Quasi Group Role Assignment With Role Awareness in Self-Service Spatiotemporal Crowdsourcing

Qian Jiang, *Graduate Student Member, IEEE*, Dongning Liu, *Member, IEEE*, Haibin Zhu, *Senior Member, IEEE*, Yan Qiao, *Senior Member, IEEE*, and Baoying Huang, *Graduate Student Member, IEEE*

***Abstract*— Self-service spatiotemporal crowdsourcing (SSC), a booming variant of spatiotemporal crowdsourcing (SC), emerges because of the vigorous development of the mobile Internet. Unlike the conventional SCs, the particularity of self-service in SSC may lead to unfinished tasks at the end of the entire assignment process, making a one-time assignment scheme ineffective. SSC is essentially an adaptive collaboration (AC) problem that requires a dynamic assignment strategy for a higher task completion rate. This article tackles this issue by establishing a quasi group role assignment (QGRA) based on a typical SSC scenario, that is, the photographing to make money problem (PMMP). First, it sheds light on a novel role awareness method, which can effectively divide tasks to accelerate the solution while, to some extent, raising the task completion rate. Second, it specifies an agent satisfaction evaluation (ASE) method to quantify the relationship between task completion rate and workers' satisfaction. This method aims at considerably ameliorating task completion rate. Last, it extends QGRA with a new AC algorithm, which can achieve AC of the workers while accomplishing the crowdsourcing task. Moreover, utilizing the ASE method can help decision-makers balance the task completion rate and the workers' satisfaction. Large-scale simulation experiments based on the real crowdsourced datasets exemplify the robustness and practicability of the proposed solutions. This article contributes a new version of the group role assignment (GRA) model, that is, quasi GRA (QGRA), a creative formalization to solve the AC problem.**

***Index Terms*— Adaptive collaboration (AC) problem, group role assignment (GRA), quasi GRA (QGRA), role awareness, role-based collaboration (RBC), self-service spatiotemporal crowdsourcing (SSC).**

Manuscript received 5 October 2021; revised 5 December 2021; accepted 17 December 2021. Date of publication 5 January 2022; date of current version 30 September 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 62072120, in part by the Research and Development Program of Guangdong Province under Grant 2020B010166006, and in part by the Natural Sciences and Engineering Research Council of Canada (NSERC) under Grant RGPIN2018-04818. *(Corresponding author: Baoying Huang.)*

Qian Jiang, Yan Qiao, and Baoying Huang are with the Institute of Systems Engineering and the Collaborative Laboratory for Intelligent Science and Systems, Macau University of Science and Technology, Taipa, Macau, China (e-mail: jiangqian@mail2.gdut.edu.cn; yqiao@must.edu.mo; baoyhuang@126.com).

Dongning Liu is with the School of Computer Science and Technology, Guangdong University of Technology, Guangzhou, Guangdong 510006, China (e-mail: liudn@gdut.edu.cn).

Haibin Zhu is with the Collaborative Systems Laboratory (CoSys Laboratory), Nipissing University, North Bay, ON P1B 8L7, Canada (e-mail: haibinz@nipissingu.ca).

Digital Object Identifier 10.1109/TCSS.2021.3137188

Nomenclature

t_{start}	Start time of the assignment scheme.
t_{end}	End time of the assignment scheme.
t_{interval}	Time interval for re-assignment.
$w_{\max}$	Maximum number of tasks a worker can accept at a time.
$\mathcal{A}$	A set of agents.
$\mathcal{R}$	A set of roles.
m	Size of the agent set $\mathcal{A}$.
n	Size of the role set $\mathcal{R}$.
$i_0, i_1, i_2, \ldots$	Indices of agents.
$j_0, j_1, j_2, \ldots$	Indices of roles.
r	A role.
a	An agent.
L	Role range vector.
L^a	Ability limit vector.
Q	Qualification matrix.
T	Role assignment matrix.
σ	Group performance.
$\bar{Q}$	Capability matrix.
$\mathcal{T}$	Workable role vector.
$\mathcal{T}^a$	Workable agent vector.
n_w	Number of workable assignment unit in a group.
$\Diamond$	Role profit vector.
Λ^a	Agent income level vector.
Ξ	Agent loss profit matrix.
τ^a	Agent satisfaction threshold vector.

I. Introduction

SELF-SERVICE spatiotemporal crowdsourcing (SSC) [1], [2] is a new form of spatiotemporal crowdsourcing (SC), and it has developed rapidly due to the popularity of the mobile Internet, for example, the photographing to make money problem (PMMP) [1] is typical in the SC scenario. Similar to traditional SC [3], numerous tasks in SSC are independent. However, the characteristic of self-service complicates this problem and brings up two main challenges. First, some tasks in SSC may not be completed by the assigned workers due to the low profits. Second, to increase the task completion rate properly, the decision-makers require an adaptive assignment strategy to redistribute the crowdsourcing tasks frequently. That is to say, SSC is essentially an adaptive collaboration (AC) problem [4]–[6] with an uncertain task completion rate. To this end, we tackle a typical SSC paradigm, PMMP, with a new perspective on role-based collaboration (RBC) [7].

RBC [8]–[10] is an efficient computational methodology for constructing collaboration systems. Its fundamental model Environments—Classes, Agents, Roles, Groups, and Objects (E-CARGO) [11]–[13] and its submodel group role assignment (GRA) [14], [15]/group multirole assignment (GMRA) [16] have been a well-specific way to solve SC problems [4], [17], especially the static ones [17]. But the GRA/GMRA model cannot directly solve the two non-trivial challenges in SSC mentioned above. For these concerns, this article revises the GRA model with potential rejections of assignment, that is, quasi GRA (QGRA). It first designs a specific role awareness method [17], [18], which utilizes task aggregation density as a metric to divide crowdsourcing tasks properly. This novel method can help formalize the SSC by the GRA model while accelerating solutions. Then, this article proposes an innovative agent satisfaction evaluation (ASE) method to clarify the relationship between task completion rate and workers' satisfaction for a high task completion rate. The ASE method also helps decision-makers make their efforts to achieve an equilibrium between crowdsourcers and workers. Lastly, it puts forward an AC algorithm to realize the adaptive assignment for those unfinished tasks.

The contributions of this article are listed as follows.

1) This article revises the GRA model for solving the AC problem, that is, QGRA.
2) To achieve AC, this article proposes a role awareness method and ASE method synergistically.
3) Lastly, our proposed ASE method also assists decision-makers in tuning task completion rate and workers' satisfaction.

The remainder of this article is arranged as follows. It first introduces a real-world scenario of SSC in Section II. Then in Section III, it formally defines the problem, that is, QGRA, via the E-CARGO and GRA/GMRA submodel. Section IV delves into a series of specific methods proposed to extend the QGRA model. In Section V, large-scale simulation experiments are carried out. The related research and efforts are described in Section VI. This article concludes and points out future work in Section VII.

II. Real-World Scenario

The PMMP [1] is a typical SSC paradigm. It aims to provide business inspection and information collection for enterprises through taking photographs actively uploaded by the crowdsourcing workers. Many companies related to the PMMP have sprung up in recent years, for example, Foap [19] and Meituan Paidian [20].

Company X, which is committed to providing optimized assignment schemes for companies, recently won a bid for a PMMP project from company Y. This project is to promote the low task completion rate in the PMMP, which is predominantly caused by the dissatisfactions of crowdsourcing workers about the pricing and spatiotemporal constraints. Ann, the chief executive officer (CEO) in company X, asks Bob, the chief technology officer (CTO) in company X, to accomplish this tough project. Based on the open dataset [1] provided by company Y, Bob obtains the distribution of the crowdsourcing tasks (1863 in total) and workers (853 in total), as elaborated in Fig. 1, which also illustrates that these crowdsourcing tasks and workers are distributed in Guangdong Province, China. Taking into account the expenditure of crowdsourcers, all the new gathering crowdsourcing tasks are updated once a day.

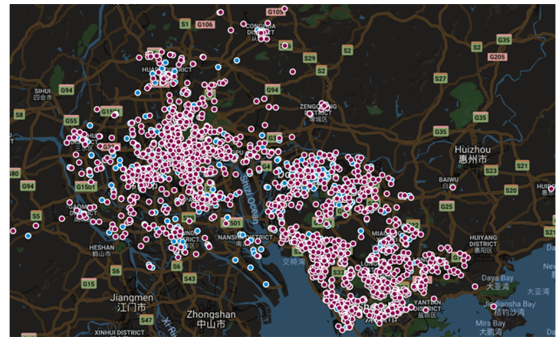


Fig. 1. Distribution of the crowdsourcing workers and tasks[a] [1]. [a]Note: The red circles represent the locations of the workers, and the blue ones represent the locations of the tasks. We will conduct large-scale simulation experiments in Section V with randomly generated crowdsourcing tasks. The datasets in [1] are accessible in Appendix A.

TABLE I
Vital Parameters in PMMP

Parameter	Meanings	Value
t_{start}	The start time of the assignment scheme.	10:00 AM
t_{end}	The end time of the assignment scheme.	18:00 PM
$t_{interval}$	The time interval for re-assignment.	0.5 hours
w_{max}	The maximum number of tasks a worker can accept at a time.	5

At the same time, Bob empirically listed a few vital parameters (see Table I) and their values in the PMMP through company Y's need. Bob also assessed the cost of all workers completing tasks based on distances, that is, Table II.

According to Bob's analysis, the PMMP is essentially a one-to-many (1-M) assignment problem with spatiotemporal constraints. That is, one crowdsourcing worker can accept multiple tasks at the same time, and one task can only be accepted by one worker at a certain time interval. Correspondingly, Bob learns that the E-CARGO model and its submodel GMRA have become an effective strategy to solve the M-M problem [21] with tough spatiotemporal constraints. Hence, he decides to use it (Definition 6) to solve the PMMP and report the assignment results to Ann.

However, Ann is unsatisfied with Bob's assignment scheme because of the following reasons. First, GMRA ignores workers' satisfaction and assumes that all the assigned tasks are accepted. Potential rejections may make the assignment results impractical. For example, GMRA assigns Task 199 to Worker 81 but the profit (see Table III) obtained by Worker 81 is −1.07 yuan (Chinese dollar), which may be unacceptable. As for SSC, due to the particularity of self-service, the workers may not perform assigned tasks due to the low profits. Second, the methodology for solving the PMMP may need to perform a re-assignment process for those unfinished

TABLE II
EVALUATION OF THE WORKERS ON TASKS[A]

Worker	Task						
	Task 1	*Task* 2	*Task* 3	…	*Task* 851	*Task* 851	*Task* 853
Worker 1	52.40	39.43	50.06	…	21.12	15.97	61.49
Worker 2	1.96	12.35	0.97	…	29.94	57.62	106.65
Worker 3	95.20	82.82	92.71	…	64.68	45.65	25.41
…	…	…	…	…	…	…	…
Worker 1861	110.06	97.23	107.65	…	78.89	56.59	30.47
Worker 1862	14.23	5.56	13.55	…	21.74	47.07	102.36
Worker 1863	98.70	87.30	96.10	…	69.83	54.89	13.29

*Note: Because the number of data items is too large to present, only the first and last three workers and the first and last three tasks are shown. The evaluation method will be depicted in Section III, and the unit of values in Table II is kilometers.

TABLE III
PROFIT OF THE WORKERS ON TASKS[A]

Worker	Task						
	Task 1	*Task* 2	*Task* 3	…	*Task* 851	*Task* 851	*Task* 853
Worker 1	-11.40	39.43	50.06	…	-4.41	-3.50	-13.38
Worker 2	0.14	-2.35	0.30	…	-6.38	-12.80	-23.46
Worker 3	-20.95	-18.09	-20.40	…	-14.14	-10.13	-5.32
…	…	…	…	…	…	…	…
Worker 1861	-24.27	-21.31	-23.73	…	-17.31	-12.57	-6.45
Worker 1862	-2.87	-0.82	-2.72	…	-4.55	-10.45	-22.50
Worker 1863	-21.73	-19.09	-21.16	…	-15.29	-12.19	-2.62

*Note: The profit evaluation method is depicted in Section IV, and the unit of values in Table III is yuan.

crowdsourcing tasks, while the GRA/GMRA model lacks an adaptive mechanism.

The aforementioned non-trivial problems make it difficult for Bob to complete the scheduling strategy before the deadline. Fortunately, we can solve this kind of problem by extending the GRA model with the appropriate adaptive property. The following of this article depicts the details of our solutions, which can help Bob deliver his project on time.

III. E-CARGO MODEL AND ITS REVISED GRA/GMRA SUBMODEL

For solving the SSC problem, we first formalize it with the E-CARGO model and its revised GRA/GMRA model. Regarding the E-CARGO model [22], [23], a system $\sum$ can be described as a nine-tuple $\sum ::= \langle \mathcal{C}, \mathcal{O}, \mathcal{A}, \mathcal{M}, \mathcal{R}, \mathcal{E}, \mathcal{G}, s_0, \mathcal{H}\rangle$, where $\mathcal{C}, \mathcal{O}, \mathcal{A}, \mathcal{M}, \mathcal{R}, \mathcal{E}, \mathcal{G}$, and $\mathcal{H}$ are E-CARGO components' sets, that is, environments, classes, agents, roles, groups, objects, messages, and human users. The initial state of $\sum$ is denoted by s_0.

When solving the role assignment problems, environments (es) and groups (gs) are simplified into vectors and matrices, respectively. Besides, we use nonnegative integers $m(= |\mathcal{A}|$, where $|\mathcal{A}|$ is the cardinality of set $\mathcal{A})$ to express the size of the agent set $\mathcal{A}$, $n(= |\mathcal{R}|)$ the size of the role set $\mathcal{R}$, i, i_1, $i_2, \ldots$ the indices of agents, and j, j_1, $j_2, \ldots$ the indices of roles.

Here, we utilize the real-world scenario mentioned in Section II, that is, the PMMP, as an example to introduce RBC and its submodel GRA/GMRA. The following brings up the specific definitions of the SSC.

Definition 1: A role [7], [9], [10] is defined as $r ::= \langle id, ®\rangle$, where id is the identification of r, that is, role j $(0 \le j < n)$, and ® represents the set of requirements or properties for agents to act r.

Note: It is easy to take the crowdsourcing tasks in the PMMP as roles because all tasks are simple and independent of each other. However, with the time requirement and completion rate in mind, here we design the role awareness method in Section IV to cluster tasks to form roles. Such an operation can convert a 1-M problem to 1-1 problem, which can accelerate the solution and encourage workers to complete the tasks through higher profits. ® in the PMMP has lower requirements since the crowdsourcing tasks are simple and unskilled.

Definition 2: An agent [24], [25] is defined as $a ::= \langle id, Ⓠ\rangle$, where id is the identification of a, that is, agent i $(0 \le i < m)$; Ⓠ is the set of a's values corresponding to the abilities required in the group.

Note: Agents refer to the crowdsourcing workers in the scenario of Section II, and Ⓠ expresses the agents' performances about ®. The specific definition of Ⓠ is illustrated in Definition 5.

Definition 3: A role range vector [11] L is a vector of the number of agents required for roles in the environment e of group g.

Note: L is a valuable component of the E-CARGO model. It indicates the minimum number of workers required for each crowdsourcing task cluster (role). In the PMMP, there exist equations $\sum_{j=0}^{n-1} L[j] = m$ $(0 \le j < n)$ and $L[j] = 1$.

Definition 4: An ability limit vector [11] L^a is an m-vector, where $L^a[i]$ $(0 \le i < m)$ indicates the number of roles that can be assigned to agent i. The superscript of L^a indicates that L^a is a definition for the agents.

Note: When formalizing the PMMP via the GMRA model, for each agent i, it satisfies $L^a[i] = w_{\max}$ since the roles are the crowdsourcing tasks. Whereas, after using the proposed role awareness method, the roles in the QGRA/GRA model are task clusters. Thus, in this situation, agent i satisfies the equation $L^a[i] = 1$.

Definition 5: A qualification matrix [9] Q is an $m \times n$ matrix, where $Q[i, j] \in [0, 1]$ expresses the qualification value of agent i $(0 \le i < m)$ for role j $(0 \le j < n)$, that is, Ⓠ in Definition 2. $Q[i, j] = 0$ indicates the lowest value and 1 the highest.

Note: In the PMMP, since each task in roles is simple and unskilled, the most important criterion for evaluating crowdsourcing workers (agents) is the central distance (unit in kilometers) between an agent and a role, for example, Euclidean distance [18]. In agent evaluation, for each role, we use the location of the farthest task from a specific agent as the position of this role. For example, if there exists role $j = \{\text{Task 1, Task 2, Task 3}\}$ $(0 \le j < n)$ and Task 2 is the farthest task to agent i $(0 \le i < m)$, then the location of

$$\begin{pmatrix} 0.29 & 0.29 & 0.27 & 0.33 & \dots & 0.09 & 0.41 & 0.31 & 0.31 \\ 0.03 & 0.05 & 0.04 & 0.16 & \dots & 0.20 & 0.63 & 0.54 & 0.51 \\ 0.51 & 0.51 & 0.49 & 0.55 & \dots & 0.31 & 0.22 & 0.13 & 0.18 \\ 0.54 & 0.55 & 0.53 & 0.58 & \dots & 0.34 & 0.22 & 0.13 & 0.20 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0.48 & 0.48 & 0.46 & 0.55 & \dots & 0.31 & 0.19 & 0.09 & 0.12 \\ 0.58 & 0.59 & 0.57 & 0.62 & \dots & 0.38 & 0.23 & 0.16 & 0.24 \\ 0.09 & 0.12 & 0.10 & 0.13 & \dots & 0.14 & 0.61 & 0.52 & 0.50 \\ 0.52 & 0.52 & 0.50 & 0.59 & \dots & 0.34 & 0.16 & 0.06 & 0.12 \end{pmatrix}$$

Fig. 2. Normalized Q matrix in our scenario. Note: For display purposes, only the first four and last four rows and the first four and last four columns are shown.

role j is the same as Task 2. Additionally, we calculate $Q[i, j]$ through the haversine formula [26] because the position information obtained by each role and agent is the sequence regarding $\langle$longitude, latitude$\rangle$ (in radian)

$$Q[i, j] = r_{\text{Earth}} \times 2\arcsin\sqrt{\sin^2\frac{y_i - y_j}{2} + \cos(y_i) \times \cos(y_j) \times \sin^2\frac{x_i - x_j}{2}} \quad (1)$$

where r_{Earth} represents the radius of the Earth, $\langle x_i, y_i\rangle$ is the location of agent i, and $\langle x_j, y_j\rangle$ is the location of role j. Lastly, we normalize the qualification matrix Q by the max-min normalization method for better calculation. Fig. 2 illustrates the normalized Q matrix in the scenario in Section II.

Definition 6: A role assignment matrix [9] T is defined as an $m \times n$ matrix, where $T[i, j] \in \{0, 1\}$ $(0 \le i < m, 0 \le j < n)$ indicates whether or not agent i is assigned to role j. $T[i, j] = 1$ means yes and 0 no.

Definition 7: The group performance σ of group g is defined as the sum of the assigned agents' qualifications, that is,

$$\sigma = \sum_{i=0}^{m-1}\sum_{j=0}^{n-1} Q[i, j] \times T[i, j].$$

The social meaning of σ is the minimum total cost of the crowdsourcing workers to finish their tasks during the whole scheduling process.

Definition 8: Role j is workable in group g if it has been assigned enough agents, that is,

$$\sum_{i=0}^{m-1} T[i, j] \ge L[j].$$

Definition 9: T is workable if each role j is workable, that is, $\sum_{i=0}^{m-1} T[i, j] = L[j] (0 \le j < n)$. Group g is workable if T is workable.

Definition 10: Given $\mathcal{A}$ $(|\mathcal{A}| = m)$, $\mathcal{R}$ $(|\mathcal{R}| = n)$, Q, and L, the revised GRA [9] problem is to find a workable T to obtain

$$\min \sigma^{\text{GRA}} = \sum_{i=0}^{m-1}\sum_{j=0}^{n-1} Q[i, j] \times T[i, j] \quad (2)$$

$$\text{s.t. } T[i, j] \in \{0, 1\}, \quad (0 \le i < m, 0 \le j < n) \quad (3)$$

$$\sum_{i=0}^{m-1} T[i,j] = L[j], \quad (0 \le j < n) \quad (4)$$

$$\sum_{j=0}^{n-1} T[i, j] \le 1, \quad (0 \le i < m) \quad (5)$$

where expressions (3)–(5) are the constraints for the control variables $T[i, j]$, and expression (4) is to ensure that each role can be executed by an agent.

Definition 11: Given $\mathcal{A}$ $(|\mathcal{A}| = m)$, $\mathcal{R}$ $(|\mathcal{R}| = n)$, Q, L, and L^a, the revised GMRA [11] problem is to find a workable T to obtain

$$\min \sigma^{\text{GMRA}} = \sum_{i=0}^{m-1}\sum_{j=0}^{n-1} Q[i,j] \times T[i, j] \quad (6)$$

$$\text{s.t. (3), (4), and } \sum_{j=0}^{n-1} T[i,j] \le L^a[i], \quad (0 \le i < m) \quad (7)$$

where expression (7) is to ensure that each agent can accept a limited number of roles at a time.

In the following discussions, we call revised GRA/GMRA GRA/GMRA if we do not expressly state it. The social meaning of GRA/GMRA is to obtain the ideal optimal assignment scheme of the SSC problem and assume that each agent enforces its assigned tasks. Note that the roles in the GMRA model are the crowdsourcing tasks, but in the QGRA model, roles are the task clusters estimated by the novel role awareness method proposed in Section IV.

Certainly, the revised GRA/GMRA model cannot solve the SSC since there exist some roles that may not be played by any agent during the whole scheduling process. Besides, GRA/GMRA also ignores agents' satisfaction (called preference in [27]) for their assigned tasks. Accordingly, we propose the QGRA model, and here we need to introduce the following definitions for better depicting this model.

Definition 11: A capability matrix $\bar{Q}$ [8] is an $m \times n$ matrix, and $\bar{Q}[i, j] \in \{0, 1\}$ $(0 \le i < m, 0 \le j < n)$. $\bar{Q}[i, j] = 1$ indicates that agent i is willing to play role j and 0 means no.

Note: The value of $\bar{Q}[i, j]$ is related to the agents' satisfactions, and the ASE method will be described in detail in Section IV, that is, expression (13).

Definition 12: A workable role vector $\mathcal{T}$ is an n-vector, where $\mathcal{T}[j] \in \{0, 1\}$ $(0 \le j < n)$, and it satisfies

$$\mathcal{T}[j] = \begin{cases} 0, & \sum_{i=0}^{m-1} \bar{Q}[i, j] = 0 \\ 1, & \sum_{i=0}^{m-1} \bar{Q}[i, j] > 0. \end{cases}$$

Note: $\mathcal{T}[j] = 1$ represents that role j will have at least one candidate agent to play it and 0 means no.

Definition 13: A workable agent vector $\mathcal{T}^a$ is an m-vector, where $\mathcal{T}^a[i] \in \{0, 1\}$ $(0 \le i < m)$. The superscript of $\mathcal{T}^a$ indicates that it is a definition for the agents. Similar to vector $\mathcal{T}$, for each agent i in the PMMP, it fulfills

$$\mathcal{T}^a[i] = \begin{cases} 0, & \sum_{j=0}^{n-1} \bar{Q}[i, j] = 0 \\ 1, & \sum_{j=0}^{n-1} \bar{Q}[i, j] > 0. \end{cases}$$

Note: $\mathcal{T}^a[i] = 1$ indicates that agent i will play one of its candidate role(s) and 0 means no.

Definition 14: The number of workable assignment unit in a group is defined as $n_w = \min\left\{\sum_{j=0}^{n-1} \mathcal{T}[j], \sum_{i=0}^{m-1} \mathcal{T}^a[i]\right\}(0 \le i < m, 0 \le j < n)$.

Note: n_w is to find the number of assignment unit ($\langle$agent i, role $j\rangle$), that is, $\min\left\{\sum_{j=0}^{n-1} \mathcal{T}[j], \sum_{i=0}^{m-1} \mathcal{T}^a[i]\right\}$.

Definition 15: Given $\mathcal{A}$ ($|\mathcal{A}| = m$), $\mathcal{R}$ ($|\mathcal{R}| = n$), Q, and L, the QGRA problem is to find a workable T to obtain

$$\min \sigma^{\text{QGRA}} = \sum_{i=0}^{m-1}\sum_{j=0}^{n-1} Q[i,j] \times T[i,j] \tag{8}$$

$$\text{s.t. (3), (5), and } \sum_{i=0}^{m-1} T[i,j] \le L[j], \quad (0 \le j < n) \tag{9}$$

$$\sum_{i=0}^{m-1}\sum_{j=0}^{n-1} T[i,j] = n_w, \quad (0 \le i < m, 0 \le j < n) \tag{10}$$

where expression (9) ensures that unworkable roles exist during the scheduling process, and (10) indicates the number of workable assignment units.

The above-mentioned QGRA model has some problems. First, since the roles in the SSC are task clusters, a proper method is required to divide tasks for forming roles. Second, the QGRA model requires an ASE method to help quantify n_w. Last, an adaptive mechanism is required to dynamically reassign that unfinished task to achieve AC. Consequently, the above method needs to be adequately promoted for solving SSC.

IV. QGRA Algorithm

As mentioned in Section III, there are three non-trivial challenges left for solving the SSC by the QGRA model. The first one is how to swiftly cluster tasks to form roles. The second is to design an appropriate strategy to evaluate agents' satisfaction, and the last one is to achieve AC. We propose three corresponding methods for these concerns: role awareness, ASE, and AC.

A. Role Awareness

The role awareness phase, aimed at determining the appropriate roles in the E-CARGO model, has developed into a vital step for RBC [17], [18]. Superficially, in SSC, it is easy to take the crowdsourcing tasks as roles because they are independent and straightforward. More pertinently, we reasonably divide these tasks into clusters to form the roles. Such an operation can increase roles' profits, thereby improving the agents' likelihoods to perform these roles. Besides, taking task clusters as roles in the PMMP can also convert the 1-M problem [28] to a 1-1 [14] problem. This conversion can considerably reduce the time complexity of the distributing process in SSC. Also, because the tasks in the SSC scenarios are independent, the clustering of tasks does not introduce new constraints.

Based on the predefined vital parameters in Table I, we designed the specific role awareness method as shown in Algorithm 1. Here, we define some special symbols as follows for a better description for Algorithm 1.

Algorithm 1 Role Awareness

input: $\mathcal{D}$, ε, Γ, w_{max}
output: $\mathcal{R}$
begin
for each d in $\mathcal{D}$ **do** /*Initialization*/
  $\Gamma[d] \leftarrow 0$;
end for
for each d in $\mathcal{D}$ **do**
  if $\Gamma[d] = 0$ **then**
    $N_d \leftarrow$ *searchNeighbors*($\mathcal{D}$, d, ε, Γ);
  end if
  if $|N_d| \le 1$ **then** /*Isolated task*/
    $\Gamma[d] \leftarrow -1$;
  elseif $|N_d| \ge w_{max}$ **then**
    $\Gamma[d] \leftarrow \Gamma[d] + 1$; /*Updating cluster's ID*/
    Select a task with the highest price and ($w_{max}-1$) lowest-priced tasks in N_d to form a new N_d;
    for each d' in N_d **do**
    Select the corresponding cluster type of d' from Γ as $\Gamma[d']$;
    $\Gamma[d'] \leftarrow \Gamma[d]$;
    end for
else
    $\Gamma[d] \leftarrow \Gamma[d] + 1$;
    count $\leftarrow$ 1; /*Temporary variables*/
    for each d' in N_d **and** count $< w_{max}$ **do**
      if $\Gamma[d'] = 0$ **then**
        $N_d \leftarrow N_d \cup$ *searchNeighbors*($\mathcal{D}$, d', ε, Γ);
      end if
      $\Gamma[d'] \leftarrow \Gamma[d]$;
      count $\leftarrow$ count + 1;
    end for
  end if
end for
Classify $\mathcal{D}$ by Γ to get $\mathcal{R}$;
return $\mathcal{R}$;
end

1) $\mathcal{D}$ represents a set of crowdsourcing tasks.
2) ε is the neighborhood radius. In the PMMP, the social meaning of ε is the acceptable distance of the crowdsourcing workers to move between these tasks. Since [29], [30] depict that the generally acceptable walking range for a person is 1.5 km, we set $\varepsilon = 1.5$ km.
3) Γ is an identifier vector, and $\Gamma[d] \in \{-1\} \cup \mathbb{N}^*(0 \le d < |\mathcal{D}|)$ represents the corresponding cluster-ID, which is a positive integer, for task d. $\Gamma[d] = -1$ represents an isolated task. Initially, $\Gamma[d]$ is initialized as 0 to indicate that task d has not been clustered.
4) $w_{\max}$ indicates the maximum number of tasks a worker can accept. It is defined in Table I.
5) N_d indicates a collection of the neighbor crowdsourcing tasks of task d.
6) $\mathcal{R}$ is the set of roles defined in Section III.
7) *searchNeighbors*($\cdot$) is a function to cluster the crowdsourcing tasks in $\mathcal{D}$, and it returns a modified Γ.

Algorithm 2 *searchNeighbors* ($\mathcal{D}$, d, ε, Γ)

input: $\mathcal{D}$, d, ε, Γ
output: N_d
begin
$N_d = \{\}$;
for each d' in $\mathcal{D}$ **do**
  Select the corresponding cluster ID of d' from Γ as $\Gamma[d']$;
  if $\Gamma[d'] \leq 0$ and *calculateGPSDistance*(d, d') $\leq \varepsilon$ **then**
    $N_d \leftarrow N_d \cup \{d'\}$;
  end if
end for
return N_d
end

One of the vital steps of Algorithm 1 is the function *searchNeighbors*($\cdot$), and its pseudo-code is illustrated as Algorithm 2. Here, for clarity, we introduce an additional symbol.

1) *calculateGPSDistance*($\cdot$) is a function to calculate Euclidean distance between crowdsourcing tasks by expression (1).

Based on the pseudo-codes of Algorithms 1 and 2, the time complexity of our proposed role awareness method is $O(k^2)$, where $k = |\mathcal{D}|$. Same as the DBSCAN algorithm [31], we utilize the task's density as the primary criterion for clustering. But the difference is that our clustering tasks are limited to the maximum number, and each cluster can only contain $w_{\max}$ tasks. To this end, we apply the principle of tying in economics [32] and maximize the price difference as the criterion for dividing those task clusters with more than $w_{\max}$ tasks. Note that such a division method may not obtain the optimal solution for a one-time assignment. Still, for SSC, the most important thing is to implement the adaptive scheduling scheme swiftly.

B. Agent Satisfaction Evaluation

Another key issue of solving SSC is quantifying the variable n_w in expression (10), which is related to the agent's satisfaction with the assigned roles. For an agent (worker), the satisfaction of playing a role (task cluster) is related to the profit of playing the roles and the agent's income level. The higher the pricing of the role, the more likely the agent will perform the role. Meanwhile, the lower the agent's income level, the higher the probability of performing the roles. Based on above analysis, we propose an ASE method. To better describe this method, we propose some new definitions.

Definition 16: A role profit vector $\Diamond$ is an n-vector, and $\Diamond[j](\Diamond[j] \in \mathbb{R}^+$ and $0 \leq j < n)$ expresses the total revenue obtained by an agent from completing role j. Additionally, the unit of $\Diamond[j]$ is yuan.

Note: In the PMMP, roles are the task clusters. We denote $\mathcal{D}_j$ as the set of crowdsourcing tasks in role j and P_d as the self-defined prices of task d in $\mathcal{D}_j$. Therefore, the element in vector $\Diamond$ can be formalized as $\Diamond[j] = \sum_{t=0}^{w} P_d$, where $w = |\mathcal{D}_j|$. For example, in the scenario of Section II, $\Diamond = [3.23, 8.35, 10.10, 7.89, 4.69, \ldots, 14.94, 10.45, 15.52, 23.55, 40.37]$.

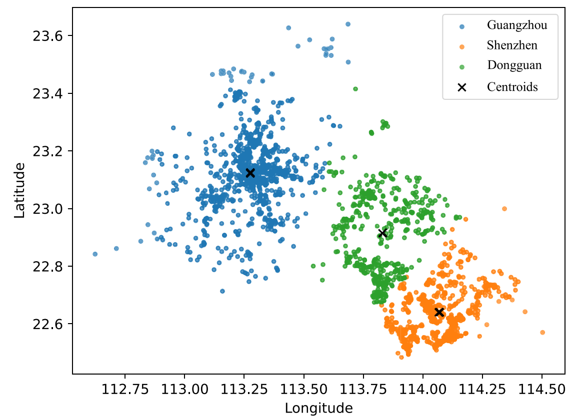


Fig. 3. Divided clusters of the crowdsourcing workers by the K-means algorithm.

Definition 17: An agent income level vector Λ^a is an m-vector, and $\Lambda^a[i]$ ($\Lambda^a[i] \in \mathbb{R}^+$ and $0 \leq i < m$) expresses the income level of agent i. The superscript of Λ^a represents that Λ^a is a definition for the agents. Moreover, the unit of $\Lambda^a[i]$ is also yuan.

Note: As shown in Fig. 1, agents are the local crowdsourcing workers in Guangdong Province, China. To this end, we hope to use the average daily incomes of the area where the agents are located to measure their income levels. Through the analysis of the location information of the crowdsourcing workers, we find that they are mainly located in three areas: Guangzhou, Dongguan, and Shenzhen, which are three cities in Guangdong Province, China. The geographical locations (see Fig. 3) and economic development levels of these three cities are different. By gathering the local statistical data of these cities [34]–[36], we can calculate that the latest average daily income of residents of these cities is 281.9, 214.4, and 376.2 yuan, respectively. Consequently, we apply the K-Means algorithm [33] to partition the agents into three clusters, which correspond to the above three cities, respectively, as illustrated in Fig. 3. Then, we empirically use the latest average daily income of residents in these cities as a measure of the income level of the agent, namely $\Lambda^a = [376.2, 214.4, 281.9, 281.9, 281.9, \ldots, 281.9, 281.9, 281.9, 214.4, 281.9]$.

Definition 18: An agent loss profit matrix Ξ is an $m \times n$ matrix and $\Xi[i, j]$ ($\Xi[i, j] \in \mathbb{R}^+$, $0 \leq i < m$, and $0 \leq j < n$) expresses the overhead for agent i to play role j.

Note: In the PMMP, the main expense is transportation, that is, by bus or cab. As mentioned in Section VI-A, an agent's acceptable walking range is 1.5 km [30], that is, $\varepsilon = 1.5$ km. We can regard that when the distance between an agent and a role is less than 1.5 km, the agent's transportation expense is 0 yuan. Otherwise, the agent's overhead increases monotonically with the distance, but the increase eventually stabilizes [37]. Therefore, we empirically estimate the relation between the distance and the transportation expense as follows:

$$\Xi[i, j] = \begin{cases} 0, & Q[i, j] \leq \varepsilon \\ \alpha \times (1 - e^{\varepsilon - Q[i, j]}), & Q[i, j] > \varepsilon \end{cases} \tag{11}$$

where $Q[i, j] \in \mathbb{R}^*$ is a qualification value, ε is the pre-defined neighborhood radius representing an agent's acceptable walking range, and α ($\alpha \in \mathbb{R}^{+}$) depicts the growth rate. In our scenario, we randomly set $\alpha = 6$, and without loss of generality, we will conduct large-scale experiments in Section V with a random value of α. Note that the form of expression (11) will not affect the generality of the QGRA model.

Definition 19: An agent satisfaction threshold vector τ^a is an m vector, and $\tau^a[i]$ ($\tau^a[i] \in [0, 1]$ and $0 \le i < m$) expresses the acceptable ratio between profit gained from completing a role and the income level of agent i.

Note: In our scenario, we randomly set $\tau^a = [0.1, 0.7, 1.0, 1.0, 0.2, \ldots, 0.5, 0.6, 0.9, 0.3, 0.1]$, and we will carry out large-scale experiments in Section V with a randomly generated value of τ^a.

The above new definitions introduce the ASE method

$$\text{ASE}(i, j) = \frac{(\Diamond[j] - \Xi[i, j])}{\Lambda^a[i] \times \frac{t_{\text{interval}}}{t_{\text{work}}}}, \quad (0 \le i < m, 0 \le j < n) \quad (12)$$

where t_{work} represents local working hours (8 h, including an hour for a break), and t_{interval} represents the time taken by agent i to finish role j, and the denominator of expression (12) represents the original working income of agent i during t_{interval}. In the PMMP, we assume that role j can be completed by agent i within t_{interval} when the transportation expense is sufficient. Additionally, for those agents that complete the role tasks swiftly, we still need them to wait until the time of t_{interval} to perform the re-assignment. Such a batching matching strategy [38] can get a more comprehensive assignment plan.

Based on the above definitions, we can estimate the capability matrix $\bar{Q}$ as follows:

$$\bar{Q}[i, j] = \begin{cases} 0, & \text{ASE}(i, j) < \tau^a[i] \\ 1, & \text{ASE}(i, j) \ge \tau^a[i]. \end{cases} \quad (13)$$

Once the capability matrix $\bar{Q}$ and relevant n_w are determined, the QGRA model can realize a one-time assignment with the help of the IBM ILOG CPLEX optimization package (CPLEX) [39]. Here, we use the scenario in Section II as an example. In the initial state of the system $\sum$ formalized in Section II, with $m = 1863$, $n = 401$, $\mathfrak{T} = [0, 0, 1, 1, 1, \ldots, 0, 1, 0, 1, 1]$, $\mathfrak{T}^a = [1, 1, 1, 0, 1, \ldots, 1, 1, 0, 1, 1]$, and $n_w = 276$, the group performance obtained by the QGRA model of the initial assignment σ^{QGRA} is 0.496. The corresponding tasks completion rate of QGRA is 78.69%. Evidently, there are still 21.31% of the crowdsourcing tasks remaining to be finished. For this concern, the next vital step is to achieve the adaptive re-assignment for these unfinished tasks.

C. Adaptive Collaboration

To achieve adaptive re-assignment for those unfinished tasks left from the last assignment phase, an AC problem, the priority is to identify the trigger mode and timing of the system's unsteady state.

In SCC, three influenced factors will trigger the re-evaluation of the Q matrix and re-assignment of the QGRA model: the completion of the roles, the time slot for a one-time assignment, and the change of agent's location. The first two factors can be determined by the last scheduling result and the predefined parameter t_{interval} in Table I, respectively. As for the change of agent's location during the scheduling process, we propose Algorithm 3 for simulation. To better describe it, we define the following symbols.

Algorithm 3 Changing Agent's Location

input: $< x_i, y_i >, < x_j, y_j >, T$
output: $< x'_i, y'_i >$
begin
$x'_i \leftarrow x_i,\ y'_i \leftarrow y_i$;
if $T[i][j] = 1$ **then** /*agent i undertakes role j*/
  $x'_i \leftarrow x_j,\ y'_i \leftarrow y_j$;
else
  /*$\mathbb{U}$ represents the uniform distribution*/
  $r_{move} \leftarrow \mathbb{U}\ (0, 2.7)$;
  $\theta \leftarrow \mathbb{U}\ (0, 2\pi)$;
  /*Constant 111 is the approximate conversion multiple between kilometer and latitude/longitude*/
  $x'_i \leftarrow x'_i + \frac{r_{move}}{111} cos\theta,\ y'_i \leftarrow y'_i + \frac{r_{move}}{111} sin\theta$;
end if
return $< x'_i, y'_i >$;
end

1) $\langle x_i, y_i \rangle$ is the current location of Agent i.
2) $\langle x_j, y_j \rangle$ is the location of Role j.
3) $\langle x'_i, y'_i \rangle$ is the predicted location of Agent i obtained by Algorithm 3.
4) T is the role assignment matrix obtained by the QGRA model.
5) r_{move} represents the moveable radius during the time slot (t_{interval}) of the current scheduling scheme. Since [29] shows that the average walking speed of a person is 5.4 km/h, we set the range of $r_{\text{move}} \in [0, 5.4 \times t_{\text{interval}}]$. In our scenario, since t_{interval} is 0.5 h, the range of r_{move} is [0, 2.7] (unit in km).

Algorithm 3 elaborates the proposed method to simulate the change of agents' positions. Obviously, its time complexity is $O(1)$.

With the above-mentioned role awareness method, ASE method, and Algorithm 3, the QGRA model can achieve AC for solving SCC, as illustrated in Algorithm 4. Accordingly, here we define some specific symbols for clarifying Algorithm 4:

1) t_{current} is the current time of the system $\sum$.
2) *roleAwareness*$(\cdot)$ represents Algorithm 1.
3) $[-](\cdot)$ is a function named $[-]$, which the CPLEX solves.
   For example, QGRA$(\mathcal{R}, \mathcal{A}, Q, L, \tau)$ represents the function of the QGRA model [expression (8)].
4) $T^{[-]}$ is a workable role assignment matrix obtained by the model $[-]$, that is, T^{QGRA} depicts that of the QGRA model.
5) $\bar{\sigma}^{[-]}$ is the overall group performance of $[-]$ obtained by Algorithm 4, that is, $\bar{\sigma}^{\text{QGRA}}$.
6) $\mathcal{D}'^{[-]}$ represents those finished tasks of $[-]$ evaluated by Algorithm 4. For example, $\mathcal{D}'^{\text{QGRA}}$ is that of the QGRA model.

Algorithm 4 illustrates the pseudo-code of achieving AC of the QGRA model; because the complexity of the revised GRA

Algorithm 4 Adaptive Collaboration[a]

input: $\mathcal{D}$, $\mathcal{A}$, ε, Γ, w_{max}, t_{start}, t_{end}, $t_{interval}$
output: $\bar{\sigma}^{QGRA}$, $\mathcal{D}'^{QGRA}$
begin
$t_{current} \leftarrow t_{start}$, $\bar{\sigma}^{QGRA} \leftarrow 0$, $\mathcal{D}'^{QGRA} \leftarrow \emptyset$;
while $t_{current} \leq t_{end}$ and $|\mathcal{D}| > 0$ **do**
  $\mathcal{R} \leftarrow roleAwareness\ (\mathcal{D}, \varepsilon, \Gamma, w_{max})$;
  Evaluate Q and L by Expression (1) and $\mathcal{R}$;
  Calculate the variable τ by Expression (13);
  σ^{QGRA}, $T^{QGRA} = QGRA\ (\mathcal{R}, \mathcal{A}, Q, L, \tau)$;
  Add those finished tasks to $\mathcal{D}'^{QGRA}$ by T^{QGRA};
  $\mathcal{D} \leftarrow \mathcal{D} - \mathcal{D}'^{QGRA}$;
  $\bar{\sigma}^{QGRA} \leftarrow \bar{\sigma}^{QGRA} + \sigma^{QGRA}$;
  Update the locations of the agents in $\mathcal{A}$ by Algorithm 3;
  $t_{current} \leftarrow t_{current} + t_{interval}$;
end while
return $\bar{\sigma}^{QGRA}$, $\mathcal{D}'^{QGRA}$;
end

[a]Note: For convenience, we denote the model [-] after applying Algorithm 4 as the [-] algorithm.

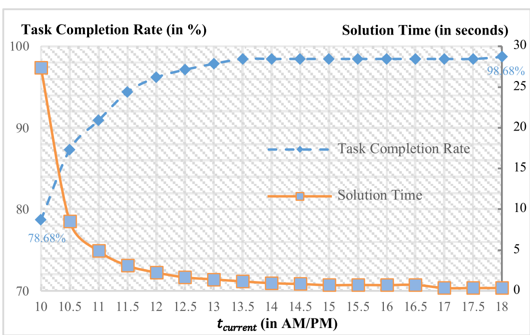


Fig. 4. Solution time and the task completion rate of Algorithm 4 in our scenario[a]. [a]Note: In our scenario in Section II, the value of the parameters are: $m = 1863$, $|\mathcal{D}| = 835$, $t_{\text{start}} = $ 10:00 AM, $t_{\text{end}} = $ 18:00 PM, and $t_{\text{interval}} = 0.5$ h.

model is $O\ (m^3)$ ($m = |\mathcal{A}|$) [13], [40], then that of the QGRA model and Algorithm 4 are also $O\ (m^3)$.

Fig. 4 shows the solution time and task completion rate of Algorithm 4 in each unsteady state phase of the scenario in Section II. As shown in Fig. 4, compared with the one-time QGRA solution with a task completion rate of 78.68%, the task completion rate of Algorithm 4 reaches 98.68%. Those unfinished tasks' names are Task 297, Task 298, Task 303, Task 384, Task 450, Task 561, Task 587, Task 745, Task 777, Task 783, and Task 795. No workers were willing to finish these tasks because of the low profits. The overall group performance $\bar{\sigma}^{\text{QGRA}}$ obtained by Algorithm 4 in this scenario is 0.78. Compared with GMRA, the average solution time of Algorithm 4 is 4.12 s, while that of GMRA is 8.46 s. Additionally, the overall average profit of agents in QGRA is 201.11 yuan, while that of GMRA is 143.77 yuan. Thus, the QGRA algorithm outperforms GMRA when solving the SSC problem.

V. Simulation Experiment

To demonstrate the viability and efficiency of our proposed method, we perform large-scale random simulation experiments on a laptop configured, as illustrated in Table IV.

TABLE IV
Configuration of the Experimental Platform

Hardware	
CPU	2.6 GHz Intel Core i7
Memory	16GB 2400 MHz DDR4
Software	
OS	macOS Big Sur Version 11.5.2
Editor	Visual Studio Code Version 1.59.1
Python	Python 3.8.5

TABLE V
Ranges of Some Self-Defined Parameters[A]

Parameters	Ranges
α	(0, 10]
$\tau^a[i]\ (0 \leq i < m)$	[0, 1]
n	$[0, \mid\mathcal{D}\mid]$
$P_d\ (0 \leq d < \mid\mathcal{D}\mid)$	(0, 30]

*Note: The ranges of these parameters are set based on the local income [34]–[36] and referenced applications [20].

Since [1] offers two different sizes of the real crowdsourcing task datasets and a worker dataset, we use them as the original datasets in the simulation, that is, ($|\mathcal{A}| = m = 1863$, $|\mathcal{D}| = 835$) and ($|\mathcal{A}| = m = 1863$, $|\mathcal{D}| = 2066$). In the meantime, for verifying our proposed method's robustness, we add random disturbances to the two original task datasets by applying Algorithm 3. Such an approach can simulate different distributions of the crowdsourcing tasks. Moreover, as mentioned above, our proposed solution is mainly concerned with some self-defined parameters: the growth rate α in expression (11), the agent satisfaction threshold vector τ^a in Definition 16, the role numbers n obtained by Algorithm 1, and the pricing P_d of the crowdsourcing task d ($0 \leq d < |\mathcal{D}|$). Based on the ranges of these parameters in Table V, we will conduct our simulation experiments for 100 random cases to get the average result.

Also, to simplify the description, we introduce the following symbols.

1) σ^{QGMRA} denotes the group performance obtained by the traditional GMRA model with an AC. That is to say, the QGMRA model is to find a workable T to obtain expression (6) subject to expressions (3), (7), (9), and (10). Note that the difference between the QGRA algorithm and the GRA algorithm is that the former takes task clusters, obtained by the role awareness method in Section IV, as roles, but the latter directly regards crowdsourcing tasks as roles.
2) $\partial[-]$ denotes the overall average profit of agents in the model $[-]$ quantified by expression (12), that is, ∂^{QGRA} representing that of the QGRA model.
3) $\mathcal{F}[-]$ represents the overall task completion rate of the model $[-]$. Namely, $\mathcal{F}^{\text{QGRA}}$ depicts the overall task completion rate of the QGRA model.
4) λ_1 is defined as $(\partial^{\text{QGRA}} - \partial^{\text{QGMRA}})/(\partial^{\text{QGMRA}})$, and λ_2 is defined as $(\partial^{\text{QGRA}} - \partial^{\text{GMRA}})/(\partial^{\text{GMRA}})$ if $\partial^{\text{GMRA}} > 0$. Otherwise, λ_2 is defined as $\partial^{\text{QGRA}} - \partial^{\text{GMRA}}$.

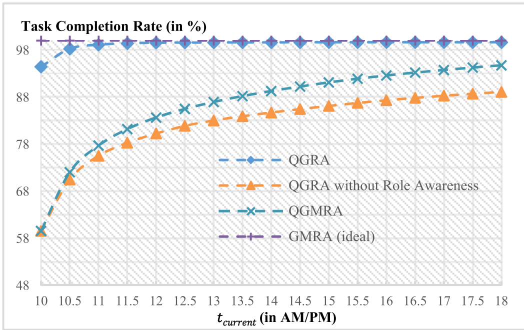


Fig. 5. Average task completion rate under two typical datasets[a]. [a]Note: The sparse dataset is with $|\mathcal{A}| = m = 1863$, $|\mathcal{D}| = 835$, and the density dataset is with $|\mathcal{A}| = m = 1863$, $|\mathcal{D}| = 2066$.

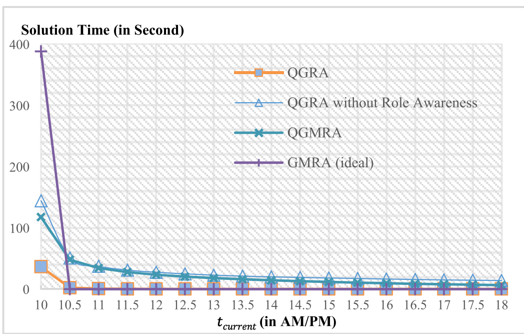


Fig. 6. Average solution time under two typical datasets[a]. [a]Note: The sparse dataset is with $|\mathcal{A}| = m = 1863$, $|\mathcal{D}| = 835$, and the density dataset is with $|\mathcal{A}| = m = 1863$, $|\mathcal{D}| = 2066$.

TABLE VI

SIMULATION AVERAGES[A]

Algorithm	Average Simulation Result			
	$\bar{\sigma}$	∂	$\mathcal{F}$	t_{solve}
GMRA (ideal)	8.58	572.65	100%	388.29s (22.84s in average)
QGMRA	2.88	771.55	94.73%	396.17s (23.30s in average)
QGRA without Role Awareness	10.44	568.34	89.08%	511.32s (30.08s in average)
QGRA	**1.47**	**824.78**	**99.64%**	**44.17s (2.60s in average)**
$\boldsymbol{\lambda_1}$ = +6.90%; $\boldsymbol{\lambda_2}$ = +44.03%; $\boldsymbol{\mu_1}$ = +4.91%; $\boldsymbol{\mu_2}$ = -0.36%.				

*Note: The ideal GMRA algorithm gets the optimal solution by ignoring the overall average profit/satisfaction of agents.

5) μ_1 is defined as $\mathcal{F}^{\text{QGRA}} - \mathcal{F}^{\text{QGMRA}}$, and μ_2 is defined as $\mathcal{F}^{\text{QGRA}} - \mathcal{F}^{\text{GMRA}}$.

First, we examine the validity of the proposed QGRA algorithm. Then, we compared the QGRA algorithm with relevant existing GRA/GMRA algorithms, as elaborated in Figs. 5 and 6. Obviously, Figs. 5 and 6 verify that the original role awareness method can help the QGRA algorithm drastically reduce the solution time while increasing nearly 10.56% of the task completion rate.

Meanwhile, Table VI supplement some overall differences among these algorithms. It demonstrates that, with the help of the role awareness method, the QGRA algorithm even has a higher task completion (i.e., $\mu_1 = +4.91\%$ in Table VI) and lower solution time than the QGMRA algorithm. It is worth mentioning that the overall group performance $\bar{\sigma}^{\text{GMRA}}$

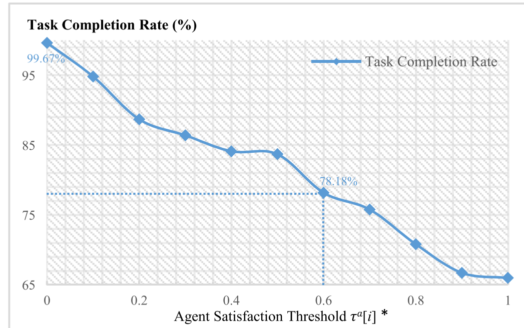


Fig. 7. Relationship between task completion rate and agents' satisfaction in QGRA algorithm[a]. [a]Note: For comparison, we assume that all workers have the same satisfaction, for example, $\tau^a = [0.6, 0.6, \ldots, 0.6, 0.6]$. For convenience, we denote the former equation as $\tau^a[i] = 0.6$ $(0 \leq i < m)$.

of the GMRA algorithm is even higher than that of QGRA or QGMRA since GMRA enforces workers to some outlier tasks. The average agents' profit ∂ of QGRA without applying the role awareness method is lower than ∂^{GMRA} because GMRA has a higher task completion rate. In fact, the former average profit $(568.34/89.08 = 6.38)$ is higher than that of the latter $(572.65/100 = 5.73)$.

Eventually, we examine the interrelation between the task completion rate and the agents' satisfaction. Expression (13) reveals that the solution of the QGRA algorithm is mainly influenced by the value of τ^a. For this concern, we take the PMMP in Section II as an example to exemplify the change of the task completion rate of the QGRA algorithm under different values of τ^a, that is, Fig. 7. Based on Fig. 7, we can draw a conclusion that when the expected profit of the agents obtained from these tasks is less than 60% of the local average income, the task completion rate will be dropped to less than 80%, that is, 78.18%. More pertinently, we justify how specific pricing of the tasks influence the tasks completion rate, that is, Table VII. Table VII demonstrates that, in the scenario of Section II, when the average price of the tasks exceeds 10 yuan, the completion rate of the task is close to 100%.

From the experiment above, we can conclude the following.

1) Figs. 5 and 6 show that, with the novel role awareness method, the QGRA algorithm can significantly reduce the average solution time while dramatically increasing the task completion rate.
2) Table VI and Figs. 5 and 6 exemplify that the QGRA outperforms existing GRA/GMRA algorithms from perspectives of both solution time and agents' profits.
3) Table VI and Fig. 7 reveal that the pricing of the crowdsourcing tasks plays a decisive role in the task completion rate. The decision-makers can trade-off between task completion rate and agents' satisfaction by checking the value of the agent satisfaction threshold τ^a based on their demands.

From the simulation results, we understand the following.

1) The innovative QGRA algorithm can complete more than 99% of the tasks and ensure that the average solution time is within 3 s, in line with the tolerant time of the actual application.
2) The task's pricing is a critical factor that influences the task completion rate. When the expected profit of the

TABLE VII

AVERAGE TASK COMPLETION RATE (%)[A]

$\tau^a[i]$	The range of P_d ($0 \le d < \lvert\mathcal{D}\rvert$)		
	0~10	10~20	20~30
0.0	99.34%	99.93%	100%
0.1	84.62%	99.92%	100%
0.2	66.23%	99.90%	100%
0.3	59.43%	99.86%	100%
0.4	52.78%	99.70%	100%
0.5	51.57%	99.70%	100%
0.6	35.56%	99.13%	100%
0.7	28.66%	98.90%	100%
0.8	14.80%	98.32%	100%
0.9	2.63%	98.05%	100%
1.0	0.15%	97.90%	100%

*Note: Table VII records the average task completion rate of two typical datasets [1]. The sparse dataset is with $|\mathcal{A}| = m = 1863$, $|\mathcal{D}| = 835$, and the density dataset is with $|\mathcal{A}| = m = 1863$, $|\mathcal{D}| = 2066$.

agents obtained from these tasks is less than 60% of the local average income, the task completion rate will be dropped to less than 80%, that is, 78.18%.

VI. RELATED WORK

A. Spatiotemporal Crowdsourcing

SC [41]–[43] is a booming machine computational paradigm, which has received increasing attention in multiple fields, that is, mobile crowdsourcing [44]–[46], event reporting [47], privacy preservation [48], task allocation [17], [49], [50], and so forth [51]–[54]. In recent years, many scholars have tended to solve the task assignment problem through multi-agent systems (agent-based method [55]–[58]). For example, Wang *et al.* [44] developed G-DPSO-based worker-centric task selection and M-ITA-based platform-centric worker selection, which can improve the truthfulness and the efficiency of the mobile crowd sensing networks (MCSNs). From the event reporting aspect, Hamrouni *et al.* [47] proposed a photograph-based spatial mobile crowdsourcing (SMCS) framework for event reporting, and their simulation results show the effectiveness of the proposed framework in reducing false submissions and delivering high-quality responses of their method. Moreover, Xu *et al.* [48] designed a blockchain-powered crowdsourcing method (BPCM) to achieve crowdsourcing workers' privacy as well as reduce the energy consumption for the providers.

The aforementioned specific studies of SC have achieved substantial results in the corresponding field. Still, SSC proposed in this article is a new type of task allocation problem in SC, and there is a shortage of research efforts for directly solving it.

B. Tasks Allocation in SC

SC and SSC are all essentially task allocation problems [22], [59]–[61] between the crowdsourcing workers and relevant crowdsourcing tasks. A typical instance of assignment problems in SC is Amazon's mechanical turk (AMT) microtask market, a crowdsourcing platform, which is mainly applied to simple and independent tasks such as adding tags to images or judging search results. There exist many corresponding types of research to it [50], [62], [63]. For example, Zheng *et al.* [50] developed a quality-aware task assignment system for crowdsourcing applications (QASCA) for the online task assignment problem by incorporating evaluation metrics of the workers into assignment strategies. Dai *et al.* [63] proposed a feature-based Bayesian task recommendation (BTR) system, which recommends tasks for workers by learning interactions between workers and task features based on neural networks.

The aforementioned research has properly tackled the AMT-related problem by analyzing the workers' skills or tasks' property. However, since the tasks in SSC are straightforward and unskilled, that is, taking pictures, the above method is not suitable for it. Besides, the relevant articles of SSC (e.g., PMMP) mainly focus on task pricing [2] instead of team assignment. Moreover, based on analyzing the related software of PMMP, that is, Foap [19] and Meituan Paidian [20], we find that the tasks in SSC have a low task completion rate due to its self-service property.

C. GRA and SSC

Due to the low task completion rate and existing competition dilemma [43] between workers, a centrally organized methodology with distributed assignment is required. Recently, some scholars have tended to solve SC-related problems by RBC [11], [17], [25], [64] and its GRA/GMRA [65], [66] submodel, which is characterized by centralized modeling and distributed execution [18]. For example, Huang *et al.* [17] deal with the LMAP for fresh produce via GRA with constraints (GRA+), which can assign proper couriers for forming a team to deliver daily orders efficiently and improve the quality of the team's service. However, their proposed model can only solve the one-time assignment problem, and the tasks in SSC require AC for reassigning those unfinished tasks.

Regarding the AC problem, there have been some successful studies to solve it in the supply chain area. For example, Peng *et al.* [67] study the effectiveness of the transshipment strategy under inventory competition and how to coordinate a supply chain in the presence of transshipment. Moreover, Liu *et al.* [68] provide guidelines for improving the cooperation between comprehensive retail platforms (CRPs) and social service platforms (SSPs). The above-mentioned studies have achieved vital results in tackling the AC problem in the supply chain field, but their ACs are mainly triggered by the goods' price. However, in SSC, the task's pricing is relatively stable, and the AC trigger is due to the locations of the crowdsourcing workers and the time window. More pertinently, although Sheng *et al.* [4] conditionally extended GRA with an AC mechanism, and their model is that the agent enforces the role without considering the agents' satisfaction. The proposed QGRA provides renovations in time consumption, agents' satisfaction, and task completion.

VII. Conclusion

This article proposes an extension of the GRA model to solve SSC (e.g., PMMP), that is, QGRA.

In order to accelerate the solution of the SSC problem while appropriately raising the task completion rate, this article first devises a role awareness method to select proper task clusters for forming the roles. Then, it assigns the tasks in SSC by evaluating the agents' satisfaction, that is, ASE method, for improving the task completion rate dramatically. Leveraging the ASE method can also reveal the relationship between task completion rate and workers' satisfaction. Moreover, it sheds light on an AC algorithm to help QGRA achieve swift re-assignment for those unfinished tasks. Lastly, thousands of varying scale simulation experiments are conducted to demonstrate the efficiency and robustness of our proposed method.

The simulation results provide a solid reference for the decision-makers to weigh the task completion rate and workers' satisfaction.

From this article, further investigations on the extended QGRA model may be conducted in the following directions.

1) More efforts should be taken to analyze the relationship between the workers' profits and the crowdsourcers' expenditure.
2) When the release cycle of new tasks becomes shorter (e.g., half an hour), these new tasks need to be considered during the re-assignment process.
3) Investigates a comprehensive role awareness method and proves its optimality for clustering the unfinished tasks.
4) Re-assignment may be conducted instantly if sufficient agents become idle. For crowdsourcing workers who complete tasks swiftly, the system may assign new tasks to them.
5) We may study a more appropriate distance evaluation method instead of directly using the Euclidean distance.
6) Merely adding n_w as a constraint (i.e., Expression (10)) of the QGRA model is actually a trade-off between task completion rate and satisfaction. For scenarios where new tasks and new crowdsourcing workers join, more constraints can be introduced to ensure the satisfaction of all crowdsourcing workers.

Appendix

A. Relevant Utilized Datasets

The original datasets in [1] and our modified datasets are accessible in the public Github project of the authors: https://github.com/jiangqian1997/E-CARGO-Codes/tree/main/QGRA_datasets. The descriptions of these datasets are in a file named "README.txt."

References

[1] *Pricing for the Task of Taking Pictures to Make Money*, Contemp. Undergraduate Math. Contest Model., Chin. Soc. Ind. Appl. Math., 2017, p. 1.

[2] Z. Li, Y. Li, W. Lu, and J. Huang, "Crowdsourcing logistics pricing optimization model based on DBSCAN clustering algorithm," *IEEE Access*, vol. 8, pp. 92615–92626, 2020.

[3] W. Wang, B. An, and Y. Jiang, "Optimal spot-checking for improving the evaluation quality of crowdsourcing: Application to peer grading systems," *IEEE Trans. Computat. Social Syst.*, vol. 7, no. 4, pp. 940–955, Aug. 2020.

[4] 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.

[5] H. Zhu, "Adaptive collaboration systems," *IEEE Syst., Man, Cybern. Mag.*, vol. 1, no. 4, pp. 8–15, Oct. 2015.

[6] H. Zhu, M. Hou, and M. Zhou, "Adaptive collaboration based on the E-CARGO model," *Int. J. Agent Technol. Syst.*, vol. 4, no. 1, pp. 59–76, Jan. 2012.

[7] H. Zhu, "Maximizing group performance while minimizing budget," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 50, no. 2, pp. 633–645, Feb. 2020.

[8] H. Zhu, "Avoiding critical members in a team by redundant assignment," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 50, no. 7, pp. 2729–2740, Jul. 2020.

[9] H. Zhu, "Avoiding conflicts by group role assignment," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 46, no. 4, pp. 535–547, Apr. 2016.

[10] X. Zhu, H. Zhu, D. Liu, and X. Zhou, "Criteria making in role negotiation," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 50, no. 10, pp. 3731–3740, Oct. 2020.

[11] H. Zhu *et al.*, "Solving the many to many assignment problem by improving the Kuhn–Munkres algorithm with backtracking," *Theor. Comput. Sci.*, vol. 618, pp. 30–41, Aug. 2016.

[12] H. Zhu and M. Zhou, "Role-based collaboration and its kernel mechanisms," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 36, no. 4, pp. 578–589, Jul. 2006.

[13] H. Zhu, D. Liu, S. Zhang, S. Teng, and Y. Zhu, "Solving the group multirole assignment problem by improving the ILOG approach," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 12, pp. 3418–3424, Dec. 2017.

[14] H. Zhu, Y. Sheng, X. Zhou, and Y. Zhu, "Group role assignment with cooperation and conflict factors," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 48, no. 6, pp. 851–863, Jun. 2018.

[15] H. Zhu, M. C. 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.

[16] H. Zhu, "Group multi-role assignment with conflicting roles and agents," *IEEE/CAA J. Autom. Sinica*, vol. 7, no. 6, pp. 1498–1510, Nov. 2020.

[17] B. Huang, H. Zhu, D. Liu, N. Wu, Y. Qiao, and Q. Jiang, "Solving last-mile logistics problem in spatiotemporal crowdsourcing via role awareness with adaptive clustering," *IEEE Trans. Computat. Social Syst.*, vol. 8, no. 3, pp. 668–681, Jun. 2021.

[18] D. Liu, Q. Jiang, H. Zhu, and B. Huang, "Distributing UAVs as wireless repeaters in disaster relief via group role assignment," *Int. J. Cooperat. Inf. Syst.*, vol. 29, nos. 1–2, pp. 2040002-1-2040002-22, 2020.

[19] (2021). *Foap*. [Online]. Available: https://www.foap.com/

[20] Meituan. (2021). *Meituan Paidian*. [Online]. Available: https://paidian.meituan.com/

[21] R. Burkard, M. Dell'Amico, and S. Martello, *Assignment Problems: Revised Reprint*. Philadelphia, PA, USA: SIAM, 2012.

[22] H. Zhu, "Agent categorization with group role assignment with constraints and simulated annealing," *IEEE Trans. Computat. Social Syst.*, vol. 7, no. 5, pp. 1234–1245, Oct. 2020.

[23] H. Zhu, "Computational social simulation with E-CARGO: Comparison between collectivism and individualism," *IEEE Trans. Computat. Social Syst.*, vol. 7, no. 6, pp. 1345–1357, Dec. 2020.

[24] H. Zhu, Z. Yu, and Y. Gningue, "Solving the exam scheduling problem with GRA+," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. (SMC)*, Toronto, ON, Canada, Oct. 2020, pp. 1485–1490.

[25] H. Zhu, "Why did Mr. Trump oppose globalization? An E-CARGO approach," *IEEE Trans. Computat. Social Syst.*, vol. 8, no. 6, pp. 1333–1343, Dec. 2021.

[26] C. C. Robusto, "The cosine-haversine formula," *Amer. Math. Monthly*, vol. 64, no. 1, pp. 38–40, 1957.

[27] 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.

[28] H. Zhu and M. Zhou, "M–M role-transfer problems and their solutions," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 39, no. 2, pp. 448–459, Mar. 2009.

[29] R. W. Bohannon, "Comfortable and maximum walking speed of adults aged 20–79 years: Reference values and determinants," *Age Ageing*, vol. 26, no. 1, pp. 15–19, 1997.

[30] P. N. Seneviratne, "Acceptable walking distances in central areas," *J. Transp. Eng.*, vol. 111, no. 4, pp. 365–376, Jul. 1985.
[31] T. Ouyang, W. Pedrycz, and N. J. Pizzi, "Rule-based modeling with DBSCAN-based information granules," *IEEE Trans. Cybern.*, vol. 51, no. 7, pp. 3653–3663, Jul. 2021.
[32] J. P. Choi and C. Stefanadis, "Tying, investment, and the dynamic leverage theory," *RAND J. Econ.*, vol. 32, no. 1, pp. 52–71, 2001.
[33] P. A. Laharotte, R. Billot, E. Come, L. Oukhellou, A. Nantes, and N. E. E. Faouzi, "Spatiotemporal analysis of Bluetooth data: Application to a large urban network," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 3, pp. 1439–1448, Jun. 2015.
[34] General Office of the People's Government of Guangdong Province. *Average Salary of Urban Residents in Guangzhou*. Accessed: Aug. 3, 2021. [Online]. Available: http://tjj.gz.gov.cn/tjsj/index.html
[35] Dongguan Bureau of Statistics. *Average Salary of Urban Residents in Dongguan*. Accessed: Aug. 3, 2021. [Online]. Available: http://www.dgtongji.cn/#/detail?code=11109
[36] Shenzhen Bureau of Statistics. *Average Salary of Urban Residents in Shenzhen*. Accessed: Aug. 3, 2021. [Online]. Available: http://tjj.sz.gov.cn/zwgk/zfxxgkml/tjsj/tjgb/content/post_8844535.html
[37] X. Hu, C. Zou, X. Tang, T. Liu, and L. Hu, "Cost-optimal energy management of hybrid electric vehicles using fuel cell/battery health-aware predictive control," *IEEE Trans. Power Electron.*, vol. 35, no. 1, pp. 382–392, Jan. 2020.
[38] J. Gao, X. Li, C. Wang, and X. Huang, "BM-DDPG: An integrated dispatching framework for ride-hailing systems," *IEEE Trans. Intell. Transp. Syst.*, early access, Aug. 30, 2021, doi: 10.1109/TITS.2021.3106243.
[39] IBM. (2021). *IBM ILOG CPLEX Optimization Package*. [Online]. Available: http://www-01.ibm.com/software/integration/optimization/cplex-optimization-studio/
[40] H. Ma, H. Zhu, Z. Hu, W. Tang, and P. Dong, "Multi-valued collaborative QoS prediction for cloud service via time series analysis," *Future Gener. Comput. Syst.*, vol. 68, pp. 275–288, Mar. 2017.
[41] P. Michelucci and J. L. Dickinson, "The power of crowds," *Science*, vol. 351, no. 6268, pp. 32–33, 2016.
[42] E. Briscoe, S. Appling, and J. Schlosser, "Technology futures from passive crowdsourcing," *IEEE Trans. Computat. Social Syst.*, vol. 3, no. 1, pp. 23–31, Mar. 2016.
[43] J. Lu, Y. Xin, Z. Zhang, S. Tang, C. Tang, and S. Wan, "Extortion and cooperation in rating protocol design for competitive crowdsourcing," *IEEE Trans. Computat. Social Syst.*, vol. 8, no. 1, pp. 246–259, Feb. 2021.
[44] Y. Wang, Z. Cai, Z. Zhan, Y. Gong, and X. Tong, "An optimization and auction-based incentive mechanism to maximize social welfare for mobile crowdsourcing," *IEEE Trans. Computat. Social Syst.*, vol. 6, no. 3, pp. 414–429, Jun. 2019.
[45] 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. Computat. Social Syst.*, vol. 7, no. 4, pp. 1033–1046, Aug. 2020.
[46] L. Wang, Z. Yu, D. Zhang, B. Guo, and C. H. Liu, "Heterogeneous multi-task assignment in mobile crowdsensing using spatiotemporal correlation," *IEEE Trans. Mobile Comput.*, vol. 18, no. 1, pp. 84–97, Jan. 2019.
[47] A. Hamrouni, H. Ghazzai, M. Frikha, and Y. Massoud, "A spatial mobile crowdsourcing framework for event reporting," *IEEE Trans. Computat. Social Syst.*, vol. 7, no. 2, pp. 477–491, Apr. 2020.
[48] X. Xu, Q. Liu, X. Zhang, J. Zhang, L. Qi, and W. Dou, "A blockchain-powered crowdsourcing method with privacy preservation in mobile environment," *IEEE Trans. Computat. Social Syst.*, vol. 6, no. 6, pp. 1407–1419, Dec. 2019.
[49] A. Tarable, A. Nordio, E. Leonardi, and M. A. Marsan, "The importance of worker reputation information in microtask-based crowd work systems," *IEEE Trans. Parallel Distrib. Syst.*, vol. 28, no. 2, pp. 558–571, Feb. 2017.
[50] Y. Zheng, J. Wang, G. Li, R. Cheng, and J. Feng, "QASCA: A quality-aware task assignment system for crowdsourcing applications," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, May 2015, pp. 1031–1046.
[51] C. Shan, N. Mamoulis, G. Li, R. Cheng, Z. Huang, and Y. Zheng, "A crowdsourcing framework for collecting tabular data," *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 11, pp. 2060–2074, Nov. 2020.
[52] S. K. Mridha and M. Bhattacharyya, "Introducing collaboration in competitive crowdsourcing markets," *IEEE Intell. Syst.*, vol. 34, no. 1, pp. 23–31, Jan. 2019.
[53] L. Tran-Thanh, T. D. Huynh, A. Rosenfeld, S. D. Ramchurn, and N. R. Jennings, "Crowdsourcing complex workflows under budget constraints," in *Proc. 29th AAAI Conf. Artif. Intell.*, 2015, pp. 1298–1304.
[54] L. Wang, Z. Yu, Q. Han, B. Guo, and H. Xiong, "Multi-objective optimization based allocation of heterogeneous spatial crowdsourcing tasks," *IEEE Trans. Mobile Comput.*, vol. 17, no. 7, pp. 1637–1650, Jul. 2018.
[55] M. Bristow, L. Fang, and K. W. Hipel, "Agent-based modeling of competitive and cooperative behavior under conflict," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 44, no. 7, pp. 834–850, Jul. 2014.
[56] R. N. Almeida and N. David, "Signs of heritage—An agent-based model of the dynamics of heritage categories," *IEEE Trans. Computat. Social Syst.*, vol. 6, no. 6, pp. 1283–1294, Dec. 2019.
[57] R. Mercuur, V. Dignum, and C. M. Jonker, "Integrating social practice theory in agent-based models: A review of theories and agents," *IEEE Trans. Computat. Social Syst.*, vol. 7, no. 5, pp. 1131–1145, Oct. 2020.
[58] A. Trivedi and S. Rao, "Agent-based modeling of emergency evacuations considering human panic behavior," *IEEE Trans. Computat. Social Syst.*, vol. 5, no. 1, pp. 277–288, Mar. 2018.
[59] J. Xu, H. Li, Y. Li, D. Yang, and T. Li, "Incentivizing the biased requesters: Truthful task assignment mechanisms in crowdsourcing," in *Proc. 14th Annu. IEEE Int. Conf. Sens., Commun., Netw. (SECON)*, Jun. 2017, pp. 1–9.
[60] W. Xiong, C. Pan, Y. Qiao, N. Wu, M. Chen, and P. Hsieh, "Reducing wafer delay time by robot idle time regulation for single-arm cluster tools," *IEEE Trans. Autom. Sci. Eng.*, vol. 18, no. 4, pp. 1653–1667, Oct. 2021.
[61] D. Deng, C. Shahabi, and U. Demiryurek, "Maximizing the number of worker's self-selected tasks in spatial crowdsourcing," in *Proc. 21st ACM SIGSPATIAL Int. Conf. Adv. Geograph. Inf. Syst.*, Nov. 2013, pp. 324–333.
[62] P. Mavridis, D. Gross-Amblard, and Z. Miklós, "Using hierarchical skills for optimized task assignment in knowledge-intensive crowdsourcing," in *Proc. 25th Int. Conf. World Wide Web*, Apr. 2016, pp. 843–853.
[63] W. Dai, Y. Wang, J. Ma, and Q. Jin, "BTR: A feature-based Bayesian task recommendation scheme for crowdsourcing system," *IEEE Trans. Computat. Social Syst.*, vol. 7, no. 3, pp. 780–789, Jun. 2020.
[64] D. Liu, B. Huang, and H. Zhu, "Solving the tree-structured task allocation problem via group multirole assignment," *IEEE Trans. Autom. Sci. Eng.*, vol. 17, no. 1, pp. 41–55, Jan. 2020.
[65] Q. Jiang, H. Zhu, Y. Qiao, Z. He, D. Liu, and B. Huang, "Agent evaluation in deployment of multi-SUAVs for communication recovery," *IEEE Trans. Syst., Man, Cybern. Syst.*, early access, Dec. 7, 2021, doi: 10.1109/TSMC.2021.3130161.
[66] Q. Jiang, H. Zhu, Y. Qiao, D. Liu, and B. Huang, "Refugee resettlement by extending group multirole assignment," *IEEE Trans. Computat. Social Syst.*, early access, Dec. 14, 2021, doi: 10.1109/TCSS.2021.3128282.
[67] P. Zhang, Y. He, and C. Shi, "Transshipment and coordination in a two-echelon supply chain," *RAIRO, Oper. Res.*, vol. 51, no. 3, pp. 729–747, Jul. 2017.
[68] W. Liu, Y. Liang, O. Tang, V. Shi, and X. Liu, "Cooperate or not? Strategic analysis of platform interactions considering market power and precision marketing," *Transp. Res. E, Logistics Transp. Rev.*, vol. 154, Oct. 2021, Art. no. 102479.

Qian Jiang (Graduate Student Member, IEEE) was born in January 1997. He received the B.S. degree in network engineering and the M.S. degree in computer science and technology from the School of Computer Science and Technology, Guangdong University of Technology, Guangzhou, China, in 2018 and 2021, respectively. He is currently pursuing the Ph.D. degree in intelligent science and system with the Institute of Systems Engineering, Macau University of Science and Technology, Taipa, Macau, China.

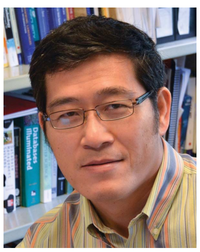
His research interests include human–machine systems, computational social simulation, and scheduling and optimization.

Mr. Jiang has reported a conference paper as the first author in the Chinese Computer-Supported Cooperative Work (CSCW) 2019. He has published a regular article on *International Journal of Cooperative Information Systems* in 2020. Recently, he has also published two regular articles on IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: SYSTEMS and IEEE TRANSACTIONS ON COMPUTATIONAL SOCIAL SYSTEMS. He has served as a Reviewer for IEEE TRANSACTIONS ON COMPUTATIONAL SOCIAL SYSTEMS.

Dongning Liu (Member, IEEE) was born in January 1979. He received the Ph.D. degree in logic from Sun Yat-sen University, Guangzhou, China, in 2007.

He has been a Full Professor with the Guangdong University of Technology, Guangzhou, since 2009. He is responsible for teaching artificial intelligent logic and discrete math at the School of Computer Science and Technology. He has been engaged in education and technology transfer on collaborative computing. He is also the Vice Dean of the School of Computer Science and Technology. He was a Post-Doctoral Fellow in mathematics at Sun Yat-sen University from 2007 to 2009. He was a Visiting Professor with Nipissing University, North Bay, ON, Canada, from 2015 to 2016. He has published more than 50 articles on computer magazines and international conferences.

Dr. Liu is a member of IEEE Systems, Man and Cybernetics (SMC) Society and a Senior Member of the China Computer Federation (CCF) Society. He has been serving as a TC Member for the Technical Committee of Distributed Intelligent Systems and the Vice Secretary-General and a Standing Committee Member for Technical Committee on Cooperative Computing of China Computer Federation. He is also a Reviewer of IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: SYSTEMS, IEEE TRANSACTIONS ON COMPUTATIONAL SOCIAL SYSTEMS, and IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY.



Haibin Zhu (Senior Member, IEEE) received the B.S. degree in computer engineering from the Institute of Engineering and Technology, Zhengzhou, China, in 1983, and the M.S. and Ph.D. degrees in computer science from the National University of Defense Technology (NUDT), Changsha, China, in 1988 and 1997, respectively.

He is currently a Full Professor and the Coordinator of the Computer Science Program and the Founding Director of the Collaborative Systems Laboratory, Nipissing University, North Bay, ON, Canada. He was a Visiting Professor and a Special Lecturer with the College of Computing Sciences, New Jersey Institute of Technology, Newark, NJ, USA, from 1999 to 2002; and a Lecturer, an Associate Professor, and a Full Professor with NUDT from 1988 to 2000. He has accomplished (published or in press) more than 200 research works including more than 30 IEEE TRANSACTIONS articles, six books, five book chapters, three journal issues, and three conference proceedings. His research interests include collaboration theory, technologies, systems, and applications, human–machine systems, computational social simulation, collective intelligence, multi-agent systems, software engineering, and distributed intelligent systems.

Dr. Zhu is a fellow of the Institute of Cognitive Informatics and Cognitive Computing (ICIC), a Senior Member of ACM, a Full Member of Sigma Xi, and a Life Member of the Chinese Association of Science and Technology (CAST), USA. He also served as a PC member for more than 100 academic conferences. He has offered more than 70 invited talks including keynote and plenary speeches on related topics internationally, for example, Australia, Canada, China, Germany, Hong Kong, Macau, Singapore, Turkey, U.K., and USA. His research was sponsored by the Natural Sciences and Engineering Research Council of Canada (NSERC), International Business Machines Corporation (IBM), DeNitrification DeComposition (DNDC), Defence Research and Development Canada (DRDC), and Overseas Private Investment Corporation (OPIC). He was a recipient of the meritorious service award from IEEE SMC Society in 2018; the Chancellor's Award for Excellence in Research in 2011 and Two Research Achievement Awards from Nipissing University in 2006 and 2012; the IBM Eclipse Innovation Grant Awards in 2004 and 2005; the Best Paper Award from the 11th ISPE International Conference on Concurrent Engineering (ISPE/CE2004); the Educator's Fellowship of OOPSLA 2003; the Second Class National Award for Education Achievement in 1997; and three First Class Ministerial Research Achievement Awards from China in 1991, 1994, and 1997. He has been serving as the Associate Vice President (AVP) of Systems Science and Engineering (SSE); the Co-Chair of the Technical Committee of Distributed Intelligent Systems; and a member of the SSE Technical Activity Committee, the Conferences and Meetings Committee, and the Electronic Communications Subcommittee of IEEE Systems, Man and Cybernetics (SMC) Society. He has been an Active Organizer of the annual IEEE International Conference on SMC since 2003, the Social Media Co-Chair since 2015, the Web Co-Chair since 2015, the Area Co-Chair since 2017, the Tutorial Chair since 2018, the Special Session Chair since 2019, the Co-Chair since 2020, the Poster Co-Chair since 2020, the Registration Co-Chair since 2021, and was the Session Chair from 2003 to 2020 and the Special Session Organizer from 2003 to 2020. He was the Program Co-Chair of the 13th International Conference on Computer Science and Information Technology, October 2020, Online (ICCSIT2020) and the 10th International Conference on Pervasive and Parallel Computing, Communication, and Sensors, November 2020, Online (PECCS2020); the Publication Chair for the 1st IEEE International Conference of Human–Machine Systems, September 2020 (online); and the Program Chair of 16th IEEE International Conference on Networking, Sensing and Control, Banff, AB, Canada, May 2019. He is the PC Chair of the 24th IEEE International Conference on Computer Supported Cooperative Work in Design, Dalian, China, May 2020, and was the PC Chair of 17th IEEE International Conference on Computer Supported Cooperative Work in Design, Whistler, BC, Canada, June 2013. He has been serving as an Associate Editor (AE) for IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: SYSTEMS, IEEE TRANSACTIONS ON COMPUTATIONAL SOCIAL SYSTEMS, *IEEE Systems, Man, and Cybernetics Magazine*, and *IEEE Canada Review*.

Yan Qiao (Senior Member, IEEE) received the B.S. and Ph.D. degrees in industrial engineering and mechanical engineering from the Guangdong University of Technology, Guangzhou, China, in 2009 and 2015, respectively.

From September 2014 to September 2015, he was a Visiting Student with the Department of Electrical and Computer Engineering, New Jersey Institute of Technology, Newark, NJ, USA. From January 2016 to December 2017, he was a Post-Doctoral Research Associate with the Institute of Systems Engineering, Macau University of Science and Technology, Taipa, Macau, China, where he has been an Assistant Professor since January 2018. He has more than 80 publications, including one book chapter and more than 30 regular articles in IEEE TRANSACTIONS. His research interests include scheduling and optimization, semiconductor manufacturing systems, and smart manufacturing.

Dr. Qiao was a recipient of the Qualtech Systems Inc. (QSI) Best Application Paper Award Finalist of 2011 IEEE International Conference on Automation Science and Engineering, the Best Student Paper Award of 2012 IEEE International Conference on Networking, Sensing and Control, the Best Conference Paper Award Finalist of 2016 IEEE International Conference on Automation Science and Engineering, the Best Student Paper Award Finalist of 2020 IEEE International Conference on Automation Science and Engineering, and the 2021 Hsue-Shen Tsien Paper Award of IEEE/CAA JOURNAL OF AUTOMATICA SINICA. He has served as a reviewer for a number of journals.

Baoying Huang (Graduate Student Member, IEEE) received the B.S. degree in network engineering and the M.S. degree in computer science and technology from the School of Computer Science and Technology, Guangdong University of Technology, Guangzhou, China, in 2016 and 2019, respectively. She is currently pursuing the Ph.D. degree in intelligent science and system with the Institute of Systems Engineering, Macau University of Science and Technology, Taipa, Macau, China.

Her research interests include scheduling and optimization, human–machine systems, computational social simulation, and semiconductor manufacturing systems.

Dr. Huang has published four regular articles in IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: SYSTEMS, IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING, and IEEE TRANSACTIONS ON COMPUTATIONAL SOCIAL SYSTEMS. She has served as a reviewer for a number of journals and conferences.