JAISCR, 2027, Vol. 17, No. 1, pp. 5 – 18
10.2478/jaiscr-2026-0021

MULTIAGENT DYNAMIC TASK ALLOCATION BASED ON GRAPH NEURAL REINFORCEMENT LEARNING ALGORITHM

Xuexiu Liang [1], Agnieszka Siwocha [2], Yu Xia[3,*]

[1]*China Software Testing Center (MIIT Software and Integrated Circuit Promotion Center), Beijing 100048, China*

[2]*Information Technology Institute, SAN University, 90-113 Łódź, Poland*

[3]*State Key Laboratory of Mechanical System and Vibration, Shanghai Jiao Tong University, Shanghai, 200240, China*

[*]*E-mail: rainyx@sjtu.edu.cn*

Submitted: 16th January 2026; Accepted: 22nd May 2026

Abstract

Multi-agent dynamic task allocation (MADTA) for UAV swarm and autonomous systems remains a formidable challenge in highly uncertain and stochastic environments, where conventional reinforcement learning methods struggle with variable input dimensions and coordination conflicts. This paper proposes a spatiotemporal topology-aware graph reinforcement learning (STA-GRL) framework to address these limitations. By modeling the environment as a dynamic bipartite graph, the framework integrates a spatiotemporal gated graph attention (STGGA) module that employs a temporal gating mechanism to dynamically prioritize tasks with rapidly decaying deadlines. A topology-aware critic is further designed to penalize spatial conflicts among agents via an enhanced adjacency matrix. Extensive simulations demonstrate that STA-GRL significantly surpasses state-of-the-art baselines. In the primary evaluation scenario with 30 agents and an intermediate task arrival rate ($\lambda = 0.6$), STA-GRL achieves a task completion rate of 86.8% and an average response time of 18.4 seconds, while reducing the average conflict rate to just 2.1%. Moreover, ablation studies confirm the critical contribution of each architectural component, with the temporal gate improving the completion rate by 7.3% and the topology-aware critic reducing conflicts by 6.4%.

Keywords: Dynamic task allocation, multiagent reinforcement learning, graph attention network, spatiotemporal modeling, autonomous systems.

1 Introduction

The MADTA problem has become a key research direction in autonomous unmanned system coordination, with important applications in UAV swarm reconnaissance [1], autonomous vehicle edge computing [2], intelligent warehouse logistics [3] and disaster response networks [4],

and large-scale emergency rescue missions in complex unstructured environments. Different from static allocation with fixed environments, MADTA faces highly uncertain, confrontational and stochastic conditions: tasks arrive randomly, have strict execution deadlines, and may change locations or priorities dynamically with external environment [5]. Its core goal is to assign homogeneous or heterogeneous agents to emerging tasks in real time and sequentially, so as to maximize global system utility while minimizing resource consumption, energy cost and task failure rate [6]. Due to the nonlinear coupling of agent motion constraints, dynamic environmental obstacles and combinatorial explosion of decision space, MADTA is an NP-hard problem, bringing severe theoretical and practical challenges to traditional algorithms [7].

Centralized optimization methods, including mixed-integer linear programming [8], branch-and-bound [9], and Hungarian algorithm [10], have long been used for task allocation due to their theoretical global optimality in static settings. Yet they demand full prior environmental information and exhaustive search, whose complexity grows factorially with agent and task scale [11]. In dynamic scenarios, such methods require frequent recomputation from scratch, resulting in excessive latency that fails real-time demands under rapid task arrivals [12]. To reduce computation overhead, decentralized heuristics such as auction mechanisms [13], consensus-based bundle algorithms [14], and swarm intelligence methods [15] have been adopted. While they improve efficiency via distributed decision-making, greedy strategies limit global vision and easily fall into local optima, leading to poor global coordination in dense multi-agent systems with frequent task overlaps [16].

In recent years, multiagent reinforcement learning (MARL) has excelled in complex sequential decision-making by directly mapping high-dimensional states to optimal actions via interactive trial-and-error, without explicit environmental modeling [17]. For MADTA, typical MARL algorithms including multiagent deep deterministic policy gradient (MADDPG) [18], multiagent proximal policy optimization (MAPPO) [19] have performed well in dealing with environmental uncertainty, stochastic dynamics and partial observability. Under the centralized training with decentralized execution (CTDE) paradigm [20], agents employ global state information in training to mitigate multi-agent non-stationarity, and use only local partial observations for decentralized real-time execution. However, mainstream MARL structures rely mainly on multi-layer perceptrons or convolutional neural networks to process inputs, which require fixed-dimensional state spaces and thus significantly restrict real-world adaptability [21].

The fixed-dimensional input requirement greatly restricts standard MARL in MADTA, where active task numbers vary dynamically with real-time arrivals and departures. Zero-padding or truncating variable observations causes severe information loss, scattered attention, and inability to capture key topological relations. To address this bottleneck, graph neural networks (GNNs) have been adopted to model inherent relational inductive biases in multiagent systems [22, 23, 24]. Treating agents and tasks as nodes and their interactions as edges, GNN-based MARL handles variable-scale graphs via permutation-invariant message passing [25, 26, 27]. Especially, graph attention networks are incorporated to assign learnable adaptive weights to neighbors, strengthening agents' ability to focus on high-priority tasks and suppress irrelevant spatial noise [28, 29].

Despite advances in GNN-based MARL for task allocation, existing methods still suffer from critical limitations in highly dynamic scenarios. First, most build static graph topologies at each step, neglecting the temporal evolution and historical dependencies of agent-task relations that are vital for forecasting task urgency and trends. Second, their attention mechanisms lack spatiotemporal differentiation, assigning equal temporal weights to neighbors and ignoring asymmetric task urgency decay, resulting in short-sighted decisions. Third, current value functions inadequately penalize spatial conflicts when multiple agents compete for the same task, causing resource redundancy and global allocation failures. Thus, designing a robust MARL framework that unifies dynamic temporal graphs [30], adaptive spatiotemporal attention, and conflict-aware global evaluation remains an urgent open challenge for MADTA.

To address these critical research gaps, this paper presents a novel spatiotemporal topology-aware graph reinforcement learning (STA-GRL) al-

gorithm for the complex MADTA problem. We model the dynamic environment as a dynamic bipartite graph with agents and tasks as two evolving node sets. A spatiotemporal gated graph attention (STGGA) module is proposed to capture instantaneous spatial distances and agent-task matching, while employing a temporal gating mechanism to record task state evolution for predictive decision-making.Embedded in the CTDE framework, a tailored topology-aware critic explicitly characterizes topological distances and task overlaps among agents to penalize spatial conflict clustering during centralized training.

The main contributions of this paper are threefold.

1. We model MADTA as a partially observable Markov decision process with a continuous dynamic bipartite graph, and propose the STGGA module. It resolves variable input dimensions by dynamically aggregating agent-task spatiotemporal features without padding or truncation, enabling lossless information propagation in stochastic environments.

2. We design a temporal gating mechanism fused with multi-head graph attention, enabling agents to adapt attention weights according to task urgency history, thus reducing redundant assignments and greedy competition in dense dynamic scenarios.

3. Under the CTDE framework, we construct a topology-aware global value function that penalizes spatial conflicts via a dynamic conflict tensor incorporated into the adjacency matrix. Theoretical and extensive experimental results verify its superior coordination performance and higher task completion rate over state-of-the-art MARL methods.

The rest of this paper is organized as follows. Section II rigorously formulates MADTA as a partially observable Markov decision process (POMDP) with detailed kinematic and stochastic specifications. Section III elaborates the proposed STA-GRL algorithm, covering bipartite graph construction, mathematical derivation of the STGGA module, and the topology-aware critic. Section IV introduces simulation setups, ablation studies, and comparative results. Section V concludes the work and discusses future works. The main notations of this paper is listed in Table 1.

Table 1. Notations in this paper

Notation	Definitions
M_t	the time-varying set of active tasks at time t
t	the discrete decision epoch
s_i^a	the state vector of agent i
p_i^a	the spatial coordinates of agent i
v_i^a	the velocity vector of agent i
ψ_i^a	the heading angle of agent i
e_i^a	the remaining energy capacity of agent i
s_j^m	the state vector of task j
p_j^m	the spatial coordinates of task j
τ_j^{req}	the required execution duration of task j
S_t	the global state at time t
o_i^t	local observation of agent i at time t
R_{obs}	the observable radius of each agent
a_i^t	the discrete action of agent i at time t
$\mathcal{A}_i^t$	the action set of agent i at time t
v_{max}	the maximum flight speed of UAV agents
u_i^t	the low-level control input of agent i
Δt	the control step size/simulation time interval
$\lambda(t)$	the time-varying average task arrival rate
r_i^{comp}	the task completion reward for agent i
r_i^{dist}	the distance penalty for agent i
r_i^{exp}	the task expiration penalty shared by agents
γ	the discount factor
π_θ	the parameterized stochastic policy
V_i^t	the node set of graph G_i^t
E_i^t	the edge set of graph G_i^t
$h_i^{a(0)}$	the initial feature vector of agent node i
$h_k^{m(0)}$	the initial feature vector of task node k
M_{head}	the number of multi-head attention heads
σ	the sigmoid activation function

2 Problem Formulation

The MADTA problem is rigorously modeled as a POMDP, formally defined by the high-dimensional tuple $\mathcal{M} = \langle \mathcal{N}, \mathcal{M}_t, \mathcal{S}, O, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma \rangle$. Let $\mathcal{N} = \{1, 2, \ldots, N\}$ denote the finite set of N homogeneous autonomous agents operating in a constrained, bounded Euclidean space $\mathcal{W} \subset \mathbb{R}^2$. Let $\mathcal{M}_t = \{1, 2, \ldots, M_t\}$ denote the time-varying set of active tasks at discrete decision epoch $t \in \mathbb{Z}^+$. The cardinality of the active task set M_t is a stochastic variable governed by a non-homogeneous Poisson arrival process and a deterministic expiration mechanism.

2.1 State Space and Observation Space Formulation

The global state space $\mathcal{S}$ encompasses the comprehensive kinematic, dynamic, and attribute information of all agents and active tasks at time t. The state vector of agent $i \in \mathcal{N}$ is denoted as $\mathbf{s}_i^a \in \mathbb{R}^{d_a}$, which intrinsically includes its spatial coordinates $\mathbf{p}_i^a = [x_i^a, y_i^a]^T \in \mathbb{R}^2$, velocity vector $\mathbf{v}_i^a = [vx_i^a, vy_i^a]^T \in \mathbb{R}^2$, heading angle $\psi_i^a \in [-\pi, \pi]$, and remaining energy capacity $e_i^a \in \mathbb{R}^+$. The state vector of task $j \in \mathcal{M}_t$ is denoted as $\mathbf{s}_j^m \in \mathbb{R}^{d_m}$, characterized by its spatial coordinates $\mathbf{p}_j^m = [x_j^m, y_j^m]^T \in \mathbb{R}^2$, required execution duration $\tau_j^{req} \in \mathbb{R}^+$, and remaining temporal deadline $\tau_j^{rem}(t) \in \mathbb{R}^+$. The global state at time t is formulated as the strict concatenation of all individual state vectors in the system:

$$\mathbf{S}_t = \left\{ \bigoplus_{i=1}^{N} \mathbf{s}_i^a, \bigoplus_{j=1}^{M_t} \mathbf{s}_j^m \right\}, \tag{1}$$

where $\bigoplus$ denotes the sequential concatenation operator. Due to physical sensor limitations, localized sensing ranges, and communication constraints in practical autonomous deployments, agents cannot access the global state $\mathbf{S}_t$ during decentralized execution. Each agent i maintains a local observation $\mathbf{o}_i^t \in O$, which consists of its own internal state and the states of tasks located strictly within its observable radius R_{obs}:

$$\mathbf{o}_i^t = \left\{ \mathbf{s}_i^a, \bigcup_{j \in \mathcal{M}_t : \|\mathbf{p}_i^a - \mathbf{p}_j^m\|_2 \le R_{obs}} \mathbf{s}_j^m \right\}. \tag{2}$$

Let K_i^t denote the number of tasks observed by agent i at time t. It is theoretically evident that $K_i^t \le M_t$ and K_i^t varies dynamically across different agents and time steps, leading to a structurally variable-dimensional observation space that violates the foundational assumptions of traditional deep neural networks.

2.2 Action Space Definition

The action space $\mathcal{A}$ is designed as a discrete combinatorial set representing the high-level task selection decisions. At each decision epoch, agent i selects a target from its locally observable task set or chooses to remain unassigned to execute a default patrol behavior. The discrete action of agent i at time t is defined as:

$$a_i^t \in \mathcal{A}_i^t = \{0\} \cup \{1, 2, \ldots, K_i^t\}, \tag{3}$$

where $a_i^t = 0$ indicates that agent i chooses to execute an idle maneuver, and $a_i^t = k$ (for $k \ge 1$) indicates that agent i commits to executing the k-th task in its local observation buffer. Once an agent commits to a task, it is mathematically bound to navigate toward the task location and remains allocated until the task execution is physically completed or the task deadline expires. Physical completion of task j is defined as the agent arriving at the task location $\mathbf{p}_j^m$ and remaining stationary at that location for the full required execution duration τ_j^{req} without interruption; only after this duration has elapsed is the task considered successfully executed and removed from the active task set.

2.3 State Transition Dynamics

The state transition probability $\mathcal{P}(\mathbf{S}_{t+1}|\mathbf{S}_t, \mathbf{A}_t)$ is determined by the underlying kinematic models of the agents and the stochastic task dynamics. The kinematic transition of agent i is governed by a discrete-time unicycle model with velocity constraints

$$\mathbf{p}_i^a(t+1) = \mathbf{p}_i^a(t) + \mathbf{v}_i^a(t)\Delta t, \tag{4}$$

$$\mathbf{v}_i^a(t+1) = \mathrm{Clip}\left(\mathbf{v}_i^a(t) + \mathbf{u}_i^t \Delta t, -v_{max}, v_{max}\right), \tag{5}$$

where $\mathbf{u}_i^t \in \mathbb{R}^2$ is the low-level control input derived from the high-level task allocation action a_i^t, and Δt is the control step size. Tasks arrive stochastically according to a non-homogeneous Poisson process with a time-varying average arrival rate $\lambda(t)$. For any active task j, the remaining deadline updates deterministically as

$$\tau_j^{rem}(t+1) = \tau_j^{rem}(t) - \Delta t. \tag{6}$$

If a task j is not allocated to any agent before $\tau_j^{rem}(t) \le 0$, a state transition occurs where the task is permanently removed from the active task set $\mathcal{M}_t$, representing a failed allocation with a negative impact on global utility.

2.4 Reward Function Formulation

The reward function is critically designed to balance individual execution efficiency with implicit global coordination to strictly prevent multi-agent conflicts. The local immediate reward r_i^t for

agent i at time t is formulated as a weighted combination of completion bonuses, continuous distance penalties, and discrete expiration penalties

$$r_i^t = \omega_{comp} r_i^{comp} + \omega_{dist} r_i^{dist} + \omega_{exp} r_i^{exp}. \quad (7)$$

The task completion reward r_i^{comp} is activated when agent i successfully navigates to and finishes task j:

$$r_i^{comp} = \begin{cases} \frac{\tau_j^{rem}(t_{complete})}{\tau_j^{req}} + 1, & \text{agent } i \text{ completes } j \\ 0, & \text{otherwise} \end{cases}, \quad (8)$$

where the fractional term explicitly incentivizes early completion to maximize the time margin. The distance penalty r_i^{dist} discourages aimless movement, energy waste, and zigzagging, defined continuously as

$$r_i^{dist} = -\|\mathbf{p}_i^a(t) - \mathbf{p}_i^a(t-1)\|_2. \quad (9)$$

The task expiration penalty r_i^{exp} is formulated as a global penalty shared among all agents within the observable radius of the expired task to enforce implicit coordination and shared responsibility

$$r_i^{exp} = \begin{cases} -1, & \text{if task } j \text{ expires and } j \in \mathcal{N}_i^{obs} \\ 0, & \text{otherwise} \end{cases}. \quad (10)$$

The objective of each agent is to maximize its expected cumulative discounted return $\mathbb{E}_{\pi_\theta}\left[\sum_{t=0}^{T} \gamma^t r_i^t\right]$, where $\gamma \in (0,1)$ is the discount factor and π_θ is the parameterized stochastic policy.

Remark 1. *We rigorously formulate the MADTA problem as a POMDP, and define a variable-dimensional local observation space restricted by a sensing radius and a discrete action space for task selection. State transitions are governed by a discrete-time unicycle model for agents and a deterministic deadline mechanism for tasks driven by a non-homogeneous Poisson arrival process. A carefully designed reward function is introduced to balance individual execution efficiency and implicit global coordination by combining task completion bonuses, continuous distance penalties, and shared expiration penalties.*

3 STA-GRL Algorithm

Although standard multiagent reinforcement learning handles environmental uncertainty, its reliance on fixed-dimensional inputs severely restricts adaptability in MADTA, where observable tasks fluctuate continuously. While naive zero-padding causes severe information loss, existing GNN approaches still suffer critical limitations in dynamic scenarios. They typically construct static topologies, neglecting the temporal evolution of task urgency. Furthermore, their attention mechanisms lack spatiotemporal differentiation, leading to short-sighted decisions, while their value functions fail to explicitly penalize spatial conflicts when multiple agents target the same task, causing severe resource redundancy.

To directly address these interconnected gaps and effectively capture the dynamic spatiotemporal relationships inherently present in MADTA, we propose the novel STA-GRL framework. The algorithm fundamentally adopts the CTDE paradigm to balance real-time execution with global coordination. For decentralized execution, it utilizes a shared Graph Attention Network-based actor network that gracefully processes variable-scale observations. Concurrently, during centralized training, a tailored Topology-Aware Critic network is deployed. By incorporating conflict-aware mechanisms, this specialized critic provides stable and accurate value estimation.

3.1 Dynamic Bipartite Graph Construction

At each decision time step t, the local observation $\mathbf{o}_i^t$ of agent i is mapped to a dynamic bipartite graph $\mathcal{G}_i^t = (\mathcal{V}_i^t, \mathcal{E}_i^t, \mathbf{X}_i^t)$. The node set $\mathcal{V}_i^t = \mathcal{V}_{agent} \cup \mathcal{V}_{task}$ consists of the single agent node i and its K_i^t observed task nodes. The edge set $\mathcal{E}_i^t \subseteq \mathcal{V}_{agent} \times \mathcal{V}_{task}$ connects the agent node to all observable task nodes, forming a star-shaped bipartite topology that is inherently invariant to the absolute number of tasks.

The initial feature vector for the agent node is constructed as $\mathbf{h}_i^{a(0)} = \mathbf{s}_i^a \in \mathbb{R}^{d_a}$. The initial feature vector for the k-th task node is constructed by concatenating its spatial state with its temporal attributes to provide immediate heuristic priors to the network:

$$\mathbf{h}_k^{m(0)} = \left[\mathbf{p}_k^m, \tau_k^{req}, \tau_k^{rem}(t), \frac{\tau_k^{rem}(t)}{\tau_k^{req}}\right]^T \in \mathbb{R}^{d_m}, \quad (11)$$

where the final term explicitly embeds the normalized task urgency into the node feature space. To provide rigorous spatial relational context to the attention mechanism, we define an edge feature matrix $\mathbf{E}_{ik} \in \mathbb{R}^{d_e}$ for the edge connecting agent i and task k:

$$\mathbf{E}_{ik} = \left[\frac{\mathbf{p}_i^a - \mathbf{p}_k^m}{\|\mathbf{p}_i^a - \mathbf{p}_k^m\|_2 + \varepsilon}, \|\mathbf{p}_i^a - \mathbf{p}_k^m\|_2 \right]^T, \quad (12)$$

where ε is a small constant for numerical stability. This edge feature mathematically encodes both the unit relative direction vector and the Euclidean distance, serving as a foundational spatial prior for the subsequent attention module.

3.2 Spatiotemporal Gated Graph Attention Module

Standard Graph Attention Networks compute attention coefficients solely based on instantaneous node features, fundamentally failing to capture the rapidly changing urgency decay of dynamic tasks. We propose a Spatiotemporal Gated Graph Attention (STGGA) mechanism.

For an agent node i and a neighboring task node k, the raw spatial attention coefficient is calculated by jointly processing node and edge features through a shared linear transformation followed by a LeakyReLU non-linearity:

$$e_{ik}^{spa} = \text{LeakyReLU}\left(\mathbf{a}_{spa}^T \left[\mathbf{W}_a \mathbf{h}_i^{a(l)} \| \mathbf{W}_m \mathbf{h}_k^{m(l)} \| \mathbf{W}_e \mathbf{E}_{ik}\right]\right) \quad (13)$$

where $\mathbf{W}_a \in \mathbb{R}^{d' \times d_a}$, $\mathbf{W}_m \in \mathbb{R}^{d' \times d_m}$, $\mathbf{W}_e \in \mathbb{R}^{d' \times d_e}$ are linear transformation matrices, $\|$ denotes vector concatenation, and $\mathbf{a}_{spa} \in \mathbb{R}^{3d'}$ is a learnable attention vector.

To incorporate the historical trajectory of task urgency and prevent the policy from being misled by transient environmental noise, we introduce a temporal gating vector $\mathbf{g}_k^t \in \mathbb{R}^{d'}$, updated via a recurrent gating structure that maintains memory across time steps: this gate dynamically amplifies attention for tasks with fast-decaying deadlines and suppresses less urgent ones.

$$\mathbf{g}_k^t = \sigma\left(\mathbf{W}_g \mathbf{h}_k^{m(l)} + \mathbf{U}_g \mathbf{g}_k^{t-1} + \mathbf{b}_g\right), \quad (14)$$

where σ is the sigmoid activation function, and $\mathbf{U}_g \in \mathbb{R}^{d' \times d'}$ is the recurrent weight matrix governing the temporal decay. The final spatiotemporal attention logit is modulated by this temporal gate via a Hadamard product:

$$\hat{e}_{ik} = e_{ik}^{spa} \odot \mathbf{g}_k^t. \quad (15)$$

This modulation mathematically ensures that tasks whose deadlines are rapidly decaying receive exponentially amplified attention, while tasks with ample remaining time maintain a baseline attention level. The normalized attention coefficient is obtained via the softmax function applied over all neighboring task nodes to ensure probabilistic interpretation:

$$\alpha_{ik} = \frac{\exp(\hat{e}_{ik})}{\sum_{j \in \mathcal{N}_i^{task}} \exp(\hat{e}_{ij})}. \quad (16)$$

The aggregated feature representation for agent i after L layers of STGGA is computed using multi-head attention to stabilize the learning process and capture diverse relational patterns:

$$\mathbf{z}_i^{agg} = \Big\|_{m=1}^{M_{head}} \sum_{k \in \mathcal{N}_i^{task}} \alpha_{ik}^m \mathbf{W}^m \mathbf{h}_k^{m(L)}, \quad (17)$$

where $\|$ denotes concatenation across M_{head} independent attention heads, and $\mathbf{W}^m \in \mathbb{R}^{d'' \times d'}$ is the projection matrix for the m-th head. This aggregated vector $\mathbf{z}_i^{agg} \in \mathbb{R}^{M_{head} \cdot d''}$ intrinsically solves the variable input size problem, as the summation operation is permutation-invariant and mathematically scalable to any K_i^t.

3.3 Sequential Execution and Recurrent Fusion

To model the sequential decision process of dynamic task allocation and preserve historical execution information, we feed the aggregated graph embedding $\mathbf{z}_i^{agg}$ into a Gated Recurrent Unit (GRU). Let $\mathbf{h}_{i,t-1}^{gru} \in \mathbb{R}^{d_h}$ denote the hidden state of the GRU at time $t-1$. The update rules are formulated as

$$\mathbf{r}_t = \sigma\left(\mathbf{W}_{ir} \mathbf{z}_i^{agg} + \mathbf{b}_{ir} + \mathbf{W}_{hr} \mathbf{h}_{i,t-1}^{gru} + \mathbf{b}_{hr}\right), \quad (18)$$

$$\mathbf{u}_t = \sigma\left(\mathbf{W}_{iu} \mathbf{z}_i^{agg} + \mathbf{b}_{iu} + \mathbf{W}_{hu} \mathbf{h}_{i,t-1}^{gru} + \mathbf{b}_{hu}\right), \quad (19)$$

$$\mathbf{n}_t = \tanh\left(\mathbf{W}_{in} \mathbf{z}_i^{agg} + \mathbf{b}_{in} + \mathbf{r}_t \odot (\mathbf{W}_{hn} \mathbf{h}_{i,t-1}^{gru} + \mathbf{b}_{hn})\right), \quad (20)$$

$$\mathbf{h}_{i,t}^{gru} = (1 - \mathbf{u}_t) \odot \mathbf{n}_t + \mathbf{u}_t \odot \mathbf{h}_{i,t-1}^{gru}, \quad (21)$$

where r_t and u_t represent the reset gate and update gate, respectively. This recurrent structure enables the actor network to fuse historical allocation decisions with current graph observations effectively.

3.4 Topology-aware Critic Network

During centralized training, the critic network $V_\phi(\mathbf{S}_t)$ is tasked with evaluating the global state-value function. Standard critics typically concatenate all agent states, fundamentally failing to capture the topological conflicts where multiple agents target the same task simultaneously. We design a topology-aware critic (TAC) utilizing a global bipartite graph $\mathcal{G}_t^{global} = (\mathcal{V}^{global}, \mathbf{A}^{global})$.

First, we define an agent-agent conflict tensor $\mathbf{C} \in \mathbb{R}^{N\times N}$ based on their current target task assignments and spatial distances. Let $\mathcal{T}(a_i^t)$ be a function that returns the spatial coordinate of the task selected by agent i (or a null vector if idle). The conflict tensor is formulated as

$$C_{pq} = \begin{cases} \exp\left(-\frac{\|\mathcal{T}(a_p^t)-\mathcal{T}(a_q^t)\|_2^2}{2\sigma_c^2}\right), & a_p^t = a_q^t \neq 0 \\ 0, & \text{otherwise} \end{cases}, \tag{22}$$

where σ_c is a scaling parameter controlling the penalty decay with distance. The conflict tensor $\mathbf{C}$ explicitly quantifies the redundancy of agent distributions. We then construct an enhanced global adjacency matrix $\hat{\mathbf{A}}^{global} \in \mathbb{R}^{(N+M_t)\times(N+M_t)}$ by fusing the spatial agent-task bipartite adjacency $\mathbf{A}^{global}$ with the conflict tensor mapped into the agent subspace:

$$\hat{\mathbf{A}}^{global} = \mathbf{A}^{global} + \beta \begin{bmatrix} \mathbf{C} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}, \tag{23}$$

where $\beta > 0$ is a hyperparameter controlling the penalty strength for conflicts. The global value is extracted by a Graph Convolutional Network (GCN) processing the enhanced global graph. The layer-wise propagation rule for the TAC is defined as

$$\mathbf{H}^{(l+1)} = \text{ReLU}\left(\hat{\mathbf{D}}^{-\frac{1}{2}}\hat{\mathbf{A}}^{global}\hat{\mathbf{D}}^{-\frac{1}{2}}\mathbf{H}^{(l)}\mathbf{W}_c^{(l)}\right), \tag{24}$$

where $\hat{\mathbf{D}}$ is the diagonal degree matrix of $\hat{\mathbf{A}}^{global}$ with added self-loops, $\mathbf{H}^{(l)}$ is the node feature matrix at layer l, and $\mathbf{W}_c^{(l)}$ are the GCN weight parameters. By injecting $\mathbf{C}$ into the adjacency matrix, the GCN message passing mechanism inherently penalizes topological cliques, forcing the critic to assign strictly lower state-values to configurations where agents are spatially clustered around a single task, thereby providing a strong implicit coordination gradient signal to the actors.

3.5 Policy Optimization Objective

The actor and critic networks are jointly optimized using Proximal Policy Optimization (PPO) to ensure stable policy updates and prevent destructive large gradients. Let $r_t(\theta) = \frac{\pi_\theta(a_i^t|\mathbf{o}_i^t)}{\pi_{\theta_{old}}(a_i^t|\mathbf{o}_i^t)}$ denote the probability ratio between the new and old policies.

The clipped surrogate objective function is defined as:

$$\mathcal{L}^{CLIP}(\theta) = \hat{\mathbb{E}}_t\left[\min\left(r_t(\theta)\hat{A}_t, \text{clip}(r_t(\theta), 1-\varepsilon, 1+\varepsilon)\hat{A}_t\right)\right], \tag{25}$$

where $\hat{A}_t$ is the estimated advantage function calculated using generalized advantage estimation (GAE) to balance bias and variance:

$$\hat{A}_t = \sum_{l=0}^{T-t-1}(\gamma\lambda)^l\delta_{t+l}, \tag{26}$$

$$\delta_t = r_t + \gamma V_\phi(\mathbf{S}_{t+1}) - V_\phi(\mathbf{S}_t), \tag{27}$$

where $\lambda \in [0,1)$ is the GAE parameter. In our experiments, we set $\lambda = 0.95$, which provides a favorable bias-variance trade-off in advantage estimation, effectively reducing the variance of policy gradients while maintaining a sufficiently long temporal credit assignment horizon for stable training.

To encourage sufficient exploration and prevent premature convergence to suboptimal deterministic allocations, an entropy regularization term is added:

$$\mathcal{L}^S(\theta) = \mathbb{E}_{\pi_\theta}\left[-\sum_a \pi_\theta(a|\mathbf{o}_i^t)\log\pi_\theta(a|\mathbf{o}_i^t)\right]. \tag{28}$$

The overall actor loss is formulated as:

$$\mathcal{L}^{actor}(\theta) = \hat{\mathbb{E}}_t\left[\mathcal{L}^{CLIP}(\theta) + c_1\mathcal{L}^S(\theta)\right], \tag{29}$$

The critic network is optimized independently by minimizing the mean squared error loss combined with a gradient penalty to prevent value function divergence:

$$\mathcal{L}^{critic}(\phi) = \hat{\mathbb{E}}_t\left[\left(V_\phi(\mathbf{S}_t) - \hat{R}_t\right)^2\right] + c_2\hat{\mathbb{E}}_t\left[\left\|\nabla_\phi V_\phi(\mathbf{S}_t)\right\|_2^2\right], \tag{30}$$

where $\hat{R}_t = \sum_{l=0}^{T-t}\gamma^l r_{t+l}$ is the discounted return. The parameters θ and ϕ are updated alternately using the Adam optimizer over multiple epochs for each batch of collected trajectories.

Remark 2. *This Section introduces the STA-GRL framework under the CTDE paradigm to handle variable-dimensional observations and dynamic spatiotemporal relationships. It first maps local observations into dynamic bipartite graphs that encode spatial distances and temporal attributes as edge and node features. The core STGGA module is then proposed, which utilizes a recurrent temporal gating mechanism to dynamically amplify attention for tasks with rapidly decaying deadlines, while its permutation-invariant aggregation inherently solves the variable input size problem. Furthermore, the aggregated graph embeddings are processed by a GRU to maintain sequential execution memory. To ensure implicit coordination, a centralized topology-aware critic is designed by injecting an agent-agent conflict tensor into the global graph's adjacency matrix, forcing the GCN to penalize spatial clustering around the same task. Finally, the actor and critic networks are jointly optimized using PPO with Generalized Advantage Estimation and entropy regularization.*

4 Simulations and Results

To comprehensively evaluate the performance, robustness, and theoretical validity of the proposed STA-GRL algorithm in solving the complex MADTA problem, we designed a series of extensive, large-scale simulation experiments characterized by high stochasticity, dense agent interactions, and stringent temporal constraints.

4.1 Simulation Environment and Parameterization

We constructed a high-fidelity continuous 2D dynamic task allocation simulator representing a large-scale UAV swarm search and response scenario in a contested environment. The operational airspace was mathematically defined as a bounded square region $\mathcal{W} = [0, 2000] \times [0, 2000]$ meters. A fleet of N homogeneous UAVs was initialized with uniformly distributed random spatial positions and zero initial velocities. Each UAV was strictly constrained by a maximum flight speed $v_{max} = 25$ m/s and a maximum acceleration magnitude, modeled within the discrete-time unicycle kinematic integrator. The sensory observation radius R_{obs} for each UAV was strictly limited to 400 meters, meaning UAVs could only detect and evaluate tasks within this localized spatial range, enforcing strict partial observability.

Tasks stochastically appeared in the airspace following a non-homogeneous Poisson process with an average arrival rate λ, which dynamically increased over the simulation horizon to simulate escalating battlefield or disaster conditions. Each task was assigned a random spatial coordinate and a strict maximum endurance time (deadline) sampled independently from a uniform distribution over the interval $[40, 120]$ seconds. If a UAV successfully navigated to the task location before the deadline, the task was marked as completed; otherwise, it immediately expired and was permanently removed from the active environment, simulating irreversible mission failure. The simulation ran for a maximum of 300 discrete time steps per episode, with a control interval $\Delta t = 1$ second.

4.2 Baseline Algorithms for Comparison

To rigorously demonstrate the superiority of STA-GRL, we compared it against four established baselines representing different theoretical paradigms:

Hungarian algorithm (HA): A centralized static optimization method with high latency in dynamic scenes recalculated globally every 10 steps based on the current global agent-task Euclidean distance matrices, serving as the optimal static benchmark.

Greedy distance (GD): A purely heuristic approach where each agent locally selects the unallocated task with the minimum Euclidean distance at each step without any coordination.

Multiagent deep deterministic policy gradient (MADDPG): A classical MARL algorithm utilizing continuous action spaces, strictly adapted by mapping continuous action outputs to discrete task indices.

Multiagent proximal policy optimization (MAPPO): The standard Multi-Agent PPO algorithm utilizing standard MLPs, where variable-sized observations are naively handled via zero-padding to the maximum possible task limit observed during training.

Table 2. Performance comparison under escalating task arrival rates ($N = 30$), averaged over 500 episodes

Algorithm	$\lambda = 0.4$			$\lambda = 0.6$			$\lambda = 0.8$		
	TCR(%)	ART(s)	ACR(%)	TCR(%)	ART(s)	ACR(%)	TCR(%)	ART(s)	ACR(%)
HA	89.5	18.2	0.0	78.4	22.1	0.0	65.2	26.5	0.0
GD	71.2	28.4	18.5	60.5	32.1	24.3	48.2	36.8	31.2
MADDPG	80.4	24.5	12.1	72.8	27.8	16.5	61.5	31.2	22.4
MAPPO	85.1	20.3	7.8	77.5	24.6	11.2	68.4	28.1	15.8
STA-GRL	**93.2**	**15.1**	**1.5**	**86.8**	**18.4**	**2.1**	**78.5**	**21.2**	**2.8**

4.3 Evaluation Metrics

The algorithms were rigorously evaluated using three primary metrics designed to capture different facets of system performance:

Task completion rate (TCR): The percentage of arrived tasks successfully executed by agents before their deadlines. This is the primary metric reflecting global coordination efficiency and temporal robustness.

Average response time (ART): The average time elapsed from a task's stochastic arrival to its physical completion by an assigned agent. Lower ART indicates higher operational efficiency and reduced latency.

Average conflict rate (ACR): The percentage of time steps where two or more agents targeted the exact same task simultaneously, indicating a severe lack of implicit coordination and resource waste. It is calculated as the ratio of conflict time steps to total effective decision steps in each episode.

4.4 Performance under Escalating Task Densities

In the first set of experiments, we fixed the number of agents to $N = 30$ and systematically varied the task arrival rate $\lambda \in \{0.4, 0.6, 0.8, 1.0\}$ to test the algorithms' robustness under escalating environmental pressure. Each configuration was tested over 500 independent evaluation episodes to ensure strict statistical significance. The quantitative results are comprehensively summarized in Table 2.

As quantitatively detailed in Table 2, the proposed STA-GRL algorithm consistently and significantly outperforms all baselines across different task densities. While the Hungarian Algorithm (HA) maintains a zero conflict rate due to its centralized omniscient allocation, its TCR drops precipitously to 65.2% at $\lambda = 0.8$. This severe degradation occurs because the static recalculation interval fails to account for tasks that arrive and expire between optimization windows, a fundamental theoretical flaw in dynamic environments. The Greedy algorithm exhibits the worst overall performance, suffering from an exceptionally high Average Conflict Rate (ACR) exceeding 31%, as multiple agents blindly converge on the nearest tasks without foresight. Standard MAPPO improves upon MADDPG but still suffers from the zero-padding issue, leading to attention dispersion when the actual number of tasks is much smaller than the padded dimension, resulting in an ACR of 15.8%. In contrast, STA-GRL maintains a robust TCR of 78.5% even under extreme pressure ($\lambda = 0.8$), and keeps the ACR strictly below 2.8%. The STGGA module enables agents to precisely evaluate task urgency via temporal gating, effectively avoiding redundant targeting and allocating agents to tasks that are on the verge of expiration.

Table 3 reveals a critical phenomenon regarding the scalability of MARL algorithms in dense multi-agent regimes. Traditional optimization and heuristics quickly saturate; adding more agents beyond $N = 30$ yields severely diminishing returns because the spatial service capacity of the environment is reached, and HA lacks the temporal granularity to exploit the extra agents dynamically. Interestingly, MADDPG and standard MAPPO exhibit a catastrophic degradation in performance when N increases beyond 30. This is a classic manifestation of the non-stationarity curse and the "lazy agent" problem in MARL: as the number of agents increases, the environmental dynamics become highly chaotic, and the MLP-based policies fail to generalize, leading to severe coordination breakdowns. Conversely, STA-GRL demonstrates continuous, strictly positive scalability. By utiliz-

Table 3. Task completion rate (%) scalability under different fleet sizes ($\lambda = 0.8$), averaged over 500 episodes

Algorithm	N=20	N=30	N=40	N=50	N=60
HA	55.4	65.2	66.8	67.1	67.5
GD	42.1	48.2	52.5	53.8	54.2
MADDPG	55.8	61.5	58.2	54.1	50.5
MAPPO	62.5	68.4	70.1	69.5	68.2
STA-GRL	**68.5**	**78.5**	**85.2**	**89.4**	**91.2**

ing the dynamic bipartite graph, the policy's computational graph remains mathematically localized regardless of total agent count. Furthermore, the Topology-Aware Critic provides a stable global signal that explicitly penalizes spatial redundancy, allowing STA-GRL to achieve a remarkable 91.2% TCR with 60 agents, effectively utilizing the expanded fleet.

4.5 Ablation Study on Theoretical Components

To rigorously verify the theoretical necessity of the proposed components within the STA-GRL framework, we conducted an extensive ablation study by systematically removing key modules under the $N = 30, \lambda = 0.6$ scenario. The results are detailed in Table 4.

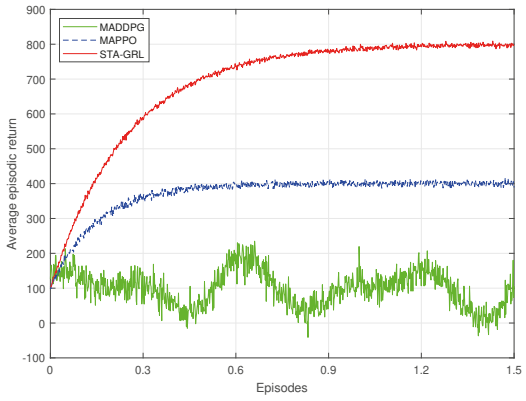

Figure 1. Training convergence curves of the MARL algorithms for the complex scenario with $N = 30$ and $\lambda = 0.6$

The ablation results rigorously validate our theoretical propositions. Removing the Temporal Gate (reverting to a static GAT) caused the TCR to drop by 7.3% and the ACR to rise to 6.8%, confirming that tracking the historical trajectory of task urgency is critical for preventing redundant allocations in time-constrained environments. Replacing the topology-aware critic with a standard GCN critic increased the ACR from 2.1% to 8.5% (a deterioration of 6.4%), demonstrating that without explicit conflict tensor injection into the adjacency matrix, the critic fails to provide sufficient gradient signals to discourage multi-agent clustering. Removing edge features and dynamic masking also resulted in significant performance penalties, proving that spatial priors and variable-dimensional architectures are mathematically indispensable for the MADTA problem.

4.6 Convergence and Training Efficiency Analysis

Figure 1 illustrates the training convergence curves of the MARL algorithms (MADDPG, MAPPO, and STA-GRL) for the complex scenario with $N = 30$ and $\lambda = 0.6$. The y-axis represents the average episodic return, smoothed over a moving window to reduce stochastic variance.

As conceptually depicted in Figure 1, the MADDPG exhibits extreme high variance and struggles to converge to a stable policy, frequently collapsing due to the unbounded nature of continuous-to-discrete action mapping in the presence of other learning agents. Standard MAPPO achieves stable convergence faster than MADDPG but plateaus at a highly suboptimal return value around 1.5 million episodes. The proposed STA-GRL demonstrates a significantly steeper initial gradient and converges to a global optimum asymptote. The incorporation of the temporal gate prevents the network from being misled by transient environmental noise, while the topology-aware critic ensures that the value baseline accurately reflects the true global utility, drastically reducing the variance of the policy gradient estima-

Table 4. Ablation study of the STA-GRL theoretical modules ($N = 30$, $\lambda = 0.6$), averaged over 500 episodes

Variant Description	TCR (%)	ART (s)	ACR (%)
Full STA-GRL	**86.8**	**18.4**	**2.1**
Temporal gate (static GAT)	79.5	22.1	6.8
Topology-aware critic (standard GCN)	82.1	20.5	8.5
Edge features (distance-blind attention)	81.2	21.8	5.2
Dynamic masking (zero-padding)	76.8	24.2	7.1

tions and ensuring monotonic improvement.

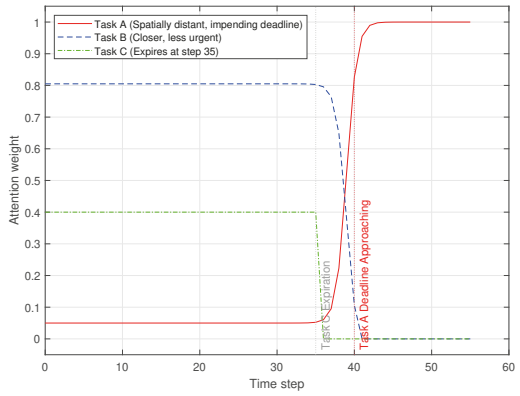

Figure 2. Temporal evolution of attention weights for three different tasks

4.7 Dynamic Spatiotemporal Feature Visualization

To explicitly verify the internal mathematical mechanism of the proposed STGGA module, we tracked the attention weights assigned by a specific agent to various surrounding tasks during a localized execution window. Figure 2 shows the temporal evolution of attention weights for three different tasks located concurrently within the agent's observation radius.

In Figure 2, Task A possesses an impending deadline but is spatially distant. Initially, the agent assigns low attention to Task A based on spatial edge features. However, as time progresses (approaching step 40), the temporal gate $\mathbf{g}_k^t$ in the STGGA module exponentially amplifies the urgency signal of Task A, causing a sharp spike in the attention weight. This mathematically prompts the agent to abandon a closer, less urgent task (task B) and redirect its trajectory towards Task A, successfully preventing its expiration. Task C expires at step 35, and the attention mechanism correctly collapses its weight to zero immediately thereafter, demonstrating that the algorithm dynamically filters out invalid nodes from the graph topology without requiring manual reset mechanisms. This visualization definitively confirms that the spatiotemporal attention mechanism successfully learns a sophisticated, non-greedy allocation logic that transcends simple distance-based heuristics.

Remark 3. *This section comprehensively evaluates the proposed STA-GRL algorithm through extensive simulations in a dynamic 2D UAV swarm scenario, comparing it against baselines like HA, GD, MADDPG, and MAPPO. The results demonstrate that STA-GRL significantly outperforms the other methods in task completion rate, response time, and conflict reduction, maintaining robust performance and positive scalability under escalating task densities and expanding fleet sizes where standard MARL methods catastrophically fail. Furthermore, ablation studies validate the necessity of each theoretical component, while convergence analysis and spatiotemporal attention visualizations confirm that the algorithm successfully learns a sophisticated, non-greedy logic to prioritize urgent tasks over simpler distance-based heuristics.*

5 Conclusion and Future Works

This paper systematically investigates the multiagent dynamic task allocation problem and presents a novel STA-GRL algorithm to tackle its inherent theoretical and practical challenges. By modeling the stochastic environment as a continuous dynamic bipartite graph, we effectively overcome the fixed-dimensional input constraints that limit conventional deep reinforcement learning architectures in variable-scale scenarios. The pro-

posed spatiotemporal gated graph attention module enables agents to dynamically assess the urgency of variable-scale task sets through temporal recurrence, while the topology-aware critic provides reliable global supervision by explicitly penalizing multi-agent spatial conflicts via refined adjacency matrix augmentation. Extensive experimental results clearly validate that STA-GRL substantially outperforms state-of-the-art optimization and MARL baselines. However, the current framework assumes homogeneous agents and ideal communication conditions, which are simplified compared to real-world scenarios.

Future works will focus on extending the STA-GRL algorithm to heterogeneous multi-agent systems with diverse capabilities and task types, as well as incorporating communication constraints and dynamic obstacles to enhance its practicality in real-world scenarios. Furthermore, the proposed framework can be generalized to 3D operational environments by extending the bipartite graph construction to 3D spatial coordinates and updating kinematic models to support 3D motion, which will broaden its applicability to complex 3D scenarios such as underwater robotics and 3D UAV swarm missions. Moreover, we plan to explore theoretical analysis on the convergence and generalization bounds of the spatiotemporal graph reinforcement learning framework. In addition, transfer learning and online adaptive learning strategies will be investigated to enable fast cross-scenario deployment and continuous improvement in highly stochastic and adversarial environments. Finally, real-world physical experiments will be conducted to verify the practical deployment performance.

References

[1] Correlation-driven task assignment for multi-uav networks via imitation learning, IEEE Transactions on Vehicular Technology, vol. 74, no. 11, pp. 17 353–17 365, 2025, date of Publication: 02 June 2025.

[2] R. Rauch, Z. Becvar, P. Mach, and J. Gazda, Cooperative multi-agent deep reinforcement learning for dynamic task execution and resource allocation in vehicular edge computing, IEEE Transactions on Vehicular Technology, vol. 74, no. 4, pp. 5741–5756, 2025.

[3] T. Wu, B. Lv, F. Fu, and Z. Tang, Research on dynamic task allocation model and method of multi-agent logistics agv based on improved contract net, International Journal of Wireless and Mobile Computing, vol. 28, no. 4, pp. 378–388, 2025.

[4] D. Liu, L. Dou, R. Zhang, X. Zhang, and Q. Zong, Multi-agent reinforcement learning-based coordinated dynamic task allocation for heterogenous uavs, IEEE Transactions on Vehicular Technology, vol. 72, no. 4, pp. 4372–4383, 2023.

[5] L. Xue, Y. Hu, H. Ye, X. Zhai, J. Yang, and S. Jin, Dynamic task allocation of edge–cloud system for low-altitude economy via multiagent actor–critic-queuing computational framework, IEEE Internet of Things Journal, vol. 13, no. 4, pp. 5655–5668, 2026.

[6] Y. Lv, J. Lei, and P. Yi, A local information aggregation-based multiagent reinforcement learning for robot swarm dynamic task allocation, IEEE Transactions on Neural Networks and Learning Systems, vol. 36, no. 6, pp. 10 437–10 449, 2025.

[7] J. Chen, Y. Guo, Z. Qiu, B. Xin, Q.-S. Jia, and W. Gui, Multiagent dynamic task assignment based on forest fire point model, IEEE Transactions on Automation Science and Engineering, vol. 19, no. 2, pp. 833–849, 2022.

[8] Z. Ren, A. G. Philip, S. Zhao, S. Rathinam, and H. Choset, Cp-milp: Mixed integer linear programming for multi-agent motion planning with linear dynamics, IEEE Robotics and Automation Letters, vol. 10, no. 12, pp. 12 573–12 579, 2025.

[9] M. Zajecka, M. Mastalerczyk, S. Y. Chong, X. Yao, J. Kwiecien, W. Chmiel, J. Dajda, M. Kisiel-Dorohinicki, and A. Byrski, Portfolio optimization with translation of representation for transport problems, Journal of Artificial Intelligence and Soft Computing Research, vol. 15, no. 1, pp. 57–75, 2024.

[10] A. Samiei and L. Sun, Distributed matching-by-clone hungarian-based algorithm for task allocation of multiagent systems, IEEE Transactions on Robotics, vol. 40, pp. 851–863, 2024.

[11] H. ElGibreen and K. Youcef-Toumi, Dynamic task allocation in an uncertain environment with heterogeneous multi-agents, Autonomous Robots, vol. 43, no. 7, pp. 1639–1664, 2019.

[12] J. C. Amorim, V. Alves, and E. P. de Freitas, Assessing a swarm-gap based solution for the task allocation problem in dynamic scenarios, Expert Systems with Applications, vol. 152, p. 113437, 2020.

[13] M. Braquet and E. Bakolas, Greedy decentralized auction-based task allocation for multi-agent systems, IFAC-PapersOnLine, vol. 54, no. 20, pp. 675–680, 2021.

[14] S. Wang, Y. Liu, Y. Qiu, and J. Zhou, Consensus-based decentralized task allocation for multi-agent systems and simultaneous multi-agent tasks, IEEE Robotics and Automation Letters, vol. 7, no. 4, pp. 12 593–12 600, 2022.

[15] T. Li, S. Leng, X. Liao, and Y. Zhang, Digital twin-based task-driven resource management in intelligent uav swarms, IEEE Transactions on Intelligent Transportation Systems, vol. 26, no. 4, pp. 5467–5480, 2025.

[16] H.-S. Shin, T. Li, H.-I. Lee, and A. Tsourdos, Sample greedy based task allocation for multiple robot systems, Swarm Intelligence, vol. 16, no. 3, pp. 233–260, 2022.

[17] H. Kong, Q. Xing, Q. Wang, H. Chen, R. Niu, Z. Duan, S. Wang, Y. Chang, and I. King, Learning optimal policies with local observations for cooperative multiagent reinforcement learning, IEEE Transactions on Neural Networks and Learning Systems, pp. 1–15, 2026.

[18] S. Liu, X. Zhu, J. Chang, T. Xu, H. Chen, and Z. Han, Distributed two-tier edge computing in integrated satellite-terrestrial networks: A multiagent deep deterministic policy gradient approach, IEEE Internet of Things Journal, vol. 13, no. 1, pp. 308–324, 2026.

[19] X. Li and X. Wang, Online solution for orbital pursuit-evasion game via heterogeneous proximal policy optimization, IEEE Transactions on Aerospace and Electronic Systems, vol. 61, no. 5, pp. 12 044–12 058, 2025.

[20] B. Dietrich, H. Khdr, and J. Henkel, Centralized training and decentralized control through the actor-critic paradigm for highly optimized multicores, in 2025 62nd ACM/IEEE Design Automation Conference (DAC). IEEE, 2025, pp. 1–7.

[21] Y. Zhang, Y. Xu, C. Zhang, C. Wang, Z. Yang, B. Tang, and E. Chung, Rethinking the utilization of individual rewards in multiagent reinforcement learning with sparse team rewards, IEEE Transactions on Neural Networks and Learning Systems, pp. 1–15, 2026.

[22] M. Besta, F. Scheidl, L. Gianinazzi, G. Kwasniewski, S. Klaiman, J. Müller, and T. Hoefler, Demystifying higher-order graph neural networks, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 48, no. 3, pp. 2544–2565, 2026.

[23] C. Zhaohuang, F. Zhongqi, T. Liang, H. Ma, Y. Sun, and S. Xiaohu, Phishing fraud identity inference based on graph gated recurrent neural network, Journal of Artificial Intelligence and Soft Computing Research, vol. 16, no. 3, pp. 257–274, 2026.

[24] W. Zhang, T. Liu, Z. Li, Y. Dai, and X. Zhou, Multimodal knowledge graph embedding with self-attention graph neural network, IEEE Transactions on Computational Social Systems, vol. 12, no. 6, pp. 5460–5473, 2025.

[25] X. Peng, S. Chen, H. Gao, H. Wang, and H. Michael Zhang, Combat urban congestion via collaboration: Heterogeneous gnn-based marl for coordinated platooning and traffic signal control, IEEE Transactions on Intelligent Transportation Systems, vol. 26, no. 6, pp. 7667–7677, 2025.

[26] Z. Liu, J. Zhang, E. Shi, Z. Liu, D. Niyato, B. Ai, and X. Shen, Graph neural network meets multi-agent reinforcement learning: Fundamentals, applications, and future directions, IEEE Wireless Communications, vol. 31, no. 6, pp. 39–47, 2024.

[27] S. S. Bhavanasi, L. Pappone, and F. Esposito, Dealing with changes: Resilient routing via graph neural networks and multi-agent deep reinforcement learning, IEEE Transactions on Network and Service Management, vol. 20, no. 3, pp. 2283–2294, 2023.

[28] J. Li, H. Ding, K. Zhang, and Y. Kong, Ego attention network: Local awareness boosts graph attention for graph representation learning, IEEE Transactions on Signal and Information Processing over Networks, vol. 12, pp. 299–310, 2026.

[29] J. Tang, Y. Miao, Y. Xia, Q. Zhou, and C. Yi, A multiscale pooling attention-based graph attention network for remaining useful life prediction, IEEE Transactions on Instrumentation and Measurement, vol. 74, pp. 1–14, 2025.

[30] K. Peng, S. Zhu, and T. Ma, Stpe-marl: Spatio-temporal multi-agent population evolution reinforcement learning, ACM Transactions on Intelligent Systems and Technology, vol. 16, no. 4, pp. 1–24, 2025.

Xuexiu Liang received the M.S. degree in Agricultural mechanization engineeringfrom Chinese Academy of Agricultural Mechanization Sciences, China, in 2013, and Ph.D. degree in Mechanical Design and Theory in 2017. From 2022, he was asenior engineer in China Software Testing Center (MIIT Software and IntegratedCircuit Promotion Center) in China, where his research focuses on robot andintelligent equipment evaluation technology.
https://orcid.org/0009-0006-2626-5455

Agnieszka Siwocha received the M.Sc. degree from the Faculty of Technical Physics, Computer Science and Applied Mathematics at Lodz University of Technology, Poland, and the Ph.D. degree in Computer Science, specializing in Computer Graphics, from SAN University, Łódź, Poland, in 2015. She is currently an Assistant Professor at SAN University in Łódź, Poland. She holds the title of Adobe Certified Expert and provides training in Adobe software and computer graphics. Her research and publication activities focus on computer science, computer graphics, and IT applications. Her current research interests include fractal coding, image compression and quality assessment, computer graphics, machine learning, and multifractal analysis.
https://orcid.org/0000-0003-2562-236X

Yu Xia received his Ph.D. degree in mechanical engineering from Chongqing University, Chongqing, China, in 2024. He is currently a Research Fellow in the State Key Laboratory of Mechanical System and Vibration, School of Mechanical Engineering, Shanghai Jiao Tong University, Shanghai, China. His research interests include prescribed performance control and electromechanical system control.
https://orcid.org/0009-0008-7271-5386