Multiattribute E-CARGO Task Assignment Model Based on Adaptive Heterogeneous Residual Networks

Zhaowei Liu, *Member, IEEE* and Zongxing Zhao, *Graduate Student Member, IEEE*

***Abstract*—Mobile crowd sensing (MCS) is an emerging approach to collect data using smart devices. In MCS, task assignment is described as assigning existing tasks to known workers outside the constraints of task demand attributes and worker attributes, and maximizing the profit of the platform. However, workers and tasks often exist in different environments and heterogeneous features such as workers with attributes are not considered, leading to nondeterministic polynomial (NP)-hard task assignment problems. To optimize such problems, this article proposes a multiattribute environments-classes, agents, roles, groups, and objects (E-CARGO) task assignment model based on adaptive heterogeneous residual networks (AHRNets). The AHRNet is integrated into deep reinforcement learning (DRL) to optimize the NP-hard problem, dynamically adjust task assignment decisions and learn the relationship between workers with different attributes and task requirements. Multiattribute E-CARGO uses group task assignment policy to obtain the ideal worker-task assignment relationship. Compared with traditional heuristic algorithms for solving NP-hard, this method has the flexibility and applicability of adaptive networks, enabling the solver to interact with and adapt to new environments and generalize its experience to different situations. Under various experimental conditions, a large number of numerical results show that this method can achieve better results than the reference scheme.**

***Index Terms*—Adaptive heterogeneous residuals network (AHRNet), group task assignment (GTA), multiattribute environments-classes, agents, roles, groups, and objects (E-CARGO) model, task assignment.**

I. Introduction

THE application of mobile crowd sensing (MCS) in collecting, processing, and analyzing environmental data with large-scale user participation shows its unique advantages [1]. In contrast to traditional sensor networks, MCS can make full use of distributed mobile device networks, get a lot of users involved, and get around the problems that come with using just one sensor in traditional sensing methods. On the other hand, compared with static sensors, MCS can provide real-time environmental information because it can collect and process data instantly. This is valuable for real-time monitoring and decision-making, such as traffic flow monitoring and urban air quality monitoring. Therefore, in order to successfully implement MCS systems, careful consideration must be given to meeting the needs of device owners and effectively handling the dynamic changes of swarm intelligence.

At this stage, there are many typical MCS models [2], [3], [4], [5], [6], but the execution process is generally consistent. First, for specific application scenarios, the task issuer converts the required problems into specific tasks and releases them to the mobile device user group. Among them, the tasks can be data acquisition, geographic location tagging, image recognition, etc. [7], [8]. After the task is successfully released, the task distribution and reception will be completed through the platform according to different constraints between workers and tasks, in order to make workers more enthusiastic about tasks and maximize the rewards workers receive for completing tasks. As well as improving workers' work efficiency and work quality, a large number of workers must be recruited to cover the existing work area. Therefore, cost issues will limit this idea and cannot make the system bear more workers.

From the above discussion, it is not difficult to find that one of the core issues currently faced by MCS is task assignment [9], [10], [11]. The second is that the vast majority of current MCS models [12], [13], [14], [15] are for workers with homogeneous attributes, implying that there are no differences between workers other than different task demands, which is clearly not in line with the current assignment scenarios of major systems. Therefore, in this article, we follow the assumption of heterogeneity, that is, the attribute relationship between workers is greater than 2, which can be interpreted as workers in different occupations also having the same task requirements or workers in the same occupation having multiple task requirements. Based on this condition, the problem of assigning tasks in various application settings is looked at from a number of different angles and attributes. In view of the NP-hard problem in the traditional MCS model when optimizing task assignment, this study uses a heuristic scheme to find the optimal solution. Therefore, this article proposes a new research plan from the perspective of group task assignment (GTA).

Manuscript received 1 August 2023; revised 31 October 2023 and 11 December 2023; accepted 14 December 2023. This work was supported in part by the National Natural Science Foundation of China under Grant 62272405, in part by the School and Locality Integration Development Project of Yantai City (2022), in part by the Youth Innovation Science and Technology Support Program of Shandong Provincial under Grant 2021KJ080, and in part by the Natural Science Foundation of Shandong Province under Grant ZR2022MF238. *(Corresponding author: Zhaowei Liu.)*

The authors are with the School of Computer and Control Engineering, Yantai University, Yantai 264005, China (e-mail: lzw@ytu.edu.cn; zhaozx_ytu@163.com).

Digital Object Identifier 10.1109/TCSS.2023.3344173

On the one hand, the traditional environments-classes, agents, roles, groups, and objects (E-CARGO) model is a useful solution to solve the problem of organizational collaboration distribution. The E-CARGO model of role-based collaboration (RBC) [16], [17], [18] is difficult to adapt to the interaction between existing complex systems, that is, there is a many-to-many relationship between task requirements and worker capabilities. The reason is that the RBC model [19], [20] focuses on analyzing the collaborative relationship between roles, and there is an urgent need for an allocation model that can handle multiattribute graph data. Therefore, this study will use the multiattribute E-CARGO model to solve such problems. This model analyzes the interactive relationship between workers and tasks into a graph structure representation and uses the multiattribute E-CARGO model to simulate the relationship between workers with different attributes and the task environment. This model benefits from the fact that multiattribute E-CARGO model can take into account the advantages of different workers (attribute differences) based on the original interaction between workers. While ensuring the collaborative relationship between workers, team managers assign corresponding tasks to subworkers through worker negotiation, task subdivision, etc.

On the other hand, the worker and task system in this study can be abstracted into a heterogeneous graph problem, benefiting from the powerful generalization ability of deep heterogeneous graph neural networks [21], [22], [23]. Meanwhile, deep reinforcement learning (DRL) applies known experience to the specific instances required through heuristic learning. However, the problem faced by existing DRL [24], [25], [26] is that traditional neural networks are difficult to adapt to multidemand scenarios, and the execution task assignment often results in unsolvable (NP-hard) problems, which usually have high-dimensional state space and complex constraints. This article handles this problem by integrating a heterogeneous adaptive residual network ($\text{AHRNet}_{\text{DRL}}$). This study uses a combination of deep adaptive heterogeneous residual network (AHRNet) and DRL to simplify the complexity of the problem. The worker structure graph representation with different attributes is embedded into real-valued, and the embedding vectors of different task requirement instances are input into the deep network to explore high-order optimal solutions and achieve task assignment for multiattribute workers.

The main contributions of this article are as follows.

1) In view of the heterogeneous characteristics of the requirements between worker abilities and task environments, this article explores the multiattribute task assignment problem in MCS. For the first time, a multiattribute E-CARGO task assignment model based on adaptive heterogeneous residual networks is proposed, which can more effectively extract the common relationships between tasks and workers.
2) This article models the task allocation problem as a continuous decision-making process based on the multiattribute E-CARGO model. Through the group task allocation strategy, the tasks and workers comply with the many-to-many relationship. When dealing with different task assignment problem instances, AHRNet is integrated into DRL to learn from various tasks in the training stage and deploy the learned model to solve new tasks.
3) This article simulates the demand problems between tasks and workers in a large number of real-world scenarios, proving the high performance and excellent generalization ability of this method in various heterogeneous scenarios.

II. Related Work

As the research on MCS [27], [28], [29], [30] gradually deepens, task assignment has increasingly become a key link in MCS performance evaluation, which determines how tasks are effectively distributed to participants to achieve efficient data collection and processing. Currently, many scholars have achieved remarkable results in single-task assignment scenarios. For example, a variant genetic algorithm is used in [31] to solve the NP-hard problem in single-task assignment and maximize the task completion rate to achieve the optimal task assignment scheme. Competition-congestion-aware stable matching (CCASM) [32] designs a stable matching mechanism for worker competition and complex behavioral selection, which can be regarded as a multiobjective optimization problem. Similarly, DPF [33] uses a differential private algorithm to allocate relevant workers while protecting task quality information, achieving worker recruitment on the basis of protecting privacy. In real-life interactive systems, workers often play multiple roles. The single-task allocation mentioned above, in which one worker can only handle one task (single attribute), is poorly applicable. However, the E-CARGO model [34], [35], [36] can effectively solve such problems. For example, Sheng et al. [37] consider the status of different groups and use adaptive collaboration (AC) to achieve team performance distribution in the collaborative system. In the E-CARGO model, group role assignment (GRA) [38] is an important task of RBC. Research [39] is based on the GRA and solves the group role allocation problem with balance problems through the correlation method. At the same time, for the problem of self-service spatiotemporal crowdsourcing (SSC), the GRA-based model [40] also shows excellent performance. Group multirole assignment (GMRA) is an extension of the GRA model that incorporates multirole issues. For example, [41] and [42] use different constraint strategies to realize the constraints and formal expression of agents. Such E-CARGO models are useful explorations to achieve multitask allocation.

In the large environment of task demand, which is often dynamic. MCS needs to make corresponding changes according to changes in tasks. Traditional heuristic solutions often mainly deal with static environments and obviously cannot exert their advantages in dynamic environments. An optimal solution, DRL, has gradually attracted attention. In [43], a DRL-based algorithm is proposed to deal with the cell selection problem, and its goal is to dynamically select a subset of cells in the sensing area to infer the data of the entire sensing area. In [44], the authors study a task assignment problem modeled with a constrained optimization problem. The problem aims to arrange

TABLE I
NUMBER OF WORKERS REQUIRED PER TASK

Tasks	Task1	Task2	Task3	Task4	Task5	Task6	Task7	Task8
Estimated finish time (days)	80	100	150	120	110	100	100	180
Minimum number of workers	10	19	20	18	18	15	16	21
Minimum number of groups	2	2	1	2	3	2	2	4

TABLE II
WORKERS NEEDED TO MATCH TASK ATTRIBUTES

Tasks	Task 1 (Web Development)		Task 2 (Hardware, Software Maintenance)		Task 3 (System Testing)	...
Required worker attributes	Front-end engineer	Back-end engineer	Hardware engineer	Software engineer	Test Engineer	...
Minimum number of workers	3	5	6	5	18	...
Maximum number of workers	5	7	14	13	20	...

routes for workers within the maximum travel distance while maximizing the total task quality of task initiators.

From the above related work, it can be analyzed that the E-CARGO model shows superior performance in processing multiattribute workers and tasks, and DRL also shows its superior performance in processing dynamic task environments. However, it is difficult to leverage their respective advantages in a heterogeneous environment with multiple roles. It can be explained as: a large number of workers compete for a small number of tasks or the number of workers cannot meet the demand for a large number of tasks. Therefore, it is particularly important to train a model that can adaptively select the demand relationship between workers and tasks in this heterogeneous environment.

III. A Real Application Scenario

In a crowd-sensing system company, people can share environmental sensing data through a cell phone application in order to gather information about air quality, traffic flow, and noise levels, among other things. The goal of the system is to provide real-time environmental information to the public by integrating various sensory data through a crowd sourcing approach.

Recently, with the increase in the number of users, the amount of perceptual data collected has increased greatly, and the system needs to assign this data to the right roles for processing and analysis. However, due to the sheer size and urgency of the system's data, a task assignment problem arose within the company. While working on this project, the company's chief technology officer (CTO) was faced with the challenge of assigning tasks appropriately to ensure the project was delivered on time. Therefore, the CTO divided the entire project into multiple tasks, which were executed by different groups. The execution time of each task is shown in Table I. The CTO requires A to organize different groups to complete these tasks and achieve the most effective performance in the shortest time. Obviously, GTA is used.

In order to be able to complete the given task as scheduled, Manager A analyzes the workers' features as well as the task demands within each group. As shown in Table I, it can be seen that the minimum work status of the workers to complete the task requirements within the given time. As shown in Table II, the minimum and maximum number of workers required for different tasks in each group are further analyzed. Based on experience, Manager A recognizes the need to align workers' capabilities with the demands of the tasks being handled, in order to maximize performance and complete the task most efficiently. Manager A needs to resolve the relationship between workers and tasks and must use workers' abilities to handle task requirements most effectively.

Known from experience, Manager A recognizes that workers' abilities change over time and further impact on group performance. He recognizes that a fixed task assignment scheme will not ensure maximum group performance. Confronted with this problem, Manager A requests that past data be used to predict trends in each worker's ability.

In short, the screening of worker abilities will be regarded as an attribute selection problem in this article, and an effective solution model will be given in detail next.

IV. The System Model

In this section, the group task allocation scheme of the multiattribute E-CARGO model will be introduced in detail, and the use of AHRNet will be integrated into DRL to handle multiattribute tasks and worker task allocation issues. Specifically, first, through the multiattribute E-CARGO model, the group allocation strategy is used to divide workers and tasks into different groups, so that the workers in each group are most beneficial to the completion of the task. Then, the assigned groups (Group 1, Group 2,..., Group n) are used as input data to obtain the feature similarity between workers and tasks through the heterogeneous adaptive residual network. Finally, the worker's degree of satisfaction with completing the task is calculated through the policy network to implement the task assignment strategy. As shown in Table III, the symbols involved in this model are defined.

A. E-CARGO Model With Multiple Attributes

Based on the concept of the E-CARGO model [45], [46], [47], this article considers the multiattribute problem of requirements between workers and tasks that we mentioned. The framework model is shown in Fig. 1. A system consisting of ten-tuples is defined to be represented as a multiattribute E-CARGO

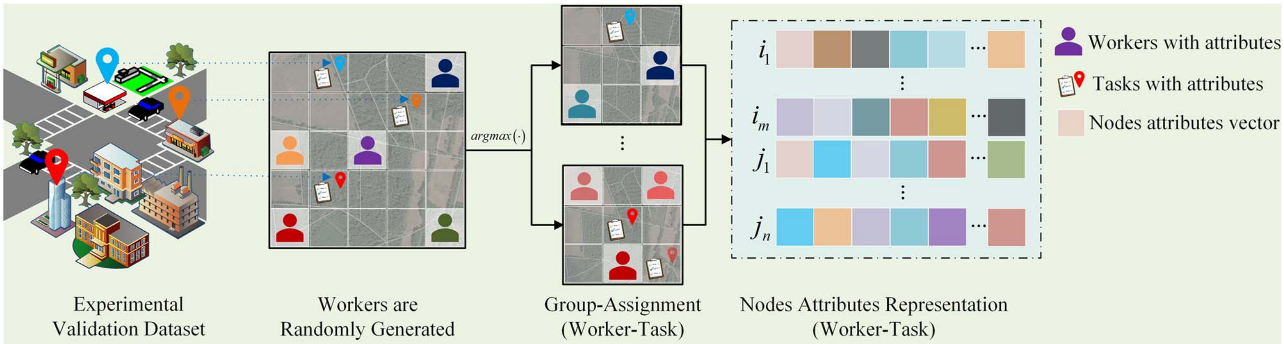

Fig. 1. Multiattributes E-CARGO model. The multiattribute E-CARGO model uses a group assignment strategy to divide workers and tasks into different groups. The workers in each group are most beneficial to the completion of tasks. The assigned groups are used as input data for AHRNet$_{\text{DRL}}$ to complete training.

TABLE III
NOTATIONS AND EXPLANATIONS

Notations	Explanations
w_i	Worker i
t_i	Task j
N_m	Set of workers with attributes
N_n	Set of tasks with attributes
$T[i,j]$	Assignment relationship between w_i and t_j
$Q[i,j]$	Quality value of w_i competence of t_j
$S[i,j]$	Similarity between w_i and t_j
T_{gw}	Results of group assignment
T'_{gw}	Matching results of workers and tasks
a_0	Worker/task attribute information
$\hat{\alpha}_{ij}$	The similarity score
h_i	Features of node i
$\mathcal{M}_\phi$	Special types of transformation matrices
β_i	The adaptive score
s_w^t	State of w_i executing t_j
a_{ij}^t	Action of w_i executing t_j
S_{level}	The satisfaction of workers

model, which is described as

$$\sum = < C\,(\text{classes}), O\,(\text{objects}), A\,(\text{works}), M\,(\text{messages}), R\,(\text{task}), E\,(\text{environment}), G\,(\text{groups}), H\,(\text{human users}), s_0\,(\text{system's initial state}), a_0(\text{attritutes}) >. \quad (1)$$

Each component of the ten-tuple represents its corresponding set. In order to unify the representation, this article expresses the meaning of A as a set of *workers* instead of a set of *agents* in the original E-CARGO, and in the original based on the E-CARGO model, this article introduces the concept of attribute set a_0.

In order to solve the problems involved in Section III, we first need to understand the special symbol definitions involved in the E-CARGO model. Among them, N is represented as a set of nonnegative integers, N_m represents the size of $|A|$ (worker set), N_n represents the size of $|R|$ (task set), $(< i_0, a_0^i >, < i_1, a_1^i >, ..., < i_{N_A - 1}, a_{N_m - 1}^i >) \in A = \{0, 1, ..., N_m - 1\}$ represents the index of the worker number and the attributes $(< a_0^i, a_1^i, ..., a_{N_m - 1}^i >)$, and $(< j_0, a_0^j >, < j_1, a_1^j >, \ldots, < j_{N_n - 1}, a_{N_n - 1}^j >) \in R = \{0, 1, \ldots, N_n - 1\}$ represents the index of the task number and its attributes $(< a_0^j, a_1^j, ..., a_{N_n - 1}^j >)$. Based on the problem analysis, this model will detail the worker-task assignment relationship. Next, specific related definitions will be given.

Definition 1: The assignment relationship between workers and tasks is denoted as $< w, t >$, with workers denoted by w and tasks by t.

Definition 2: The task scope vector L is the minimum number of workers required for the task in group (g) of environment (e), indicating the minimum number of workers required for a single task $L[j] \in N$ to perform the task. Such relationships in groups and environments are defined as ge.

Definition 3: Q is used to represent the worker's competency quality matrix (evaluation value matrix) for the task, where $Q[i,j]$ represents the quality value of w_i $(0 \le i < N_m)$ competency t_j $(0 \le j < N_n)$. Q can be obtained by comparing the worker with the set of tasks and is used to characterize the correlation relationship between the worker and the task.

Definition 4: T denotes whether the assignment between a worker and a task has been completed or not, denoted as $T[i,j] \in \{0,1\}$ $(0 \le i < N_m; 0 \le j < N_n)$. When $T[i,j] = 1$, it means that worker w_i is assigned to task t_j, and the state of worker at this time is denoted as assigned; if $T[i,j] = 0$, it means that worker w_i is not assigned to task t_j. If the number of workers in the same group is sufficient, then task t_j is assignable in this group (g), and the matrix T must exist, that is, $\sum_{i=0}^{N_m - 1} T[i,j] \ge L[j] (0 \le j < n)$. It can be seen through reasoning that the existence of T means that group (g) can work normally.

Definition 5: In order to better reflect the completion satisfaction of the entire group, group performance ϑ is defined to represent the sum of performance obtained by workers completing tasks. Specifically expressed as

$$\vartheta = \sum_{i=0}^{N_m - 1} \sum_{j=0}^{N_n - 1} Q\,[i,j] \times T\,[i,j]\,. \quad (2)$$

Definition 6: When there are two or more workers with the same ability competing for the same task, this phenomenon is defined as worker conflict. The worker conflict matrix is defined as a matrix of order $m \times m$, expressed as W^C. When $W^C\,[i_1, i_2] = 1$, it means there is a conflict between two different workers. When the value is 0, it means there is no conflict.

A conflict situation can often be expressed as $W^C[i_1, i_2] = W^C[i_2, i_1]$. Based on the results of the experiments, conflict is a low probability event because it usually happens when the characteristics of the two workers stay the same. In the data currently used, this will also inevitably occur. With the use of the random value mechanism, two workers can only take a random value of 0 or 1, one of whom is 1 and the other is 0. The model will choose to assign the value of 1 worker to complete this task to solve the problem of worker conflict.

Definition 7: In order to better satisfy the need for multiple workers to complete a single task in most environments, a necessary condition is needed to discuss GTA issues. When there are enough workers in a group, it is expressed as: $N_m \geq \sum_{j=0}^{N_n-1} L[j]$; when $N_m \leq \sum_{j=0}^{N_n-1} U[j]$, indicating that there are still vacancies for the workers required for the task. Therefore, it is necessary to consider the upper and lower limits of the number of workers required for a task. The following *Definition 8* will give a detailed definition.

Definition 8: The matrix $L\_\mathrm{gw}[l, j]$ is expressed as the $N_{(g)} \times N_{(n)}$ minimum number of workers matrix required for task $j\,(0 \leq j < N_n)$ in the lth group. Similarly, the matrix $U\_\mathrm{gw}[l, j]$ is expressed as the $N_{(g)} \times N_{(n)}$ maximum number of workers matrix required for task $j\,(0 \leq j < N_n)$ in the lth group. Here, $N(g)$ represents the number of groups. The number of workers satisfied by the same task should comply with the following constraints: L_gw[l,j] $\leq$ gw_L_num[l] $\leq$ U_gw[l,j].

Definition 9: In order to better describe the benefits that can be achieved by completing this task, the minimum group benefit vector matrix $Min\_P_G[l]$ is set.

Based on the clarity of the above nine definitions, the assignment relationship between workers with heterogeneity and tasks will be analyzed in detail next. At the beginning of this section, the attribute content between workers and tasks has been clearly expressed. In order to describe the multiattribute situation, this article constructs a matrix of size $N_m \times K$, where K represents the number of attributes. Define $A^w[i, k]$ to represent the value of the kth attribute of the ith worker, and $A^t[j, p]$ to represent the value of the pth attribute of the jth task. In order to have a better matching relationship between workers and tasks, a multiattribute similarity function is used to measure. Specifically expressed as

$$S[i, j] = \sqrt{\sum_{k}^{K} \sum_{p}^{P} \left(A^w[i, k] - A^t[j, p]\right)^2}. \tag{3}$$

Specifically, the assignment program is expressed as follows:

$$T^s[i, j] = \frac{S[i, j]}{\sum_k^K S[i, j]}. \tag{4}$$

Here, $T^s[i, j] \in (0, 1)$, denotes the weight of worker i assigned to task j and satisfies that the sum of the weights of each worker is 1. It can be seen that each worker will be assigned to the corresponding task and maximize the utilization of workers.

Definition 10: In order to describe the assignment relationship between workers and tasks in different groups more clearly, the assignment matrix for the lth group is defined as $T^s\_\mathrm{gw}[l]$. Similarly, $P_G\_\mathrm{gw}[l]$ is defined to represent the benefits obtained when workers and tasks are assigned. The specific calculation method is

$$P_G\_\mathrm{gw}[l] = \sum_{i=0}^{N_m-1} \sum_{i=0}^{N_n-1} Q\left[A_i^w, A_j^t\right] \times T^s\_\mathrm{gw}[l, i, j]. \tag{5}$$

In the above equation, $Q\left[A_i^w, A_j^t\right]$ represents the benefits that workers with attributes can obtain from completing tasks with attributes. In view of this, a feasible group assignment T_{gw} of the GTA can be obtained

$$T_{\mathrm{gw}} = \underset{T' \in \{0,1\}^{l \times N_m \times N_n}}{\operatorname{argmax}} \sum_{l}^{N_g-1} \sum_{i=0}^{N_m-1} \sum_{i=0}^{N_n-1} Q\left[A_i^w, A_j^t\right] \times T'[l, i, j]. \tag{6}$$

Through the above analysis, the multiattribute E-CARGO model has obtained different groups $\{g_0, g_1, \ldots, g_l\}$ between workers and tasks. What is included in this group is that completing the task smoothly and on schedule can maximize benefits. There are often a large number of workers and tasks in real systems. Therefore, this research needs to extend this assignment method to deep heterogeneous networks and use the learnable method of the network to complete the assignment relationship between a large number of workers and tasks. Next, this article will introduce in detail the use of AHRNet integrated into DRL to expand its generalization capabilities based on multiattribute E-CARGO model assignment and optimize the NP-hard problems caused by task assignment in traditionally large amounts of data.

B. Adaptive Heterogenous Residual Networks With Integrated DRL (AHRNet$_{DRL}$)

Traditionally, GAT [48] has demonstrated strong capabilities in the field of homogeneous graph neural networks, but its neglect of node attribute aspects is often unscalable for heterogeneous networks. To address this issue, this study extends the GAT attention mechanism to heterogeneous networks, and the framework is shown in Fig. 2. To find the best assignment vector representation relationship between workers and task nodes, an adaptive approach is used to build a deep residual network. Each of these nodes corresponds to different performance states as well as operations, workers[assigned/unassigned, taskselected/pending], the former corresponding to the worker's performance state and the latter to the corresponding operation. At the same time, these nodes will be fed into the Q-learning layer of DRL with these two features to predict the Q-values between the worker and the task and compute whether the two can be paired successfully or not. We create a heterogeneous graph $G = (H, E)$ from the multiattribute E-CARGO model's distribution relationship between workers and tasks output. This graph is then fed into AHRNet$_{\mathrm{DRL}}$.

In order to be able to obtain optimal Q-values, AHRNet$_{\mathrm{DRL}}$ is essential for training the state-action accuracy of workers and

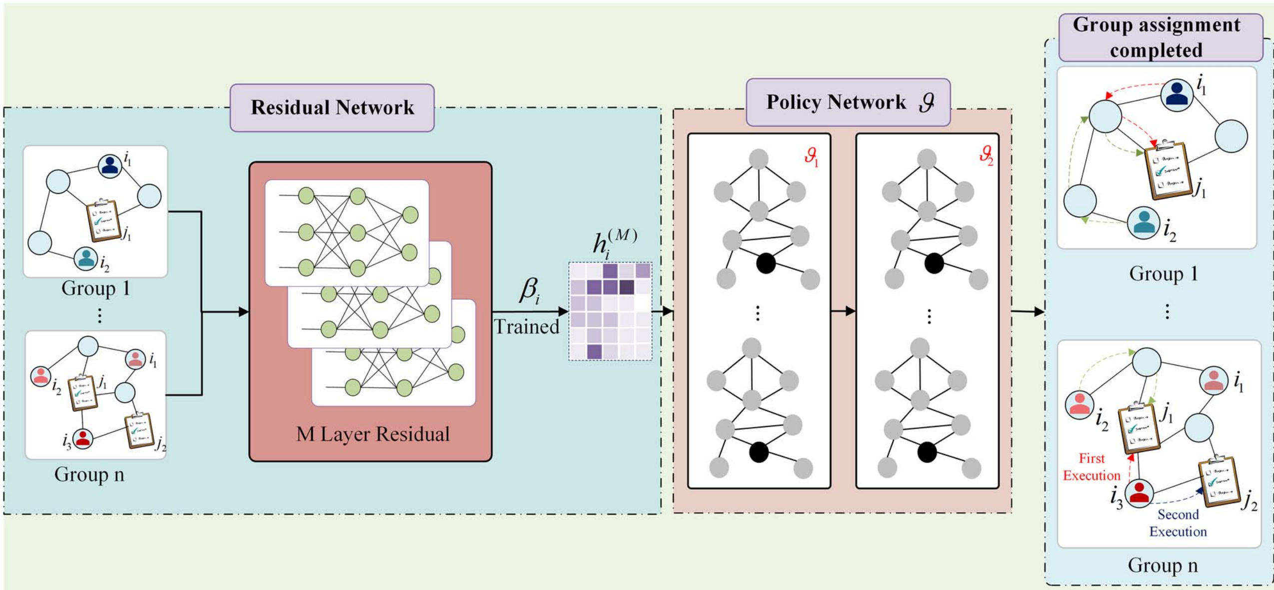

Fig. 2. Heterogeneous adaptive residual networks with integrated DRL. Obtain the feature similarity between workers and tasks in the group (Group 1,..., Group n) through AHRNet$_{\mathrm{DRL}}$ model, and then calculate the degree of satisfaction of workers completing tasks through the policy network to implement the task assignment strategy.

tasks. Next, i and j will be used to denote the worker-task index numbers within the group that are modeled by the multiattribute E-CARGO model in order to be assigned well, respectively. This study takes a more in-depth consideration of the edge-type interaction relationship between workers and tasks, expressing the degree of attraction between workers as

$$\hat{\alpha}_{ij} = \frac{\exp\left(\mathrm{LR}\left(a^T\left[Wh_i^l \middle\| Wh_p^l \middle\| W_r r_{\psi(<i,j>)}\right]\right)\right)}{\sum_{p\in\mathcal{N}_i, l\in N_g}\exp\left(\mathrm{LR}\left(a^T\left[Wh_i^l \| Wh_p^l \| W_r r_{\psi(<i,p>)}\right]\right)\right)}. \tag{7}$$

Here, the assigned similarity scores $\hat{\alpha}_{ij}$ between worker i and task j within the lth group are computed, LR$(\cdot)$ represents the LeakyReLU function, $r_{\psi(<i,j>)}$ is the assigned d_p- dimensional edge-type $\psi(<i,j>)$ embedding, and W_r is a feature type transformation matrix.

Due to the problem of too few training layers, traditional GNNs [49], [50], [51], [52] often find it difficult to obtain a more in-depth representation of high-order node features. The key to effectively solving such problems is the residual network layer. Because of the need for data interaction between each layer, the dimensions of the data trained between each layer will be inconsistent. This study sets up a preactivation feature representation

$$h_i^{(m)} = \sigma\left(\sum_{j\in\mathcal{N}_i}\alpha_{ij}^{(m)}W^{(m)}h_j^{(m-1)} + W_{\mathrm{res}}^{(m)}h_i^{(l-1)}\right). \tag{8}$$

In the above equation, $\alpha_{ij}^{(m)}$ is used to measure whether worker i is better qualified for task j. In the graph representation [53], [54], workers and tasks remain adjacent to each other. Through preactivation, the feature dimensions of the lth layer are consistent with the feature dimensions of the $(l-1)$th layer. When the dimension changes, a learnable linear transformation matrix $W_{\mathrm{res}}^{(m)} \in \mathbb{R}^{d_{l+1}}$ is needed to make the dimension consistent.

Obviously, when the number of layers changes, the distribution similarity scores between nodes will also change accordingly. In order to more effectively enhance the learning ability of the model, Z independent attention mechanisms are used. The change rules are as follows:

$$\alpha_{ijz}^{(m)} = (1-\beta)\,\hat{\alpha}_{ijz}^{(m)} + \beta\alpha_{ijz}^{(m-1)}. \tag{9}$$

Thereby, the mth layer feature $\hat{h}_{iz}^{(m)}$ can be represented as

$$\hat{h}_{iz}^{(m)} = \sum_{j\in\mathcal{N}_i}\alpha_{ijz}^{(m)}W_z^{(m)}h_j^{(m-1)}. \tag{10}$$

For each layer, the output dimensions are often difficult to be rounded off, and the node feature representations $h_i^{(M)}$ of the final layer (Mth) are obtained by averaging through multiple heads of attention

$$h_i^{(M)} = \frac{1}{Z}\sum_{z=1}^{Z}\hat{h}_{iz}^{(M)}. \tag{11}$$

Finally, this algorithm will use the aggregated feature deviations between workers and tasks to calculate an adaptive score β_i that measures the feasibility of task assignment

$$\beta_i = \max\left(1 - \frac{\tau\lambda}{\left\|h_i^{(M)} - h_j^{(M)}\right\|_2}, 0\right). \tag{12}$$

Among them, $\left\|h_i^{(M)} - h_j^{(M)}\right\|_2$ represents the feature deviation between the two. The smaller the deviation, the more theoretically the worker is suitable for the task. Next, β_i will be used to select workers suitable for the task. The most critical point is that this study considers that the number of workers is saturated, and it is obvious that multiple workers are often needed to satisfy a task. However, there are differences in ability levels between workers, and β_i adaptive selection will be used to select workers whose characteristic abilities best match the tasks. The calculation is as follows:

$$T_{\text{gw}}^{'} = h_j^{(M)} \odot \left[\left(1-\beta_i\right) h_{i_1}^{(M)} + \beta_i h_{i_2}^{(M)} \right]. \tag{13}$$

Here, i_1 and i_2 are sorted according to their ability values. After the experimental analysis of this study, when different workers perform the task for the first time at the same time, β_i is relatively small, which shows that i_1 is more beneficial to the task; when i_1 performs multiple tasks, the β_i value will increase, and other workers will have a larger weight. The main reason is that the deviation of worker i_1 from the new task will be greater than the deviation from the initial task. The state-action relationship between workers and tasks will be analyzed below.

C. Q-Learning Layer

AHRNet$_{\text{DRL}}$ implements the local information vector feature representation of workers and task nodes. Based on these node vectors, the execution status s_w^t of the worker is further obtained

$$s_w^t = \text{relu}\left[\omega_{ij} \text{cat} \left(\sum_{i \in \mathcal{N}_j} h_i^{(M)}, \sum_{j \in \mathcal{N}_i} \alpha_{sj} h_j^{(M)} \right) \right]. \tag{14}$$

In the equation, ω_{ij} represents the network parameters. The cat$(\cdot)$ series method is used to obtain the worker's working status s_w^t for the task. The attention score α_{ij}^t mainly affects the worker's ability to perform the task

$$\alpha_{ij}^t = \frac{\exp\left(\text{LR}\left(a^T \left[W_d h_i \,\|W h_j\|\right]\right)\right)}{\sum_{i \in \mathcal{N}_j} \exp\left(\text{LR}\left(a^T \left[W_d h_i \,\|W h_j\|\right]\right)\right)}. \tag{15}$$

It can be seen from the equation that when there are insufficient workers, one worker will need to complete multiple tasks. However, as the number of tasks increases, the worker's ability α_{ij}^t will also decrease. The learnable decay matrix W_d is used to determine this change in the worker.

In addition to the change in the worker's state s_w^t, the consequent action will also change. The state-action change process will be done in the Mth layer residual network

$$a_{ij}^t = \text{relu}\left(\rho \text{cat}\left(h_i, h_j\right) + \varpi\right) \tag{16}$$

where ρ and ϖ denote the d-dimensional hyperparameters in the network.

For the state-action pair (s_w^t, a_{ij}^t), if the action between the two is interrupted during the execution of the task, the next state s_{w+1}^t will also change. Therefore, the Q-value with the stop action is assigned 0, otherwise, when proceeding normally, the worker will get the corresponding reward (Q-value). The specific expression is as follows:

$$\mathcal{Q}\left(s_w^t, a_{ij}^t; \vartheta\right) = \begin{cases} 0, & \text{if } s_{w+1}^t \text{ is termnal} \\ \vartheta_2^T \text{relu}\left(\vartheta_1 \text{cat}\left(s_w^t, a_{ij}^t\right)\right), & \text{otherwise.} \end{cases} \tag{17}$$

In the above equation, $\vartheta = (\vartheta_1, \vartheta_2) \in \mathbb{R}^{d \times d}$ are two layers of different training strategy networks. $\vartheta = (\vartheta_1, \vartheta_2) \in \mathbb{R}^{d \times d}$ will be used to represent the corresponding rewards obtained by the status and actions performed by the worker. The rewards obtained are utilized to obtain the satisfaction level S_{level} of the worker, which leads to the evaluation of the model performance

$$S_{\text{level}} = \frac{\mathcal{Q}_{\text{worst}}\left(s_w^t, a_{ij}^t; \vartheta\right) - \mathcal{Q}_{\text{target}}\left(s_w^t, a_{ij}^t; \vartheta\right)}{\mathcal{Q}_{\text{worst}}\left(s_w^t, a_{ij}^t; \vartheta\right) - \mathcal{Q}_{\text{best}}\left(s_w^t, a_{ij}^t; \vartheta\right)}. \tag{18}$$

Q_{worst}, Q_{target}, and Q_{best}, respectively, correspond to the worst, target, and best satisfaction levels of performing the task. Globally, consider the status of workers throughout the entire task processing process to evaluate the most realistic worker satisfaction. Obviously, when the worker satisfaction level is higher, it indicates that the assignment scheme of this model is the most reasonable.

D. Model Training

The main framework of this model can be divided into the following three parts: a multiattribute E-CARGO model that realizes the grouping of workers and tasks; an AHRNet$_{\text{DRL}}$ that adaptively assigns workers for different tasks; and a Q-learning layer that calculates the rewards that workers receive for completing the corresponding tasks. The algorithm description of the model is shown in Algorithm 1. End-to-end training is implemented between the three training layers, and for each real case of task assignment, the optimal assignment strategy can be achieved by forward propagation from input to output only once, and the training model is optimized again by back-propagation.

The total number of workers and tasks are denoted by N_m and N_n, respectively. Workers in this model need to make up to N_m decisions, each of which needs to be passed forward once based on a heterogeneous adaptive residual network. Similar to [12], the time complexity of one forward pass through the network can be approximated as $O\left(N_m + \left(N_m + N_n\right)^2 + N_n\right)\left(\hat{W}_1 + Z\hat{W}_2 + \hat{W}_3\right)$. Here, $\hat{W}_1$, $\hat{W}_2$, and $\hat{W}_3$ denote the time consumed by the three functional layers to process a graph with only one worker and one task, respectively. If we set $\hat{W}_g = \hat{W}_1 + \hat{W}_2 + \hat{W}_3$, the time complexity of one forward pass of a general graph instance can also be written as $O\left(\left(N_m + N_n\right)^2 \cdot Z\hat{W}_g\right)$. Since action decisions have at most N_n forward passes, the overall time complexity of this method to solve a problem instance is $O\left(\left(N_m + N_n\right)^2 \cdot N_n \cdot Z\hat{W}_g\right)$. It can be seen that processing graphs for task assignment has a higher complexity than processing homogeneous graphs [55], [56] for data uploading. The reason is that the attribute information of work nodes and task nodes is also a key factor affecting training. Similarly, the deepening of the model increases the complexity and is more conducive to capturing global information.

V. Experimental Analyses

All the experiments were run on a GPU server with two NVIDIA GeForce RTX 3090 GPUs and a 12th Gen Intel(R)

Algorithm 1 Algorithmic Implementation of the Model

Input: N_n, N_m, policy network ϑ, hyperparameters ρ, ϖ
Output: Worker satisfaction S_{level}

```
Multi-attribute E-CARGO
Calculate the worker-task assignment weight through (4);
Calculate the worker-task assignment relationship through (6);
Construct the obtained (Group 1 ,..., Group n) into a heterogeneous graph G = (H, E).
AHRNet_DRL
for i = 1, ..., n do
    for h_i ∈ H do
        Find the node neighbors N_i;
        for h_j ∈ N_i do
            Calculate the attraction α̂_ij of h_i to h_j according to (7);
            if l ≥ 2 then
            Update α_ijz^(m) according to (9);
            end if
        end for
        Calculate the M^th layer node features h_i^M according to (11);
        Calculate the adaptive score β_i of h_i on h_j according to (12);
    end for
end for
for i, j = 1, ..., n do
    Calculate the execution status s_w^t of h_i according to (14);
    Calculate the rewards Q(s_w^t, a_ij^t; ϑ) obtained by h_i after executing h_j according to (17);
end for
Calculate worker satisfaction S_level for each h_i.
```

Core(TM) i9-12900K 24-core processor. Python and PyTorch have respective versions of 3.8.0 and 1.11.0.

In order to verify the effectiveness of this model algorithm and conform to the interactive relationship between workers and tasks in real systems, the number of workers and tasks was randomly set, and a large number of experiments were completed in the existing mainstream task assignment models. For the adaptive heterogeneous residual network model integrated with DRL, the learning rate is set to 0.01, dropout is set to 0.8, weight decay is set to 0.0005, and K = 10. In this experiment, the average of ten classification accuracies will be recorded as the result.

A. Datasets and Baseline Models

Three realistically acquired worker-task system datasets are used in this model. Including: Berlin52 [57], 52 Berlin landmarks in the dataset are used as task objects for workers, and each landmark contains the specific attributes of its task; NRW1379 [57], in the same way as the Berlin52 dataset, but this dataset contains 1379 landmark information in North Westphalia; and GeoLife [57], because the dataset is too large, 39.92–40.02 north latitude and 116.01–116.05 east longitude were selected in the simulation. This dataset is different from Berlin52 and NRW379 in that the tasks and workers' locations are randomly generated according to the specific needs in the selected area, while only the workers' locations are randomly generated in Berlin52 and NRW379.

This study investigates the popular task assignment models in recent years and believes that the research work of CCASM [32] is currently more prominent. For the sake of fairness, the simulation experiment content will use the CCASM model as the comparison baseline and the comparison model [traditional stable matching (TSM) [32], random allocation (RA) [32], greedy for worker satisfaction (GWS) [32], greedy for platform benefit (GPB) [32], asynchronous task selection (ATS) [58], greedy-enhanced genetic algorithm (GGA) [59]] involved in the article. The specific model parameters will retain the original article settings. The selected comparison model is a more advanced model for dealing with the MCS task assignment scheme, and the model does not use an assignment strategy when dealing with the worker-task relationship, which can form a well-compared experiment and serve as an excellent control for the validation of our model. The baseline model is explained as follows.

1) TSM [32]: This baseline does not consider the impact of competition and congestion on workers' decisions in the MCS system.
2) RA [32]: This baseline randomly assigns tasks to different workers based on the number of task requirements and worker capabilities.
3) ATS [58]: This baseline simulates a way for workers to participate in MCS task assignment via mobile devices. According to the task sequence, workers select tasks that satisfy them in order.
4) GWS [32]: This baseline uses a greedy algorithm to maximize worker satisfaction in the MCS system.
5) GGA [59]: This baseline combines the greedy algorithm and the genetic algorithm, using the results of GWS as initialization input to achieve task assignment.
6) GPB [32]: This baseline uses a greedy algorithm in the MCS system to maximize platform benefits.

B. Performance Comparison of Group Assignment

The same model configuration parameters are followed in different datasets, as shown in Fig. 3. Randomly selecting the same number of workers and tasks showed similar experimental results in different datasets. Here, only the experimental results of the Berlin52 dataset are shown; Ours w/GTA and Ours wo/GTA, respectively, indicate that group assignment was used and group assignment was not used in the task assignment experiment. Fig. 3(a) shows the change in average worker satisfaction as the number of tasks increases when the number of workers remains constant, verifying whether group assignment is adopted. In contrast, Fig. 3(b) shows the scenario when the number of tasks remains constant. The experiment shows that the model with group assignment reaches convergence faster and is more accurate than the comparison model without group

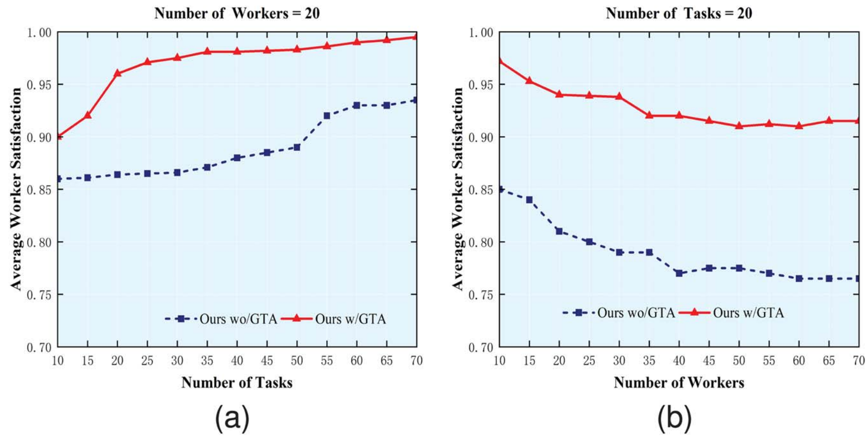

Fig. 3. Comparison of average worker satisfaction using group assignment for the Ours model or not. (a) When the number of workers remains constant. (b) When the number of tasks remains constant.

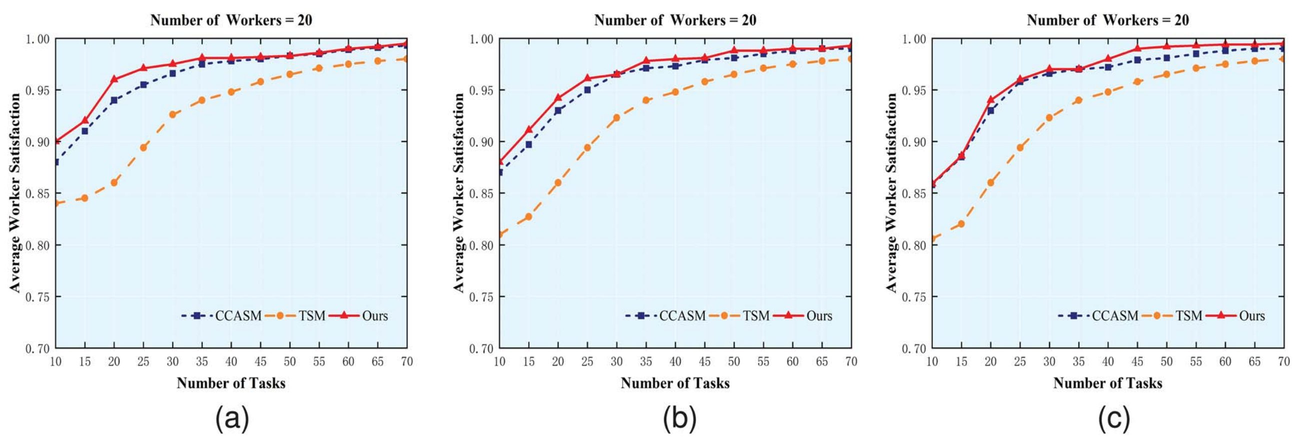

Fig. 4. Comparing the effect of worker growth on average worker satisfaction on the CCASM, traditional stable matching (TSM), and Ours models when the number of tasks is constant: (a) Berlin52 dataset; (b) GeoLife dataset; and (c) NRW1379 dataset.

assignment. It is clear that the comparison model without group assignment is worse in every way. The reason is that GTA can make workers and tasks with similar features, needs, and attributes appear in the same group, and they are obviously attracted to each other. When tasks raise requirements, workers can more quickly choose tasks that suit them; it is expected that users of models that do not perform group assignment operations have a low level of satisfaction. When a task puts forward corresponding requirements, workers need to choose the one that suits them among all tasks. Analysis from different angles is inferior to the performance of GTA. Therefore, the multiattribute E-CARGO model can demonstrate more effective performance than the traditional task assignment model.

C. Performance Analysis

1) Adaptive Group Assignment: As shown in Figs. 4 and 5, respectively, when workers and tasks are fixed, another factor to observe is the impact on average worker satisfaction. It can be seen that this model shows strong advantages compared to the other two models. As shown in Fig. 4, as the number of tasks increases, average worker satisfaction will also start to increase from a certain initial value. When the number of tasks reaches a certain number, it will reach a convergence state. The degree of convergence of this model and CCASM is almost consistent. The reason is that when the number of tasks is smaller than the number of workers, there will be competition between workers, and it is obvious that some workers are unable to complete tasks that satisfy themselves. As the number of tasks increases, this model can better help workers choose tasks that are suitable for them, which will lead to worker satisfaction continuing to increase until convergence. The results in Fig. 5 can be explained for this reason. It is worth noting that when the number of tasks remains constant, as more and more workers join its ranks, the number of workers is greater than the number of tasks. At this time, workers will be more inclined to choose tasks that are more advantageous to them, thereby stabilizing satisfaction. To sum up, this model shows strong advantages in handling multiattribute GTAs in different scenarios.

2) Average Worker Satisfaction: Figs. 6 and 7 show the effect of a change in the number of tasks and workers, respectively, on worker satisfaction and compare it with the other six

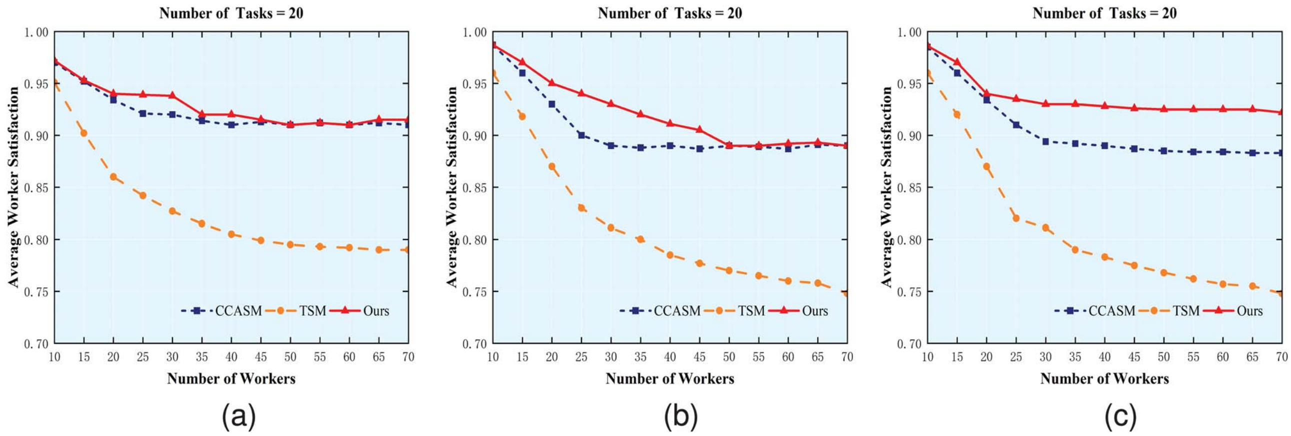

Fig. 5. Comparing the effect of task growth on average worker satisfaction on the CCASM, TSM, and Ours models when the number of workers is constant: (a) Berlin52 dataset; (b) GeoLife dataset; and (c) NRW1379 dataset.

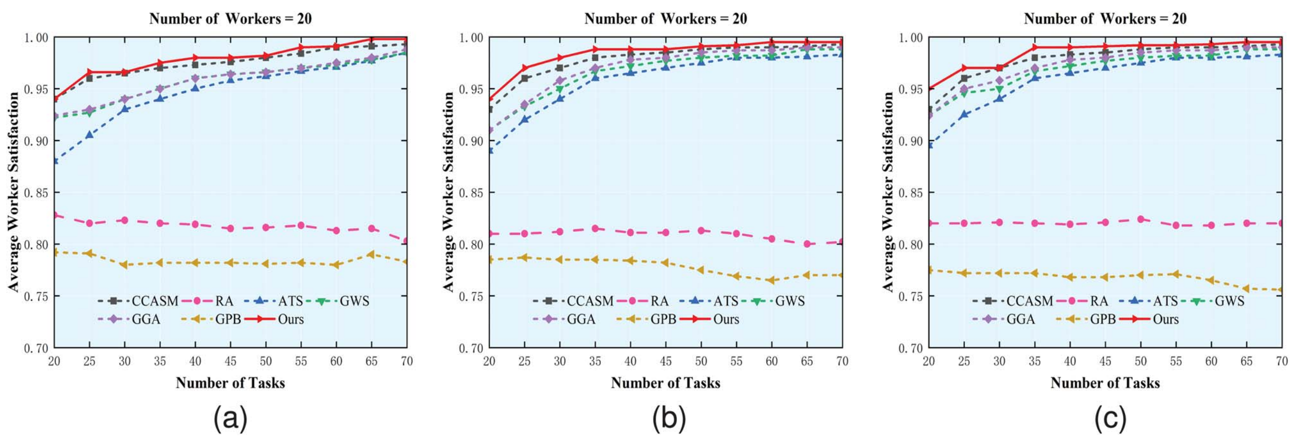

Fig. 6. Comparing the effect of growth in the number of workers on average worker satisfaction on the baseline model when the number of tasks is constant: (a) Berlin52 dataset; (b) GeoLife dataset; and (c) NRW1379 dataset.

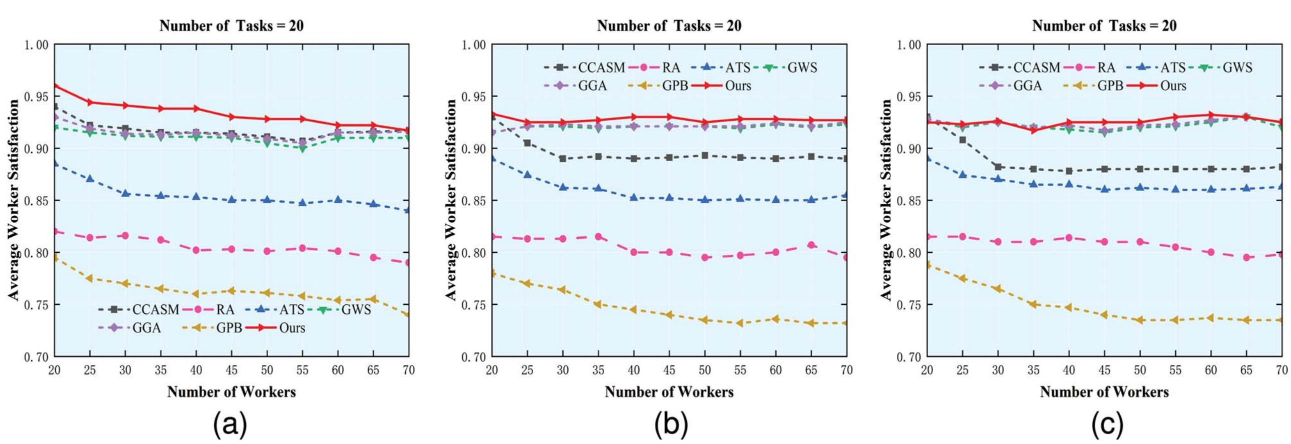

Fig. 7. Comparing the effect of growth in the number of tasks on average worker satisfaction on the baseline model when the number of workers is constant: (a) Berlin52 dataset; (b) GeoLife dataset; and (c) NRW1379 dataset.

baseline models. As can be seen in Fig. 6, the Ours model performs the best, while GPB is the worst, and the reason for this is that GPB focuses only on the maximum benefit that the worker can bring to the group by completing the task and ignores the worker's satisfaction level. The reason for the fluctuating worker satisfaction in RA is due to random assignment, and therefore the assignment of tasks that suit the worker is also random. As the number of tasks increases, worker satisfaction

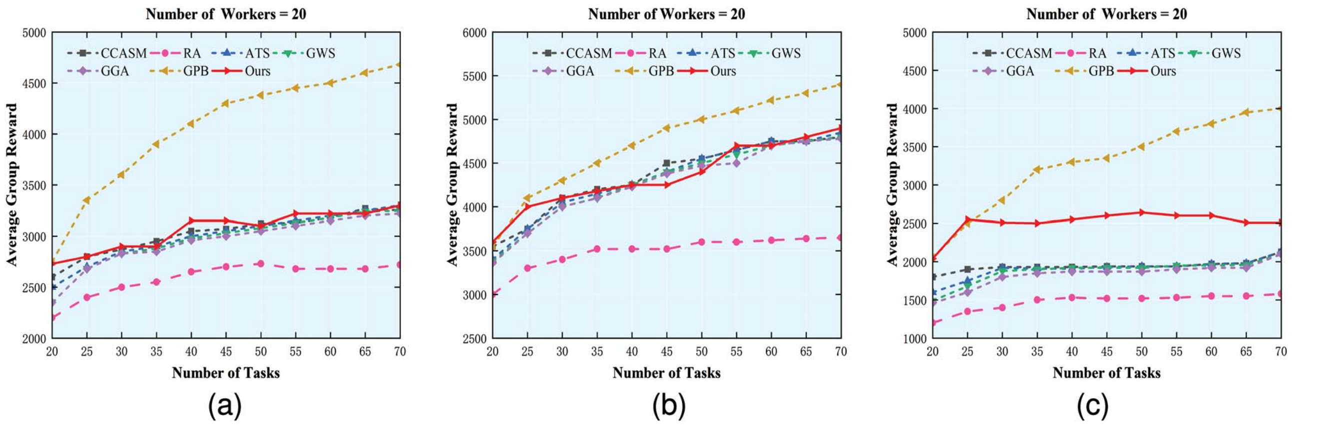

Fig. 8. Compare the effect of growth in the number of workers on average group rewards on the baseline model when the number of tasks is constant: (a) Berlin52 dataset; (b) GeoLife dataset; and (c) NRW1379 dataset.

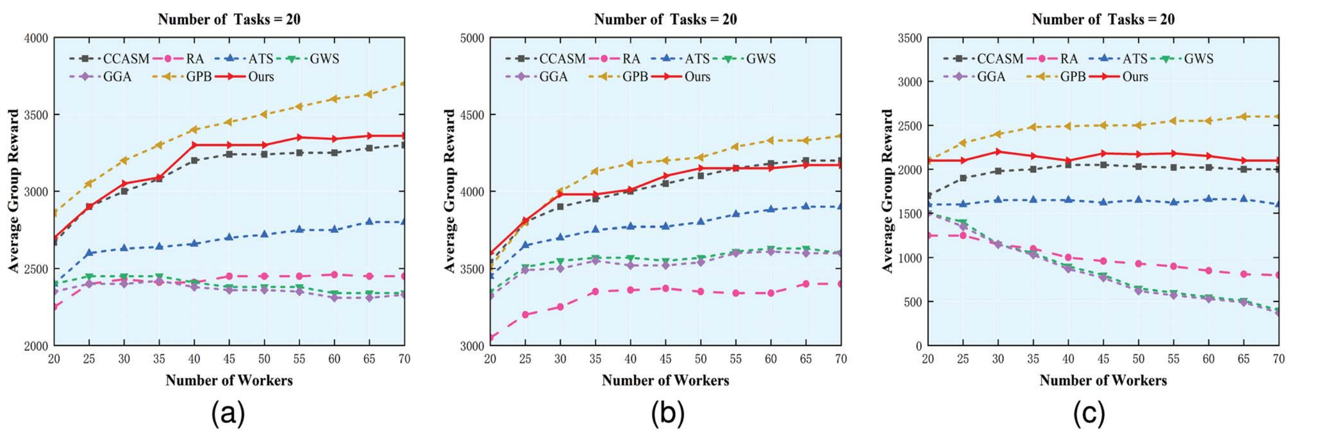

Fig. 9. Compare the effect of growth in the number of tasks on average group rewards on the baseline model when the number of workers is constant: (a) Berlin52 dataset; (b) GeoLife dataset; and (c) NRW1379 dataset.

increases for Ours, ATS, GWS, GGA, and CCASM, reflecting the fact that with an increase in the number of tasks, workers are able to choose tasks that satisfy them more easily. The same is true for Fig. 7.

3) Average Group Reward: In this experiment, group rewards will be used to represent the MCS platform [60] benefits. Figs. 8 and 9, respectively, show the changes in contribution to group rewards when the number of tasks and workers changes, and the Ours model is compared with the other six baseline models. The reason why GPB's benefits are better than other models is that only group rewards are taken into account. The method of group assignment makes workers and task attributes highly similar, and it is easier for workers to choose to satisfy both benefits and satisfaction. Obviously, the Ours model is better than other baseline models.

In summary, the Ours model comprehensively considers worker satisfaction and team interest relationships in task assignment. No matter how the number of tasks and workers changes, workers will get the optimal assignment plan, making this model of significant practical significance.

VI. Discussion

In this section, this research will introduce the problems that have been solved in this work and will continue to study in this direction. In this work, under the conditions of multiattribute workers and tasks, worker satisfaction, group benefits, and task preference of workers in different groups are studied, and a stable assignment scheme is formulated in this article. In addition, this article only focuses on the number of workers and tasks in terms of constraints. At present, the group plans to explore more fine-grained condition division, such as the maximum and minimum execution times of workers and tasks.

VII. Conclusion and Future Work

Existing task assignment schemes are difficult to satisfy both worker satisfaction and group benefits, and existing algorithms suffer from mismatches when dealing with multiattribute assignment schemes. This article considers the multiple attribute requirements that often exist between workers and tasks and the problem of task division in the group. It is the first model ($\text{AHRNet}_{\text{DRL}}$) to combine the E-CARGO model with the graph learning method. This study uses the multiattribute E-CARGO model to complete group assignments between workers and tasks. Classifying workers and tasks with high similarity into the same group can be more conducive to achieving optimal assignment. Second, AHRNet is integrated into DRL, and the deep network learnable method realizes feature selection of global workers and task nodes, improving the overall accuracy of the model. Finally, comparing this model with the traditional baseline model, the excellent performance of this algorithm is proven through simulation experiments.

Our team believes that E-CARGO is a very potential task allocation model, and its application research in the field of graph learning is far from sufficient. Next, the team will continue to explore the algorithm research of E-CARGO in the heterogeneous graph neural network method. It is a very challenging topic to address the existence of abnormal situations in common tasks and how to achieve large-scale task allocation.

References

[1] Y. Luo, Z. Yu, H. Yin, H. Cui, and B. Guo, "Multi-agent mobile crowdsensing by pervasive machines: A robust task allocation approach," *CCF Trans. Pervasive Comput. Interact.*, vol. 5, no. 1, pp. 13–30, 2023.

[2] C. Welty, L. Aroyo, F. Korn, S. M. McCarthy, and S. Zhao, "Rapid instance-level knowledge acquisition for Google maps from class-level common sense," in *Proc. AAAI Conf. Human Comput. Crowdsourc.*, vol. 9, 2021, pp. 143–154.

[3] Z. Li, X. Zhao, Z. Zhao, and T. Braun, "WiFi-RITA positioning: Enhanced crowdsourcing positioning based on massive noisy user traces," *IEEE Trans. Wireless Commun.*, vol. 20, no. 6, pp. 3785–3799, 2021.

[4] A. J. Perez and S. Zeadally, "Secure and privacy-preserving crowdsensing using smart contracts: Issues and solutions," *Comput. Sci. Rev.*, vol. 43, 2022, Art. no. 100450.

[5] W. Wang, Y. Wang, P. Duan, T. Liu, X. Tong, and Z. Cai, "A triple real-time trajectory privacy protection mechanism based on edge computing and blockchain in mobile crowdsourcing," *IEEE Trans. Mobile Comput.*, vol. 22, no. 10, pp. 5625–5642, Oct. 2023.

[6] Q. Zhang, Y. Wang, G. Yin, X. Tong, A. M. V. V. Sai, and Z. Cai, "Two-stage bilateral online priority assignment in spatio-temporal crowdsourcing," *IEEE Trans. Services Comput.*, vol. 16, no. 3, pp. 2267–2282, May/Jun. 2023.

[7] P. William, P. Kumar, G. S. Chhabra, and K. Vengatesan, "Task allocation in distributed agile software development using machine learning approach," in *Proc. Int. Conf. Disruptive Technol. Multi-Discip. Res. Appl. (CENTCON)*, vol. 1, 2021, pp. 168–172.

[8] M.-L. Lee, S. Behdad, X. Liang, and M. Zheng, "Task allocation and planning for product disassembly with human–robot collaboration," *Robot. Comput.-Integr. Manuf.*, vol. 76, 2022, Art. no. 102306.

[9] Y. Wang, Z. Cai, Z.-H. Zhan, B. Zhao, X. Tong, and L. Qi, "Walrasian equilibrium-based multiobjective optimization for task allocation in mobile crowdsourcing," *IEEE Trans. Comput. Social Syst.*, vol. 7, no. 4, pp. 1033–1046, Aug. 2020.

[10] F. Li, Y. Wang, Y. Gao, X. Tong, N. Jiang, and Z. Cai, "Three-party evolutionary game model of stakeholders in mobile crowdsourcing," *IEEE Trans. Comput. Social Syst.*, vol. 9, no. 4, pp. 974–985, Aug. 2022.

[11] W. Yan et al., "Collaborative structure and feature learning for multi-view clustering," *Inf. Fusion*, vol. 98, 2023, Art. no. 101832.

[12] C. Xu and W. Song, "An adaptive data uploading scheme for mobile crowdsensing via deep reinforcement learning with graph neural network," *IEEE Internet Things J.*, vol. 9, no. 18, pp. 18064–18078, Sep. 2022.

[13] J. Liu, J. Wang, Y. Yan, and G. Zhao, "A task allocation method based on data fusion of multimodal trajectory in mobile crowd sensing," *Peer-to-Peer Netw. Appl.*, 2023.

[14] C. Xu and W. Song, "Decentralized task assignment for mobile crowdsensing with multi-agent deep reinforcement learning," *IEEE Internet Things J.*, vol. 10, no. 18, pp. 16564–16578, Sep. 2023.

[15] Z. Lu, Y. Wang, X. Tong, C. Mu, Y. Chen, and Y. Li, "Data-driven many-objective crowd worker selection for mobile crowdsourcing in industrial IoT," *IEEE Trans. Ind. Inform.*, vol. 19, no. 1, pp. 531–540, Jan. 2023.

[16] H. Zhu, M. Zhou, and R. Alkins, "Group role assignment via a Kuhn–Munkres algorithm-based solution," *IEEE Trans. Syst., Man, Cybern. A, Syst. Humans*, vol. 42, no. 3, pp. 739–750, May 2012.

[17] P. Zhang, H. Zhu, and D. Liu, "Equality or equity? E-CARGO perspectives on the fairness of education," *IEEE Trans. Comput. Social Syst.*, 2023, doi: 10.1109/TCSS.2023.3276917.

[18] T. Yang and H. Zhu, "New employee training scheduling using the e-CARGO model," in *Proc. 26th Int. Conf. Comput. Supported Cooperative Work Des. (CSCWD)*, 2023, pp. 691–696.

[19] H. Zhu, "Fundamental issues in the design of a role engine," in *Proc. Int. Symp. Collaborative Technol. Syst.*, Piscataway, NJ, USA: IEEE, 2008, pp. 399–407.

[20] H. Zhu, *E-CARGO and Role-Based Collaboration: Modeling and Solving Problems in the Complex World*. Hoboken, NJ, USA: Wiley, 2021.

[21] Z. Guo, L. Tang, T. Guo, K. Yu, M. Alazab, and A. Shalaginov, "Deep graph neural network-based spammer detection under the perspective of heterogeneous cyberspace," *Future Gener. Comput. Syst.*, vol. 117, pp. 205–218, 2021.

[22] J. Li et al., "Higher-order attribute-enhancing heterogeneous graph neural networks," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 1, pp. 560–574, Jan. 2023.

[23] Y. Wang, Z. Liu, J. Xu, and W. Yan, "Heterogeneous network representation learning approach for Ethereum identity identification," *IEEE Trans. Comput. Social Syst.*, vol. 10, no. 3, pp. 890–899, Jun. 2023.

[24] E. Benhamou, D. Saltiel, J. J. Ohana, J. Atif, and R. Laraki, "Deep reinforcement learning (DRL) for portfolio allocation," in *Proc. Mach. Learn. Knowl. Discovery Databases Appl. Data Sci. Demo Track: Eur. Conf., ECML PKDD, Part V*, Ghent, Belgium, Sep. 14–18, 2020. Springer, 2021, pp. 527–531.

[25] P. Dong, Z.-M. Chen, X.-W. Liao, and W. Yu, "A deep reinforcement learning (DRL) based approach for well-testing interpretation to evaluate reservoir parameters," *Petrol. Sci.*, vol. 19, no. 1, pp. 264–278, 2022.

[26] Y. Peng, G. Tan, H. Si, and J. Li, "DRL-GAT-SA: Deep reinforcement learning for autonomous driving planning based on graph attention networks and simplex architecture," *J. Syst. Archit.*, vol. 126, 2022, Art. no. 102505.

[27] A. Ali, M. A. Qureshi, M. Shiraz, and A. Shamim, "Mobile crowd sensing based dynamic traffic efficiency framework for urban traffic congestion control," *Sustain. Comput., Inform. Syst.*, vol. 32, 2021, Art. no. 100608.

[28] Y. Sun et al., "On enabling mobile crowd sensing for data collection in smart agriculture: A vision," *IEEE Syst. J.*, vol. 16, no. 1, pp. 132–143, Mar. 2022.

[29] C. Zhang, M. Zhao, L. Zhu, T. Wu, and X. Liu, "Enabling efficient and strong privacy-preserving truth discovery in mobile crowdsensing," *IEEE Trans. Inf. Forensics Secur.*, vol. 17, pp. 3569–3581, 2022.

[30] F. Abbondati, S. A. Biancardo, R. Veropalumbo, and G. Dell'Acqua, "Surface monitoring of road pavements using mobile crowdsensing technology," *Measurement*, vol. 171, 2021, Art. no. 108763.

[31] X. Tao and W. Song, "Efficient task allocation for mobile crowd sensing based on evolutionary computing," in *Proc. IEEE Int. Conf. Internet Things (iThings); IEEE Green Comput. Commun. (GreenCom); IEEE Cyber, Phys. Social Comput. (CPSCom); IEEE Smart Data (SmartData)*, Piscataway, NJ, USA: IEEE, 2018, pp. 374–380.

[32] G. Yang, B. Wang, X. He, J. Wang, and H. Pervaiz, "Competition-congestion-aware stable worker-task matching in mobile crowd sensing," *IEEE Trans. Netw. Service Manage.*, vol. 18, no. 3, pp. 3719–3732, Sep. 2021.

[33] H. Zhao, M. Xiao, J. Wu, Y. Xu, H. Huang, and S. Zhang, "Differentially private unknown worker recruitment for mobile crowdsensing using multi-armed bandits," *IEEE Trans. Mobile Comput.*, vol. 20, no. 9, pp. 2779–2794, Sep. 2021.

[34] H. Ma, J. Li, Y. Tang, H. Zhu, Z. Huang, and W. Tang, "Universal optimization framework: Leader-centered learning team formation based on fuzzy evaluations of learners and E-CARGO," *IEEE Syst., Man, Cybern. Mag.*, vol. 9, no. 2, pp. 6–17, Apr. 2023.

[35] C. Peng, H. Zhu, L. Liu, and R. Grewal, "Optimal data allocation in the environment of edge and cloud servers," in *Proc. IEEE Int. Conf. Netw., Sens. Control (ICNSC)*, 2022, pp. 1–6.

[36] B. Akbari and H. Zhu, "Fault-resilience role engine for an autonomous cooperative multi-robot system using E-CARGO," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. (SMC)*, 2022, pp. 730–735.

[37] Y. Sheng, H. Zhu, X. Zhou, and W. Hu, "Effective approaches to adaptive collaboration via dynamic role assignment," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 46, no. 1, pp. 76–92, Jan. 2016.

[38] H. Zhu and R. Alkins, "Group role assignment," in *Proc. Int. Symp. Collaborative Technol. Syst.*, Piscataway, NJ, USA: IEEE, 2009, pp. 431–439.

[39] D. Liu, Y. Yuan, H. Zhu, S. Teng, and C. Huang, "Balance preferences with performance in group role assignment," *IEEE Trans. Cybern.*, vol. 48, no. 6, pp. 1800–1813, Jun. 2018.

[40] Q. Jiang, D. Liu, H. Zhu, Y. Qiao, and B. Huang, "Quasi group role assignment with role awareness in self-service spatiotemporal crowdsourcing," *IEEE Trans. Comput. Social Syst.*, vol. 9, no. 5, pp. 1456–1468, Oct. 2022.

[41] D. Liu, B. Huang, and H. Zhu, "Solving the tree-structured task allocation problem via group multirole assignment," *IEEE Trans. Automat. Sci. Eng.*, vol. 17, no. 1, pp. 41–55, Jan. 2020.

[42] L. Liang, J. Fu, H. Zhu, and D. Liu, "Solving the team allocation problem in crowdsourcing via group multirole assignment," *IEEE Trans. Comput. Social Syst.*, vol. 10, no. 3, pp. 843–854, Jun. 2023.

[43] W. Liu, L. Wang, E. Wang, Y. Yang, D. Zeghlache, and D. Zhang, "Reinforcement learning-based cell selection in sparse mobile crowdsensing," *Comput. Netw.*, vol. 161, pp. 102–114, 2019.

[44] L. Chen et al., "Decision transformer: Reinforcement learning via sequence modeling," in *Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 15084–15097.

[45] H. Zhu, "Group role assignment with constraints (GRA+): A new category of assignment problems," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 53, no. 3, pp. 1670–1683, Mar. 2023.

[46] L. Zhang, Z. Yu, S. Wu, H. Zhu, and Y. Sheng, "Adaptive collaboration with training plan considering role correlation," *IEEE Trans. Comput. Social Syst.*, 2022, doi: 10.1109/TCSS.2022.3204052.

[47] Q. Jiang, D. Liu, H. Zhu, Y. Qiao, and B. Huang, "Equilibrium means equity? An E-CARGO perspective on the golden mean principle," *IEEE Trans. Comput. Social Syst.*, vol. 10, no. 4, pp. 1443–1454, Aug. 2023.

[48] P. Velickovic et al., "Graph attention networks," *Stat*, vol. 1050, no. 20, pp. 10–48550, 2017.

[49] M. Réau, N. Renaud, L. C. Xue, and A. M. Bonvin, "DeepRank-GNN: A graph neural network framework to learn patterns in protein–protein interfaces," *Bioinformatics*, vol. 39, no. 1, 2023, Art. no. btac759.

[50] Y.-C. Lin, B. Zhang, and V. Prasanna, "HP-GNN: Generating high throughput GNN training implementation on CPU-FPGA heterogeneous platform," in *Proc. ACM/SIGDA Int. Symp. Field-Programmable Gate Arrays*, 2022, pp. 123–133.

[51] L. Yang, Z. Liu, Y. Dou, J. Ma, and P. S. Yu, "ConsisRec: Enhancing GNN for social recommendation via consistent neighbor aggregation," in *Proc. 44th Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2021, pp. 2141–2145.

[52] Z. Liu, D. Yang, Y. Wang, M. Lu, and R. Li, "EGNN: Graph structure learning based on evolutionary computation helps more in graph neural networks," *Appl. Soft Comput.*, vol. 135, 2023, Art. no. 110040.

[53] Z. Liu, D. Yang, S. Wang, and H. Su, "Adaptive multi-channel Bayesian graph attention network for IoT transaction security," *Digit. Commun. Netw.*, early access, Dec. 16, 2022, doi: 10.1016/j.dcan.2022.11.018.

[54] R. Li, Z. Liu, Y. Ma, D. Yang, and S. Sun, "Internet financial fraud detection based on graph learning," *IEEE Trans. Comput. Social Syst.*, vol. 10, no. 3, pp. 1394–1401, Jun. 2023.

[55] D. Zeng, W. Liu, W. Chen, L. Zhou, M. Zhang, and H. Qu, "Substructure aware graph neural networks," in *Proc. AAAI Conf. Artif. Intell.*, vol. 37, no. 9, 2023, pp. 11129–11137.

[56] W. Fan et al., "Graph neural networks for social recommendation," in *Proc. World Wide Web Conf. (WWW)*, 2019, p. 417–426.

[57] G. Yang, B. Wang, X. He, J. Wang, and H. Pervaiz, "Competition-congestion-aware stable worker-task matching in mobile crowd sensing," *IEEE Trans. Netw. Service Manag.*, vol. 18, no. 3, pp. 3719–3732, 2021.

[58] M. H. Cheung, F. Hou, J. Huang, and R. Southwell, "Distributed time-sensitive task selection in mobile crowdsensing," *IEEE Trans. Mobile Comput.*, vol. 20, no. 6, pp. 2172–2185, Jun. 2021.

[59] B. Guo, Y. Liu, W. Wu, Z. Yu, and Q. Han, "ActiveCrowd: A framework for optimized multitask allocation in mobile crowdsensing systems," *IEEE Trans. Human-Mach. Syst.*, vol. 47, no. 3, pp. 392–403, Jun. 2017.

[60] Y. Wang, Y. Gao, Y. Li, and X. Tong, "A worker-selection incentive mechanism for optimizing platform-centric mobile crowdsourcing systems," *Comput. Netw.*, vol. 171, 2020, Art. no. 107144.

Zhaowei Liu (Member, IEEE) received the Ph.D. degree in automation from Shandong University, Jinan, China, in 2018.

He is currently a Professor with Yantai University, Yantai, China. His research direction is artificial intelligence. His main research interests are distributed artificial intelligence, machine learning.

Dr. Liu is a member of the China Computer Federation (CCF).

Zongxing Zhao (Graduate Student Member, IEEE) received the B.Sc. degree in software engineering and currently working toward the M.Sc. degree in electronic information with the School of Computer and Control Engineering, Yantai University, Yantai, China.

His current research interests include heterogeneous graph representation learning and task allocation.