Group Multi-Role Assignment With Conflicting Roles and Agents

Haibin Zhu, *Senior Member, IEEE*

***Abstract*—Group role assignment (GRA) is originally a complex problem in role-based collaboration (RBC). The solution to GRA provides modelling techniques for more complex problems. GRA with constraints (GRA$^+$) is categorized as a class of complex assignment problems. At present, there are few generally efficient solutions to this category of problems. Each special problem case requires a specific solution. Group multi-role assignment (GMRA) and GRA with conflicting agents on roles (GRACAR) are two problem cases in GRA$^+$. The contributions of this paper include: 1) The formalization of a new problem of GRA$^+$, called group multi-role assignment with conflicting roles and agents (GMAC), which is an extension to the combination of GMRA and GRACAR; 2) A practical solution based on an optimization platform; 3) A sufficient condition, used in planning, for solving GMAC problems; and 4) A clear presentation of the benefits in avoiding conflicts when dealing with GMAC. The proposed methods are verified by experiments, simulations, proofs and analysis.**

***Index Terms*—Agents, environments-classes, agents, roles, groups, and objects (E-CARGO) model, group multi-role assignment (GMRA), group role assignment with constraints (GRA$^+$), roles, role-based collaboration (RBC).**

I. Introduction

CONFLICTS are major reasons why collaborative efforts can become inefficient and ineffective. Such conflicts can result in verbal arguments, quarrels, emotional disappointment, and even physical confrontation [1]–[4]. Resource sharing and personal interaction can be sources of conflict. This paper discusses a typical role assignment problem considering possible conflicts. The proposed method aims at carefully designed assignment strategies and policies to avoid potential conflicts.

Role-based collaboration (RBC) [5] has been investigated for almost two decades and verified as a promising way to deal with collaboration problems. The environments-classes, agents, roles, groups, and objects (E-CARGO) model is designed as the fundamental model of RBC, and has been proved possessing significant power of modelling. RBC and E-CARGO are confirmed to be applicable in dealing with real-world problems.

In support of this claim, we have examined, formalized and solved many assignment cases that extend general assignment problems [6]. For example, we have formalized and solved the problems as follows (Fig. 1):

1) Group role assignment (GRA) [7];
2) GRA with constraints (GRA$^+$) including:
   i) GRA with conflicting agents on roles (GRACAR) [8];
   ii) Group multi-role assignment (GMRA) [9], 10];
   iii) GRA with conflict and cooperation factors [11];
   iv) GRA with balance between preferences and performance [12]; and
   v) Tree-structured task allocation [13];
3) GRA with multiple objectives (GRA^{++}) including:
   i) GRA with budget constraints [14];
   ii) Good at many and expert one [15];
   iii) GRA with agents' busyness degrees [16];
   iv) GRA with agents' preferences [17]; and
   v) Most economic redundant assignment [18].

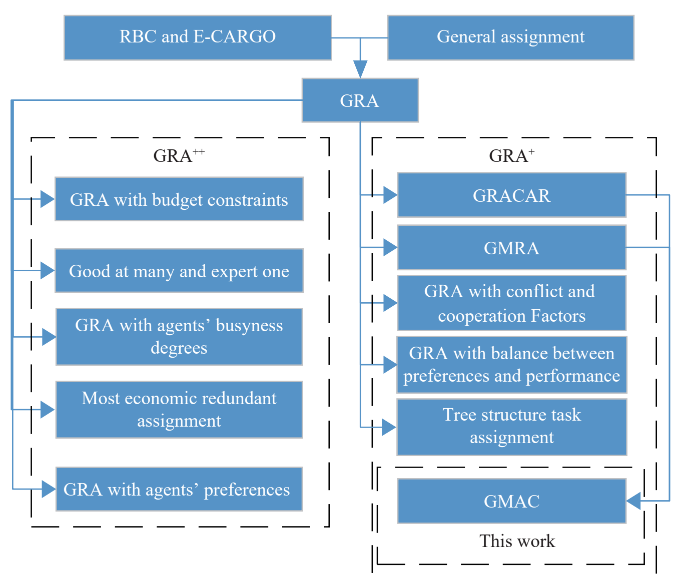

Fig. 1. The roadmap of this work.

These efforts provide solutions to real-world problems. They also broaden the scope of operational research [19] into the development of additional specific and efficient algorithms. It is important to note that not all GRA problems can be expressed by integer linear programming (ILP) [6],

Manuscript received April 14, 2020; revised May 27, 2020; accepted June 18, 2020. This work was supported in part by Natural Sciences and Engineering Research Council, Canada (NSERC) (RGPIN-2018-04818) and the funding from the Innovation for Defence Excellence and Security (IDEaS) Program from the Canadian Department of National Defence (DND). Recommended by Associate Editor MengChu Zhou.

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

H. B. Zhu is with the Department of Computer Science and Mathematics, Nipissing University, North Bay, Ontario, P1B 8L7, Canada (e-mail: haibinz@nipissingu.ca).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JAS.2020.1003354

[19]–[24] due to the extended scope of assignment problems (Fig. 2). From Fig. 2, it can be seen that a fundamental way to handle GRA, which includes GRA^+ and GRA^{++}, is a transformation into ILP problems and then a solution through using an optimization platform.

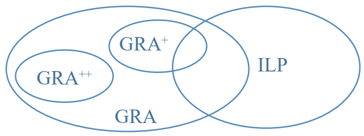

Fig. 2. Overlap of GRA with ILP.

This paper discusses a new type of GMRA problem. It introduces two key conflict types, i.e., agent conflicts and role conflicts, and is called group multi-role assignment with conflicting roles and agents (GMAC). GMAC differs from the previous work [5], [7]–[18] in that the solutions to the previous GRA problems cannot be applied to this situation without paying a significant effort. The discovery route to the GMAC problem is illustrated in Fig. 1.

Although the proposed problem is investigated from a theoretical and abstract way, it maps many industrial problems. Section II presents one case. Another common example is scheduling persons driving trucks, i.e., agent (*driver*, *truck*), is scheduled for a city on a day, i.e., role (*city, day*). Agent conflict happens because one person cannot drive two different trucks at the same time, i.e., agent ($driver_1$, $truck_1$) and agent ($driver_1$, $truck_2$) are in conflict. At the same time, role conflict occurs because a truck cannot go to different cities on the same day due to long distances, i.e., role ($city_1$, day_1) and role ($city_2$, day_1) are in conflict. Therefore, the solution to the proposed problem is significant.

This paper assumes that the human power is not enough to conduct GRA, which allows each person to take at most one role. This assumption is acceptable when GRA has no solutions due to insufficient staff. With this consideration, there are many variations when we conduct group multi-role assignment (GMRA). GMRA in fact opens a scope to manage a team to avoid future cooperation problems in many different aspects. If we compare general role assignment (GA), group role assignment (GRA), and GMRA, we may find that the introduction of L and L^a (Section III) can be used to model more complex team situations. Solutions to these problems make unique contributions to the literature definitely.

The remainder of this paper is outlined as follows. Section II describes a real-world scenario. Section III specifies the elements used in formalization with E-CARGO and defines the proposed problem, i.e., GMAC. Section IV illustrates experiments on the GMAC problem with the IBM ILOG CPLEX optimization package (CPLEX) and provides efficiency analysis. Section V presents a sufficient condition for GMAC to be feasible. Section VI illustrates the benefits of avoiding the aforementioned conflicts in GMAC. Section VII outlines related work. Conclusions and suggestions for future research topics appear in Section VIII. A nomenclature is added in the Appendix to ease reading.

II. A Real-World Scenario

In company X, Adam, the chief executive officer (CEO), wins a bid for a software development project. He seeks the assistance of Christine, the human resources (HR) officer by asking her to establish an appropriate team from among the company employees. Christine composes a task list (Table I) for the team and a list of candidates (the leftmost column in Table II). Next, she uses routine criteria to evaluate the qualification of each employee for specific project tasks (Table II). Table III informs that more than one task can be assigned to one employee due to staffing shortage but such numbers should be limited to guarantee quality. Past performance is used to set the task limits for each team member. Based on Tables I–III, Christine can assign the tasks to candidates shown in Table II (numbers in the parentheses) by using the GMRA algorithm [10] to maximize the sum of all the assigned qualification values and satisfy all task requirements given by Tables I and III.

TABLE I
REQUIRED POSITIONS

Position	Project manager	Analyst	Designer	Coder	Tester	Client service
Required number	1	2	2	4	2	2

The assumption of Table II is valid and has its rationale. For example, if the roles are courses to take, C++, operating systems, data structure, and networking, and an agent (student) is evaluated by his performance on each course, we may acquire the numbers of 85%, 78%, 83%, and 91%, i.e., 0.85, 0.78, 0.83, and 0.91. Therefore, the sum of all the qualification values of an agent can be larger than 1.

Table III assumes the number of tasks for each person to take at most. Therefore, they are supposed of integers. The setting of these numbers should be based on historical data and each person's past performance. It is reasonable because different people have different energy, capacity, and personality; and some may deal with many tasks well but some others may not.

To guarantee overall project quality, Adam reminds Christine to address any conflicts from task assignments among team members. Conflicting tasks cannot be assigned to the same person. Conflicting persons cannot be assigned to the same task. On this basis, Christine creates Tables IV and V.

Faced with these new requirements, Christine encounters a new challenge that is more complex than GRMA [10] and GRACAR [8]. She requests, and receives, additional time to establish an acceptable task assignment. Now, Christine has to deal with the GMAC problem before the extended deadline.

III. E-CARGO Model and Related Problems

An RBC system Σ can be depicted as a 9-tuple $\Sigma ::= < \mathcal{C}, \mathcal{O}, \mathcal{A}, \mathcal{M}, \mathcal{R}, \mathcal{E}, \mathcal{G}, s_0, \mathcal{H} >$, where $\mathcal{E}$, $\mathcal{C}$, $\mathcal{A}$, $\mathcal{R}$, $\mathcal{G}$, $\mathcal{O}$, $\mathcal{M}$, and $\mathcal{H}$ are sets of E-CARGO components, i.e., environments, classes, agents, roles, groups, objects, messages, and human users. The system's initial state is denoted by s_0.

Following the formalization of E-CARGO, $\mathcal{N}$ is used to

TABLE II
THE QUALIFICATIONS

Candidate	Position					
	Project manager	Analyst	Designer	Coder	Tester	Client service
Ann	0.18	0.82	(0.75)	**0.20**	0.50	0.40
Bob	0.35	**(0.80)**	0.58	0.35	**(0.70)**	0.55
Chris	0.84	**(0.85)**	**(0.86)**	**(0.90)**	0.65	0.35
Doug	0.96	0.51	0.45	(0.64)	**0.85**	**(0.80)**
Ed	0.22	0.33	**0.68**	**(0.33)**	(0.78)	0.60
Fred	**(0.96)**	0.50	0.10	**(0.73)**	0.65	**(0.85)**

TABLE III
ALLOWED NUMBER OF ASSIGNED TASKS

Candidate	Ann	Bob	Chris	Doug	Ed	Fred
Limit number	1	2	3	2	2	3

TABLE IV
AGENT CONFLICTS

Candidate	Ann	Bob	Chris	Doug	Ed	Fred
Ann	0	1	0	0	0	0
Bob	1	0	0	0	1	0
Chris	0	0	0	1	0	0
Doug	0	0	1	0	0	0
Ed	0	1	0	0	0	0
Fred	0	0	0	0	0	0

denote the set of non-negative integers, i.e., $\{0, 1, 2, 3, \ldots\}$, m ($=|\mathcal{A}|$) the size of the agent set $\mathcal{A}$, n ($= |\mathcal{R}|$) the size of the role set $\mathcal{R}$. To simplify descriptions, we suppose that $\mathcal{A}$, $\mathcal{R}$, and other derived sets are ordered sets, i, i_1, i_2, …, the indices of agents, and j, j_1, j_2, …, the indices of roles. Agents, roles, groups, and assignments are concentrated in this paper. The *Environment* component of the E-CARGO model is expressed by several symbols and mathematical structures.

Definition 1 [5], [7]: For group g, a *role (agent) assignment* is a tuple $<a, r>$ of $g.\mathcal{J}$.

Definition 2 [5], [7]: L is *the lower role range vector* of group g to express the number of agents required for a role, i.e., $L[j] = g.e.\mathcal{B}[j].q.l$, $L[j] \in \mathcal{N}$, $0 \le j < n$.

Definition 3 [5], [7]: Q is the *qualification matrix* for a group to express agent suitability for assignment to a role, i.e., $Q[i, j] \in [0, 1]$ indicates agent i's qualification value for role j ($0 \le i < m$, $0 \le j < n$), 0 means the lowest and 1 the highest.

Definition 4 [5], [7]: T is the matrix expression of $g.\mathcal{J}$, i.e., $T[i, j] = 1$ means that agent i is assigned to role j (i.e., $<a_i, r_j> \in g.\mathcal{J}$) and $T[i, j] = 0$ means the opposite (i.e., $<a_i,r_j> \notin g.\mathcal{J}$).

Definition 5 [5], [7]: The *performance of a* group σ is defined as the sum of the assigned agents' qualifications, i.e., $\sigma = \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} Q[i, j] \times T[i, j]$.

Definition 6 [5], [7]: Role j is *workable* if it is assigned with sufficient agents, i.e., $\sum_{i=0}^{m-1} T[i, j] \ge L[j]$ $(0 \le j < n)$.

Definition 7 [5], [7]: We say T is *workable* if each role j is workable, i.e., $\sum_{i=0}^{m-1} T[i, j] \ge L[j]$ $(0 \le j < n)$. Group g is *workable* if T is workable.

Definition 8 [10]: Agent ability vector L^a expresses the allowable number of role assignments per agent, i.e., $L^a[i] \in \mathcal{N}$ $(0 \le i < m)$. That is to say, agent i can be assigned to a maximum of $L^a[i]$ roles.

Note that, L^a can be formed by extracting information from agents' properties in the original E-CARGO model.

Definition 9 [10]: Given $\mathcal{A}$ ($|\mathcal{A}| = m$), $\mathcal{R}$ ($|\mathcal{R}| = n$), Q, L, and L^a, *GMRA* is to find T to obtain

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

s.t.

$$T[i, j] \in \{0, 1\} (0 \le i < m, 0 \le j < n) \tag{1}$$

$$\sum_{i=0}^{m-1} T[i, j] = L[j] (0 \le j < n) \tag{2}$$

$$\sum_{j=0}^{n-1} T[i, j] \le L^a[i] (0 \le i < m) \tag{3}$$

where (1) is a 0–1 constraint; (2) means that each role should be workable; and (3) indicates role assignment limits for each agent.

We use T^* to represent the feasible T that satisfies Definition 9. We then obtain the optimum group performance of GMRA $\sigma^* = \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} Q[i, j] \times T^*[i, j]$.

By solving the above GMRA problem, we establish an assignment scheme shown in Table II (numbers in parentheses), where the obtained group performance is 9.95.

To understand the differences between GA, GRA, and GMRA, we need to restate the GA and GRA problems.

Definition 10: Given $\mathcal{A}$ ($|\mathcal{A}| = m$), $\mathcal{R}$ ($|\mathcal{R}| = n$), $n = m$, and Q, *GA* is to find T to obtain

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

s.t. (1), and

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

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

where (4) means that each role should be assigned to one agent; and (3) indicates that each agent is to be assigned only

TABLE V
ROLE CONFLICTS

Position	Project manager	Analyst	Designer	Coder	Tester	Client service
Project manager	0	0	0	1	0	0
Analyst	0	0	0	0	0	0
Designer	0	0	0	0	0	0
Coder	1	0	0	0	1	0
Tester	0	0	0	1	0	0
Client service	0	0	0	0	0	0

one role.

Definition 11 [7]: Given m, n, Q, *GRA* is to find T to obtain

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

s.t. (1), (2), and

$$\sum_{i=0}^{n-1} T[i,j] \le 1\,(0 \le i < m) \tag{6}$$

where (6) means that one agent is assigned, at most, one role.

Compared with GA, GRA raises the possibility of conflicting agents, because one role may be assigned to multiple agents. Compared with GA and GRA, GMRA raises the possibility of conflicting roles, because one agent may be assigned to multiple roles.

We now need to introduce two new structures in order to formalize GMAC.

Definition 12 [8]: An *agent conflict matrix* is an $m\times m$ matrix A^c ($A^c[i_1, i_2]\in\{0,1\}$, $0 \le i_1, i_2 < m$), where $A^c[i_1, i_2] = 1$ indicates that agents i_1 and i_2 are in conflict. 0 means no conflict. We define $A^c[i, i] = 0$ ($0 \le i < m$) to mean that an agent has no conflict with itself.

With the introduction of A^c, we may define the GRACAR problem [8].

Definition 13 [8]: Given $\mathcal{A}$ ($|\mathcal{A}| = m$), $\mathcal{R}$ ($|\mathcal{R}| = n$), Q, L, and A^c, *GRACAR* is to find T to obtain

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

s.t. (1), (2), (6), and

$$\begin{gathered} A^c[i_1,i_2]\times(T[i_1,j]+T[i_2,j]) \le 1 \\ (0 \le i_1 \ne i_2 < m)\ (0 \le j < n) \end{gathered} \tag{7}$$

where (7) means no conflict due to agents being assigned to the same role.

Definition 14: A *role conflict matrix* is an $n\times n$ matrix R^c ($R^c[j_1, j_2]\in\{0,1\}$, $0 \le j_1, j_2 < n$), where $R^c[j_1, j_2] = 1$ expresses that roles j_1 and j_2 are in conflict while 0 means no conflict. We define $R^c[j, j] = 0$ ($0 \le j < n$).

Evidently, A^c and R^c are symmetric along the diagonal from $A^c[0, 0]$ to $A^c[m–1, m–1]$ and $R^c[0, 0]$ to $R^c[n–1, n–1]$, i.e., ($A^c[i_1, i_2] = A^c[i_2, i_1]$) ($0 \le i_1, i_2 < m, i_1 \ne i_2$) and ($R^c[j_1, j_2] = R^c[j_2, j_1]$) ($0 \le j_1, j_2 < n, j_1 \ne j_2$).

Definition 15: Given $\mathcal{A}$ ($|\mathcal{A}| = m$), $\mathcal{R}$ ($|\mathcal{R}| = n$), Q, L, L^a, A^c, and R^c, ***GMAC*** is to find a workable T to

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

s.t. (1)–(3), (7) and

$$\begin{gathered} R^c[j_1,j_2]\times(T[i,j_1]+T[i,j_2]) \le 1 \\ (0 \le i < m)\ (0 \le j_1 \ne j_2 < n) \end{gathered} \tag{8}$$

where (8) requires that conflicting roles are not assigned to the same agent.

We use T^*_c and σ^*_c to mean the assignment matrix and the group performance obtained by GMAC, i.e.,

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

After the problem is formalized, we can solve it using the CPLEX [25].

By solving the GMAC problem, we obtain a new assignment scheme shown in Table II (bold numbers). Although the group performance is 9.51 and lower than that of GMRA, i.e., 9.95, this scheme avoids conflicts. Based on GRMA assignment, underlined values in Table II indicate three pairs of conflicts: Bob and Ed, Chris and Doug on role *Coder*, and *Coder* and *Tester* for agent Ed. Considering the involved qualification values, 0.9, 0.64, 0.33, 0.70, and 0.78, which are 3.35 in total. If these conflicts cause a performance decrease of 15%, actual group performance for GMRA becomes 9.95 – 3.35 × 15% = 9.95 – 0.5025 = 9.4475, which is less than the new group performance value of 9.51. If the conflicts are not severe and lead to a performance decrease of only 10%, then GMRA performs better than GMAC.

To assist decision makers to well plan and organize a team before solving the GMAC problem, we need to state a necessary condition.

Theorem 1: A necessary condition for GMAC to have a feasible solution is that

$$\sum_{j=0}^{n-1} L[j] \le \sum_{i=0}^{m-1} L^a[i]. \tag{9}$$

Proof: Note that (9) is a necessary condition of the corresponding GMRA problem [10], which does not consider constraints (7) and (8), to have a feasible solution.

Therefore, (9) is also a necessary condition for GMAC. ■

IV. Experiments

To verify the performance of the CPLEX solution to GMAC, we implement a Java program based on the CPLEX package [25]. The platform is shown in Table VI.

The first experiment is conducted to establish the trend of consumed time vs. the increase of problem scales. In each step we test for 100 random cases. In each case, Q, L, L^a, A^c, and R^c are randomly generated (uniform distributions) based on

TABLE VI
EXPERIMENT PLATFORM CONFIGURATION

Hardware	
CPU	Intel core i5-8350U CPU @1.7 GHz 1.9 GHz
MM	8 GB
Software	
OS	Windows 10 Pro, 64-bit OS
Eclipse	Version: 2018-12 (4.10.0)
JDK	Java version "1.8.0" Java (TM) SE Runtime Environment

the designated conflict rate $p_{ca} = p_{cr} = 30\%$ (see Definitions 19 and 20), where m varies from 20 to 200 in increments of 20, and $m = n$.

1) $Q[i,j] \in [0, 1]$ $(0 \le i < m, 0 \le j < n)$, i.e., the qualification value of each agent for each role;

2) $L[j]$ $(0 \le j < n)$, $1 \le L[j] \le 3$, and

3) $L^a[i]$ $(0 \le i < m)$, $1 \le L^a[i] \le 3$.

Unfortunately, we could not obtain the results after eight hours. From the experience, CPLEX uses much time in searching for problem solutions where none exists. To address the matter, we apply the necessary condition for GMAC to have a feasible solution (Theorem 1).

In the next experiment, we added one step to check whether the problem case satisfies (9). If the necessary condition is not satisfied, we do not start the CPLEX search. Fig. 3 shows the experiment results, where we only show the maximum and average time, because the minimum time is always 0 when not satisfying (9). The feasible rates for the random cases are shown in Table VII.

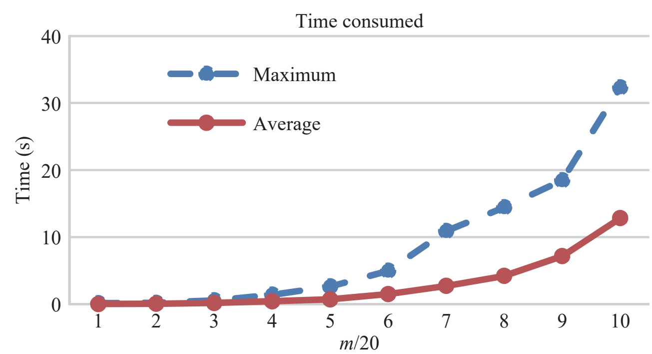

Fig. 3. Time consumed by problems in different scales.

TABLE VII
FEASIBLE RATES FOR DIFFERENT SCALES

Scale m	20	40	60	80	100	120	140	160	180	200
Feasible rate (%)	44	45	43	45	38	43	42	41	46	54

In the third experiment, using the same settings from the second one, we conduct simulations (Fig. 4) by guaranteeing that each case satisfies the necessary condition, i.e., (9). We notice that the maximum consumed time is similar to the second experiment but the average and minimum times increase due to the searching efforts of CPLEX. A side effect of applying the necessary condition is that all 2000 cases have feasible solutions. This experiment also indicates that the CPLEX based solutions are practical for solving problems with up to 200 agents, or up to 400 virtual agents, with respect to the L setting.

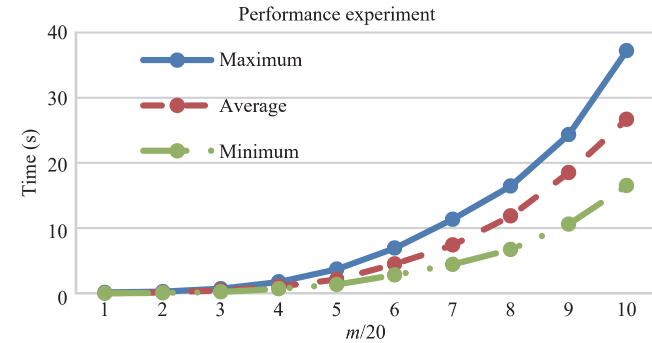

Fig. 4. Time consumed by problems in different scales (necessary conditions guaranteed).

V. FEASIBILITY OF GMAC

To show the performance distribution of problems, regardless of solution feasibility, the 4th experiment is set as

1) $m = 20$, $n = 20$, $p_{ca} = p_{cr} = 50\%$;

2) $L[j]$ $(0 \le j < n)$, $1 \le L[j] \le 5$;

3) $L^a[i]$ $(0 \le i < m)$, $1 \le L^a[i] \le 5$ and make sure $\sum_{j=0}^{n-1} L[j] \le \sum_{i=0}^{m-1} L^a[i]$, and other values remain unchanged.

Fig. 5 shows the time used by 100 different cases. In the figure, we put all the problems not satisfying (9) on the left and the others on the right. It suggests that CPLEX deals with problems evenly and reports optimal solution results regardless of feasibility. Such a situation wastes a lot of time.

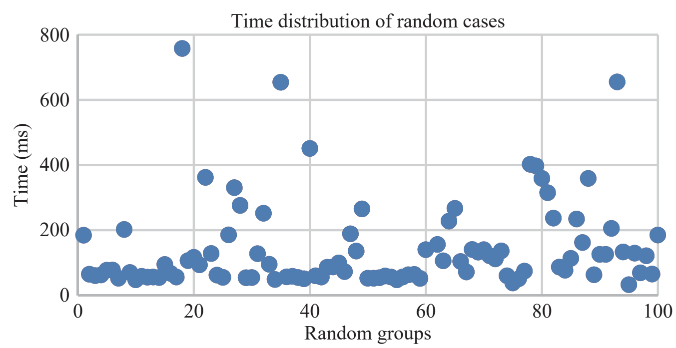

Fig. 5. Distribution of time consumed for not satisfying (9) cases (left half) and the other cases (right half).

If problem feasibility can be predetermined, time can be saved when using CPLEX to obtain the optimal assignment. Following this line of investigation, a sufficient condition for GMAC can be established.

Definition 16: Given $\mathcal{A}$ $(|\mathcal{A}| = m)$, $\mathcal{R}$ $(|\mathcal{R}| = n)$, Q, L, L^a, A^c, and R^c, a simplified GMAC (SGMAC) problem is to determine assignment T' to

$$\max \sigma = \sum_{i'=0}^{m'-1} \sum_{j'=0}^{n'-1} Q'[i', j'] \times T'[i', j']$$

s.t.

$$\mathcal{A}' = \left\{a_{i'} | a_{i'} = \{a_i\} \cup \left\{a_{i_1} \mid a_{i_1} \in \mathcal{A}, A^c[i, i_1] = 1\right\}\right\} \tag{10}$$

$$\mathcal{R}' = \{r_{j'} | r_{j'} = \{r_j\} \cup \{r_{j_1} \mid r_{j_1} \in \mathcal{R}, R^c[j, j_1] = 1\}\} \tag{11}$$

$$m' = |\mathcal{A}'| \tag{12}$$

$$n' = |\mathcal{R}'| \tag{13}$$

$$Q'[i', j'] = \begin{cases} Q[i', j'] \ (a_{i'} \in \mathcal{A}, r_{j'} \in \mathcal{R}) \\ \max\{Q[i_1, j'] | a_{i1} \in a_{i'}\} \ (r_{j'} \in \mathcal{R}) \\ \max\{Q[i', j_1] | r_{j_1} \in r_{j'}\} (a_{i'} \in \mathcal{A}) \\ \max\{Q[i_1, j_1] | a_{i1} \in a_{i'}, r_{j_1} \in r_{j'}\} (\neg a_{i'} \in \mathcal{A}, \neg r_{j'} \in \mathcal{R}) \end{cases}$$
$$(0 \le i' < m', 0 \le j' < n') \tag{14}$$

$$L'[j'] = \sum\nolimits_{r_j \in r_{j'}} L[j] (0 \le j' < n') \tag{15}$$

$$L^{a'}[i'] = \min\left(\sum\nolimits_{a_i \in a_{i'}} L^a[i], n\right) (0 \le i' < m') \tag{16}$$

$$T'[i', j'] \in \{0, 1\} (0 \le i' < m', 0 \le j' < n') \tag{17}$$

$$\sum_{i'=0}^{m'-1} T'[i', j'] = L'[j'] \ (0 \le j' < n') \tag{18}$$

$$\sum_{j'=0}^{n'-1} T'[i', j'] \le L^{a'}[i'] \ (0 \le i < m') \tag{19}$$

where (10)–(16) provide a way to form the parameters of the new problem, and (17) and (18) apply the constraints of GMRA to the new problem.

Evidently, an SGMAC problem is a GMRA problem. Certainly, SGMAC is not equivalent to GMAC. However, it presents hints for establishing the sufficient condition for GMAC to have a feasible solution.

We provide Definition 16 to simplify the GMAC problem in order to quickly determine if GMAC has a feasible solution. From the definition, if we transfer GMAC to SGMAC, we need combining. If we need to get a feasible solution to GMAC from that to SGMAC, which is not definitely an optimum solution to GMAC, we need splitting.

Lemma 1: SGMAC is a GMRA problem.

Proof: We use replacement.

If we replace $\mathcal{A}'$ ($|\mathcal{A}'| = m'$), $\mathcal{R}'$ ($|\mathcal{R}'| = n'$), Q', L', and $L^{a'}$ by $\mathcal{A}$ ($|\mathcal{A}| = m$), $\mathcal{R}$ ($|\mathcal{R}| = n$), Q, L, and L^a, respectively. SGMAC follows all the expressions in Definition 9. ■

Theorem 2: GMAC has a feasible solution if SGMAC has a feasible solution.

Proof:

Suppose we have a solution for SGMAC, i.e., $T'[i, j]$.

We can use Algorithm 1 (Appendix) to obtain T.

Because the T meets all the constraints in GMRA (See Theorem 3), T is a feasible solution to the corresponding GMAC problem. ■

The basic idea of Algorithm 1 is to split the combined agent of SGMAC into the original agents of GMAC based on the assignment T' in order to satisfy the original $L[j]$ $(0 \le j < n)$ requirement. If a role in SGMAC is combined, it can also be split into the original roles then designating a required agent by $T'[i', j'] (0 \le i' < m', 0 \le j' < n')$.

To demonstrate the requirements of Theorems 3 and 4, we simply modify the problem in Section II by changing L to satisfy the condition. If we set L = [1 2 2 1 1 2]. We can restate the corresponding SGMAC problem as

$m' = 3$, $n' = 4$

$\mathcal{A}' = \{\{0, 1, 4\}, \{2, 3\}, \{5\}\}$

$\mathcal{R}' = \{\{0, 3, 4\}, \{1\}, \{2\}, \{5\}\}$

$L' = [3\ 2\ 2\ 2]$

$L^{a'} = [4\ 4\ 3]$

$$Q' = \begin{bmatrix} 0.78 & 0.82 & 0.75 & 0.60 \\ 0.96 & 0.85 & 0.86 & 0.80 \\ 0.96 & 0.50 & 0.10 & 0.85 \end{bmatrix}$$

The result is

$$T' = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 \end{bmatrix}.$$

We can get an assignment for the GMAC problem, i.e.,

$$T = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}.$$

Note that this T represents a feasible but not necessarily optimum assignment scheme.

Theorem 3: The T obtained from Algorithm 1 meets all the constraints of the corresponding GMAC problem.

Proof: From Definition 16, we know that T' is the solution matrix for the SGMAC problem, it meets constraints (17)–(19).

Therefore, we know that

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

Note that, if role $r_{j'}$ is a combined role, the corresponding $L[j'] = \sum_{r_j \in r_{j'}} L[j] (0 \le j' < n')$.

Algorithm 1 is to split the 1s in column j' of T' into columns in T ($r_j \in r_{j'}$). Hence, it guarantees

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

Also because

$$\sum_{j'=0}^{n'-1} T'[i', j'] \le L^{a'}[i'] \ (0 \le i' < m')$$

and Algorithm 1 is also used to split the 1s in row i' of T' into rows in T ($a_i \in a_{i'}$), as well as

$$L^{a'}[i'] = \min\left(\sum\nolimits_{a_i \in a_{i'}} L^a[i], n\right) (0 \le i' < m').$$

Therefore, we have

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

Because we have combined all the conflicting agents into one single agent and the conflicting roles into one single role in SGMAC, and $T[i,j]$ is at most 1, assignment T' in SGMAC does not produce any conflicts in T.

In summary, the obtained T from Algorithm 1 (Appendix) satisfies constraints (1)–(3), (7) and (8). ■

Now we can apply the sufficient condition of GMRA to GMAC.

Given vector V, $|V|$ is the cardinality of V; max^kV is the set of the k biggest elements in V ($k \ge 0$), where $|V| \ge k$; V^* is the first element in V; especially, V^* of an empty set is defined as 0; $sub(max^kV)$ is the function such that every k biggest elements in V minus 1; W^* is the initial W, W_i is $W_{i-1}-\{W_{i-1}{}^*\}$ ($0 < i \le |W|$); and WV_0 is the initial V, WV_i is $sub(max^{Li\text{-}1^*}V_{i-1})$ ($0 < i \le |V|$).

Theorem 4: A sufficient condition for a GMAC problem is that the corresponding SGMAC problem has a feasible solution, i.e., there exists an integer q, such that L'_q is empty and for each i, $L'^*_i \le |L'_{i-1}L^{a'}_i|$ ($0 < i \le q$).

Proof: Reference [10] proves that the aforementioned condition is a necessary and sufficient condition for a GMRA problem to have a feasible solution.

Therefore, Theorem 4 is a direct conclusion. ■

The real-world meaning of Theorem 4 is that if we rearrange the order of the roles in $\mathcal{R}'$ by $L'[j']$ from the largest to the smallest, and starting from column 0, we assign the agents currently having the most allowed numbers of roles, i.e., $L^{a'}[i']$ to $L'[j']$ roles and change the $L'[j']$ agents' abilities $L^{a'}[i']$s by subtracting 1. One by one column to $n'-1$, if for each column we have sufficient agents to satisfy the role requirement $L'[j']$ ($0 \le j' < n'$), a feasible solution exists.

Note that, Theorem 4 is a sufficient, but not necessary, condition for GMAC to have a feasible solution. For example, the problem stated in Section II has a feasible solution as presented. However, its corresponding SGMAC problem has no solutions because $m' < \max(L'[j']\ (j' = 0, 1, 2))$. Note: $m' = 3$, and $\max(L'[j']\ (j' = 0, 1, 2)) = L'[0] = L[0]+L[3]+L[4] = 6$.

The significance of Theorem 4 is that we can check the conditions before we organize a team. This provides confidence in establishing a team that can be successfully assigned to tasks, in an acceptable time, by the provided CPLEX solution.

One real-world significance of Theorem 4 is to provide a reliable criterion for decision-makers to use in task assignment while avoiding conflicts that can arise in complex projects.

To solve real-world problems, decision-makers may at first try to meet the sufficient condition by specific organizations of tasks and personnel. If it is too difficult to satisfy the sufficient condition, decision-makers can use the provided CPLEX solution for task assignment. The CPLEX result and time may then inform the solution feasibility.

As for complexity, the complexity of Theorem 3 is $O(m\times n) \approx O(m^2)$ and that of Theorem 4 is $O(m^2\log(m))$ [10], which are much more efficient than that of CPLEX searching.

Because GMAC is an abstract problem, it can be extended to a broad range of application scenarios with similar components and architecture.

VI. Benefits of Avoiding Conflicts

Not all assignment conflicts produce a severe loss of group performance. To quantitatively present the benefits of solving the GMAC problem, we conduct comparisons with GMRA by working on random groups.

Definition 17 [8]: The *number of agent conflicts* n_{ca} is defined as $\sum_{i_1=0}^{m-1}\sum_{i_2=i_1+1}^{m-1} A^c[i_1, i_2]$.

Definition 18: The *number of role conflicts* n_{cr} is defined as $\sum_{j_1=0}^{n-1}\sum_{j_2=j_1+1}^{n-1} R^c[j_1, j_2]$.

Definition 19: The *agent conflict rate* p_{ca} is defined as $2\times n_{ca}/m\times(m-1)$, i.e., the number of conflicts divided by the total number of pairs of agents.

Note that, $n_{ca}/[n\times(n-1)/2] = 2\times n_{ca}/n\times(n-1)$ because the definition of n_{ca} only accounts 1/2 of the 1s in A^c. For example, the p_{ca} in Table IV is $6/(6\times5) = 20\%$.

Definition 20: The *role conflict rate* p_{cr} is defined as $2\times n_{cr}/n\times(n-1)$, i.e., the number of conflicts divided by the total number of pairs of agents.

Note that, $n_{cr}/[n\times(n-1)/2] = 2\times n_{cr}/n\times(n-1)$ because the definition of n_{cr} only accounts 1/2 of the 1s in R^c. For example, the p_{cr} in Table V is $4/(6\times5) \approx 13.33\%$.

To estimate the benefits of conflict avoidance, we need to classify the degrees of loss due to conflicts [4]. For example, we must be able to differentiate trivial and serious conflicts.

Definition 21: The *conflict loss rate* $p_{cl} \in [0,1]$ is the rate of individual agent's qualification loss, i.e., the actual qualification of agent i on role j is $Q[i,j]\times(1-p_{cl})$ if agent i or role j is in conflict with other assigned agents or roles.

To determine the *conflict loss rate* p_{cl}, we may borrow the ideas of classification of bugs in a software product. Bugs in software [26] can be classified as ignorable, mild, annoying, disturbing, serious and extreme. Others may produce even larger losses. Therefore, we may have different degrees of conflicts as described in Table VIII. Ignorable conflicts can be recorded as 0 s in the two conflict matrix.

TABLE VIII
THE CLASSIFICATION OF CONFLICTS

Conflict degree	Loss rate p_{cl}
Mild	20%
Annoying	40%
Disturbing	60%
Serious	80%
Extreme	100%

As for losses, we mean that both the individual qualifications will be decreased if conflicts between two agents or roles happen. For example, in Table II, if Chris and Doug are in mild conflict, their actual qualification values on role *Coder* should be decreased to $0.90\times(1-20\%) = 0.72$, and $0.64\times(1-20\%) = 0.512$, respectively. Similarly, if *Coder* and *Tester* are in annoying conflict, then Ed's qualifications for these two positions will be reduced to 0.33×(1–40%) = 0.198 and 0.78×(1–40%) = 0.468.

Suppose that Ed and Doug are also in mild conflict, which is not the case in Section II. The GMRA assignment results in two overlapping conflicts on Ed (agent conflict with Doug) and *Coder* (role conflict with *Tester*), his qualification value on *Coder* reduces to 0.33×(1–20%)×(1–40%) = 0.1584.

In collaboration, we do not consider conflicts worse than extreme because such roles and agents should not be included in team building. As a rule of thumb, we assume that annoying conflicts (40% qualification penalty) should be avoided. Under such conditions, both parties in the conflicting pair lose 40% of their original qualification values. In summary, the loss of an agent's qualification for a role is determined by the number of assigned conflicts. This requires a new structure for collecting the number of assignment conflicts.

Definition 22: The *number of assigned conflicts* for agent i on role j $(0 \le i < m, 0 \le j < n)$ $C_{AR}[i, j]$ (see Definition 24) is $\sum_{i_1=0}^{m-1} A^c[i, i_1] \times T^*[i, j] \times T^*[i_1, j] + \sum_{j_1=0}^{n-1} R^c[j, j_1] \times T^*[i, j] \times T^*[i, j_1]$, where T^* is obtained without considering A^c and R^c, i.e., the result of GMRA.

Definition 23: The *total number of assigned conflicts* n_c is $\sum_{j=0}^{n-1} \sum_{i_1=0}^{m-1} \sum_{i_2=i_1+1}^{m-1} A^c[i_1, i_2] \times T^*[i_1, j] \times T^*[i_2, j] + \sum_{i=0}^{m-1} \sum_{j_1=0}^{n-1} \sum_{j_2=j_1+1}^{n-1} R^c[j_1, j_2] \times T^*[i, j_1] \times T^*[i, j_2]$.

Definition 24: The *assigned conflict number matrix* C_{AR} is defined as a $m \times n$ matrix, the elements of which are the numbers of assigned conflicts for agents on roles, i.e., $C_{AR}[i, j] \in \mathcal{N}$ means the number of assigned conflicts for agent i on role j.

To compare the losses of GMRA and the benefits of GMAC, we need a way to find the conflicting agents and roles in the GMRA assignment scheme, Algorithm 2 (Appendix) is to collect all the numbers of conflicts incurred on each agent for a role.

After we have found the assigned conflict numbers for each agent relative to each role, we can recalculate agent qualification values and form Q', where

$$Q'[i, j] = Q[i, j] \times (1 - p_{cl})^{C_{AR}[i, j]} (0 \le i < m, 0 \le j < n). \quad (20)$$

Now, we conduct experiments for the comparison, without loss of generality, we use a group size of $m = 50$, and $n = 40$, values typical to a mid-sized company. For each group, we randomly choose the following data within the range of the definitions:

1) $Q[i, j] \in [0, 1]$ $(0 \le i < m, 0 \le j < n)$, i.e., the qualification value of each agent for each role;

2) $A^c[i_1, i_2]$ $(0 \le i_1, i_2 < m)$, i.e., the conflicts between agents;

3) $R^c[j_1, j_2]$ $(0 \le j_1, j_2 < n)$, i.e., the conflicts between roles; and

4) $L[j]$ $(0 \le j < n)$, $1 \le L[j] \le 5$, $L^a[i]$ $(0 \le i < m)$, $1 \le L^a[i] \le 4$ and make sure $\sum_{j=0}^{n-1} L[j] \le \sum_{i=0}^{m-1} L^a[i]$.

For each group, we collect several data items:

1) σ^*, the ideal group performance without considering conflicts, i.e., GMRA;

2) σ_c^*, the maximum group performance with GMAC;

3) σ_a^*, the actual group performance under T^*, i.e.,

$$\sigma_a^* = \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} Q'[i, j] \times T^*[i, j].$$

The benefit λ is calculated as follows:

$$\lambda = \frac{(\sigma_c^* - \sigma_a^*)}{\sigma_a^*}.$$

To understand the different effects of the conflict rates, we set p_{ca} and p_{cr} from 10% to 50%, with a step of 10%. To make the data convincing, we create 100 groups for each conflict rate pair. Finally, we compute the averages of the 100 numbers for each of the 25 cases. Table IX shows the simulation results.

TABLE IX
SIMULATION AVERAGES ($m = 50$, $n = 40$, $p_{cl} = 0.4$)

p_{ca}	p_{cr}	σ^*	σ_a^*	σ_c^*	σ_c^*/σ^*	λ
10%	10%	112.11	103.02	111.34	99.31%	8.33%
	20%	112.83	98.29	111.53	98.85%	13.24%
	30%	113.27	94.01	111.26	98.23%	17.25%
	40%	112.77	89.05	109.94	97.49%	20.89%
	50%	112.22	84.45	108.11	96.34%	23.66%
20%	10%	113.39	100.86	112.29	99.03%	11.43%
	20%	113.72	96.15	112.05	98.53%	15.90%
	30%	112.74	90.55	110.24	97.78%	19.69%
	40%	113.03	86.26	109.62	96.98%	23.36%
	50%	113.17	82.26	108.42	95.80%	26.16%
30%	10%	113.13	98.03	111.71	98.74%	13.67%
	20%	113.96	92.93	111.74	98.05%	18.80%
	30%	112.08	87.38	109.14	97.38%	21.76%
	40%	112.93	83.87	108.98	96.50%	25.12%
	50%	111.97	79.18	106.67	95.27%	27.49%
40%	10%	112.71	94.57	110.95	98.44%	16.38%
	20%	111.73	89.16	109.17	97.71%	20.01%
	30%	112.46	84.91	108.90	96.83%	24.00%
	40%	113.29	81.88	108.67	95.92%	26.78%
	50%	112.59	77.74	106.39	94.49%	28.65%
50%	10%	113.07	92.07	110.82	98.01%	18.75%
	20%	112.22	86.85	109.08	97.20%	22.23%
	30%	112.98	82.68	108.82	96.32%	26.14%
	40%	113.05	78.61	107.35	94.96%	28.75%
	50%	114.07	75.19	106.13	93.04%	30.95%

By observing the simulation results, it is evident that conflict avoidance obtains benefits in all the cases if $p_{cl} \ge 0.4$ and the benefit λ increases with additional conflicts, i.e., p_{ca} and p_{cr} increase. Column σ_c^*/σ^* reveals that we do not sacrifice much ideal group performance (from 6.96% = 1 – 93.04% to 0.69% = 1 – 99.31%) by trying to avoid conflicts. That is, even though we are not sure what the negative effects of conflicts are, we are sure that avoiding conflicts does not have much effect on ideal group performance. The qualitative benefit is that we have avoided conflicts.

VII. Related Work

Conflicts are common phenomena in the real-world [1], [4], [27]–[41]. Conflict-related research has been undertaken for decades [2], [4], [35], [37], [39] and is attracting increased attention from engineering researchers [1], [3], [27]–[31], [33]–[36], [38], [40], [41], especially so in the area of traffic management [31]–[33], [36]. However, little research on role assignment conflict exists beyond the previous work.

Bristow *et al.* [1] propose an agent-based framework for modeling behavior under possible agent conflicts. They suppose that agents' actions can be in conflict. By allowing agents to view their individual situation from a system's perspective, their framework provides rules to improve the quality of decision making.

Tessier *et al.* [4] review multi-agent systems with probable conflicting agents. They define conflict based on propositional attributes. They also describe the categorization of human conflict handling methods. Finally, they propose a conflict-handling action model. In fact, our proposed solution is one way of conflict avoidance in their model or can be considered as one instance of the "flight" method in human conflict handling.

Cai *et al.* [27] present a measurement for conflicts in order to implement a belief assignment. They generalize the classical pignistic probability transform as a pignistic belief transform. They also propose a betting distance of two pignistic belief transforms to evaluate the conflict degree of basic probability assignment. Their work can be used to evaluate the benefits of conflict avoidance related to this work.

Canbaz *et al.* [28] intend to resolve design conflicts in distributed design, because a complex design requires various designers to work together at the same time but in different places. They believe that the inconsistencies of design objectives and working procedures are reasons to cause design conflicts, which may lead to imprecise management. They put forward a conflict management model and use the constraint satisfaction problem solution to detect and rectify design conflicts.

Hong *et al.* [31] design a framework for conflict resolution in air traffic control by using air traffic complexity, which ensures conflict avoidance among airplanes. These conflicts relate to disasters that may lead to fatalities, an unacceptable situation. Their work provides a typical scenario for our future work on conflict avoidance based on group role assignment. We believe that the proposed modelling method in this paper, i.e., GMAC, can simplify the modeling of the problem discussed in [31], which presents over 30 constraint expressions. After the conflict matrices are determined, our proposed modelling makes practitioners concentrate on more important issues. Applying GMAC to air traffic control is interesting future research work.

Jiang *et al.* [32] analyze the differences in conflict situation between driving cultures, and field traffic data have been collected by video recording and image processing at urban midblock crosswalks both in Beijing, China, and Munich, Germany. With observation, conflict analysis, and time-to-collision (TTC) calculation, they create a trajectory-based data matrix to understand the entire conflict process. They also provide an intercultural comparison of TTC distribution and relationships between TTC and other parameters. Their work can be applied to conflict analysis in similar real-world scenarios.

Katrakazas *et al.* [33] propose a method for conflict prediction through the use of highly disaggregated vehicle-based traffic data from a traffic micro-simulation and the corresponding traffic conflicts data generated by the surrogate safety assessment model. Their results show the viability of using traffic micro-simulation along with the surrogate safety assessment model for real-time conflict prediction and the superiority of random forests with 5-min temporal aggregation in the classification results.

Kinsara *et al.* [34] discuss systems methodologies to model third-party intervention in international conflicts within the framework of the graph model for conflict resolution. They introduce an inverse graph model to utilize the graph model for conflict resolution as a negotiation tool by altering the procedure. This inversion helps decision makers understand the conflict situation, undertake certain actions within the conflict, and specify preferences for a particular resolution. Such conflicts belong to disturbing and serious categories.

Lv *et al.* [36] investigate vehicle-pedestrian conflicts and their relationship to traffic safety. They present a method to generate high-resolution traffic trajectories from sensor data, then identify vehicle-pedestrian conflicts from the trajectories using a rule-based method. Such conflicts may lead to loss of life and require thorough resolution.

Nyanchama and Osborn [37] describe a role-graph model for role-based access control. In discussing the model, they clarify roles in conflict. They use the graph model to provide a taxonomy for conflict types. They solve the complex problem of role assignment with potential conflicts by partitioning the role graph into role collections that are not in conflict and can be assigned to an agent all together.

Rajan *et al.* [3] study the verbal conflict of human-interaction behavior. They believe that the detection of such conflicts would enable monitoring and feedback in a variety of applications. To address the limitations of traditional verbal conflict detection methods, they propose an end-to-end convolutional-recurrent neural network architecture that learns conflict-specific features directly from raw speech waveforms, without using explicit domain knowledge or metadata. The conflicts mentioned in their work are annoying and their work is beneficial to some applications but their goal and research methods are completely different from this work.

Such and Criado [38] propose a way to merge multiple users' privacy preferences by considering preference conflicts. Such conflicts belong to the annoying category and must be resolved when conducting multi-party privacy management in Social Media. After such conflicts are resolved, users of

TABLE X
THE COMPARISON AMONG ROLE ASSIGNMENT PROBLEMS

Problem	Input								
	$\mathcal{A}$, $m = \lvert\mathcal{A}\rvert$	$\mathcal{R}$, $n = \lvert\mathcal{R}\rvert$	Q	L	L^a	A^c	R^c	P	Λ, ◇
General assignment	√	√ (= m)	√						
GRA	√	√	√	√					
GRA with balance between preferences and performance	√	√	√	√				√	
GRACAR	√	√	√	√		√			
GMRA	√	√	√	√	√				
Tree-structure task allocation	√	√	√	√	√				√
GMAC	√	√	√	√	√	√	√		

Note: √ means the corresponding problem needs corresponding input data.

Social Media may reach an agreement in merging privacy preferences.

Yang *et al.* [40] discuss the onboard locally centralized conflict resolution problem of heterogeneous unmanned aerial vehicles (UAVs) with variable headings. Such conflicts may damage agents (UAVs), cause loss of expensive equipment, and are considered extreme. They map the nonlinear safe separation constraint to the sine value space, and make a linear constraint of safe separation for pairwise conflicting UAVs. Then, a mixed integer linear programming model is built to cope with the conflict resolution problem of heterogeneous UAVs to minimize the overall costs. The modeled problem is finally solved by CPLEX.

Zhang *et al.* [41] formalize the multi-robot task allocation with definite path-conflict-free handling problem based on grid maps. In their model, two subtasks of each cooperative task must be executed by two robots. Therefore, path conflict becomes an obstacle. They design a task allocation algorithm, which minimizes the completion time and realizes conflict-free path planning. In the proposed method, they deal with both the common path conflicts and the path conflicts between robots when they exchange positions.

The aforementioned research indicates a strong need to fundamentally investigate GRA$^+$ including the problem discussed in this paper.

Liu *et al.* [12] study a problem of GRA, called GRA with Balance between preferences and performance (Table X), and propose a way to balance decision-maker preferences with agent qualifications. Liu *et al.* [13] also investigate a different case of GMRA, a tree-structured task allocation problem (Table X), and propose formalizations and practical solutions based on CPLEX. They speed up the CPLEX solution and provide better planning techniques aimed at solution feasibility by introducing necessary conditions and sufficient conditions.

Damiano-Teixeira [25] discusses the requirement of managing role conflicts for female faculties. Thomas [42] concerns role conflicts in managing different styles of educations. The contributions related role or role relation network also present an important application scope for the proposed problem and solution [43], [44]. Deadlock avoidance [45], [46] in automated guided vehicle systems can also be an application of the proposed modeling.

The previous work [5], [7]–[18] established a solid foundation for conducting the investigations of this paper. The related problems can be compared with each other as listed in Table X, which provides more details about the related role assignment problems mentioned in Fig. 2, where P is an m-vector to express the decision-makers' preference degrees for agents, Λ is a promotion role matrix and ◇ is a peer role matrix.

VIII. CONCLUSIONS

In this paper, we continue the effort to provide more accurate and efficient assignment schemes as a part of research in RBC and E-CARGO. The problem of GMAC is formalized, a solution based on CPLEX is implemented then tested and analyzed with regard to efficiency, planning, and benefits. The work of this paper again confirms the discovery power of RBC, E-CARGO and GRA.

It is meaningful to further investigate the following issues:

1) Conflict expression as real numbers in a structure that conveys relative degrees of conflict damage. This can improve the assignment process.

2) Evaluation of conflicts to form more exact numbers that reflect their degree of severity.

3) More formal expression of the relationships among roles and agents when conducting GRA.

4) Application of E-CARGO and GRA to more complicated scenarios [29], [39], [46]–[49] by introducing more formalized factors and constraints.

5) Effective algorithms for classification [50] used to solve similar problems to GMAC.

ACKNOWLEDGMENT

Any opinions and conclusions in this work are strictly those of the author(s) and do not reflect the views, positions, or policies of – and are not endorsed by – IDEaS, DND, or the Government of Canada. Thanks go to Mike Brewes of Nipissing University for his assistance in proofreading this article.

APPENDIX

TABLE XI
NOMENCLATURE

Symbol	Meaning
$\mathcal{A}$	The agent set.
$\mathcal{R}$	The role set.
m	The size of the agent set.
n	The size of the role set.
a_i	An element in $\mathcal{A}$.
r_j	An element in $\mathcal{R}$.
$0 \le i, i_0, i_1, \ldots < m$	The indices of agents.
$0 \le j, j_0, j_1, \ldots < n$	The indices of roles.
L	A role requirement vector.
L^a	An agent ability vector.
Q	A qualification matrix.
T	An assignment matrix.
σ	A group performance.
GA	General assignment.
GRA	Group role assignment.
GRA$^+$	Group role assignment with constraints.
GRA^{++}	Group role assignment with multiple objectives.
GMRA	Group multi-role assignment.
GMAC	Group multi-role assignment with conflicting roles and agents.
SGMAC	Simplified GMAC.
n_{ca}	The number of agent conflicts.
n_{cr}	The number of role conflicts.
p_{ca}	The agent conflict rate.
p_{cr}	The role conflict rate.
p_{cl}	The conflict loss rate.
n_c	The total number of assigned conflicts.
C_{AR}	The assgined conflcit number matrix.
$C_{AR}[i, j]$	The assgined conflcit number for agent i on role j.
Q'	The revised qualification matrix by considering the assigned conflicts.
σ^*	The ideal group performance without considering conflicts, i.e., GMRA.
σ_c^*	The maximum group performance with GMAC.
σ_a^*	The actual group performance under GMRA.
λ	The group performance benefit of GMAC compared with GMRA.

Algorithm 1 Obtain T from T'

Input: $\mathcal{A}$ ($|\mathcal{A}| = m$), $\mathcal{R}$ ($|\mathcal{R}| = n$), $\mathcal{A}'$ ($|\mathcal{A}'| = m'$), $\mathcal{R}'$ ($|\mathcal{R}'| = n'$), L, L^a, L', $L^{a\prime}$, and T'
Output: T
begin
  $T = \{0\}$;
  for $(0 \le j < n')$
    $r' = \mathcal{R}'[j]$;
    $i = 0$;
    while $(i < m')$
      $a' = \mathcal{A}'[i]$;
      if $(T'[i, j] == 1)$
        if $((a' \ne \phi)\&\&(r' \ne \phi))$
          $i'' = a'[0]$; //Split the agent
          $j'' = r'[0]$; //Split the role
          $T[i'', j''] = 1$;
          $T'[i, j] = 0$;
          $L^a[i''] = L^a[i'']-1$;
          if $(L^a[i''] == 0)$
            $a' - \{i''\}$;
          endif
          $L[j''] = L[j'']-1$;
          if $(L[j''] == 0)$
            $r' - \{j''\}$;
          endif
        endif
      endif
      $i = i+1$;
    endwhile
  endfor
end

Algorithm 2 Conflict numbers

Input: T^*, A^c, and R^c
Output: C_{AR}
begin
  $C_{AR}[i, j] = 0$ $(0 \le i < m, 0 \le j < n)$;
  for $(0 \le j < n)$
    for $(0 \le i_1 < m)$
      for $(0 \le i_2 < m)$
        if $((T^*[i_1, j] = 1)$ and $(T^*[i_2, j] = 1)$ and $(A^c[i_1, i_2] = 1))$
          $C_{AR}[i_1, j]{+}{+}$;
          $C_{AR}[i_2, j]{+}{+}$;
        endif
      endfor
    endfor
endfor
for $(0 \le i < m)$
  for $(0 \le j_1 < m)$
    for $(0 \le j_2 < m)$
      if $((T^*[i, j_1] = 1)$ and $(T^*[i, j_2] = 1)$ and $(R^c[j_1, j_2] = 1))$
        $C_{AR}[i, j_1]{+}{+}$;
        $C_{AR}[i, j_2]{+}{+}$;
      endif
    endfor
  endfor
endfor
end

REFERENCES

[1] M. Bristow, L. P. 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.

[2] T. Malsch and G. Weiss, "Conflicts in social theory and multi-agent systems: On importing sociological insights into distributed AI," in *Conflicting Agents: Conflict Management in Multi-Agent Systems*, C. Tessier, L. Chaudron, and H. J. Müller, Eds. Boston, USA: Springer,

2002, pp. 111−149.

[3] V. Rajan, A. Brutti, and A. Cavallaro, "Conflict NET: End-to-end learning for speech-based conflict intensity estimation," *IEEE Signal Process. Lett.*, vol. 26, no. 11, pp. 1668–1672, Nov. 2019.

[4] C. Tessier, H. J. Müller, H. Fiorino, and L. Chaudron, "Agents' conflicts: New issues," in *Conflicting Agents: Conflict Management in Multi-Agent Systems*, C. Tessier, L. Chaudron, and H. J. Mülle, Eds. Boston, USA: Springer, 2002, pp. 1−30.

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

[6] R. E. Burkard, M. Dell'Amico, and S. Martello, *Assignment Problems, Revised Reprint*. Philadelphia, USA: SIAM, 2009.

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

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

[9] H. B. Zhu, D. N. Liu, S. Q. Zhang, Y. Zhu, L. Y. Teng, and S. H. Teng, "Solving the many to many assignment problem by improving the Kuhn-Munkres algorithm with backtracking," *Theor. Comput. Sci.*, vol. 618, pp. 30–41, Mar. 2016.

[10] H. B. Zhu, D. N. Liu, S. Q. Zhang, S. H. 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.

[11] H. B. Zhu, Y. Sheng, X. Z. 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.

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

[13] D. N. Liu, B. Y. Huang, and H. B. 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.

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

[15] H. B. 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.

[16] H. B. Zhu and Y. Zhu, "Group role assignment with agents' busyness degrees," in *Proc. IEEE Int. Conf. Systems, Man, and Cybernetics*, Banff, Canada, 2017, pp. 3201−3206.

[17] H. B. Zhu and Y. Zhu, "Group role assignment with agents' preferences," in *Proc. 14th IEEE Int. Conf. Networking, Sensing and Control*, Calabria, Italy, 2017, pp. 605−610.

[18] H. B. Zhu, "The most economical redundant assignment," in *Proc. IEEE Int. Conf. Systems, Man and Cybernetics*, Bari, Italy, 2019, pp. 146−151.

[19] R. L. Rardin, *Optimization in Operations Research*. Upper Saddle River, USA: Prentice Hall, 1997.

[20] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*. New York, USA: W. H. Freeman and Company, 1979.

[21] H. W. Kuhn, "The Hungarian method for the assignment problem," *Nav. Res. Log.*, vol. 52, no. 1, pp. 7–21, Feb. 2005.

[22] J. Munkres, "Algorithms for the assignment and transportation problems," *J. Soc. Ind. Appl. Math.*, vol. 5, no. 1, pp. 32–38, Mar. 1957.

[23] C. H. Papadimitriou, "On the complexity of integer programming," *J. ACM*, vol. 28, no. 4, pp. 765–768, Oct. 1981.

[24] L. A. Wolsey and G. L. Nemhauser, *Integer and Combinatorial Optimization*. New York, USA: Wiley-Interscience, 1999.

[25] IBM, "IBM ILOG CPLEX optimization studio," [Online]. Available: http://www-01.ibm.com/software/integration/optimization/cplex-optimization-studio/

[26] R. S. Pressman, *Software Engineering: A Practitioner's Approach*. 6th ed. New York, USA: McGraw-Hill Higher Education, 2005.

[27] Q. X. Cai, X. Z. Gao, and Y. Deng, "Pignistic belief transform: A new method of conflict measurement," *IEEE Access*, vol. 8, pp. 15265–15272, Jan. 2020.

[28] B. Canbaz, B. Yannou, and P. A. Yvars, "Resolving design conflicts and evaluating solidarity in distributed design," *IEEE Trans. Syst.*, *Man*, *Cybern.*: *Syst.*, vol. 44, no. 8, pp. 1044–1055, Aug. 2014.

[29] K. M. Damiano-Teixeira, "Managing conflicting roles: A qualitative study with female faculty members," *J. Fam. Econ. Iss.*, vol. 27, no. 2, pp. 310–334, Jun. 2006.

[30] W. Q. Guo, W. M. Healy, and M. C. Zhou, "Impacts of 2.4-GHz ISM band interference on IEEE 802.15.4 wireless sensor network reliability in buildings," *IEEE Trans. Instrum. Meas.*, vol. 61, no. 9, pp. 2533–2544, Sept. 2012.

[31] Y. Hong, B. Choi, K. Lee, and Y. Kim, "Conflict management considering a smooth transition of aircraft into adjacent airspace," *IEEE Trans. Intell. Trans. Syst.*, vol. 17, no. 9, pp. 2490–2501, Sept. 2016.

[32] X. B. Jiang, W. H. Wang, and K. Bengler, "Intercultural analyses of time-to-collision in vehicle–pedestrian conflict on an urban midblock crosswalk," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 2, pp. 1048–1053, Apr. 2015.

[33] C. Katrakazas, M. Quddus, and W. H. Chen, "A simulation study of predicting real-time conflict-prone traffic conditions," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 10, pp. 3196–3207, Oct. 2018.

[34] R. A. Kinsara, D. M. Kilgour, and K. W. Hipel, "Inverse approach to the graph model for conflict resolution," *IEEE Trans. Syst.*, *Man*, *Cybern.*: *Syst.*, vol. 45, no. 5, pp. 734–742, May 2015.

[35] M. Klein, "Supporting conflict resolution in cooperative design systems," *IEEE Trans. Syst.*, *Man*, *Cybern.*, vol. 21, no. 6, pp. 1379–1390, Dec. 1991.

[36] B. Lv, R. J. Sun, H. B. Zhang, H. Xu, and R. Yue, "Automatic vehicle-pedestrian conflict identification with trajectories of road users extracted from roadside LiDAR sensors using a rule-based method," *IEEE Access*, vol. 7, pp. 161594–161606, Jul. 2019.

[37] M. Nyanchama and S. Osborn, "The role graph model and conflict of interest," *ACM Trans. Inform. Syst. Secur.*, vol. 2, no. 1, pp. 3–33, Feb. 1999.

[38] J. M. Such and N. Criado, "Resolving multi-party privacy conflicts in social media," *IEEE Trans. Knowledge Data Eng.*, vol. 28, no. 7, pp. 1851–1863, Jul. 2016.

[39] C. Tomlin, G. J. Pappas, and S. Sastry, "Conflict resolution for air traffic management: A study in multiagent hybrid systems," *IEEE Trans. Autom. Control*, vol. 43, no. 4, pp. 509–521, Apr. 1998.

[40] J. Yang, X. Xu, D. Yin, Z. W. Ma, and L. C. Shen, "A space mapping based 0–1 linear model for onboard conflict resolution of heterogeneous unmanned aerial vehicles," *IEEE Trans. Veh. Technol.*, vol. 68, no. 8, pp. 7455–7465, Aug. 2019.

[41] H. G. Zhang, H. Luo, Z. Wang, Y. H. Liu, and Y. A. Liu, "Multi-robot cooperative task allocation with definite path-conflict-free handling," *IEEE Access*, vol. 7, pp. 138495–138511, Sept. 2019.

[42] G. Thomas, "Facilitator, teacher, or leader? Managing conflicting roles in outdoor education" *J. Exp. Educ.*, vol. 32, no. 3, pp. 239–254, May 2010.

[43] J. H. Ye, Z. W. Li, K. Yi, and A. Al-Ahmari, "Mining resource community and resource role network from event logs," *IEEE Access*, vol. 6, pp. 77685–77694, Nov. 2018.

[44] Q. T. Zeng, J. Liu, C. H. Zhou, C. Liu, and H. Duan, "A novel approach for business process similarity measure based on role relation network mining," *IEEE Access*, vol. 8, pp. 60918–60928, Mar. 2020.

[45] N. Q. Wu and M. C. Zhou, "Modeling and deadlock control of automated guided vehicle systems," *IEEE/ASME Trans. Mech.*, vol. 9, no. 1, pp. 50–57, Mar. 2004.

[46] G. Fortino, W. Russo, C. Savaglio, W. M. Shen, and M. C. Zhou, "Agent-oriented cooperative smart objects: From IoT system design to implementation," *IEEE Trans. Syst.*, *Man*, *Cybern.*: *Syst.*, vol. 48, no. 11, pp. 1939–1956, Nov. 2018.

[47] G. Fortino, A. Guerrieri, W. Russo, and C. Savaglio, "Towards a development methodology for smart object-oriented IoT systems: A metamodel approach," in *Proc. IEEE Int. Conf. Systems, Man, and Cybernetics*, Hong Kong, China, 2015, pp. 1297−1302.

[48] G. Fortino, A. Garro, and W. Russo, "An integrated approach for the development and validation of multi-agent systems," *Comput. Syst. Sci.*

Eng., vol. 20, no. 4, pp. 259–271, Jul. 2005.

[49] S. M. M. Rahman, "Cyber-physical-social system between a humanoid robot and a virtual human through a shared platform for adaptive agent ecology," *IEEE/CAA J. Autom. Sinica*, vol. 5, no. 1, pp. 190–203, Jan. 2018.

[50] S. C. Gao, M. C. Zhou, Y. R. Wang, J. J. Cheng, H. Yachi, and J. H. Wang, "Dendritic neuron model with effective learning algorithms for classification, approximation, and prediction," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 2, pp. 601–614, Feb. 2019.

Haibin Zhu (M'02–SM'04) received the B.S. degree in computer engineering from Institute of Engineering and Technology in 1983, and M.S. and Ph.D. degrees in computer science from the National University of Defense Technology (NUDT) in 1988 and 1997, respectively. He is a Full Professor and the Chair of the Department of Computer Science and Mathematics, Founder and Director of Collaborative Systems Laboratory, Nipissing University, Canada, and a Visiting Professor of the Department of Control Science and Engineering, Nanjing University. He was a Visiting Professor and a Special Lecturer in the College of Computing Sciences, New Jersey Institute of Technology, USA (1999–2002) and a Lecturer, an Associate Professor, and a Full Professor at NUDT (1988–2000). He has published more than 190 research papers, 4 books, 4 book chapters and 3 conference proceedings on object-oriented programming, distributed systems, collaborative systems, and computer architecture. He is serving as Co-Chair of the Technical Committee of Distributed Intelligent Systems of IEEE SMC Society, Associate Editor of *IEEE Trans. on Systems, Man and Cybernetics: Systems*, *IEEE Trans. on Computational Social Systems*, *IEEE SMC Magazine*, and *IEEE Canada Review*, Associate Editor-in-Chief for *Int'l J. of Advances in Information and Service Sciences*, and Member of the editorial board of *Int'l J. of Software Science and Computational Intelligence*. He served as Associate Editor of *Int'l J. of Distributed Artificial Intelligence* (2017–2019). He has been an active Organizer for the annual *IEEE Int'l Conf. on SMC* since 2003, Special Session Chair, Tutorial Chair, Area Co-Chair, Social Media Co-Chair, Web Co-Chair, Poster Co-Chair, Session Chair, and Special Session Organizer. He was a Program Chair for *17th IEEE Int'l Conf. on Computer Supported Cooperative Work in Design, Dalian, China, May 6–8, 2020*, was the Program Chair for *16th IEEE Int'l Conf. on Networking, Sensing and Control, Banff, AB, Canada, May 8–11, 2019*, and was a Program Chair for *17th IEEE Int'l Conf. on Computer Supported Cooperative Work in Design, Whistler, BC, Canada, June 27–29, 2013*. He also served as PC Members for more than 90 academic conferences. He is the Recipient of the 2018 Meritorious Service Award from IEEE SMC Society, 2011 Chancellors' Award for excellence in research and 2006–2007/2011–2012 research achievement awards from Nipissing University, the 2004 and 2005 IBM Eclipse Innovation Grant Awards, the Best Paper Award from the *ISPE Int'l Conf. on Concurrent Engineering (ISPE/CE2004)*, the Educator's Fellowship of OOPSLA'03, a 2nd Class National Award of Education Achievement from Ministry of Education of China (1997), a 2nd Class National Award of Excellent Textbook from the Ministry of Education of China (2002), three 1st Class Ministerial Research Achievement Awards from The Commission of Science Technology and Industry for National Defense of China (1997, 1994, and 1991), and a 2nd Class Excellent Textbook Award of the Ministry of Electronics Industry of China (1996). He is a Member of ACM, and a Life Member of the Chinese Association for Science and Technology. He is the Founding Researcher of Role-Based Collaboration (RBC) and Adaptive Collaboration (AC).