

Research Article

Open Access



Heterogeneous multi-agent task allocation based on graph neural network ant colony optimization algorithms

Ziyuan Ma, Huajun Gong

College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 210000, Jiangsu, China.

Correspondence to: Prof. Huajun Gong, College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 210000, Jiangsu, China. E-mail: ghj301@nuaa.edu.cn

How to cite this article: Ma Z, Gong H. Heterogeneous multi-agent task allocation based on graph neural network ant colony optimization algorithms. *Intell Robot* 2023;3(4):581-95. <http://dx.doi.org/10.20517/ir.2023.33>

Received: 1 Aug 2023 **First Decision:** **Revised:** 18 Oct 2023 **Accepted:** 24 Oct 2023 **Published:** 31 Oct 2023

Academic Editor: Simon X. Yang, Haibin Duan **Copy Editor:** Yanbin Bai **Production Editor:** Yanbin Bai

Abstract

Heterogeneous multi-agent task allocation is a key optimization problem widely used in fields such as drone swarms and multi-robot coordination. This paper proposes a new paradigm that innovatively combines graph neural networks and ant colony optimization algorithms to solve the assignment problem of heterogeneous multi-agents. The paper introduces an innovative Graph-based Heterogeneous Neural Network Ant Colony Optimization (GHNN-ACO) algorithm for heterogeneous multi-agent scenarios. The multi-agent system is composed of unmanned aerial vehicles, unmanned ships, and unmanned vehicles that work together to effectively respond to emergencies. This method uses graph neural networks to learn the relationship between tasks and agents, forming a graph representation, which is then integrated into ant colony optimization algorithms to guide the search process of ants. Firstly, the algorithm in this paper constructs heterogeneous graph data containing different types of agents and their relationships and uses the algorithm to classify and predict linkages for agent nodes. Secondly, the GHNN-ACO algorithm performs effectively in heterogeneous multi-agent scenarios, providing an effective solution for node classification and link prediction tasks in intelligent agent systems. Thirdly, the algorithm achieves an accuracy rate of 95.31% in assigning multiple tasks to multiple agents. It holds potential application prospects in emergency response and provides a new idea for multi-agent system cooperation.

Keywords: Graph isomerism, neural network, enhanced ant colony optimization algorithms, heterogeneous multi-agent, task allocation



© The Author(s) 2023. **Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License (<https://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, sharing, adaptation, distribution and reproduction in any medium or format, for any purpose, even commercially, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.



1. INTRODUCTION

Multi-agent systems have been widely used in various fields, such as robot cooperation, responding to intelligent city emergencies, and distributed sensor networks. The multi-agent system in this paper is composed of unmanned aerial vehicles (UAVs), unmanned ships, and unmanned vehicles, which work together efficiently to respond to emergencies^[1].

In these systems, multiple agents collaborate to complete tasks, thereby improving the efficiency and performance of the system. As a key component of the multi-agent system, task allocation directly affects the overall performance of the system^[2,3]. However, due to the heterogeneity among agents and the dynamic nature of tasks, the assignment problem becomes complex and challenging. Therefore, we need an efficient method to solve the assignment problem of heterogeneous multi-agents^[4,5].

Graph heterogeneous neural network ant colony optimization (GHNN-ACO) algorithms are a new method that combines graph heterogeneous neural networks (GHNN) with ant colony optimization (ACO) algorithms to solve the assignment problem of heterogeneous multi-agents^[6,7]. In this method, the heterogeneous multi-agent system is modeled as a graph structure, where the agents are represented as nodes within the graph. The communication and collaboration relationships between agents are denoted as edges in the graph^[8,9]. Through the joint optimization of ACO algorithms and the neural network, this method can achieve efficient and accurate task allocation, thus improving the performance of the entire multi-agent system^[10].

In a heterogeneous Multi-agent system, the heterogeneity of agents is reflected in their characteristics, capabilities, and task requirements^[11-14]. To account for this heterogeneity, we model the Multi-agent system as a graph, where each agent is represented as a node, and the connection edges between nodes represent the communication and collaboration relationships between agents. This graph structure accurately reflects the relationships and interactions between intelligent agents^[15-17]. ACO algorithms are optimization algorithms that simulate the behavior of an ant colony when searching for the optimal path. In the task allocation of graph heterogeneous multi-agent systems, we use ACO algorithms to simulate the process of information transmission and cooperation between agents. The ACO algorithms have the ability for global search, enabling agents to quickly find the preliminary solution for task assignment. Their basic idea is that ants release pheromones when searching for the path and choose the next moving direction based on the pheromone concentration^[8,18]. Agents simulate the cooperation process by releasing pheromones in the task allocation process and selecting task allocation strategies according to the pheromone concentration^[19-22]. Through the iterative process of the ACO algorithms, the agents gradually optimize the task allocation strategy to achieve collaborative cooperation in task allocation.

To further optimize the task allocation results, we introduced a GHNN. GHNNs are deep learning models capable of processing graph-structured data^[23]. They can encode the information of nodes and edges in the graph into vectors and learn the relationships and features between nodes through the neural network learning process. In heterogeneous multi-agent task allocation, the GHNN takes the state, task features, and graph structure information of the agent as inputs and outputs the optimal task allocation strategy after calculation by the neural network. Through repeated iterative optimization, GHNNs can learn more accurate task allocation strategies, thereby further improving system performance^[24].

The application of GHNN-ACO algorithms in heterogeneous multi-agent task allocation is a joint optimization method^[25]. We apply GHNNs and ACO algorithms to heterogeneous multi-agent task allocation, forming a collaborative optimization process^[25]. The ACO algorithms are responsible for global search and self-organization, aiding agents in quickly finding preliminary task allocation schemes. On the other hand, the GHNNs handle fine learning and optimization of task allocation to obtain more accurate optimal task allocation strategies^[26]. By combining the two approaches, their respective limitations can be overcome, resulting

in an efficient and accurate task allocation process for heterogeneous multi-agent systems^[27].

Heterogeneous multi-agent systems involve agents with different capabilities, resources, and expertise, necessitating effective task allocation to optimize system performance. While traditional ACO algorithms have been successfully applied to solve combinatorial optimization problems, they may face challenges in complex multi-agent environments^[28]. To address these issues, we propose a novel paradigm that combines GHNNs with ACO algorithms to enhance the task allocation process in heterogeneous multi-agent systems^[29,30]. The GHNN-ACO algorithms present an innovative approach to solving the assignment problem in heterogeneous multi-agent systems^[31]. Through the integration of ACO algorithms and GHNNs, this method effectively addresses heterogeneity and dynamics, achieving efficient and accurate task allocation. We expect this method to find widespread applications in various fields, such as intelligent transportation systems, robot cooperation, and distributed sensor networks in the future.

The main contributions are summarized as follows:

1. Use graph neural networks to learn the relationships among heterogeneous multi-agent systems. By constructing a graph neural network, tasks and agents are treated as nodes, and edges represent the strength of the relationship between tasks and agents. The graph neural network can learn the similarity and matching degree between tasks and agents, providing more meaningful heuristic information for ACO algorithms. The GHNN-ACO structure can effectively handle heterogeneous graph data, fully utilizing the heterogeneity of nodes and edges and modeling different types of nodes and edges separately to improve the adaptability of the algorithm to heterogeneous graph data.
2. Graph neural-enhanced ACO algorithms are used to optimize the task allocation process. The graph neural network enhances the optimal matching between ACO algorithms and search tasks, as ants select the next task or agent for matching based on the heuristic information learned by the graph neural network. The pheromone in ACO algorithms represents the degree of execution, adaptability, or collaboration between tasks and agents. Ants perform task allocation and agent selection according to the pheromone concentration and the prediction results of the graph neural network to achieve a better task allocation effect.
3. Integrate global and local information and adaptability. The GHNN-ACO algorithm converts graph data into a low-dimensional vector representation through graph feature learning and fuses global and local information. Graph feature learning considers the relationship and topology between nodes, enhancing the representation vector of nodes to contain more semantic information and improving the performance of classification and prediction tasks. ACO algorithms are adaptive, and path search adjusts path selection according to pheromone concentration and node characterization information, which helps to find better paths. The GHNN-ACO algorithm finds better node classification and link prediction solutions in graph data.
4. Neural networks accelerate the task allocation process of heterogeneous multi-agent systems. The graph-enhanced neural network is used to accelerate the calculation and decision-making, such as accelerating the pheromone update in ACO algorithms or the calculation of the graph neural network to improve the efficiency of task allocation. The GHNN and ACO algorithms of the GHNN-ACO algorithm have a certain degree of parallelism. The sub-network processes different types of nodes and edges in parallel to speed up graph feature learning. In ACO algorithms, the ant search paths are independent of each other, and parallel computing can be used to accelerate the search process.

Through the above innovative combination method, the advantages of graph neural networks and ACO algorithms can be combined to improve the efficiency and accuracy of solving heterogeneous multi-agent assignment problems. This method can be applied to complex task allocation scenarios, such as drone group task allocation, logistics systems task allocation, etc., providing intelligent optimization solutions for practical application scenarios. In the experiment, it is necessary to fully verify the performance of this method in the heterogeneous multi-agent assignment problem to ensure that it can effectively solve practical problems.

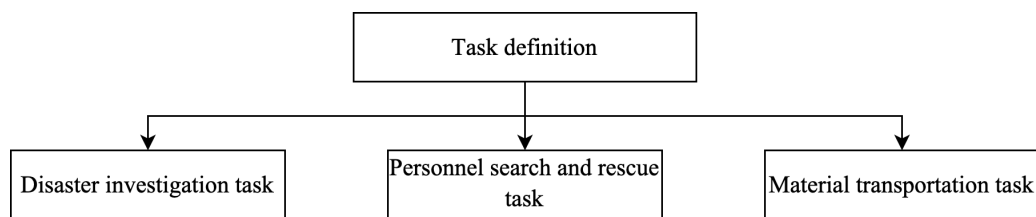


Figure 1. Structure of task definition.

2. PROBLEM FORMULATION

In modern intelligent systems, the heterogeneous multi-agent assignment problem is a research field of great concern. This problem involves various types of intelligent agents, such as drones, unmanned ships, and unmanned vehicles, deployed in unknown environments to perform various tasks. These intelligent agents have different sensors, actuators, motion capabilities, and various resource constraints. In complex, dynamic, and unfamiliar environments, such as exploring unknown areas or seas, these agents need to collaborate effectively to accomplish tasks.

2.1. Description of heterogeneous multi-agent models

The heterogeneous multi-agent model is a research model that involves task allocation and collaboration issues for different types of agents, such as drones, unmanned ships, and unmanned vehicles. In this model, different types of agents have varying task execution capabilities, resource allocations, and perception abilities, and they need to cooperate to efficiently complete a complex set of emergency rescue tasks.

Definition of intelligent agent types:

- Drone: Equipped with high-resolution cameras and sensors, capable of swift flying to obtain images and data of disaster areas. It can also search for personnel and survey dangerous areas.
- Unmanned vessel: Equipped with diving sensors and life-saving equipment, it can carry out search and rescue missions in the water, rescuing trapped personnel and providing emergency rescue supplies.
- Unmanned vehicle: With strong off-road capability and the ability to carry materials, it can enter affected areas for personnel evacuation, material transportation, and medical support.

Task Definitions:

- Disaster investigation task: After a disaster occurs, drones are dispatched to the affected area for rapid investigation. They can capture high-resolution images, monitor disaster severity, and provide real-time data to the command center to help formulate rescue plans.
- Personnel search and rescue task: Unmanned ships and vehicles are responsible for searching for trapped personnel in the affected waters and on the ground. Unmanned ships can detect underwater conditions through diving sensors, while unmanned vehicles can search for buried personnel using sensors.
- Material transportation task: Unmanned vehicles are responsible for transporting emergency rescue materials from the command center to the disaster area, providing disaster relief materials, and offering medical support.

Communication and collaboration mechanism: Establish a high-speed wireless communication network between UAVs, unmanned ships, and unmanned vehicles, using satellite communication or dedicated frequency bands to enable real-time data and status information sharing. By employing distributed decision algorithms, intelligent agents can exchange task information and sensor data in real time, enabling them to determine task allocation and execution strategies through consensus algorithms.

The disaster investigation mission is carried out by drones equipped with fast flight and high-resolution sensors

Table 1. A binary variable of task allocation

A binary variable $X(i, j)$	Definition
$X(i, j) = 1$	Agent i is assigned to execute task j
$X(i, j) = 0$	Agent i does not execute task j

to swiftly obtain images and information of the affected area. Personnel search and rescue tasks are assigned to unmanned ships and unmanned vehicles, which use underwater sensors for search and rescue missions in water, while unmanned vehicles conduct personnel search and rescue on the ground. Unmanned vehicles are responsible for handling the material transportation task, efficiently transporting rescue materials from the command center to the disaster area using their off-road capabilities and load-carrying ability.

Path planning and conflict avoidance: UAVs, unmanned ships, and unmanned vehicles utilize Fast path planning algorithms, such as A* algorithm or RRT algorithm, to plan the optimal path based on task urgency and agent speed. During task execution, sensors are used to perceive the surrounding environment in real time, and obstacle avoidance algorithms are employed to prevent collisions with other intelligent agents or obstacles.

Task execution monitoring and adjustment: The intelligent agents periodically report the task execution status and progress to the emergency rescue command center, including the percentage of completed tasks and encountered problems. The emergency rescue command center adjusts task allocation and collaboration strategies in real time based on the reports from the intelligent agents and the arrival of new tasks. The most urgent tasks are given priority, and the most suitable intelligent agents are scheduled to maximize rescue efficiency.

2.2. Heterogeneous multi-agent task scenarios

In the heterogeneous multi-agent task allocation scenario of this article, we have three types of agents: UAVs, unmanned ships (USVs), and unmanned vehicles (AVs). They are deployed in an unknown environment, such as a remote exploration area or an unknown sea area. The task objective is to perform a series of collaborative tasks in an unknown environment, such as exploration, target search, data collection, etc., to achieve specific task objectives. Each intelligent agent has its own resource limitations, such as battery energy (for drones and unmanned vehicles), fuel (for unmanned ships), and computing power. Tasks can be divided into investigating targets, collecting data, transporting items, and so on.

We model the assignment problem in the scenario as a multi-agent optimization problem. Assuming we have n intelligent agents, n represents the number of drones, unmanned ships, and unmanned vehicles, respectively. We use i to represent the i -th agent and j to represent the j -th task. Each agent i can execute task j , and each task has a specific reward or value $R(i, j)$.

To represent task allocation, we introduce a binary variable $X(i, j)$, which is defined as follows:

$X(i, j) = 1$, if agent i is assigned to execute task j ;

$X(i, j) = 0$, if agent i does not execute task j .

Our goal is to maximize the effectiveness of the entire team, which is the sum of the total rewards for tasks performed by all intelligent agents. It can be expressed as:

$$\text{Maximize } \sum_{i=1}^n \sum_{j=1}^m R(i, j) \times X(i, j) \tag{1}$$

Among them, $X(i, j)$ is a binary variable that represents whether agent i is assigned to execute task j . When

$X(i, j)=1$, it indicates that agent i is assigned to execute task j , otherwise $X(i, j)=0$.

The constraints of the Assignment problem include the following aspects:

Task constraint: Each task can only be executed by one agent.

$$\sum_{i=1}^n X(i, j) = 1, \text{ for all tasks } j = 1, 2, m \quad (2)$$

Agent constraint: Each agent can execute at most one task.

$$\sum_{j=1}^m X(i, j) \leq 1, \text{ for all agents } i = 1, 2, n \quad (3)$$

Resource constraints: Each intelligent agent needs to comply with its own resource constraints when performing tasks, such as considering battery energy for drones and unmanned vehicles and fuel constraints for unmanned ships.

Task type constraint: Some tasks may require specific types of intelligent agents to execute; for example, only drones can perform aerial reconnaissance tasks, and only unmanned vehicles can perform tasks on land.

Avoiding collision constraints: In order to ensure that there is no collision between intelligent agents, it is necessary to plan and coordinate the paths of intelligent agents to avoid their intersection.

In practical applications, this assignment problem may require real-time and adaptive performance. Because the environment is unknown and tasks may change at any time, intelligent agents need to be able to make decisions and adjustments based on real-time information.

3. HETEROGENEOUS AGENT TASK ALLOCATION ALGORITHM BASED ON GRAPH HETEROGENEOUS NEURAL NETWORK ANT COLONY OPTIMIZATION ALGORITHMS

The GHNN-ACO algorithms proposed in this paper are a hybrid algorithm integrating GHNN and ACO algorithms. These algorithms are mainly used to address the problems of node classification, link prediction, and graph feature learning in the heterogeneous data of a multi-agent graph.

3.1. Improvement and application of GHNN-ACO in heterogeneous multi-agent emergency rescue mission scenarios

In the context of a heterogeneous multi-agent economic rescue mission, we will refine the application of ACO algorithms to a heterogeneous neural network. Let us assume that a natural disaster requires emergency economic rescue. We have multiple intelligent agents, including drones, unmanned ships, and unmanned vehicles, with various tasks such as searching for and rescuing disaster victims, transferring affected residents, and distributing rescue supplies. Our objective is to maximize overall efficiency by intelligently allocating tasks and facilitating collaboration within limited resources.

In a specific multi-agent task scenario, the first step is initialization: setting the number of agents, the number of tasks, the structure of the graph for the heterogeneous neural network, the parameters of ACO algorithms,

and initializing the weights of ant colony pheromones and the heterogeneous neural network of graphs. Next, we proceed with GHNN learning: utilizing training datasets to train GHNNs. Tasks and agents serve as nodes in the graph, with edges representing the strength of the relationship between tasks and agents.

During the training process, the GHNN will learn the similarity and matching degree between tasks and agents. We then proceed with the initialization of the improved ACO algorithm: for each agent, we randomly initialize an ant and initialize the pheromone concentration matrix.

Algorithm 1 Improvement of ant colony optimization algorithms in graph-based heterogeneous neural network

```

1: Initialize the pheromone concentration matrix
2: Initialize the status of tasks and proxy nodes
3: for each iteration, do
4:   for each ant, do
5:     Select the next task or proxy node based on the current state and pheromone concentration
6:     Update the path and task allocation of ants
7:     Calculate the economic benefits or rescue efficiency of the current path as the fitness value of
      ants
8:   end for
9:   Update pheromone concentration matrix
10:  for each path  $j$ , do
11:    for each ant  $i$ , do
12:      Calculate  $\Delta_{\text{Pheromone}}(i, j) = Q/\text{fitness}(i, j)$ 
13:      Update  $\text{Pheromone}(i, j) = (1 - \text{evaporation\_rate}) * \text{Pheromone}(i, j) + \sum(\Delta_{\text{Pheromone}}(i, j))$ 
14:    end for
15:  end for
16: end for

```

3.2. Improvement of ant colony optimization algorithms based on graph heterogeneous neural network

The method proposed in this paper includes three main steps: GHNN learning, task allocation using ACO algorithms, and collaboration enhancement. Firstly, we use GHNNs to learn meaningful representations of the relationships between tasks and agents in the system topology. Neural networks capture the similarity, correlation, and collaboration potential between tasks and agents, providing valuable heuristic information for ACO algorithms. Secondly, we integrate the learned graph representation into ACO algorithms as pheromone information. This rich pheromone guides the decision-making process of ants and improves the efficiency and effectiveness of task allocation. Finally, we utilize graph neural networks to promote collaborative decision-making among intelligent agents, learning their communication and cooperation strategies.

GHNNs are a type of neural network used to process heterogeneous graph data. In heterogeneous graphs, nodes can have different attributes and types, and edges can also have different relationships. GHNNs transform the information in a graph into low-dimensional vectors by learning the representations of nodes and edges, thereby better dealing with complex graph structures and heterogeneous information. They typically consist of multiple subnetworks, each responsible for handling a type of node or edge. Each subnetwork can use different neural network structures, such as graph convolutional neural network (GCN), graph attention network (GAT), etc., to adapt to different types of nodes and edges. The subnetwork learns the representation

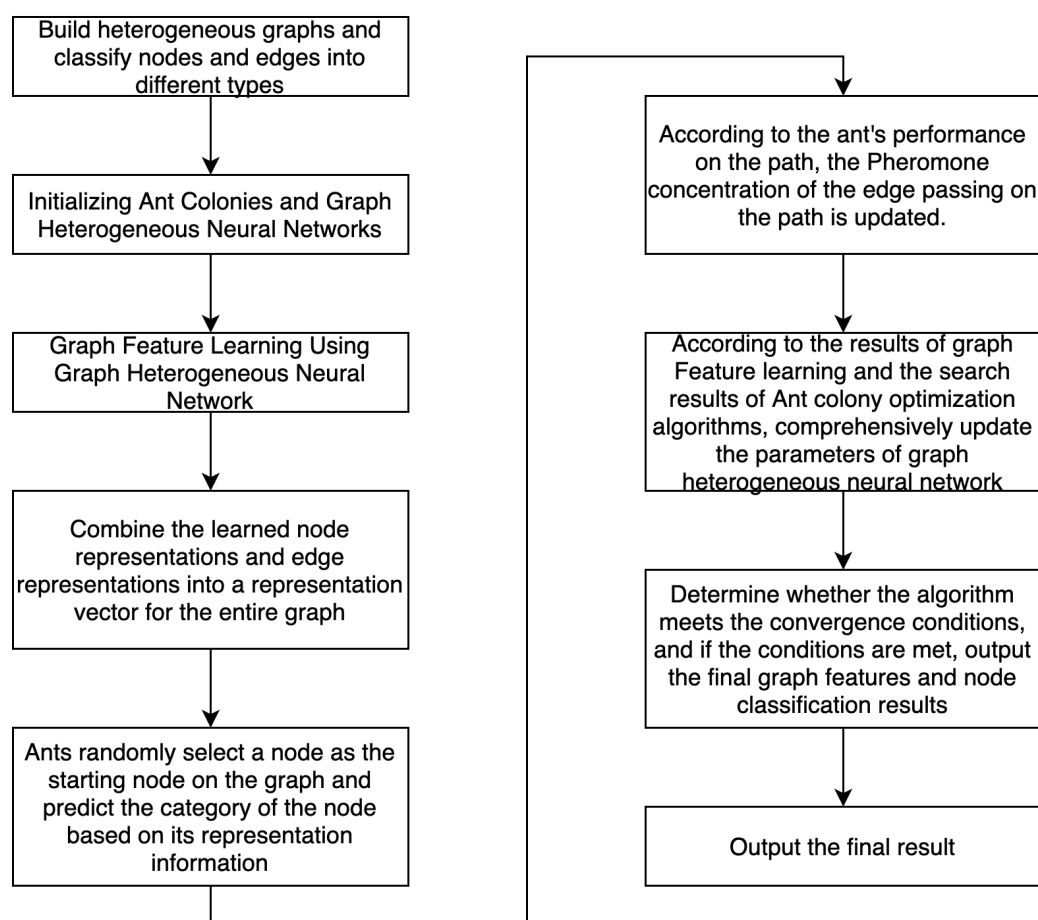


Figure 2. Structure diagram of GHNN-ACO algorithm model. GHNN-ACO: Graph-based Heterogeneous Neural Network Ant Colony Optimization.

of the entire graph through information transmission and feature aggregation.

ACO algorithms are heuristic optimization algorithms that simulate the behavior strategy of ants in the process of searching for food. In these algorithms, the ants guide other ants to choose the path by releasing pheromones, thus achieving the collaborative search of all ants. The pheromone concentration on the path will increase or decrease according to the quality of the path, and ants tend to choose the path with a higher pheromone concentration.

GHNN-ACO combines GHNNs and ACO algorithms to improve the classification, prediction, and feature learning of graph data.

When it comes to solving complex problems in graph data, traditional machine learning algorithms and neural networks often face challenges. The complexity of graph data lies in the fact that the relationships and features between nodes may be heterogeneous, meaning that different types of nodes and edges have different attributes. In order to effectively handle this heterogeneity, a GHNN has been introduced, which is a neural network structure suitable for heterogeneous graph data, as shown in Figure 2.

A GHNN typically consists of multiple subnetworks, each responsible for handling a type of node or edge. This structure allows subnetworks to focus on different types of information without confusing or ignoring

critical heterogeneous information. For example, in social networks, nodes can be users, pages, or points of interest, and edges can represent different types of interactions such as following and liking. The subnetwork of a GHNN can handle user nodes, page nodes, and interest point nodes separately while considering different types of edge information in order to better capture the characteristics of each node type and edge type.

In the graph learning phase, the GHNN subnetwork uses the common graph neural network structure (such as GCN or GAT) for forward propagation and back propagation to learn the representation of nodes and edges. These representations are low-dimensional vector representations of nodes and edges, capturing the semantic information of nodes and edges throughout the entire graph. Through collaborative learning of multiple subnetworks, GHNNs can effectively fuse information of different types of nodes and edges in heterogeneous graphs.

After the graph feature learning is completed, the GHNN-ACO algorithm introduces ACO algorithms to further optimize the classification and prediction of graph data. ACO algorithms are Swarm intelligence algorithms inspired by the behavior of ants in the process of searching for food. Ants guide other ants to choose a path by releasing pheromones so as to realize the collaborative search of all ants. In GHNN-ACO, ants represent a path search process. The pheromone concentration on the path will increase or decrease according to the quality of the path. Ants tend to choose the path with higher pheromone concentration.

The GHNN-ACO algorithm combines the node representation information obtained from graph feature learning with path search in ACO algorithms. Each ant starts from a randomly chosen starting node and uses the learned node representation information to predict the category of that node. Then, the ant selects the next node based on the pheromone concentration and node characterization information, where the pheromone concentration is determined based on the classification probability of the node. Ants perform path selection based on a specific strategy and update the pheromone concentration of the edges on the chosen path. Throughout the search process of ACO algorithms, GHNN-ACO effectively utilizes the combination of node representations and path search to optimize tasks, such as node classification and link prediction.

The structure of GHNN-ACO algorithms has many advantages but also some disadvantages. Let us analyze one by one:

1. Processing heterogeneous graph data: The GHNN-ACO algorithm can effectively process heterogeneous graph data, where nodes and edges can have different types and attributes. Through the structure of GHNNs, each subnetwork specifically processes a type of node or edge, which can better capture the complex information of heterogeneous graphs.
2. Fusion of local and global information: GHNNs learn the representation of graphs through local subnetworks and global information aggregation. ACO algorithms can be regarded as a global search method through path search, which combines global and local information. This fusion can help the algorithm better handle the relationships between structures and nodes in the graph.
3. Adaptive search strategy: path search in ACO algorithms is an adaptive strategy, and ants will select the next node according to pheromone concentration and node characterization information. This strategy enables the algorithm to concentrate on searching high-quality paths during the search process, which is beneficial for improving classification accuracy and link prediction accuracy.
4. Collaborative optimization: two parts of GHNN-ACO algorithms, namely GHNNs and ACO algorithms, are mutually collaborative optimization. GHNNs provide node representation information, and ACO algorithms use this information by searching paths. Ant path selection will affect the update of pheromone concentration. This collaborative optimization can enable the algorithm to converge faster and find more optimal solutions.

In the iterative update process of the GHNN-ACO algorithm, the parameters of the GHNN and the pheromone

in the ACO algorithms are comprehensively updated. This dual optimization process enables graph feature learning and path search to promote each other and gradually improve the effectiveness of graph data processing. After multiple iterations, when the algorithm meets the convergence conditions, GHNN-ACO outputs the final graph features and node classification results.

4. RESULTS

In this study, we explore the application of GHNN-ACO algorithms in real-world scenarios involving monitoring and responding to emergencies. By testing the algorithm in these scenarios, we have conducted an in-depth analysis of its performance in node classification and link prediction tasks within a multi-agent system.

4.1. Parameters setting and experimental subjects

In this design, we set up an emergency rescue task environment to simulate the emergency economic rescue tasks after a natural disaster occurs. Assuming that the affected area is represented by a 10x10 grid map, with each grid representing a task location. There are three different types of tasks to be carried out: searching and rescuing disaster victims, transferring affected residents, and distributing rescue supplies. The priority of tasks is divided into three levels: high, medium, and low.

In the task environment, we define the following parameters:

- (1) Map size: 10 × 10 grid map with a total of 100 task locations.
- (2) Task quantity: There are a total of ten tasks to be executed, including three tasks for searching and rescuing disaster victims, three tasks for transferring disaster-affected residents, and four tasks for distributing rescue supplies.
- (3) Task type: Search and rescue disaster victims, transfer disaster-affected residents, and distribute rescue supplies.
- (4) Task Priority: Each task is assigned one of three priority levels: high, medium, or low.
- (5) The experimental settings and initial conditions are as follows: Drone count: 2; Unmanned ship count: 1; Unmanned vehicle count: 2. In the initial state, all agents and tasks are randomly positioned and assigned states. The intelligent agents can move freely on the map but cannot occupy the same task location simultaneously. The priorities and types of tasks are also randomly assigned.

4.2. Experimental test results

Our research results indicate that the GHNN-ACO algorithm performs well in processing heterogeneous multi-agent data. By utilizing GHNNs, this algorithm efficiently handles different types of agents and relationships in intelligent agent systems. The ability to process heterogeneous graph data enables the GHNN-ACO algorithm to better capture complex associations and features between intelligent agents, thereby effectively improving the accuracy of node classification and link prediction tasks.

The algorithm used in this paper can quickly allocate different tasks to the required executing agents in five task scenarios. Figure 3(A) shows the task allocation between intelligent vehicles and UAVs. It can be seen that the algorithm in this article can autonomously allocate tasks to the first two scenarios, while the allocation efficiency for the third scenario involving unmanned ships is not as effective. Figure 3(B) and 3(C) represents tasks that focus on unmanned vehicles and unmanned ships and emergency rescue tasks that involve UAVs and unmanned ships, respectively.

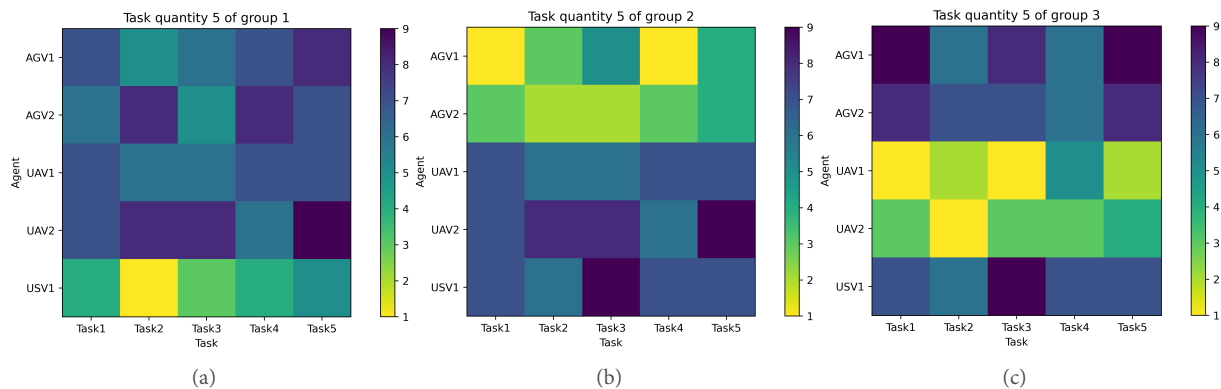


Figure 3. Algorithm test results in multi-task scenarios.

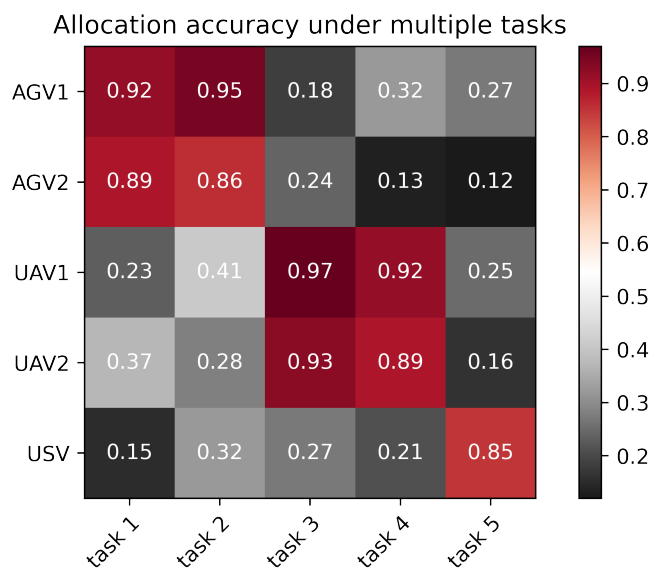


Figure 4. Accuracy of task allocation algorithm under multi-agent strategy.

We conducted tests to evaluate the accuracy of multi-agent deployment strategies under different task assignments using the proposed algorithms, as illustrated in Figure 4. Tasks 1 and 2 were assigned to unmanned vehicles for execution, tasks 3 and 4 were assigned to UAVs for execution, and task 5 was assigned to unmanned ships for execution. The innovative algorithm demonstrated in this article showcases improved accuracy in allocating different tasks to different multi-agent executions.

In the node classification task, we successfully classified the agents in the multi-agent system. For example, we categorized the UAV, unmanned ship, and unmanned vehicle into different types, such as reconnaissance type and rescue type. This accurate classification enables intelligent agent systems to better understand and respond to emergency events, providing robust support for decision-making.

The algorithm structure proposed in this article has an excellent mechanism for fusion recognition of global and local features. By learning global and local features, it can focus on some main intelligent agents that are executing tasks. Figure 5 shows the intelligent recognition of tasks of the algorithm focused on drones and

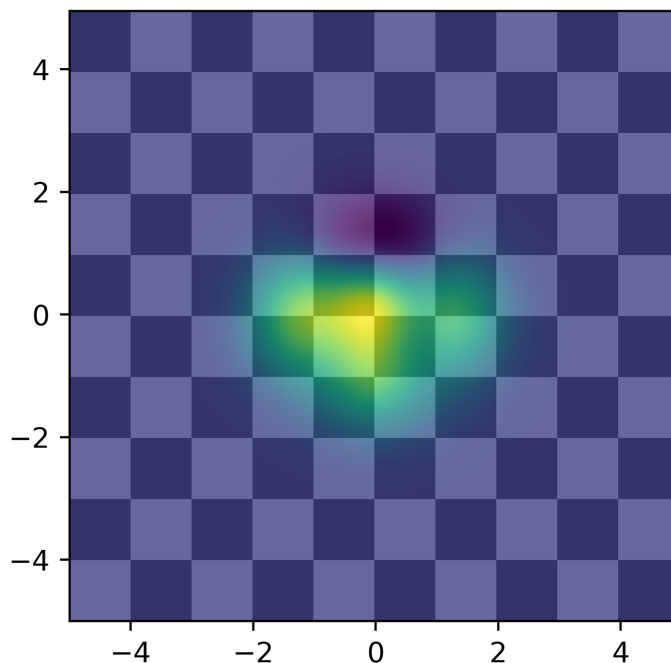


Figure 5. Algorithm recognition diagram for local and global feature fusion.

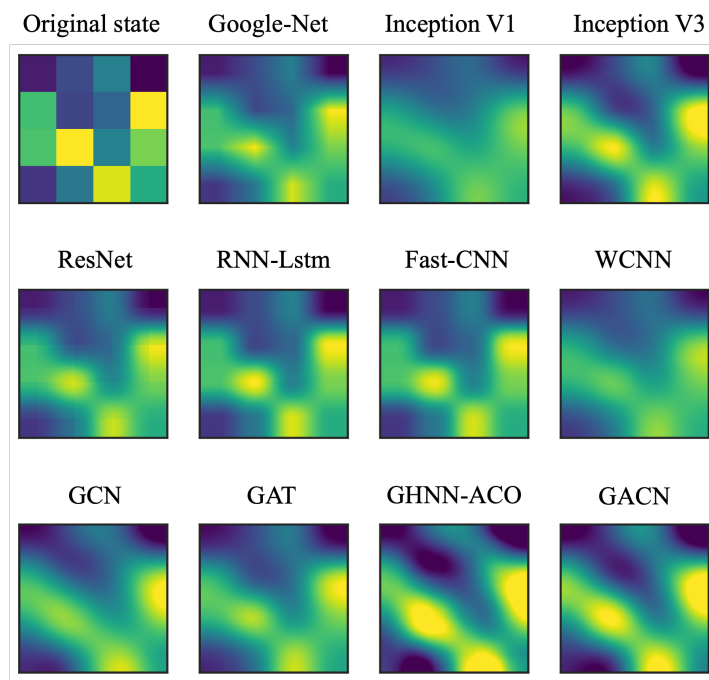


Figure 6. Algorithm recognition diagram for local and global feature fusion.

unmanned vehicles. The bright color represents the executing task intelligent agent, and the dark black color represents the intelligent agent that is not executing tasks.

This paper tests many different algorithms in task allocation, and the results show that the algorithm struc-

Table 2. Comparison of various indicators of different algorithms

Algorithms	AUC	ACC	F1	Recall
Google-Net	0.8974	0.9029	0.8812	0.8765
Fast-CNN	0.9018	0.8917	0.8995	0.9121
GCN	0.9327	0.9470	0.9321	0.9274
GAT	0.9401	0.9123	0.9247	0.9338
GHNN-ACO	0.9531	0.9379	0.9409	0.9387

ture used in this paper can better identify the task progressive aspect and non-task progressive aspect, and through local and global intelligent feature learning and recognition, this algorithm can better learn the main characteristics and properties of task allocation, as shown in Figure 6.

Meanwhile, in the link prediction task, the GHNN-ACO algorithm achieves efficient collaborative cooperation between intelligent agents. Through the path search of ACO algorithms, agents can find the optimal communication path, thus achieving more efficient task execution and cooperation. This kind of effective collaboration between agents is crucial for emergency response, which enables the entire Multi-agent system to respond to different emergencies more efficiently.

Moreover, although the GHNN-ACO algorithm has shown certain advantages, we have also identified some potential limitations. This algorithm is sensitive to initial parameter selection and requires careful adjustment to avoid falling into local optima. Accordingly, the algorithm in this paper has an accuracy rate of 95.31% in assigning multiple tasks to multiple agents, as shown in Table 2.

In addition, due to the iterative updating of GHNNs and ACO algorithms, the computing cost is high, which may pose challenges to the real-time application of large-scale Multi-agent systems. In future research, we will strive to further optimize the GHNN-ACO algorithm. By improving the initial parameter selection and optimizing the parallel computing ability of the algorithm, we hope to further improve the efficiency and performance of the algorithm. In addition, we will also explore its application in other types of multi-agent scenarios to expand its applicability and generalization.

5. CONCLUSIONS

In conclusion, the research results of this paper demonstrate that the GHNN-ACO algorithm performs well in heterogeneous multi-agent scenarios, providing an effective solution for node classification and link prediction tasks in agent systems. This algorithm has potential application prospects in emergency response of multi-agent systems and also offers a new idea for further research and optimization of multi-agent cooperation. The algorithm in this paper achieves an accuracy rate of 95.31% in assigning multiple tasks to multiple agents. We are optimistic about the application of the GHNN-ACO algorithm in heterogeneous multi-agent scenarios and believe that it will play an important role in future intelligent agent systems.

DECLARATIONS

Acknowledgments

Dr. Ma expresses his gratitude to Prof. Duan for his remarks and discussion of multi-agent formation control theory and paper structure issues.

Authors' contributions

Made substantial contributions to the research, idea generation, and software development and conducted the GHNN-ACO experiments. Wrote and edited the original draft: Ma Z

Performed process guidance responsibility for the planning and execution of the study and the evolution of overarching research aims, critical review, and material support. Review and revise the original draft: Gong H

Availability of data and materials

Not applicable.

Financial support and sponsorship

None.

Conflicts of interest

All authors declared that there are no conflicts of interest.

Ethical approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Copyright

© The Author(s) 2023.

REFERENCES

1. Zhang C, Song D, Huang C, Swami A, Chawla NV. Heterogeneous graph neural network. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. Anchorage, AK, USA: ACM; 2019. pp. 793–803. [DOI](#)
2. Fu X, Zhang J, Meng Z, King I. MAGNN: Metapath aggregated graph neural network for heterogeneous graph embedding. In: WWW '20: The Web Conference 2020. Taipei, Taiwan: ACM / IW3C2; 2020. pp. 2331–41. [DOI](#)
3. Jing Y, Yang Y, Wang X, Song M, Tao D. Amalgamating knowledge from heterogeneous graph neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Online: Computer Vision Foundation / IEEE; 2021. pp. 15709–18. [DOI](#)
4. Zhao J, Wang X, Shi C, et al. Heterogeneous graph structure learning for graph neural networks. In: Proceedings of the 35th AAAI Conference on Artificial Intelligence. Online: AAAI Press; 2021. pp. 4697–705. [DOI](#)
5. Liu Z, Chen C, Yang X, et al. Heterogeneous gGraph neural networks for malicious account detection. In: Proceedings of the 27th ACM International Conference on Information and Knowledge Management. Torino, Italy: ACM; 2018. pp. 2077–85. [DOI](#)
6. Jin D, Huo C, Liang C, Yang L. Heterogeneous graph neural network via attribute completion. In: WWW '21: The Web Conference 2021. Online: ACM / IW3C2; 2021. pp. 391–400. [DOI](#)
7. Guo J, Yang C. Learning power allocation for multi-cell-multi-user systems with heterogeneous graph neural networks. *IEEE Trans Wireless Commun* 2021;21:884–97. [DOI](#)
8. Bhatta K, Chang Q. An integrated control strategy for simultaneous robot assignment, tool change and preventive maintenance scheduling using Heterogeneous Graph Neural Network. *Robot Comput Int Manuf* 2023;84:102594. [DOI](#)
9. Ru J, Hao D, Zhang X, Xu H, Jia Z. Research on a hybrid neural network task assignment algorithm for solving multi-constraint heterogeneous autonomous underwater robot swarms. *Front Neurorobot* 2023;16:1055056. [DOI](#)
10. Zhang X, Zhao H, Xiong J, et al. Scalable Power control/beamforming in heterogeneous wireless networks with graph neural networks. In: Proceedings of the IEEE Global Communications Conference. Madrid, Spain: IEEE; 2021. pp. 1–6. [DOI](#)
11. Wei Y, Fu X, Sun Q, et al. Heterogeneous graph neural network for privacy-preserving recommendation. In: Proceedings of the IEEE International Conference on Data Mining. Orlando, FL, USA: IEEE; 2022. pp. 528–37. [DOI](#)
12. Barbosa VC, Huang HK. Static task allocation in heterogeneous distributed systems. *COPPE, Universidade Federal do Rio de Janeiro, ES-149/88 (June 1988)* 1988. [DOI](#)
13. Wang Z, Liu C, Gombolay M. Heterogeneous graph attention networks for scalable multi-robot scheduling with temporospatial constraints. *Auton Robot* 2022;46:249–68. [DOI](#)
14. Zeng Q. Characteristic analysis and route optimization of heterogeneous neural network in logistics allocation system. *Computational Neuroscience* 2022;2022:1713183. [DOI](#)
15. Salcedo-Sanz S, Xu Y, Yao X. Hybrid meta-heuristics algorithms for task assignment in heterogeneous computing systems. *Comput Operat Res* 2006;33:820–35. [DOI](#)
16. Liu Y, Wang W, Hu Y, et al. Multi-agent game abstraction via graph attention neural network. In: Proceedings of the 34th AAAI Conference on Artificial Intelligence. New York, NY, USA: AAAI Press; 2020. pp. 7211–18. [DOI](#)
17. Yang S, Yang B, Kang Z, Deng L. IHG-MA: Inductive heterogeneous graph multi-agent reinforcement learning for multi-intersection

- traffic signal control. *Neural Netw* 2021;139:265–77. DOI
18. Zheng P, Xia L, Li C, Li X, Liu B. Towards Self-X cognitive manufacturing network: An industrial knowledge graph-based multi-agent reinforcement learning approach. *J Manuf Syst* 2021;61:16–26. DOI
 19. Wang J, Yuan M, Li Y, Zhao Z. Hierarchical attention master–slave for heterogeneous multi-agent reinforcement learning. *Neural Netw* 2023;162:359–68. DOI
 20. Wang X, Zhang L, Lin T, et al. Solving job scheduling problems in a resource preemption environment with multi-agent reinforcement learning. *Robot Comput-Int Manuf* 2022;77:102324. DOI
 21. Bettini M, Shankar A, Prorok A. Heterogeneous multi-robot reinforcement learning. *arXiv preprint arXiv:230107137* 2023. DOI
 22. Lee ES, Zhou L, Ribeiro A, Kumar V. Graph neural networks for decentralized multi-agent perimeter defense. *Front Control Eng* 2023;4:1104745. DOI
 23. Fernando M, Senanayake R, Choi H, Swamy M. Graph Attention Multi-Agent Fleet Autonomy for Advanced Air Mobility. *arXiv preprint arXiv:230207337* 2023. DOI
 24. Huang J, Su J, Chang Q. Graph neural network and multi-agent reinforcement learning for machine-process-system integrated control to optimize production yield. *J Manuf Syst* 2022;64:81–93. DOI
 25. Mo X, Huang Z, Xing Y, Lv C. Multi-agent trajectory prediction with heterogeneous edge-enhanced graph attention network. *IEEE Trans Intell Transport Syst* 2022;23:9554–67. DOI
 26. Jia X, Wu P, Chen L, et al. Hdgt: Heterogeneous driving graph transformer for multi-agent trajectory prediction via scene encoding. *IEEE Trans Pattern Anal Mach Intell* 2023;45:13860-75. DOI
 27. Deka A, Sycara KP. Natural emergence of heterogeneous strategies in artificially intelligent competitive teams. In: Proceedings of the 12th International Conference on Swarm Intelligence. vol. 12689. Qingdao, China: Springer; 2021. pp. 13–25. DOI
 28. Ji X, Li H, Pan Z, Gao X, Tu C. Decentralized, unlabeled multi-agent navigation in obstacle-rich environments using graph neural networks. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems. Prague, Czech Republic: IEEE; 2021. pp. 8936–43. DOI
 29. Li M, Chen S, Shen Y, et al. Online multi-agent forecasting with interpretable collaborative graph neural networks. *IEEE Trans Neural Netw Learn Syst* 2022;1–15. DOI
 30. Ezugwu AE, Frincu ME, Adewumi AO, Buhari SM, Junaidu SB. Neural network-based multi-agent approach for scheduling in distributed systems. *Concurrency and Computation: Practice and Experience* 2017;29:e3887. DOI
 31. Li J, Ma H, Zhang Z, Li J, Tomizuka M. Spatio-temporal graph dual-attention network for multi-agent prediction and tracking. *IEEE Trans Intell Transport Syst* 2021;23:10556–69. DOI