation, without considering the information exchange among agents. The proposed MADRL framework adopts an alternative approach, where agents (such as EVs

and BSSs) can exchange information with their adjacent agents during training and execution.[1]

The authors of [4] model the real-time scheduling problem in a battery swapping-charging system as a Dec-POMDP and solve it using MADRL algorithms. The authors of [33] adopt the MA-DRL approach to learn the charging pricing strategies of multiple EVCSs and approximate the Nash equilibrium of the pricing game under incomplete information. However, the aforementioned works adopt CTDE based MADRL approaches, which require access to full system information during training. In this work, we propose to incorporate the HGAT network into MADRL in a practical setting where only adjacent agents can exchange information.

The authors of [34] propose the graph convolutional reinforcement learning, where the dynamics of the underlying graph in a multi-agent environment are accommodated by the adaptability of graph convolution so as to capture the interplay between agents. The authors of [35] propose a multi-agent graph-attention communication (MAGIC) algorithm where GATs are adopted to achieve communication and integrate messages between agents. The authors of [36] develop a heterogeneous graph-attention-based architecture to enhance distributed learning over a graph. In the above studies, an edge of a GNN connects only two nodes, whereas in our context, multiple nodes (EVs and BSSs) may be connected by a single edge (road). We therefore propose a new heterogeneous multiagent hypergraph attention actor-critic (HMA-HGAAC) framework, where an edge in the hypergraph attention (HGAT) network can connect arbitrary number of nodes.

Unlike GNNs, hypergraph neural networks [37], [38] can connect multiple nodes with a single hyperedge, so as to better capture high-order correlation among these nodes through a more sophisticated graph representation. A special type of hypergraph neural networks, hypergraph attention (HGAT) networks, incorporate the attention mechanism to provide flexible information propagation and aggregation through hypergraphs by computing the attention weights among hyperedges and nodes [39].

The main contribution of this work is two-fold. First, we propose to model the heterogeneous agents (EVs and BSSs) in a transportation network through a hypergraph structure, and formulate the dynamic joint scheduling problem of EV routing and battery charging as a decentralized partial observation Markov decision process (Dec-POMDP) [40]. The formulated Dec-POMDP explicitly incorporates the unknown dynamics of EV swapping demand, electricity prices, and renewable generation, and is solved by HGAT enhanced MADRL. The hypergraph structure can capture the interplay among adjacent agents in the transportation network, and the Dec-POMDP provides a framework for dealing with distributed decision making in a highly dynamic environment. To our knowledge, this work establishes the first MADRL framework of the joint scheduling of EV routing and battery charging for multiple BSSs, which is enhanced by a hypergraph attention (HGAT) network.

[1]We note that emerging technologies such as mobile edge computing have enabled EVs and BSSs to share and exchange information [3].

Further, we propose a new heterogeneous multiagent hypergraph attention actor-critic (HMA-HGAAC) framework with a HGAT network and the actor-critic architecture (shown in Fig. 2(a)). For the first time, the HGAT network (originally proposed for text classification [39]) is incorporated into heterogeneous multiagent reinforcement learning for distributed decision making of heterogeneous agents. For our application, the proximal policy optimization (PPO) algorithm [41] is adopted by the actor-critic module in the proposed HMA-HGAAC framework, leading to a Heterogeneous MultiAgent HyperGraph Attention Proximal Policy Optimization (HMA-HGAPPO) approach.

Compared with conventional GNNs, HGAT networks offer some advantages on the information exchange for distributed decision making in a transportation network. For GNN, edges can only represent binary relationships. In real-world applications, however, there may exist many-to-many relationships, such as the interaction among multiple EVs located at the same location. Combining hypergraphs and the attention mechanism, HGAT enables more flexible structures of information sharing and facilitates learning efficient communication among adjacent agents, with a more accurate hypergraph representation of the transportation network.

We substantiate the efficacy of the proposed HMA-HGAPPO approach by numerical simulations on real-world (EV swapping demand, electricity pricing, and renewable generation) data. Compared to IPPO and MAPPO, where each agent outputs the action based only on its own observation during execution, the proposed HMA-HGAPPO (with HGAT) achieves 25.6% and 13.1% lower system cost on a 44-node transportation network and 23.5% and 18.9% lower system cost on a 180-node transportation network.

Compared with the heterogeneous multiagent graph attention proximal policy optimization (HMA-GAPPO) approach that adopts a *graph*-attention-based architecture for the information exchange among adjacent agents, the proposed HMA-HGAPPO approach reduces 3.9% and 13.3% of system cost on the 44-node and 180-node transportation network, respectively, mainly due to the incorporation of HGAT that captures high-order feature representation for adjacent agents with a *hypergraph* representation of the transportation network.

The rest of the paper is organized as follows. In Section II, we formulate the joint scheduling problem of EV routing and battery charging in a BSS network as a Dec-POMDP. Section III describes the proposed HMA-HGAPPO approach. In Section IV, we conduct numerical experiments based on real-world data to validate the effectiveness of the proposed approach.

II. Problem formulation

We consider the coordinated operation of M BSSs in a transportation network. There are N EVs equipped with K types of batteries that can be swapped at the BSSs. The objective is to minimize the system cost, which is the sum of battery charging cost (incurred by BSSs) and the travel and waiting time cost of EV users.

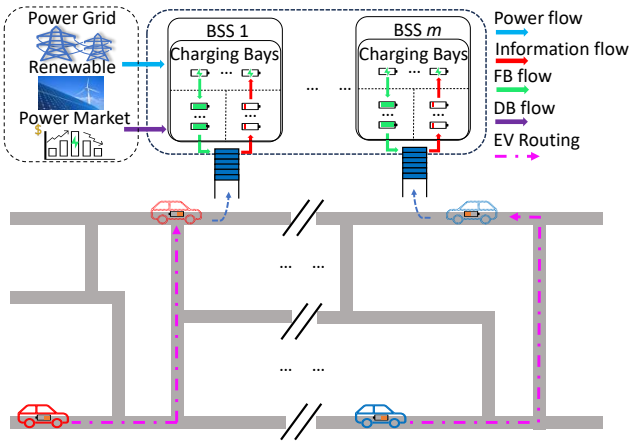


Fig. 1. The transportation network in San Francisco.

A. System Model

Fig. 1 provides an illustration for the system model.

1) **BSS observation**: Let $\mathcal{M}, \mathcal{K}$ denote the set of BSSs and types of batteries, respectively. Each BSS has inventories of fully-charged batteries (FBs) and depleted batteries (DBs), as well as charging bays (CBs) that can charge all types of batteries.
We use $\boldsymbol{x}_{m,t} = \{x_{b,t}\}_{b\in\mathcal{B}_{m,t}}$ to denote the energy levels (in kWh) of all batteries at BSS m. For each $k \in \mathcal{K}$, let $\mathcal{B}^k_{m,t} \subset \mathcal{B}_{m,t}$ denote the set of type-k batteries at BSS m, and $\mathcal{CB}^k_{m,t} \subset \mathcal{B}^k_{m,t}$ denote the set of type-k batteries of CBs at BSS m.
At timeslot t, for each $m \in \mathcal{M}$, let $\mathcal{Q}^k_{m,t}$ denote the set of type-k EVs in the queue of BSS m. When there is not enough type-k FBs at BSS m, newly arrived type-k EVs have to wait in the queue $\mathcal{Q}^k_{m,t}$ until being served. The observation of BSS m consists of the current time, its location ℓ^{bss}_m, the energy level of all batteries at the BSS, the number of batteries in CBs and the queue length of each battery type, as well as the real-time electricity price p_t and renewable generation y_t:

$$\boldsymbol{o}^{bss}_{m,t} = \{t, \ell^{bss}_m, \boldsymbol{x}_{m,t}, \{|\mathcal{CB}^k_{m,t}|, |\mathcal{Q}^k_{m,t}|\}_{k\in\mathcal{K}}, p_t, y_t\}. \tag{1}$$

2) **Charging action**: At timeslot t, for each BSS $m \in \mathcal{M}$, depleted batteries in the DB inventory are loaded to idle CBs. Let $a^{bss,k}_{m,t}$ denote the number of type-k batteries that are charged at BSS m, which satisfies

$$a^{bss,k}_{m,t} \le |\mathcal{CB}^k_{m,t}|. \tag{2}$$

We use

$$A^{bss}_{m,t} = \sum_{k\in\mathcal{K}} a^{bss,k}_{m,t} \tag{3}$$

to denote the total number of batteries that are charged at BSS m at timeslot t. To prolong battery life, the target SoC of all FBs is set to σ instead of 100% [42]: if the SoC of a battery in $\mathcal{CB}^k_{m,t}$ is greater than or equal to the target SoC ψ, it is loaded to the FB inventory for swapping.

3) **EV routing**: Let $\mathcal{N}$ denote the set of EVs that require a battery swapping service, and $\mathcal{N}^k \subset \mathcal{N}$ be the set of type-k EVs. In our model, each type-k EV is equipped with a type-k battery. At timeslot t, let $\ell^{ev}_{n,t}$ and $x_{n,t}$ denote the location and battery energy level of a type-k EV n, respectively. Each EV n requires to receive the battery swapping service at a BSS before reaching its destination d_n.
Following [43], [44], we incorporate the estimated travel time to BSSs, $\{\tau_{n,m,t}\}_{m\in\mathcal{M}}$, into EV n's observation at timeslot t, which consists of the current time, its type, location, destination, battery energy level, and the shortest travel time to BSSs:

$$\boldsymbol{o}^{ev}_{n,t} = \{t, k, \ell^{ev}_{n,t}, d_n, x_{n,t}, \{\tau_{n,m,t}\}_{m\in\mathcal{M}}\}. \tag{4}$$

4) **Routing action**: At timeslot t, each EV n takes a routing action that represents the designated BSS:

$$a^{ev}_{n,t} = m \in \mathcal{M}, \tag{5}$$

and drives through a planned routing path $L^m_{n,t}$,

$$L^m_{n,t} = (L^{m,1}_{n,t}, \ldots, L^{m,\text{end}}_{n,t}), \tag{6}$$

which is the shortest (travel-time) path from the EV's current location $L^{m,1}_{n,t} = \ell^{ev}_{n,t}$ to the designated BSS $L^{m,\text{end}}_{n,t} = \ell^{bss}_m$.
If EV n's swapping demand has not been fulfilled at t, we let $\eta_{n,t} = 0$, and the location and battery energy level of EV n evolve as:

$$\ell^{ev}_{n,t+1} = L^{m,2}_{n,t}, \tag{7}$$

$$x_{n,t+1} = x_{n,t} - \zeta, \tag{8}$$

where ζ (in kWh) represents EV n's energy consumption in each timeslot.

5) **Charging cost at BSSs**: At timeslot t, for BSS $m \in \mathcal{M}$, charging batteries in CBs results in the charging cost as follows:

$$r^{bss}_{m,t}(\boldsymbol{o}^{bss}_{m,t}, A^{bss}_{m,t}) = p_t \Big[\sum_{k\in\mathcal{K}} a^{bss,k}_{m,t} - y_t\Big]^+, \tag{9}$$

where $(x)^+ = \max(0, x)$; the term in the square bracket represents the net energy consumption of BSS m at timeslot t, *i.e.,* total charging energy minus renewable generation.[2]

6) **EVs' cost of time**: At timeslot t, for each EV $n \in \mathcal{N}$, traveling to the designated BSS or waiting at the BSS for swapping service incurs the following cost:

$$r^{ev}_{n,t}(\boldsymbol{o}^{ev}_{n,t}, a^{ev}_{n,t}) = \begin{cases} c, & \eta_{n,t} = 0, \\ c \cdot \tau_{a^{ev}_{n,t}, d_n}, & \eta_{n,t} = 1, \end{cases} \tag{10}$$

[2]It is true that the capital cost of PV system could be high [45], [46]. However, the focus of this work is on the efficient operation of existing charging and PV infrastructures, with an objective to minimize the long-term expected operation cost. The operation cost of solar generation is usually considered to be negligible, compared with conventional thermal generation [45], [46].

where c is the monetary value of time. If EV n receives the battery swapping service at t, we let $\eta_{n,t} = 1$. EV n is removed from $\mathcal{N}$ and incurs a one-time cost caused by the travel time $\tau_{a^{ev}_{n,t},d_n}$ from the current BSS $a^{ev}_{n,t}$ to the EV's destination d_n.

7) **Heterogeneous transportation hypergraph**: To formulate the interaction among EVs, we model the transportation network as a hypergraph with agents ($i.e.$, BSSs and EVs) as nodes, and the roads as hyperedges.
We build an undirected hypergraph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where $\mathcal{E} = \{\boldsymbol{e}_j\}_{j=1}^{E}$ represents the set of E hyperedges ($i.e.$, roads) that can connect two or more nodes in the transportation network. Each hyperedge $\boldsymbol{e} \in \mathcal{E}$ is a vector consisting of all nodes it connects.
The topology of the hypergraph $\mathcal{G}$ is expressed as $\mathcal{G} = (\mathbf{A}_t, \mathbf{O}_t)$, where $\mathbf{A}_t \in \mathbb{R}^{(N+M)\times E}$ is an incidence matrix illustrating the connections among the hyperedges and nodes. At timeslot t, for each node $v \in \mathcal{V}$ and hyperedge $\boldsymbol{e} \in \mathcal{E}$, $\mathbf{A}_{v\boldsymbol{e},t} = 1$ if node $v \in \boldsymbol{e}$ ($i.e.$, node v is connected with hyperedge $\boldsymbol{e}$), and $\mathbf{A}_{v\boldsymbol{e},t} = 0$ otherwise. Let $\mathbf{O}_t = (\mathbf{O}^{bss}_t, \mathbf{O}^{ev}_t)$ represent the observations of all agents at t, where $\mathbf{O}^{bss}_t = \{\boldsymbol{o}^{bss}_{m,t}\}_{m\in\mathcal{M}}$ and $\mathbf{O}^{ev}_t = \{\boldsymbol{o}^{ev}_{n,t}\}_{n\in\mathcal{N}}$, respectively.

B. Decentralized Partially Observable Markov Decision Process (Dec-POMDP)

In this subsection, we formulate the EV battery charging and routing scheduling problem as a Dec-POMDP [40] that is defined by a tuple $\langle \mathcal{S}, \mathcal{O}, \mathcal{A}, \mathcal{R}, \mathbb{P}, \mathcal{M}\cup\mathcal{N}, \gamma\rangle$. For each agent $i \in \mathcal{M}\cup\mathcal{N}$, let $\xi_i \in \{bss, ev\}$ denote the type of agent i. $\mathcal{O}$ is the observation space of agents. For each $i \in \mathcal{M}\cup\mathcal{N}$, it receives only a local observation $\boldsymbol{o}^{\xi_i}_{i,t} \in \mathcal{O}$, which is $\boldsymbol{o}^{bss}_{m,t}$ for a BSS agent (cf. Eq. (1)) or $\boldsymbol{o}^{ev}_{n,t}$ for an EV agent (cf. Eq. (4)).

At timeslot t, let

$$\boldsymbol{s}_t = \{\boldsymbol{o}^{\xi_i}_{i,t}\}_{i\in\mathcal{M}\cup\mathcal{N}} \in \mathcal{S} \tag{11}$$

denote the system state of the environment, where $\mathcal{S}$ is the state space of the environment.

For each $i \in \mathcal{M}\cup\mathcal{N}$, it takes an action according to a decentralized policy $a^{\xi_i}_{i,t} \sim \pi^{\xi_i}_{i,t}(\cdot \mid \boldsymbol{o}^{\xi_i}_{i,t})$, in which $a^{\xi_i}_{i,t}$ is $A^{bss}_{m,t}$ for a BSS agent (cf. Eq. (3)) or $a^{ev}_{n,t}$ for an EV agent (cf. Eq. (5)). At timeslot t, let

$$\boldsymbol{a}_t = \{a^{\xi_i}_{i,t}\}_{i\in\mathcal{M}\cup\mathcal{N}} \in \mathcal{A} \tag{12}$$

denote the joint actions of agents, where $\mathcal{A}$ is the joint action space.

$\mathbb{P}(\boldsymbol{s}_{t+1} \mid \boldsymbol{s}_t, \boldsymbol{a}_t)$ is the transition probability from $\boldsymbol{s}_t$ to $\boldsymbol{s}_{t+1}$ given the joint action $\boldsymbol{a}_t$ for all agents. The observation transition of agent i not only depends on its own observation but also the system state $\boldsymbol{s}_t$.

Let

$$\boldsymbol{r}_t = \mathcal{R}(\boldsymbol{s}_t, \boldsymbol{a}_t) = \{r^{\xi_i}_{i,t}\}_{i\in\mathcal{M}\cup\mathcal{N}} \tag{13}$$

denote the stage cost where $r^{\xi_i}_{i,t}$ is $r^{bss}_{m,t}$ for a BSS agent (cf. Eq. (9)) or $r^{ev}_{n,t}$ for an EV agent (cf. Eq. (10)).

Agents learn a parameterized policy $\pi(\boldsymbol{a}_t \mid \boldsymbol{s}_t) \coloneqq \prod_{i\in\mathcal{M}\cup\mathcal{N}} \pi^{\xi_i}_i(a^{\xi_i}_i \mid \boldsymbol{o}^{\xi_i}_{i,t})$ to jointly minimize the sum of all agents' expected discounted system cost:[3]

$$\mathbb{E}_{\boldsymbol{s}_t,\boldsymbol{a}_t}\left[\sum\nolimits_{t=0}^{\infty} \gamma^t \sum\nolimits_{i\in\mathcal{M}\cup\mathcal{N}} r^{\xi_i}_{i,t}\right], \tag{14}$$

where $\gamma \in [0, 1)$ is a discount factor. We note that by adding weights to the stage cost of different agents $r^{\xi_i}_{i,t}$ in (14), one can pursue various tradeoffs among agents' costs, *e.g.*, the minimization of all EVs' cost of time.

III. The Proposed Approach

In this section, we propose a HMA-HGAAC framework (as shown in Fig. 2(a)) that takes the advantage of hypergraph attention (HGAT) networks for distributed decision making of actor critic methods. As an example, we introduce the Heterogeneous Multi-Agent HyperGraph Attention Proximal Policy Optimization (HMA-HGAPPO) approach, which is composed of three modules: the encoder module, the hypergraph attention (HGAT) module, and the actor-critic module.

A. Encoder module

First, we construct an encoder module to embed observation features. The encoder module consists of deep neural networks (DNNs) that project the observation features of each agent into the embedding feature of the same dimension [47], which is processed by the HGAT module to be introduced in Section III-B.

For each agent $i \in \mathcal{M}\cup\mathcal{N}$, given its observation $\boldsymbol{o}^{\xi_i}_{i,t}$, the encoder procedure is given by

$$\boldsymbol{h}_{i,t} = \mathbf{W}^{\xi_i}\boldsymbol{o}^{\xi_i}_{i,t}, \tag{15}$$

where $\mathbf{W}^{\xi_i}$ is a linear learnable transformation weight matrix (of type-ξ_i agents) that transforms the features of agents of diverse types into a uniform feature space, and $\boldsymbol{h}_{i,t}$ is the embedding feature of agent i at timeslot t. All agents have the same embedding feature dimension. We use $\boldsymbol{h}_t = \{\boldsymbol{h}_{i,t}\}_{i\in\mathcal{M}\cup N}$ to denote the embedding features of all agents.

B. Hypergraph attention module

The hypergraph attention module aims to aggregate the embedded features of adjacent agents. Based on the established hypergraph model in Section II-A7, we adopt the hypergraph attention (HGAT) network [39] to process the exchanged information among adjacent agents, through both agent-level attention and edge-level attention.

1) Agent-level attention: Let $\mathcal{E}_i \subset \mathcal{E}$ denote the set of hyperedges connected to agent i. For agent i, the HGAT aims to acquire the representations of all hyperedges in $\mathcal{E}_i$. For each hyperedge $\boldsymbol{e} \in \mathcal{E}_i$, we incorporate the attention mechanism to compute the normalized attention coefficient $\alpha_{j\boldsymbol{e}}$ for each agent $j \in \boldsymbol{e}$ (agent j is connected by hyperedge $\boldsymbol{e}$):

$$\alpha_{j\boldsymbol{e}} = \frac{\exp\Big(\boldsymbol{w}_1^{\mathsf{T}}\sigma\big(\mathbf{W}_{g_1}\boldsymbol{h}_{j,t}\big)\Big)}{\sum_{q\in\boldsymbol{e}}\exp\Big(\boldsymbol{w}_1^{\mathsf{T}}\sigma\big(\mathbf{W}_{g_1}\boldsymbol{h}_{q,t}\big)\Big)}, \tag{16}$$

[3]Minimizing the expected system cost is mathematically equivalent to maximizing an accumulative reward that is the negative of the expected system cost.

This article has been accepted for publication in IEEE Transactions on Smart Grid. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSG.2024.3386609

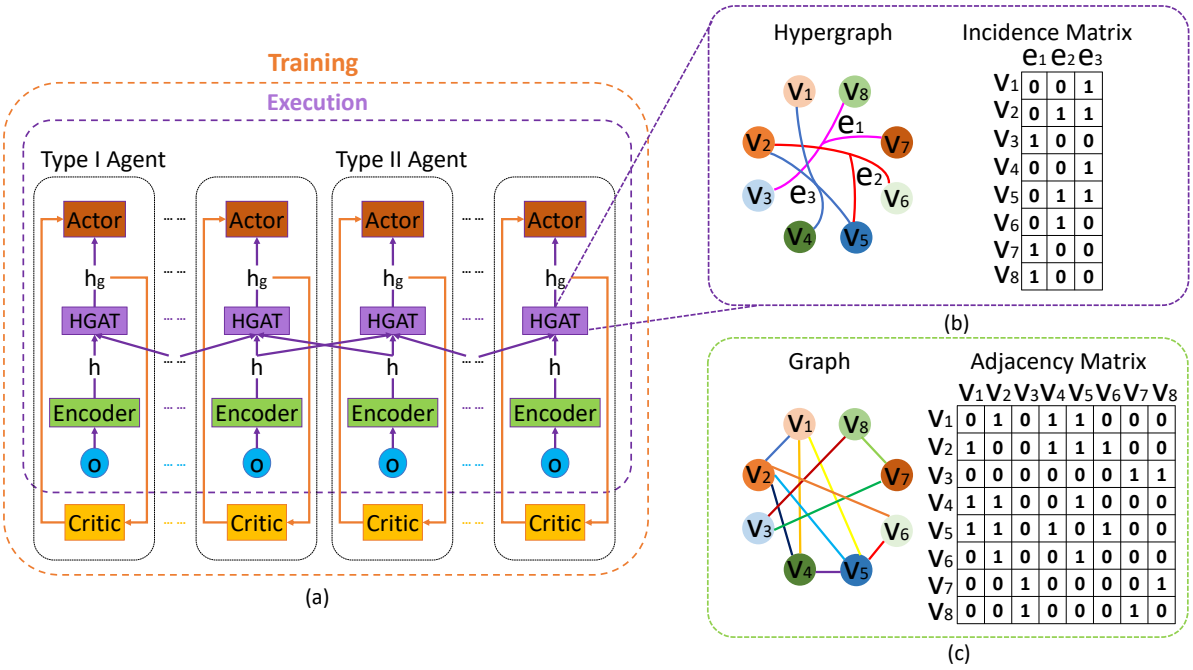


Fig. 2. The proposed HMA-HGAAC framework. (a) The HMA-HGAAC structure with an encoder module that enables information exchange among adjacent agents by unifying the dimensions of heterogeneous agents' observations. In (b), the hypergraph structure connects more than two nodes, which is represented by the incidence matrix; in (c), the graph structure connects only two nodes, which is represented by the adjacency matrix.

where $\boldsymbol{w}_1^{\mathsf{T}}$ is a weight vector, $\mathbf{W}_{g_1}$ is a linear learnable weight matrix, $\boldsymbol{h}_{j,t}$ is given in (15), and σ is an activation function (ReLU or LeakyReLU).

Given $\alpha_{j\boldsymbol{e}}$ defined in (16), the aggregated embedding feature $\boldsymbol{f}_{\boldsymbol{e},t}^{g}$ for the hyperedge $\boldsymbol{e}$ is given by

$$\boldsymbol{f}_{\boldsymbol{e},t}^{g} = \sigma\left(\sum_{j\in\boldsymbol{e}} \alpha_{j\boldsymbol{e}} \mathbf{W}_{g_1} \boldsymbol{h}_{j,t}\right). \tag{17}$$

2) Edge-level attention: After obtaining all representations of hyperedges, we adopt the edge-level attention mechanism to compute the normalized attention coefficient $\beta_{i\boldsymbol{e}}$ for agent i on hyperedge $\boldsymbol{e}$:

$$\beta_{i\boldsymbol{e}} = \frac{\exp\left(\boldsymbol{w}_2^{\mathsf{T}}\sigma\left([\mathbf{W}_{g_2}\boldsymbol{f}_{\boldsymbol{e},t}^{g} \parallel \mathbf{W}_{g_1}\boldsymbol{h}_{i,t}]\right)\right)}{\sum_{\boldsymbol{\varepsilon}\in\mathcal{E}_i}\exp\left(\boldsymbol{w}_2^{\mathsf{T}}\sigma\left([\mathbf{W}_{g_2}\boldsymbol{f}_{\boldsymbol{\varepsilon},t}^{g} \parallel \mathbf{W}_{g_1}\boldsymbol{h}_{i,t}]\right)\right)}, \tag{18}$$

where $\boldsymbol{w}_2^{\mathsf{T}}$ is a weight vector, $\mathbf{W}_{g_2}$ is a linear learnable weight matrix, $\boldsymbol{f}_{\boldsymbol{e},t}^{g}$ is defined in (17), and $\parallel$ denotes the concatenation operation.

Given $\beta_{i,\boldsymbol{e}}$ defined in (18), the aggregated embedding feature $\boldsymbol{z}_{i,t}^{g}$ for agent i is given by:

$$\boldsymbol{z}_{i,t}^{g} = \sigma\left(\sum_{\boldsymbol{e}\in\mathcal{E}_i} \beta_{i\boldsymbol{e}} \mathbf{W}_{g_2} \boldsymbol{f}_{\boldsymbol{e},t}^{g}\right). \tag{19}$$

To further stabilize the attention learning process, we propose to adopt the multi-head attention [48] structure to explore the information correlation among different aspects of adjacent agents. Eq. (19) is converted to:

$$\boldsymbol{z}_{i,t}^{g} = \left\{\sigma\left(\sum\nolimits_{\boldsymbol{e}\in\mathcal{E}_i} \beta_{i\boldsymbol{e}}^{u} \mathbf{W}_{g_2}^{u} \boldsymbol{f}_{\boldsymbol{e},t}^{g}\right)\right\}_{u=1}^{U}, \tag{20}$$

where U is the number of heads, $\beta_{i\boldsymbol{e}}^{u}$ is the normalized attention score of uth head, and $\mathbf{W}_{g_2}^{u}$ is the uth linear learnable weight matrix.

Remark 3.1: The HGAT network was proposed in [39] for text classification tasks. In this study, we innovate by integrating the HGAT into a heterogeneous multiagent reinforcement learning framework for distributed decision making, for the first time. We apply the proposed hypergraph-based heterogeneous multiagent reinforcement learning framework to an emerging topic on the joint scheduling of EV routing and battery charging for multiple BSSs in a transportation network.

Different from existing multiagent reinforcement learning works that adopt conventional GNNs (see Fig. 2(c)) [34]–[36], our proposed HMA-HGAAC framework leverages a hypergraph structure (in Fig. 2(b)), which enables flexible structures of information sharing among adjacent agents and improve the learning efficiency by exploring high-order correlation (cf. the numerical results in Section IV-D).

C. Actor-critic module

PPO is an on-policy DRL algorithm based on the actor-critic framework. The structure of PPO in our multi-agent framework is similar to that in singe-agent settings, which learns a parameterized policy (the actor-network $\pi_{\boldsymbol{\theta}}$) and a parameterized value function $V_{\boldsymbol{\phi}}$. $V_{\boldsymbol{\phi}}$ aims to reduce the variance in the training process.

In the single-agent setting, the parameterized policy is updated by differentiating the policy loss [41] as follows:

$$J^{policy}(\boldsymbol{\theta}) = \mathbb{E}\Big[\min\big(\mu_t(\boldsymbol{\theta})\hat{A}_t, \text{clip}(\mu_t(\boldsymbol{\theta}), 1-\epsilon, 1+\epsilon)\hat{A}_t\big)\Big], \tag{21}$$

where ϵ is a hyperparameter that controls the clip range, $\mu_t(\boldsymbol{\theta})$ is the probability ratio:

$$\mu_t(\boldsymbol{\theta}) = \frac{\pi_{\boldsymbol{\theta}}(\boldsymbol{a}_t|\boldsymbol{s}_t)}{\pi_{\boldsymbol{\theta}_{\text{old}}}(\boldsymbol{a}_t|\boldsymbol{s}_t)}, \tag{22}$$

and $\hat{A}_t$ is the estimation of the advantage function.

According to [41], the value function (the critic network) is updated by minimizing the value loss:

$$J^{critic}(\boldsymbol{\phi}) = \mathbb{E}\Big[\min\big(V_{\boldsymbol{\phi}}(\boldsymbol{s}_t) - \hat{V}_t\big)^2\Big], \tag{23}$$

where $\hat{V}_t = \hat{A}_t + V_{\boldsymbol{\phi}}(\boldsymbol{s}_t)$.

In our multi-agent setting, for each agent $i \in \mathcal{M} \cup \mathcal{N}$, the algorithm learns a decentralized policy and a local observation-based value function.

The policy updates are clipped based on the Eq. (21) as follows,

$$J_i^{policy}(\boldsymbol{\theta}) = \mathbb{E}\Big[\min\big(\mu_{i,t}(\boldsymbol{\theta})\hat{A}_{i,t}, \text{clip}(\mu_{i,t}(\boldsymbol{\theta}), 1-\epsilon, 1+\epsilon)\hat{A}_{i,t}\big)\Big], \tag{24}$$

where the probability ratio $\mu_{i,t}(\boldsymbol{\theta})$ between its current and updated policies (different from Eq. (22)) is given by:

$$\mu_{i,t}(\boldsymbol{\theta}) = \frac{\pi_{\boldsymbol{\theta}}^{\xi_i}(a_{i,t}^{\xi_i}|\boldsymbol{z}_{i,t}^g)}{\pi_{\boldsymbol{\theta}_{\text{old}}}^{\xi_i}(a_{i,t}^{\xi_i}|\boldsymbol{z}_{i,t}^g)}, \tag{25}$$

where $\pi_{\boldsymbol{\theta}}^{\xi_i}$ represents the actor network of a ξ_i-type agent, and $\boldsymbol{z}_{i,t}^g$ is defined in (20). The advantage function $\hat{A}_{i,t}$ can be computed by the generalized advantage function (GAE) [49]:

$$\hat{A}_{i,t} = \sum_{\ell=0}^{\infty}(\gamma\lambda)^\ell \delta_{i,t+\ell}, \tag{26}$$

where $\delta_{i,t+\ell} = r_{i,t} + \gamma V_{\boldsymbol{\phi}}(\boldsymbol{z}_{i,t}^g) - V_{\boldsymbol{\phi}}(\boldsymbol{z}_{i,t}^g)$ is the TD error of $V_{\boldsymbol{\phi}}$ with a discount factor $\gamma \in [0,1]$, and λ is an exponential weight discount controlling the bias-variance tradeoff.

Apart from clipping the policy updates (in Eq. (24)), to avoid overfitting, the value clipping trick is adopted to limit the value function update to be smaller than ϵ [32]:

$$J_i^{critic}(\boldsymbol{\phi}) = \mathbb{E}\Bigg[\min\Bigg\{\Big(V_{\boldsymbol{\phi}}(\boldsymbol{z}_{i,t}^g) - \hat{V}_{i,t}\Big)^2, \Big(V_{\boldsymbol{\phi}_{\text{old}}}(\boldsymbol{z}_{i,t}^g) + \text{clip}\big(V_{\boldsymbol{\phi}}(\boldsymbol{z}_{i,t}^g) - V_{\boldsymbol{\phi}_{\text{old}}}(\boldsymbol{z}_{i,t}^g), -\epsilon, \epsilon\big) - \hat{V}_{i,t}\Big)^2\Bigg\}\Bigg], \tag{27}$$

where $\boldsymbol{\phi}_{\text{old}}$ are old parameters of value function before the update and $\hat{V}_{i,t} = \hat{A}_{i,t} + V_{\boldsymbol{\phi}}(\boldsymbol{z}_{i,t}^g)$.

Combining Eqs. (24) and (27), we obtain the total minimization objective:

$$J^{tot}(\boldsymbol{\theta}, \boldsymbol{\phi}) = \sum_{i \in \mathcal{M}\cup\mathcal{N}} (J_i^{policy}(\boldsymbol{\theta}) + \lambda_{\text{critic}} J_i^{critic}(\boldsymbol{\phi})), \tag{28}$$

where λ_{critic} is the coefficient of critic.

D. Algorithm

The proposed approach is summarized in Algorithm 1. In lines 1 and 2, the parameters of policy and critic networks are randomly initialized and the corresponding hypermeters such as the learning rate are set. In each iteration, all agents receive the initial observations from the environment. The observation of each agent is processed by the encoder and hypergraph attention modules, and then serves as the input of the policy network to output the action (in lines 6-12). In lines 13-14, all actions are executed in the environment and the information obtained by agents is stored in the replay buffer $\mathcal{D}$.

In each epoch of the optimization process (lines 17-22), the parameter $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$ are updated by conducting stochastic gradient descent on the sampled mini-batch data from the replay buffer $\mathcal{D}$ with respect to the objective function (28).

Algorithm 1 Heterogeneous Multi-Agent HyperGraph Attention PPO (HMA-HGAPPO)

1: Use Orthogonal initialization [50] to initialize the parameters $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$ for policy $\pi_{\boldsymbol{\theta}}$ and critic $V_{\boldsymbol{\phi}}$, respectively
2: Set λ, λ_{critic}, learning rate α
3: **for** each iteration **do**
4: set replay buffer $\mathcal{D}$
5: **for** 1 **to** $batch\_size$ **do**
6: Receive initial observation $\{\boldsymbol{o}_{i,0}^{\xi_i}\}_{i\in\mathcal{M}\cup\mathcal{N}}$ for all agents
7: **for** $t = 1$ **to** T **do**
8: **for all** agents i **do**
9: Compute $\boldsymbol{h}_{i,t}$ according to (15)
10: Compute $\boldsymbol{z}_{i,t}^g$ according to (20)
11: $a_{i,t}^{\xi_i} = \pi_{\boldsymbol{\theta}}^{\xi_i}(\boldsymbol{z}_{i,t}^g)$
12: **end for**
13: Execute actions $\{a_{i,t}^{\xi_i}\}_{i\in\mathcal{M}\cup\mathcal{N}}$ and get the cost $\{r_{i,t}^{\xi_i}\}_{i\in\mathcal{M}\cup\mathcal{N}}$, next observation $\{\boldsymbol{o}_{i,t+1}^{\xi_i}\}_{i\in\mathcal{M}\cup\mathcal{N}}$
14: Store $[\{\boldsymbol{o}_{i,t}^{\xi_i}, a_{i,t}^{\xi_i}, r_{i,t}^{\xi_i}, \boldsymbol{o}_{i,t+1}^{\xi_i}\}_{i\in\mathcal{M}\cup\mathcal{N}}]$ into replay buffer $\mathcal{D}$
15: **end for**
16: **end for**
17: **for** each epoch **do**
18: Sample mini-batch from $\mathcal{D}$
19: Compute advantage $\hat{A}$ by GAE on τ
20: Compute cost-to-go $\hat{R}$ on τ
21: Update $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$ on $J^{tot}(\boldsymbol{\theta}, \boldsymbol{\phi})$ by Adam
22: **end for**
23: **end for**

IV. Numerical Results

In this section, we numerically benchmark the performance of the proposed method with other approaches on real-world data.

A. Experiment Setup

1) Dataset: We obtain 365 days' real-world electricity pricing and renewable generation data from CAISO [51] in 2022. The data from the first 20 days of each month are used for training, and the remaining days' data of each month are used for testing.

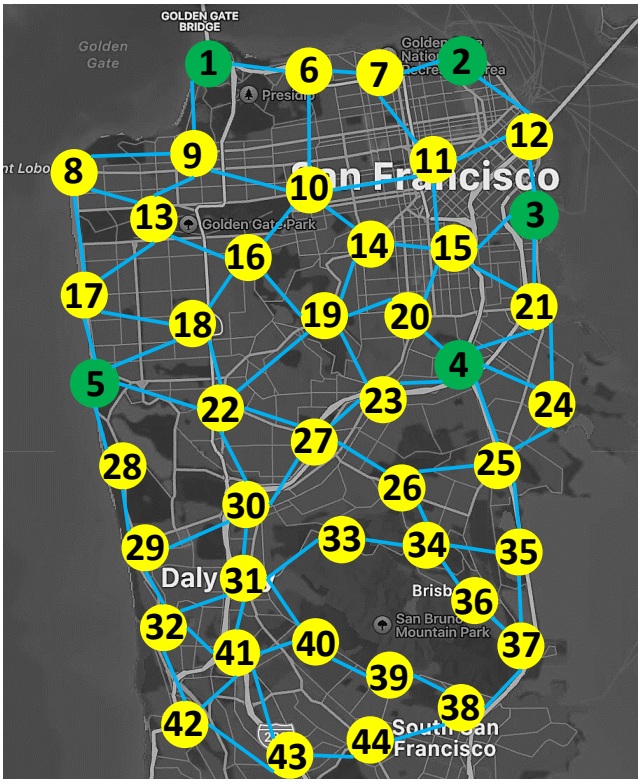


Fig. 3. A 44-node transportation network in San Francisco.

2) Environment Settings: As shown in Fig. 3, the transportation network is abstracted from the center of San Francisco. It has 44 transportation nodes and 73 roads, with 5 green nodes representing five real-world BSSs. In Fig. 4, the larger map covers an extended area of the San Francisco city with 180 transportation nodes and 280 roads, where the 11 green nodes represent 11 real-world BSSs. The approximated travel time between nodes is obtained from Google Maps.

Each BSS can charge three types of batteries, *i.e.*, Nissan Leaf [52], Audi Q4 e-tron [53], and Tesla Model S [54] with capacity of 40kWh, 80 kWh, and 100 kWh, respectively. On the 44-node transportation network, initially there are 14 batteries of each type with 7 FBs and 7 batteries in CBs at each BSS. There are in total N = 900 EVs with 300 EVs of each type in the system. Over the 180-node transportation network, initially each BSS has 26 batteries of each type with 13 FBs and 13 batteries in CBs. There are in total N = 1800 EVs with 600 EVs of each type in the system.

Each episode is one day with each timeslot lasting for 10 minutes ($T = 144$). The charging power of each CB is 7.5 kWh/timeslot and the target SoC = 90% [3].

The arrival of battery swapping demands follows the Poisson distribution with a rate of $N/100$. The initial locations and destinations of EVs are randomly selected from the transportation network [55]. The initial SoCs of EVs follow a uniform distribution over $[0.3, 0.85]$ [56].

We use the monetary value of time to measure the travel and waiting cost with $c = 1.4$ \$/timeslot [55]. We set the energy consumption per unit timeslot as $\zeta = 1$ kWh/timeslot, according to the approximate distances between nodes and the EVs' driving range [53], [54].

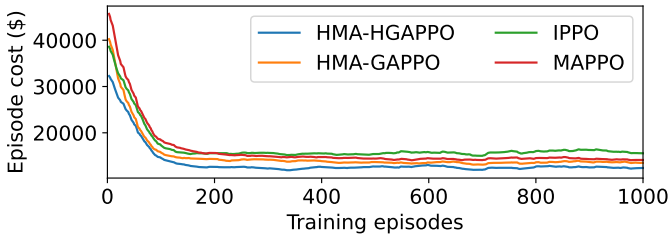


Fig. 5. Training curves smoothed with a moving average of 40 points on the 44-node transportation network.

3) Algorithms: We benchmark the following methods where all homogeneous agents share the same networks [57].

- HMA-HGAPPO: Our proposed approach in Section III where adjacent agents can communicate with each other (cf. Algorithm 1).
- Heuristic1: HMA-HGAPPO makes only charging decisions and all EVs go to the nearest BSSs for battery swapping.
- Heuristic2: HMA-HGAPPO makes only decisions for routing and BSSs greedily charge batteries in CBs.
- IPPO [32] is a PPO based DTDE approach for MADRL, where all agents are independent.
- MAPPO [31] is a CTDE algorithm for MADRL. The actor takes actions based on its own observation, while the critic can access system state during training.
- HMA-GAPPO is a variant of HMA-HGAPPO that adopts the traditional GAT network [58] instead of HGAT. The adjacent matrix is manually defined according to [59].

4) Implementation Details: All algorithms are implemented on a Windows server with 6-core Intel(R) Core(TM) i5-10600K CPU @ 4.10GHz and NVIDIA GeForce RTX 3080. For HMA-HGAPPO, there are three fully-connected hidden layers (with 64 neurons in each layer) in the encoder module. The settings of the hypergraph attention module are the same as in [39]. For the actor-critic module, hyperparameter settings are set as the default settings in [31].

The hyperparameters of other algorithms are the same as in HMA-HGAPPO, except that each fully-connected layer in MAPPO adopts 8 units due to the computational burden. All algorithms are trained for 1000 episodes and the testing phase estimates the average system cost in 100 episodes.

B. Simulation Results on the 44-node Transportation Network

In this subsection, we conduct numerical experiments on a 44-node transportation network to verify the effectiveness of our proposed algorithm, HMA-HGAPPO.

Fig. 5 illustrates the training curves of four algorithms. All methods converge after 200 training episodes. The proposed method, HMA-HGAPPO, achieves the best learning performance.

Table I compares the average system cost achieved over 100 iterations on the testing environment. Compared with Heuristic1 and Heuristic2, the proposed joint scheduling algorithm

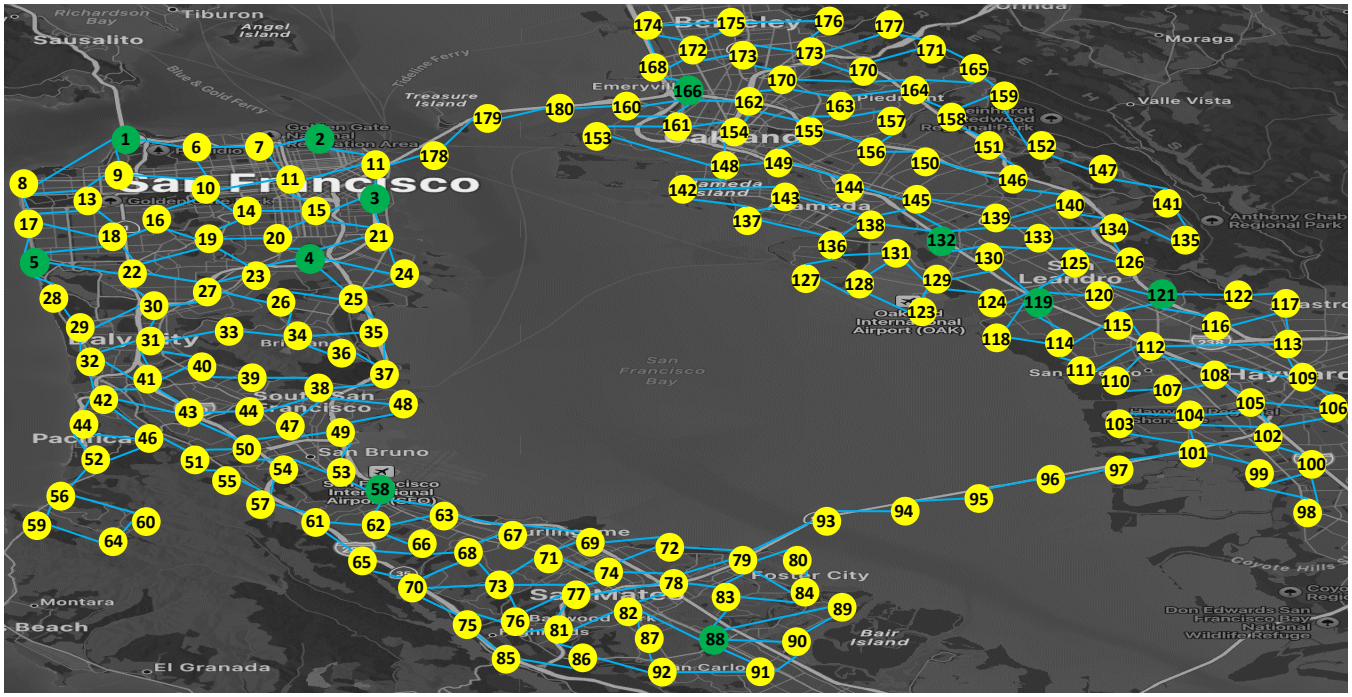


Fig. 4. A 180-node transportation network in San Francisco.

TABLE I
AVERAGE TOTAL SYSTEM COST COMPARISON ON THE 44-NODE TRANSPORTATION NETWORK

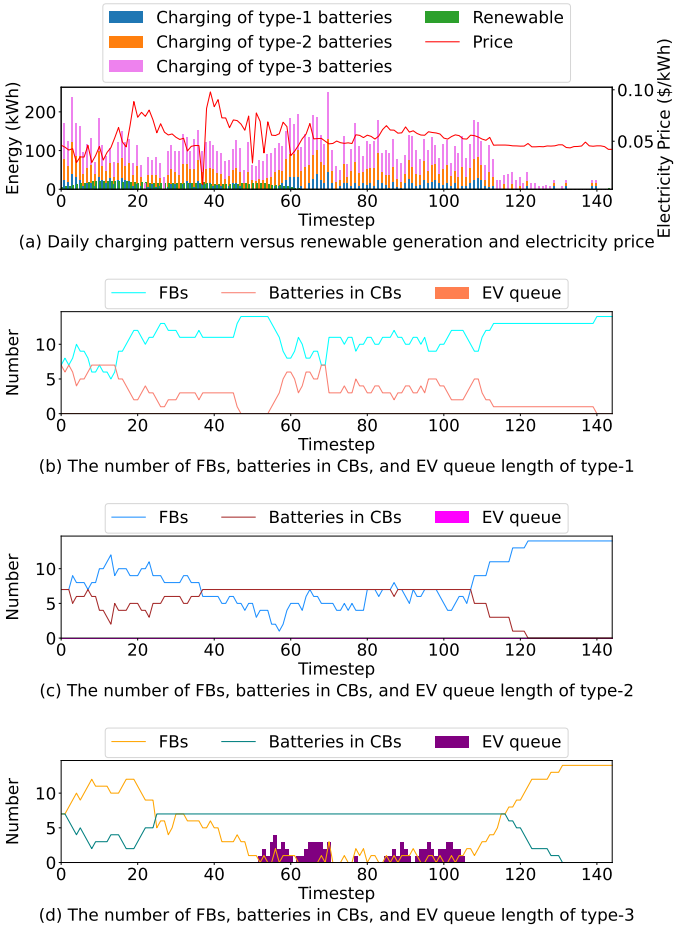
Algorithm	Cost ($)	Training time (h)
Heuristic1	34562.21	12.74
Heuristic2	25493.64	12.95
IPPO	18425.36	10.71
MAPPO	15780.98	15.25
HMA-GAPPO	14262.74	13.85
HMA-HGAPPO	13698.25	12.30

HMA-HGAPPO reduces 60.6% and 45.3% of the average system cost, respectively. In comparison with IPPO, HMA-HGAPPO reduces 25.6% of the average system cost. This is due to the information sharing among adjacent agents enabled by HGAT. HMA-HGAPPO reduces 13.1% of the average system cost achieved by MAPPO.

Compared to HMA-GAPPO, HMA-HGAPPO reduces 3.9% of the average system cost. This is due to i) HMA-GAPPO needs to manually define the connections and relationships of agents, and ii) simple GNNs cannot efficiently discover higher-order correlation among agents as the adopted hypergraph.

C. Case Study on the 44-node Transportation Network

1) Charging pattern: In Fig. 6, for BSS 3, we present the daily energy purchase from the grid, the number of batteries, and the EV queue length (resulting from HMA-HGAPPO) under random renewable generation and electricity prices. The HMA-HGAPPO approach procures a high amount of energy and greedily allocates the energy to batteries so as to reduce the queue length ($i.e.$, waiting cost) until the EV queue is empty.



(a) Daily charging pattern versus renewable generation and electricity price

(b) The number of FBs, batteries in CBs, and EV queue length of type-1

(c) The number of FBs, batteries in CBs, and EV queue length of type-2

(d) The number of FBs, batteries in CBs, and EV queue length of type-3

Fig. 6. The daily charging pattern (under the proposed HMA-HGAPPO approach) at BSS 3 on the 57th day (starting from 7am) on the 44-node transportation network. Each timeslot lasts for 10 minutes.

Over the operation horizon of 24 hours, in the first plot of Fig. 6, when the electricity price is high (from 10:00 to 11:30 and from 13:20 to 15:10), the HMA-HGAPPO procures less energy to reduce the charging cost. We note from the fourth plot that when the queue of type-3 EVs (from 15:40 to 18:40 and from 21:10 to 00:10 on the next day) is long, the HMA-HGAPPO algorithm greedily charges batteries to provide FBs to meet the swapping demands. At the end of the operation horizon, all batteries at the BSS are fully charged so as to provide enough FBs in the beginning of the next horizon.

2) Routing pattern: Fig. 7 shows the routing trajectory (node $16-10-6-1$) of a type-3 EV from timeslot $t=54$ (16:00) to $t=57$ (16:30). At $t=54$, in the first subplot, an EV with swapping demand enters the system at node 16. Although BSS 5 is closer to the EV, it has a long queue of battery swapping demands. The EV chooses to route to BSS 1 to avoid the long waiting time at BSS 5.

D. Simulation Results on the 180-node Transportation Network

Fig. 8 illustrates the training curves of four algorithms on the 180-node transportation network. All methods converge after 400 training episodes. The proposed method, HMA-HGAPPO, achieves the best learning performance.

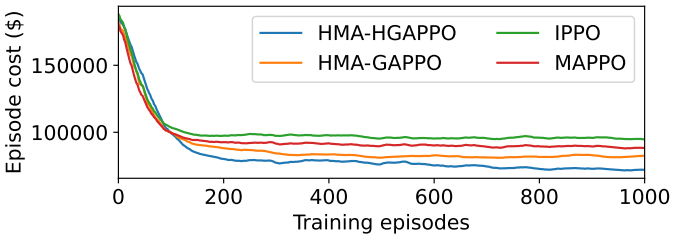


Fig. 8. Training curves smoothed with a moving average of 40 points on the 180-node transportation network.

Table II compares the average system cost achieved over 100 iterations on the testing environment. Compared with Heuristic1 and Heuristic2, the proposed joint scheduling algorithm HMA-HGAPPO reduces 51.0% and 46.3% of the average system cost, respectively. In comparison with IPPO, HMA-HGAPPO reduces 23.5% of the average system cost. This is due to the information sharing among adjacent agents enabled by HGAT. HMA-HGAPPO reduces 18.9% of the average system cost achieved by MAPPO.

TABLE II
AVERAGE TOTAL SYSTEM COST COMPARISON ON THE 180-NODE TRANSPORTATION NETWORK

Algorithm	Cost ($)	Training time (h)
Heuristic1	150234.2	28.52
Heuristic2	136985.7	28.74
IPPO	96265.13	25.29
MAPPO	90745.32	40.45
HMA-GAPPO	84854.54	30.25
HMA-HGAPPO	73549.63	28.18

Compared to HMA-GAPPO, HMA-HGAPPO reduces 13.3% of the average system cost. This is due to the facts that HMA-GAPPO needs to manually define the connections and relationships of agents, and that the hypergraph (adopted in HMA-HGAPPO) can discover higher-order correlation among agents much more efficiently than the simple GNNs (in HMA-GAPPO).

E. Case Study on the 180-node Transportation Network

1) Charging pattern: In Fig. 9, for BSS 9 and the proposed approach HMA-HGAPPO, we present the daily energy purchase from the grid, the dynamics of the number of batteries and the EV queue length under random renewable generation and electricity prices. Over the operation horizon of 24 hours, in the first plot of Fig. 9, when the electricity price is high (from 10:00 to 11:30 and from 13:20 to 15:10), the HMA-HGAPPO procures less energy to reduce the charging cost. We note from the third and fourth plots that when the queues of type-2 EVs (from 18:40 to 01:00 on the next day) and type-3 EVs (from 17:20 to 23:10) are long, the HMA-HGAPPO algorithm greedily charges batteries to provide FBs to meet the swapping demands. At the end of the operation horizon, all batteries at the BSS are fully charged so as to provide enough FBs at the beginning of the next horizon.

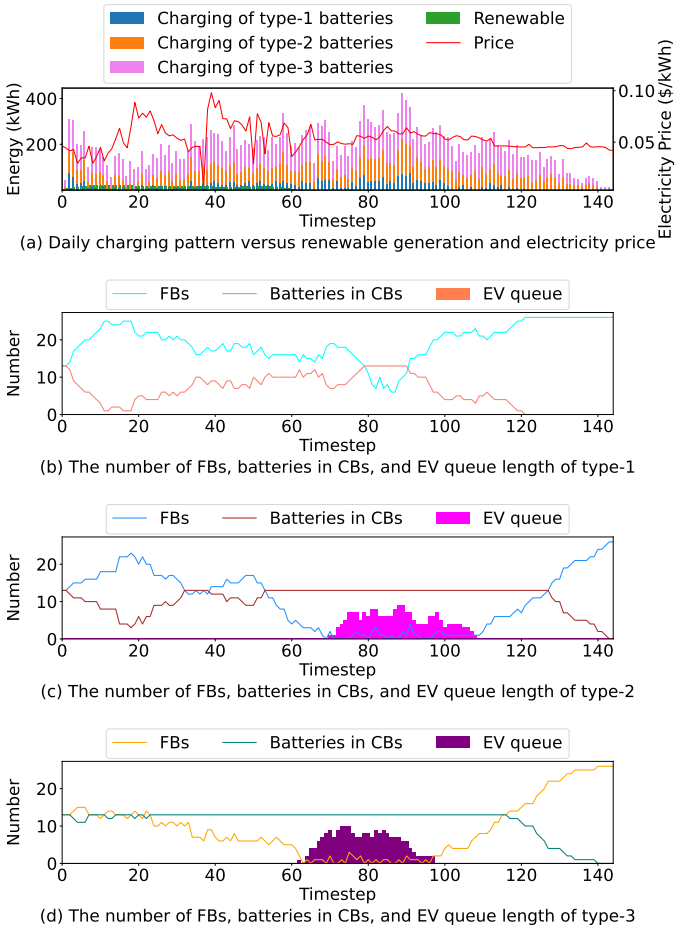


Fig. 9. The daily charging pattern (under the proposed HMA-HGAPPO approach) at BSS 9 on the 57th day (starting from 7am) on the 180-node transportation network. Each timeslot lasts for 10 minutes.

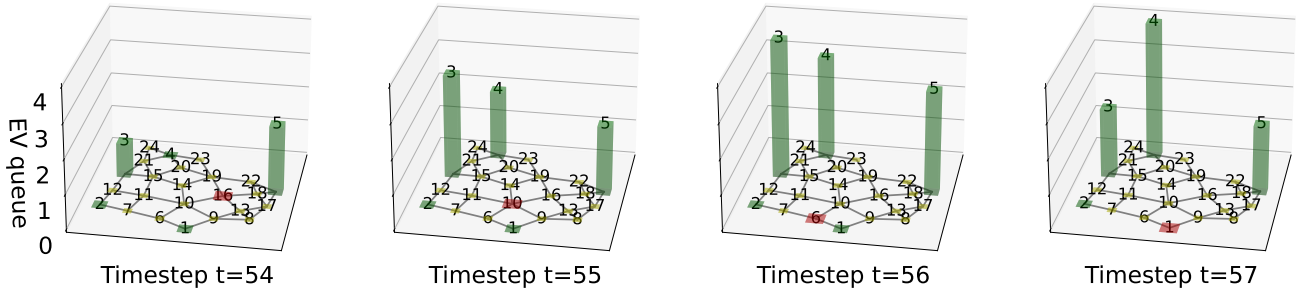


Fig. 7. The 57th day's routing trajectory of a type-3 EV on the 44-node transportation network where BSSs are located at nodes 1-5. The red diamond represents the EV's current location. The green cuboids represent the current EV queue lengths at the BSSs.

2) Routing pattern: Fig. 10 shows a full routing trajectory (node $141-135-134-133-125-119$) of type-3 EV from timeslot $t = 71$ (18:50) to $t = 76$ (19:40). At $t = 71$, in the first subplot, an EV with swapping demand enters the system at node 141. Although BSS 9 is closer to the EV, it has a long queue of battery swapping demands. The EV chooses to route to BSS 8 to avoid the long waiting time at BSS 9.

F. Case Study under Different Target SoCs

In addition to the numerical results under the target SoC of 90%, in this subsection we conduct numerical experiments on the 180-node transportation network under a target SoC of 95% [3], [60] to demonstrate the performance of the proposed algorithm HMA-HGAPPO.

Fig. 11 shows the training curves of four algorithms with target SoC 95%. All methods converge after 400 training episodes. The proposed method, HMA-HGAPPO, achieves the best learning performance.

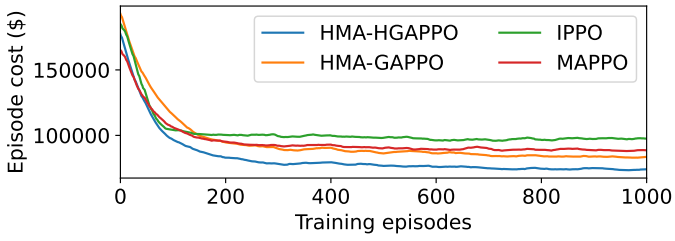


Fig. 11. Training curves smoothed with a moving average of 40 points with target SoC = 95% on the 180-node transportation network .

Table III compares the average system cost under different target SoCs of 90% and 95%. Higher target SoC leads to higher system cost, because the system needs to procure more energy to charge batteries to meet the higher target SoC. The proposed method HMA-HGAPPO achieves the best performance under different target SoCs.

TABLE III
AVERAGE TOTAL SYSTEM COST COMPARISON UNDER DIFFERENT TARGET SOCS ON THE 180-NODE TRANSPORTATION NETWORK

Algorithms	Cost ($)	
	Target SoC = 90%	Target SoC = 95%
Heuristic1	150234.2	158964.4
Heuristic2	136985.7	142671.0
IPPO	96265.13	99012.27
MAPPO	90745.32	92145.36
HMA-GAPPO	84854.54	85479.21
HMA-HGAPPO	73549.63	76127.86

G. The Incorporation of Estimated Shortest Travel Time

In this subsection, we conduct numerical experiments on the 180-node transportation networks to demonstrate the efficacy of incorporating estimated shortest travel time to BSSs, $\{\tau_{n,m,t}\}$, into the proposed approach HMA-HGAPPO.

Fig. 12 shows the training curves of HMA-HGAPPO with and without incorporating the estimated shortest travel time. We observe that the incorporation of estimated shortest travel time significantly enhances the performance.

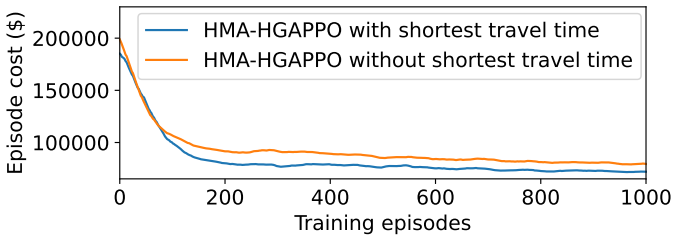


Fig. 12. Training curves smoothed with a moving average of 40 points with/without the incorporation of estimated shortest travel time, on the 180-node transportation network.

Table IV compares the average system cost achieved over 100 iterations on the testing environment. With the incorporation of estimated shortest travel time, HMA-HGAPPO reduces 15.3% of the average system cost, due to the efficient EV routing decisions based on the estimated travel time.

TABLE IV
AVERAGE TOTAL SYSTEM COST OF THE PROPOSED HMA-HGAPPO APPRAOCH WITH AND WITHOUT THE ESTIMATED TRAVEL TIME

	Cost ($)	Training time (h)
Without estimated travel time	84526.36	28.02
With estimated travel time	73549.63	28.18

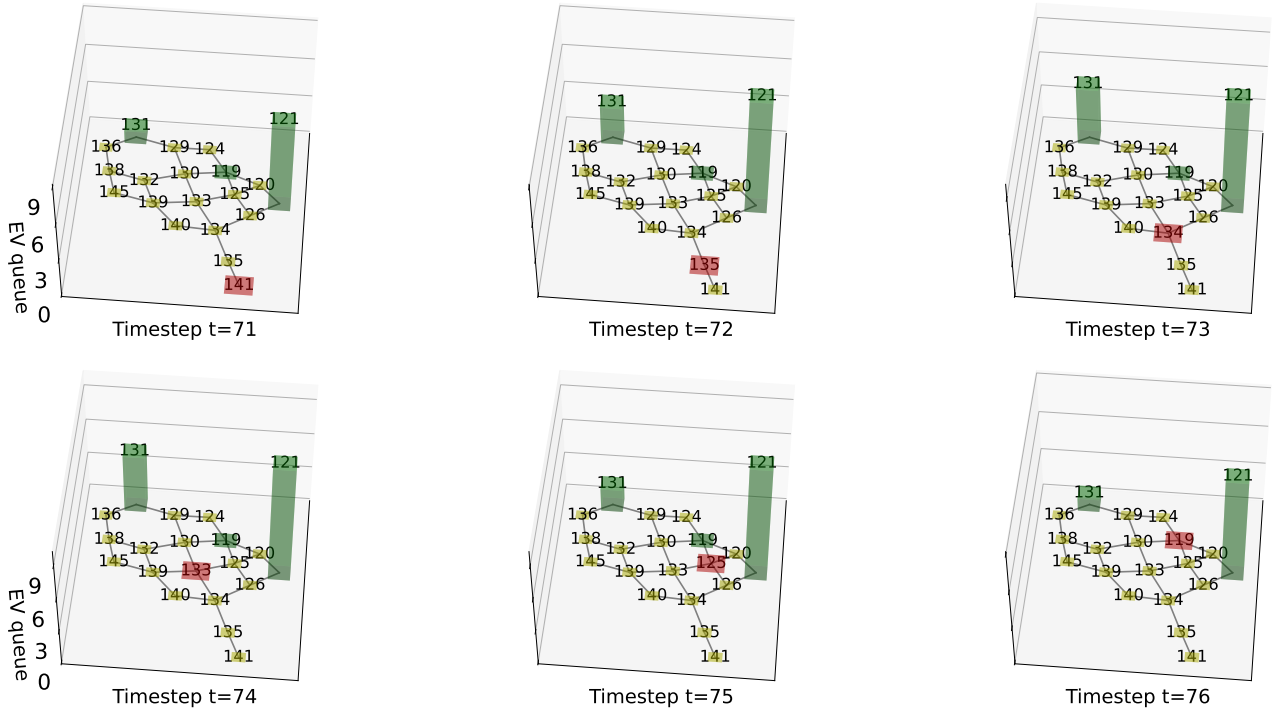


Fig. 10. The 57th day's routing trajectory of a type-3 EV on the 180-node transportation network where BSSs are located at nodes 119, 121 and 131. The red diamond represents the EV's current location. The green cuboids represent the current EV queue lengths at the BSSs.

V. Conclusion

We study the dynamic joint scheduling of EV routing and battery charging in a transportation network with multiple BSSs. We formulate the joint scheduling problem as a Dec-POMDP, with explicit incorporation of random EV swapping demand, renewable generation, and electricity prices. We leverage the hypergraph structure to depict the connection of multiple agents (EVs and BSSs) at the same location of a transportation network. We propose a new HMA-HGAAC framework that integrates hypergraph attention (HGAT) networks into a heterogeneous multiagent actor-critic framework to enhance the learning efficiency for distributed decision making. Numerical experiments on real-world data demonstrate the advantage of the proposed approach over multiple benchmarks.

We briefly discuss several future research directions. It would be interesting to simulate the proposed HMA-HGAPPO approach over larger transportation networks of practical size, leveraging more powerful GPUs with larger Video Random Access Memory (VRAM). The proposed approach can be modified to output continuously controllable charging rates for a large number of EVs, by letting the policy network output the mean and standard deviation of the Gaussian policy. We mention this as another interesting extension of the present work.

References

[1] A. Guterres, "Carbon neutrality by 2050: the worlds most urgent mission— united nations secretary-general," 2020.

[2] IEA, "World energy outlook 2022," 2022.

[3] H. Wu, "A survey of battery swapping stations for electric vehicles: Operation modes and decision scenarios," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 8, pp. 10 163–10 185, 2021.

[4] Y. Liang, Z. Ding, T. Zhao, and W.-J. Lee, "Real-time operation management for battery swapping-charging system via multi-agent deep reinforcement learning," *IEEE Trans. Smart Grid*, vol. 14, no. 1, pp. 559–571, 2022.

[5] H. Ko, S. Pack, and V. C. Leung, "An optimal battery charging algorithm in electric vehicle-assisted battery swapping environments," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 5, pp. 3985–3994, 2020.

[6] M. Ban, M. Shahidehpour, J. Yu, and Z. Li, "A cyber-physical energy management system for optimal sizing and operation of networked nanogrids with battery swapping stations," *IEEE Trans. Sustain. Energy.*, vol. 10, no. 1, pp. 491–502, 2017.

[7] A. A. Shalaby, M. F. Shaaban, M. Mokhtar, H. H. Zeineldin, and E. F. El-Saadany, "A dynamic optimal battery swapping mechanism for electric vehicles using an lstm-based rolling horizon approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 9, pp. 15 218–15 232, 2022.

[8] Q. Kang, J. Wang, M. Zhou, and A. C. Ammari, "Centralized charging strategy and scheduling algorithm for electric vehicles under a battery swapping scenario," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 3, pp. 659–669, 2015.

[9] H. Wu, G. K. H. Pang, K. L. Choy, and H. Y. Lam, "An optimization model for electric vehicle battery charging at a battery swapping station," *IEEE Trans. Veh. Technol.*, vol. 67, no. 2, pp. 881–895, 2017.

[10] Y. Li, Z. Yang, G. Li, Y. Mu, D. Zhao, C. Chen, and B. Shen, "Optimal scheduling of isolated microgrid with an electric vehicle battery swapping station in multi-stakeholder scenarios: A bi-level programming approach via real-time pricing," *Appl. Energy*, vol. 232, pp. 54–68, 2018.

[11] J. Zhang, X. Li, D. Jia, and Y. Zhou, "A bi-level programming for union battery swapping stations location-routing problem under joint distribution and cost allocation," *Energy*, vol. 272, p. 127152, 2023.

[12] X. Liu, D. Wang, Y. Yin, and T. Cheng, "Robust optimization for the electric vehicle pickup and delivery problem with time windows and uncertain demands," *Comput. Oper. Res.*, vol. 151, p. 106119, 2023.

[13] M. Schiffer and G. Walther, "Strategic planning of electric logistics fleet networks: A robust location-routing approach," *Omega*, vol. 80, pp. 31–42, 2018.

[14] M. Mahoor, Z. S. Hosseini, and A. Khodaei, "Least-cost operation of a battery swapping station with random customer requests," *Energy*, vol. 172, pp. 913–921, 2019.

[15] T. Zhang, X. Chen, Z. Yu, X. Zhu, and D. Shi, "A monte carlo simulation approach to evaluate service capacities of ev charging and battery

swapping stations," *IEEE Trans. Industr. Inform.*, vol. 14, no. 9, pp. 3914–3923, 2018.

[16] J. Jin, Y. Xu, and Z. Yang, "Optimal deadline scheduling for electric vehicle charging with energy storage and random supply," *Automatica*, vol. 119, p. 109096, 2020.

[17] J. Jin, S. Mao, and Y. Xu, "Optimal priority rule enhanced deep reinforcement learning for charging scheduling in an electric vehicle battery swapping station," *IEEE Trans. Smart Grid*, 2023.

[18] L. Ni, B. Sun, X. Tan, and D. H. Tsang, "Inventory planning and real-time routing for network of electric vehicle battery-swapping stations," *IEEE Trans. Transp. Electrif.*, vol. 7, no. 2, pp. 542–553, 2020.

[19] F. Schneider, U. W. Thonemann, and D. Klabjan, "Optimization of battery charging and purchasing at electric vehicle battery swap stations," *Transp. Sci.*, vol. 52, no. 5, pp. 1211–1234, 2018.

[20] D. Khosrowshahi, "Autonomous rides are arriving on uber with waymo," https://www.uber.com/newsroom/waymo-on-uber/, 2023.

[21] J. Zigoris, "Even robots need rest: Where waymos go to roost in san francisco," https://sfstandard.com/2023/10/06/san-francisco-waymo-robotaxi-maintenance-parking-lot/, 2023.

[22] Cruise, "Delivery made driverless," https://getcruise.com/delivery/, 2024.

[23] T. eDRV, "Ev charging for self-driving fleets: Webinar with darren hau," https://www.edrv.io/blog/ev-charging-for-self-driving-fleets-webinar-with-darren-hau, 2023.

[24] Z. Ding, W. Tan, W.-J. Lee, X. Pan, and S. Gao, "Integrated operation model for autonomous mobility-on-demand fleet and battery swapping station," *IEEE Trans. Ind. Appl.*, vol. 57, no. 6, pp. 5593–5602, 2021.

[25] Z. Ding, W. Tan, W. Lu, and W.-J. Lee, "Quality-of-service aware battery swapping navigation and pricing for autonomous mobility-on-demand system," *IEEE Trans. Industr. Inform.*, vol. 18, no. 11, pp. 8247–8257, 2022.

[26] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[27] T. Rashid, M. Samvelyan, C. S. De Witt, G. Farquhar, J. Foerster, and S. Whiteson, "Monotonic value function factorisation for deep multi-agent reinforcement learning," *The Journal of Machine Learning Research (JMLR)*, vol. 21, no. 1, pp. 7234–7284, 2020.

[28] Y. Wang, D. Qiu, and G. Strbac, "Multi-agent reinforcement learning for electric vehicles joint routing and scheduling strategies," in *2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2022, pp. 3044–3049.

[29] M. Alqahtani, M. J. Scott, and M. Hu, "Dynamic energy scheduling and routing of a large fleet of electric vehicles using multi-agent reinforcement learning," *Comput. Ind. Eng.*, vol. 169, p. 108180, 2022.

[30] D. Qiu, Y. Wang, M. Sun, and G. Strbac, "Multi-service provision for electric vehicles in power-transportation networks towards a low-carbon transition: A hierarchical and hybrid multi-agent reinforcement learning approach," *Appl. Energy*, vol. 313, p. 118790, 2022.

[31] C. Yu, A. Velu, E. Vinitsky, J. Gao, Y. Wang, A. Bayen, and Y. Wu, "The surprising effectiveness of ppo in cooperative multi-agent games," *Advances in Neural Information Processing Systems (NIPS)*, vol. 35, pp. 24 611–24 624, 2022.

[32] C. S. de Witt, T. Gupta, D. Makoviichuk, V. Makoviychuk, P. H. Torr, M. Sun, and S. Whiteson, "Is independent learning all you need in the starcraft multi-agent challenge?" *arXiv preprint arXiv:2011.09533*, 2020.

[33] T. Qian, C. Shao, X. Li, X. Wang, Z. Chen, and M. Shahidehpour, "Multi-agent deep reinforcement learning method for ev charging station game," *IEEE Trans. Power Syst.*, vol. 37, no. 3, pp. 1682–1694, 2021.

[34] J. Jiang, C. Dun, T. Huang, and Z. Lu, "Graph convolutional reinforcement learning," *arXiv preprint arXiv:1810.09202*, 2018.

[35] Y. Niu, R. R. Paleja, and M. C. Gombolay, "Multi-agent graph-attention communication and teaming." in *International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2021, pp. 964–973.

[36] E. Seraj, Z. Wang, R. Paleja, D. Martin, M. Sklar, A. Patel, and M. Gombolay, "Learning efficient diverse communication for cooperative heterogeneous teaming," in *International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2022, pp. 1173–1182.

[37] Y. Feng, H. You, Z. Zhang, R. Ji, and Y. Gao, "Hypergraph neural networks," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, 2019, pp. 3558–3565.

[38] S. Bai, F. Zhang, and P. H. Torr, "Hypergraph convolution and hypergraph attention," *Pattern Recognit.*, vol. 110, p. 107637, 2021.

[39] K. Ding, J. Wang, J. Li, D. Li, and H. Liu, "Be more with less: Hypergraph attention networks for inductive text classification," *arXiv preprint arXiv:2011.00387*, 2020.

[40] F. A. Oliehoek and C. Amato, *A concise introduction to decentralized POMDPs*. Springer, 2016.

[41] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.

[42] M. A. H. Rafi and J. Bauman, "A comprehensive review of DC fast-charging stations with energy storage: Architectures, power converters, and analysis," *IEEE Trans. Transp. Electrif.*, vol. 7, no. 2, pp. 345–368, 2021.

[43] T. Qian, C. Shao, X. Wang, and M. Shahidehpour, "Deep reinforcement learning for ev charging navigation by coordinating smart grid and intelligent transportation system," *IEEE Trans. Smart Grid*, vol. 11, no. 2, pp. 1714–1723, 2019.

[44] J. Jin and Y. Xu, "Shortest-path-based deep reinforcement learning for ev charging routing under stochastic traffic condition and electricity prices," *IEEE Internet Things J.*, vol. 9, no. 22, pp. 22 571–22 581, 2022.

[45] A. Rahman, O. Farrok, and M. M. Haque, "Environmental impact of renewable energy source based electrical power plants: Solar, wind, hydroelectric, biomass, geothermal, tidal, ocean, and osmotic," *Renew. Sustain. Energy Rev.*, vol. 161, p. 112279, 2022.

[46] A. El Hammoumi, S. Chtita, S. Motahhir, and A. El Ghzizal, "Solar pv energy: From material to use, and the most commonly used techniques to maximize the power output of pv systems: A focus on solar trackers and floating solar panels," *Energy Reports*, vol. 8, pp. 11 992–12 010, 2022.

[47] C. Wakilpoor, P. J. Martin, C. Rebhuhn, and A. Vu, "Heterogeneous multi-agent reinforcement learning for unknown environment mapping," *arXiv preprint arXiv:2010.02663*, 2020.

[48] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in Neural Information Processing Systems (NIPS)*, vol. 30, 2017.

[49] J. Schulman, P. Moritz, S. Levine, M. Jordan, and P. Abbeel, "High-dimensional continuous control using generalized advantage estimation," *arXiv preprint arXiv:1506.02438*, 2015.

[50] H. Hu and J. N. Foerster, "Simplified action decoder for deep multi-agent reinforcement learning," *arXiv preprint arXiv:1912.02288*, 2019.

[51] CAISO, "Real-time price," http://www.energyonline.com/Data/, 2022, accessed April 13, 2021.

[52] E. Database, "Nissan leaf performance," https://ev-database.org/uk/car/1656/Nissan-Leaf, 2020.

[53] ——, "Audi Q4 e-tron 4 performance," https://ev-database.org/car/1490/Audi-Q4-e-tron-40, 2020.

[54] ——, "Tesla model s performance," https://ev-database.org/car/1324/Tesla-Model-S-Performance-efficiency, 2020.

[55] X. Li, Y. Cao, S. Wan, S. Liu, H. Lin, and Y. Zhu, "A coordinated battery swapping service management scheme based on battery heterogeneity," *IEEE Trans. Transp. Electrif.*, 2023.

[56] J. Yang, W. Liu, K. Ma, Z. Yue, A. Zhu, and S. Guo, "An optimal battery allocation model for battery swapping station of electric vehicles," *Energy*, vol. 272, p. 127109, 2023.

[57] J. K. Gupta, M. Egorov, and M. Kochenderfer, "Cooperative multi-agent control using deep reinforcement learning," in *Autonomous Agents and Multiagent Systems: AAMAS 2017 Workshops*. Springer, 2017, pp. 66–83.

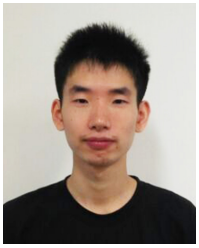
[58] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," *arXiv preprint arXiv:1710.10903*, 2017.

[59] Q. Xing, Y. Xu, Z. Chen, Z. Zhang, and Z. Shi, "A graph reinforcement learning-based decision-making platform for real-time charging navigation of urban electric vehicles," *IEEE Trans. Industr. Inform.*, 2022.

[60] B. Wang, D. Zhao, P. Dehghanian, Y. Tian, and T. Hong, "Aggregated electric vehicle load modeling in large-scale electric power systems," *IEEE Trans. Ind. Appl.*, vol. 56, no. 5, pp. 5796–5810, 2020.

This article has been accepted for publication in IEEE Transactions on Smart Grid. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSG.2024.3386609

Shuai Mao received the M.Sc. degree in power systems engineering from University College London, London, U.K., in 2019. He is currently pursuing the Ph.D. degree in the Mechanical and Automation Engineering department at the Chinese University of Hong Kong (CUHK), Hong Kong SAR. His research interests include deep reinforcement learning and its applications in intelligent transportation systems.



Jiangliang Jin (S'16) received the B.S. degree from Fudan University, Shanghai, China, in 2010, the M.S. degree from Nanyang Technological University, Singapore, in 2013, and the Ph.D. degree from the Singapore University of Technology and Design, Singapore, in 2018.

He is currently an Associate Professor with the College of Information Science and Technology, Donghua University. Before joining Donghua University, he was a Research Postdoctoral Fellow with the Chinese University of Hong Kong from 2018 to 2022. His research interest includes stochastic optimal control and demand-side management for power systems. He was a recipient of the SHIAE Fellowship and the Shanghai Pujiang Talent Award.

Yunjian Xu (S'06-M'10) received the B.S. and M.S. degrees in Electronic Engineering from Tsinghua University, Beijing, China, in 2006 and 2008, respectively, and the Ph.D. degree from the Massachusetts Institute of Technology (MIT), Cambridge, MA, USA, in 2012. Dr. Xu was a CMI (Center for the Mathematics of Information) postdoctoral fellow at the California Institute of Technology, Pasadena, CA, USA, in 2012-2013.

He is now an Associate Professor at the Department of Mechanical and Automation Engineering, the Chinese University of Hong Kong (CUHK). He was an assistant professor at the Singapore University of Technology and Design in 2013-2017. His research interests lie in robust reinforcement learning, stochastic optimal control, power system control and optimization, and wholesale electricity market. Dr. Xu was a recipient of the MIT-Shell Energy Fellowship.