Group Role Assignment With Minimized Agent Conflicts

Qian Jiang, *Member, IEEE*, Dongning Liu, *Senior Member, IEEE*, Haibin Zhu, *Fellow, IEEE*, Baoying Huang, Naiqi Wu, *Fellow, IEEE*, and Yan Qiao, *Senior Member, IEEE*

***Abstract*—In role-based collaboration (RBC) methodology, eliminating agent conflicts during the role assignment process is crucial for establishing a sustainable cooperative system. However, when agent resources are scarce, assignment strategies aimed at eliminating agent conflicts become infeasible. Consequently, there is a need to select the optimal assignment with a minimal number of agent conflicts, which is essentially a nonlinear bilevel optimization problem. To tackle this issue, we first design the group role assignment with minimized agent conflicts (GRAMAC) model to formalize this problem. It converts this problem into an extended integer linear programming (x-ILP) one and finds the optimal solution. Then, we prove that solving the GRAMAC model is an $\mathcal{NP}-$complete task. Moreover, we identify the sufficient and necessary condition under which the GRAMAC model has the optimal conflict-free solution. Finally, extensive experiments demonstrate that, compared to existing strategies, our proposed method reduces the number of agent conflicts by an average of approximately 30% while ensuring the group performance of the collaborative system.**

***Index Terms*—Agent conflicts, decision making, group role assignment with minimized agent conflicts (GRAMAC), group role assignment (GRA), $\mathcal{NP}-$ complete, role assignment, role-based collaboration (RBC), task assignment.**

Nomenclature

$\mathcal{R}$	Set of roles.
$\mathcal{A}$	Set of agents.
$\mathcal{L}$	Role range vector.
m	Number of agents.
n	Number of roles.
n_a	Number of required agents for roles.
Q	Qualification matrix.
T	Role assignment matrix.
$\widehat{T}$	Agent-role pair assignment matrix.
σ	Group performance.
A^c	Conflicting agent matrix.
c	Minimum number of agent conflicts.
ϵ	Conflict penalty coefficient.
K_j	Set of nonconflict optimal solution results of role j.
S_j	Collecting all the feasible K_js.
RBC	Role-based collaboration.
TAP	Team assignment problem.
BM-TAP	Bilevel model for TAP.
GRA	Group role assignment.
GRA+	GRA with constraints.
GRACAG	GRA with conflicting agents on group.
GRAMAC	GRA with minimized agent conflicts.

Received 14 July 2024; accepted 29 November 2024. Date of publication 19 December 2024; date of current version 19 February 2025. This work was supported in part by the National Key Research and Development Program of China under Grant 2022YFB3304400; in part by the Science and Technology Development Fund (FDCT), Macau, SAR, under Grant 0018/2021/A1 and Grant 0083/2021/A2; and in part by the Natural Sciences and Engineering Research Council of Canada (NSERC) under Grant RGPIN2018-04818. This article was recommended by Associate Editor L. C. Rego. *(Corresponding author: Yan Qiao.)*

Qian Jiang is with the Macau Institute of Systems Engineering and Collaborative Laboratory for Intelligent Science and Systems, Macau University of Science and Technology, Macau, China, and also with the School of Computer Science and Technology, Guangdong University of Technology, Guangzhou 510006, Guangdong, China (e-mails: qjiang.ieee@gmail.com).

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

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

Baoying Huang, Naiqi Wu, and Yan Qiao are with the Macau Institute of Systems Engineering and Collaborative Laboratory for Intelligent Science and Systems, Macau University of Science and Technology, Macau, China (e-mail: baoyhuang@126.com; nqwu@must.edu.mo; yqiao@must.edu.mo)

This article has supplementary material provided by the authors and color versions of one or more figures available at https://doi.org/10.1109/TSMC.2024.3510588.

Digital Object Identifier 10.1109/TSMC.2024.3510588

I. Introduction

ROLE-BASED collaboration (RBC) [1], [2], [3], [4] is a burgeoning cybernetics methodology, designed to utilize roles as the foundational mechanism for optimizing collaborative activities [5], [6]. It encompasses a range of activities, such as abstraction, classification, separation of concerns, dynamic behavior, resource allocation, interactions, coordination, and decision making [3]. Due to its inherent interpretability, it has developed into a new problem-solving paradigm to solve engineering problems [7], [8], [9], [10], such as constructing multiautonomous aerial vehicle (multi-AAV) systems [6], [11], optimizing the spatiotemporal crowdsourcing [7], [8] strategies, and solving TAP [2], [5], [12].

As illustrated in Fig. 1, the RBC methodology primarily encompasses five stages: 1) role negotiation; 2) agent evaluation; 3) role assignment; 4) role playing; and 5) role transfer. The core of the RBC methodology lies in the first three stages. Initially, the role negotiation stage [7], [11] is responsible for decomposing tasks into their smallest executable units, termed roles. Once tasks have been segmented into roles, the agent evaluation process [6], [13] is performed on all agents (i.e., task executors) responsible for executing these roles. Relying on the results of this agent evaluation, the

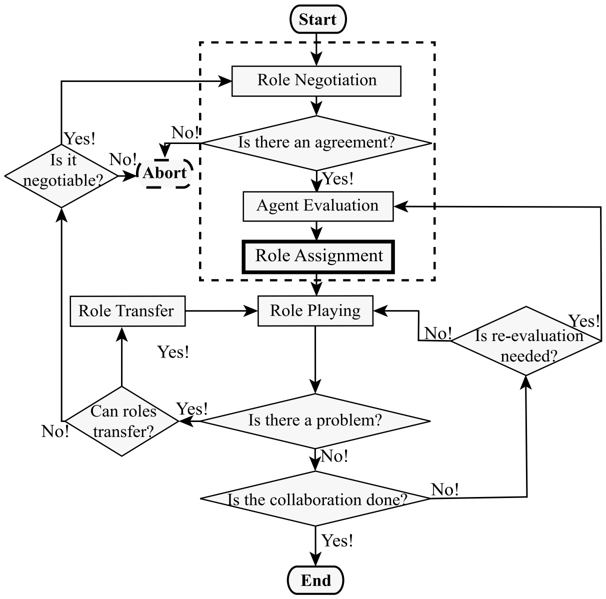


Fig. 1. Relationship of this work to RBC [6]. Note: The dashed black box is the core part of RBC, while the solid black box is the concentration of this work.

role assignment stage subsequently employs its submodels GRA+ [1], [8], [14], [15], [16], [17] to assign the optimal agents to roles. Eventually, role playing [1] and role transfer [18], [19] are stages that occur post-assignment and involve execution and dynamic adjustments.

In the aforementioned processes, role assignment serves as the linchpin, as it entails the abstraction of engineering problems and their constraints for formal modeling and researching for the optimal solution. The role assignment process also constitutes the primary focus of this article.

During the role assignment process, it is essential to consider conflicts among agents (i.e., task executors). In practical engineering problems, such conflicts between agents are frequently encountered [20]. For example, in cold chain logistics, it is generally not advisable to place pharmaceuticals and food items (agents) in the same refrigerated vehicle (role) due to the risk of cross-contamination. Similarly, in computer or mobile operating systems, conflicts arise when different programs (agents) vie for limited input/output resources (roles). In the organizational context, when assembling a new business group (BG) (role) by reallocating individuals from existing BGs, conflicts may already exist among the candidates (agents) based on their prior work experience.

In the state-of-the-art literature [1], [3], [14], [16], [21], [22] on avoiding conflicts during the role assignment process, the GRACAG model [3] stands as a seminal work. This model operates on the premise that agents have ample resources, aiming to eliminate interagent conflicts to achieve an optimal, conflict-free role assignment. However, in real-world applications, resource scarcity among agents—often due to budgetary constraints—precludes the complete elimination of conflicts. In such circumstances, it becomes crucial to seek an optimal role assignment with the premise of minimal agent conflicts.

Tackling this issue is intuitively a nonlinear, integer bilevel optimization problem [23], which cannot be directly solved by optimization platforms, e.g., IBM ILOG CPLEX optimization package (CPLEX) [24] and Gurobi optimizer [25]. In response to this challenge, we utilize a real-world TAP as an example to introduce the GRAMAC model. This novel model rigorously formalizes TAP, converting it from a bilevel, nonlinear integer programming dilemma into an extended integer linear programming (x-ILP) framework. We further prove that solving the GRAMAC model is an $\mathcal{NP}$-*complete* [14], [16] task. Moreover, we identify the sufficient and necessary conditions for the GRAMAC model to yield an optimal solution that is conflict-free.

In summary, the contributions of this article are listed as follows.

1) We utilize a real-world TAP to elucidate the formal modeling process of the GRAMAC model.
2) We prove that the GRAMAC problem is $\mathcal{NP}$-*complete*, and find the sufficient and necessary condition requisite for obtaining an optimal, conflict-free solution for GRAMAC.
3) The proposed solution broadens the applicability of the RBC methodology and its associated GRA+ model.

The remainder of this article is arranged as follows. We first introduce a real-world scenario related to the proposed problem in Section II. Following that, Section III provides an exposition of the GRACAG model. Section IV formally delineates the newly proposed GRAMAC model to solve the proposed problem. In Section V, we rigorously prove the computational complexity of the GRAMAC problem and outline the sufficient and necessary conditions for obtaining an optimal, conflict-free solution. In Section VI, simulation experiments are carried out. We introduce related research and efforts in Section VII, and finally draw the conclusion in Section VIII.

II. REAL-WORLD SCENARIO

Company X, dedicated to providing premium information technology (IT) services, recently secured a contract with Company Y for an outsourced project to develop an industry chain management system after a successful bidding process. This project primarily aims to assist Company Y's core business in establishing a digital simulation platform that offers operators a virtual training environment, enabling them to familiarize themselves with the equipment and procedures before actual operations. Considering the importance of Company Y and the urgent timeline of the project, Ann, the CEO of Company X, urged the company's CTO, Bob, to quickly select from the current staff, particularly those with lighter duties, to establish a new BG dedicated to this outsourced project, ensuring timely delivery.

Initially, leveraging his experience, Bob identifies the positions and the number of personnel required for this new BG, as detailed in Table I. Meanwhile, as Company Y has multiple projects underway concurrently, Bob is constrained to select potential candidates for this new BG from other BGs whose members have relatively lighter responsibilities.

TABLE I
REQUIRED POSITIONS FOR THE NEW BG

Position	Algorithm Engineer	Data Engineer	Software Engineer	Automation Test Engineer
Required Number	2	3	4	2

TABLE II
EVALUATION OF CANDIDATES FOR DESIGNATED POSITIONS

Candidates	Positions			
	Algorithm Engineer	Data Engineer	Software Engineer	Automation Test Engineer
Alton	0.66 [a]	0.32	0.22	0.49
Blythe	0.37	0.68	0.27	0.80
Caela	0.46	0.35	0.66	0.44
Dax	0.90	0.61	0.53	0.65
Elvin	0.33	0.22	0.86	0.45
Faela	0.89	0.70	0.33	0.53
Gaige	0.41	0.81	0.31	0.93
Hael	0.76	0.54	0.41	0.32
Isra	0.39	0.06	0.53	0.68
Jacek	0.81	0.45	0.42	0.50
Kael	0.91	1.00	0.38	0.55
Lior	0.81	0.53	0.62	0.88
Maela	0.15	0.00	0.46	0.27

[a] Values in Table II range from 0 to 1, where 0 represents the lowest and 1 denotes the highest suitability.

Bob then evaluates these candidates based on their current performance and assigns values reflecting their suitability for various positions, as presented in Table II. The values in Table II range from 0 to 1, where 0 represents the lowest and 1 denotes the highest suitability.

Considering the urgency of the project, Bob aims to assemble the BG whose members can complete the project quickly and cooperatively. He asked the leaders of existing BGs to summarize the past working relationships of the candidates and list any conflicts between them, as delineated in Table III. Based on this, Bob hopes to build the new BG, ensuring that there are no existing conflicts among members working in the new BG. This is crucial as different positions within the BG require close collaboration.

Based on Bob's analysis, the formation of this new BG fundamentally represents an one-to-many (1-M) TAP [5] with conflict constraints. Specifically, when considering the conflicting relations among candidates, a single candidate can only undertake tasks for one position, whereas multiple candidates are required to simultaneously collaborate on the tasks of a single position. Correspondingly, by searching relevant research about TAP [1], [3], Bob learns that the RBC and its submodel GRACAG have become an effective strategy to solve the 1-M TAP with conflict constraints. He decides to use them to solve it.

Bob follows the initial steps of the RBC and formalizes this TAP with the GRACAG model (see Section II in detail). However, Bob is perplexed to find that the GRACAG model is unable to yield a feasible solution for this TAP. He was uncertain about the root cause of this issue. Fortunately, our analysis revealed that the limited number of candidates rendered it impossible to ensure an agent conflict-free team formation. In such scenarios, it is imperative to achieve an optimal assignment solution while minimizing the number of agent conflicts. To address this issue, we devise a novel GRA+ model, i.e., the GRAMAC model. The subsequent sections of this article provide a detailed description of our proposed solution.

III. GRACAG MODEL

In this section, we utilize the real-world scenario introduced in Section II to introduce the GRACAG model [3] for solving TAP with conflict constraints. To facilitate understanding of the definition of the GRACAG model, we use non-negative 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 $\mathcal{R}$, $i \in \{0, 1, \ldots, m-1\}$, and $j \in \{0, 1, \ldots, n-1\}$ the indices of agents and roles, respectively. To better describe the GRACAG model, we introduce the following definitions.

Definition 1: A role [5], [26] (i.e., tasks) is defined as $r ::= < id, ® >$, where id is the identification of r, i.e., *role* j $(0 \le j < n)$; and ® represents the set of requirements or properties for agents to perform r.

Note: In the context of the TAP presented in Section II, roles are the positions required by the new BG, with n equalling 4.

Definition 2: An agent [27] (i.e., task executors) is defined as $a ::= < id, Ⓠ >$, where id is the identification of a, i.e., *agent* i $(0 \le i < m)$; Ⓠ is the set of a's values corresponding to the role requirements.

Note: Ⓠ expresses the agents' performances about ®. The specific definition Ⓠ of is illustrated in Definition 4. In the context of the TAP outlined in Section II, agents are identified as the candidates poised to take on roles corresponding to the positions necessitated by the new BG. Based on Table II, n is equal to 13.

Definition 3: A *role range vector* [15], [28] L is an n-dimensional vector of the number of agents required for roles, i.e., $L[j] \in \{1, 2, \ldots, m\} (0 \le j < n)$.

Note: In the context of the TAP mentioned in Section II, L denotes the number of candidates needed for the positions within the new BG. As per Table I in Section II, $L = [2, 3, 4, 2]$. Here, we define n_a as $\sum_{j=0}^{n-1} L[j]$ for ease of subsequent analysis.

Definition 4: A *qualification matrix* [3] Q is an $m \times n$ matrix, where $Q[i, j]$ $(0 \le i < m,\ 0 \le j < n)$ expresses the qualification value of agent i for role j, i.e., Ⓠ in *Definition* 2.

Note: In the lifecycle of RBC, the Q matrix is produced during the agent evaluation phase (see Fig. 1). The values in the Q matrix reflect the suitability of candidates for positions in the new BG, gauged by their contributions to software development and algorithm design in the past projects.

In the real-world scenario, the Q matrix is presented in Table II. Note that the candidates come from different BGs, and different BGs have distinct employee evaluation metrics and weights. To balance these metrics, we normalize the Q

TABLE III
CONFLICTING RELATIONSHIPS BETWEEN CANDIDATES

Candidate \ Candidate	Alton	Blythe	Caela	Dax	Elvin	Faela	Gaige	Hael	Isra	Jacek	Kael	Lior	Maela
Alton	0 [a]	1	0	0	0	0	0	0	1	0	0	0	0
Blythe	1	0	1	0	0	1	0	0	0	0	0	1	0
Caela	0	1	0	0	0	0	0	0	0	0	0	0	1
Dax	0	0	0	0	0	0	0	0	0	0	0	0	0
Elvin	0	0	0	0	0	0	0	1	0	0	0	0	0
Faela	0	1	0	0	0	0	0	0	1	0	1	0	0
Gaige	0	0	0	0	0	0	0	0	0	0	0	0	0
Hael	0	0	0	0	1	0	0	0	0	0	0	0	0
Isra	1	0	0	0	0	1	0	0	0	0	0	0	0
Jacek	0	0	0	0	0	0	0	0	0	0	0	0	0
Kael	0	0	0	0	0	1	0	0	0	0	0	0	0
Lior	0	1	0	0	0	0	0	0	0	0	0	0	0
Maela	0	0	1	0	0	0	0	0	0	0	0	0	0

[a] Note: In Table III, the values are either 0 or 1. A value of 0 indicates that there is no conflict between two candidates, while a value of 1 indicates the presence of a conflict between them.

matrix, meaning $Q[i, j]$ lies between 0 and 1. $Q[i, j] = 0$ indicates the lowest value, and 1 is the highest.

Definition 5: A *role assignment matrix* [3], [29] T is defined as an $m \times n$ matrix, where $T[i, j] \in \{0, 1\}$ $(0 \leq i < m, 0 \leq j < n)$ indicates whether or not agent i is assigned to role j. $T[i, j] = 1$ means yes, and 0 no.

Note: The matrix T represents a matrix about the control variables.

Definition 6: The *group performance* $\sigma^{[-]}$ [3], [15] is defined as the sum of the assigned agents' qualifications for the [-] problem or model.

Note: For example, we utilize σ^{GRACAG} to identify the corresponding group performance of the GRACAG model.

Definition 7: A *conflicting agent matrix* A^c is a $m \times m$ matrix, and $A^c[i, i'] \in \{0, 1\}$ $(0 \leq i, i' < m)$.

Note: $A^c[i, i'] = 0$ (1) signifies a nonconflict (conflict) effect when agents i and i' concurrently work in the new BG.

Definition 8: Given Q, L, and A^c, the GRACAG model [3] is to find a matrix T to obtain

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

subject to

$$T[i, j] \in \{0, 1\} \; (0 \leq i < m, \; 0 \leq j < n) \tag{2}$$

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

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

$$A^c[i, i'] \times \left(T[i, j] + T[i', j']\right) \leq 1 (0 \leq i, \; i' < m, \; 0 \; \leq j, \; j' < n) \tag{5}$$

where Expression (2) expresses the 0-1 variable, Expression (3) indicates the number of agents required for roles, Expression (4) represents that one agent can only perform a role at most, and Expression (5) ensures that there are no conflicting agents assigned to the same group.

We attempt to formalize the TAP mentioned in Section II leveraging the previously defined GRACAG model, and employ the commercial ILP solvers (e.g., CPLEX [24] or Gurobi optimizer [25]) for its resolution. However, the output information from ILP solvers indicates that the GRACAG model cannot give a feasible solution for the TAP. As inferred from Tables I and II, $n_a = \sum_{j=0}^{n-1} L[j] < m$, suggesting that the constraints (2)–(4) of the GRACAG model are plausible. The unavailability of a solution for the GRACAG model is attributed to the limited number of agents, rendering the conditions of (4) unsatisfiable. Thus, a conflict-free solution is unattainable. At this juncture, the GRACAG model is inadequate for formalizing this TAP, prompting the need to devise a new model to address the issue.

IV. PROBLEM FORMALIZATIONS

When the number of resources available to agents is limited, an agent conflict-free solution for the TAP might not exist. Given the real-world scenario and requirements outlined in Section II, it is imperative to derive the optimal assignment with the premise of minimal agent conflicts. Namely, there are two nested objectives for the TAP. Based on this, the TAP can be formulated as the mathematical representation defined in Definition 9.

Definition 9: Given Q, L, and A^c, the TAP mentioned in Section II is to find a matrix T to obtain

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

subject to (2)–(4), and

$$T \in \underset{T}{\operatorname{argmin}} \left\{ \sum_{i=0}^{m-1} \sum_{i'=0}^{m-1} \sum_{j=0}^{n-1} \sum_{j'=0}^{n-1} A^c[i, i'] \times T[i, j] \times T[i', j'] \text{ subject to } (2)-(4) \right\} \tag{6}$$

where $\sum_{i=0}^{m-1} \sum_{i'=0}^{m-1} \sum_{j=0}^{n-1} \sum_{j'=0}^{n-1} A^c[i, i'] \times T[i, j] \times T[i', j']$ represents the number of agent conflicts. Expression (6) implies that the optimal group performance σ^{TAP} must be achieved on the premise of minimizing the number of conflicts. Intuitively, the TAP problem is a nonlinear, bilevel integer optimization problem, making it intractable to obtain the optimal solution leveraging commercial ILP solvers.

To solve the TAP in Definition 9, we first put forth a BM-TAP. To elucidate this solution more comprehensively, we present the following new definitions.

Definition 10: The *agent-role pair assignment matrix* $\widehat{T}$ is an $(m\times n)\times(m\times n)$ matrix, where $\widehat{T}[i, j, i', j'] = T[i, j] \times T[i', j']$ $(0 \le i, i' < m, 0\le j, j' < n)$.

Note: TAP in Definition 9 is a nonlinear integer programming formulation because of the expression $T[i, j] \times T[i', j']$. To address this challenge, we adopt a spatial dimensionality reduction approach, converting the two-dimensional variable matrix into one-dimensional counterparts.

Definition 11: The *minimum number of agent conflicts* c is defined as a precalculated constant, where $c \in \mathbb{N}^{+}$.

Note: Given Q, L, and A^c, calculating c is to find a matrix $\widehat{T}$ to obtain

$$c = \min \sum_{i=0}^{m-1}\sum_{i'=0}^{m-1}\sum_{j=0}^{n-1}\sum_{j'=0}^{n-1} A^c\left[i, i'\right] \times \widehat{T}\left[i, j, i', j'\right]$$

subject to (2)–(4) and

$$\widehat{T}\left[i, j, i', j'\right] \in \{0, 1\} \left(0 \le i, i' < m, \ 0 \le j, j' < n\right) \quad (7)$$

$$2\times\widehat{T}\left[i, j, i', j'\right] \le T\left[i, j\right] + T\left[i', j'\right] \quad (8)$$

$$\widehat{T}\left[i, j, i', j'\right] + 1 \ge T\left[i, j\right] + T\left[i', j'\right] \quad (9)$$

where Expressions (7)–(9) ensure the constraints that $T[i, j]\times T[i', j'] = 1$for $(0 \le i, i' < m, 0\le j, j' < n)$. It is easy to verify these transformations because the elements in T belong to the set $\{0, 1\}$.

In Expression (6), every element in the set to which T belongs has a common characteristic: their objective functions equate to the minimum number of conflicts. Therefore, we consider introducing a parameter c to serve as a constraint that equivalently represents Expression (6), as illustrated in Definition 12.

Definition 12: Given Q, L, A^c, and c, BM-TAP is to find a matrix T to obtain

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

subject to (2)–(4), (7)–(9), and

$$\sum_{i=0}^{m-1}\sum_{i'=0}^{m-1}\sum_{j=0}^{n-1}\sum_{j'=0}^{n-1} A^c\left[i, i'\right] \times \widehat{T}\left[i, j, i', j'\right] = c\left(0 \le i, i' < m, \ 0 \le j, j' < n\right). \quad (10)$$

With Definitions 11 and 12, we successfully transformed TAP into a bilevel integer programming problem, which can be solved utilizing the commercial CPLEX solver. Still, BM-TAP involves repeating searches in the $\widehat{T}$-matrix space, which can easily lead to a combinatorial explosion [30] as the scale of TAP increases. To address this issue, we aim to equivalently transform the bi-level model into a single-level one, i.e., the GRAMAC model. To elucidate this equivalent model, subsequent definitions are introduced.

Definition 13: The *conflict penalty coefficient* ϵ is defined as a sufficiently large constant, where $\epsilon = \max\{Q\}\times \sum_{j=0}^{n-1} L[j]$ and $\max\{Q\}$ represents the maximum value of the Q matrix.

Note: The conflict penalty coefficient ϵ represents the substantial cost incurred when selecting agents with conflicting relationships in the assignment scheme. To retain as much information from the Q matrix as possible, it is desirable to keep ϵ as small as possible. Based on this, we utilize $\max\{Q\}\times \sum_{j=0}^{n-1} L[j]$ to signify this substantial value, which corresponds to the theoretical maximum of the expression $\sum_{i=0}^{m-1} \sum_{j=0}^{n-1} Q[i, j] \times T[i, j]$.

Definition 14: Given Q, L, A^c, and ϵ, GRAMAC is to find a matrix T to obtain

$$\max\sigma^{\text{GRAMAC}} = \sum_{i=0}^{m-1}\sum_{j=0}^{n-1} Q\left[i, j\right] \times T\left[i, j\right] - \sum_{i=0}^{m-1}\sum_{i'=0}^{m-1}\sum_{j=0}^{n-1}\sum_{j'=0}^{n-1} \epsilon \times A^c\left[i, i'\right] \times \widehat{T}\left[i, j, i', j'\right] \quad (11)$$

subject to (2)–(4) and (7)–(9).

In Expression (11), the constant ϵ serves as a high selection cost to ensure that, when the group is comprised of nonconflicting agents, the value of the objective function is maximized. Here, we propose Theorem 1 to demonstrate that BM-TAP and GRAMAC are equivalent.

Theorem 1: BM-TAP and GRAMAC are equivalent.

Proof: The proof can be found in the supplementary file. ■

Note: Theorem 1 demonstrates that the GRAMAC model equivalently transforms a bilevel optimization problem into an x-ILP problem.

Both the GRAMAC and BM-TAP models can be directly solved using the Gurobi commercial optimization solver, with solution times of 0.16 and 0.18 s, respectively. It should be noted that the solution times for the two models are relatively close in this instance. This is because the scale of the TAP is relatively small. A more comprehensive performance analysis for these models under varying TAP scales will be presented in the experimental section.

Fig. 2 illustrates the same T matrix derived from both models. At this juncture, the optimal assignment we procured, with the agent conflicts minimized, is: <Alton, Algorithm Engineer>, <Caela, Software Engineer>, <Dax, Software Engineer>, <Elvin, Software Engineer>, <Faela, Data Engineer>, <Gaige, Automation Test Engineer>, <Hael, Data Engineer>, <Jacek, Algorithm Engineer>, <Kael, Data Engineer>, <Lior, Automation Test Engineer>, <Maela, Software Engineer>. Concurrently, the group performance σ^{TAP} stands at 8.03 and c is equal to 3. The conflicting agent pairs are identified as <Caela, Maela>, <Elvin, Hael>, and <Faela, Kael>.

While the GRACAG model finds no solution for the TAP problem described in Section II, our designed GRAMAC model can achieve an optimal assignment with a minimal number of agent conflicts. Nonetheless, several nontrivial

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

Fig. 2. T matrix in the real-world scenario.

challenges persist: 1) what is the computational complexity of solving the GRAMAC problem; 2) how do the GRAMAC and GRACAG models mathematically relate to each other; and 3) how can we further optimize the solution process for the GRAMAC model.

V. Problem Complexity and Provement

A. Problem Complexity

In this section, we begin by elucidating the computational complexity of the GRAMAC problem. In Definition 14, the formulation of the GRAMAC model is well-suited for computational solving but poses challenges for complexity analysis. Therefore, we introduce a more general form of the GRAMAC model as presented in Definition 15.

Definition 15: Given Q, L, A^c, and ϵ, the general form of GRAMAC is to find a matrix T to obtain

$$\max \sigma^{\text{GRAMAC}} = \sum_{i=0}^{m-1}\sum_{j=0}^{n-1} Q[i, j] \times T[i, j] - \sum_{i=0}^{m-1}\sum_{i'=0}^{m-1}\sum_{j=0}^{n-1}\sum_{j'=0}^{n-1} \epsilon \times A^{c\prime}[i, j, i', j'] \times T[i, j] \times T[i', j']$$

subject to (2)–(4), where $A^{c\prime}[i, j, i', j'] = A^c[i, i'](0 \le i, i' < m, 0 \le j, j' < n)$.

Note: In Definition 14, the matrix A^c is a simplified form of the GRAMAC model, as in the TAP the values in the matrix A^c are independent of the roles j and j'. Therefore, we use $A^c[i, i']$ to replace what should have been $A^{c\prime}[i, j, i', j']$ ($A^{c\prime}[i, j, i', j'] = A^c[i, i']$, $0 \le i, i' < m$, and $0 \le j, j' < n$). However, the overall number of iterations of the values in A^c is the same as that of in $A^{c\prime}$. Consequently, to facilitate the proof the complexity of GRAMAC, we modify objective function of the GRAMAC model in Definition 15 to $\max \sigma^{\text{GRAMAC}} = \sum_{i=0}^{m-1}\sum_{j=0}^{n-1} Q[i, j] \times T[i, j] + \sum_{i=0}^{m-1}\sum_{i'=0}^{m-1}\sum_{j=0}^{n-1}\sum_{j'=0}^{n-1} -\epsilon \times A^{c\prime}[i, j, i', j'] \times T[i, j] \times T[i', j']$.

Drawing from the mathematical formalization presented in Definition 15, we find that GRAMAC bears a resemblance to the 0-1 maximum quadratic assignment problem (0-1 Max-QAP). The latter has been demonstrated as a strongly $\mathcal{NP}$-*complete* problem [31], [32], [33]. Therefore, we propose Theorem 2 to prove that the GRAMAC problem is $\mathcal{NP}$-*complete* by reducing the 0-1 Max-QAP to it.

Algorithm 1: Nondeterministic Algorithm to Verify $\sigma^{\text{GRAMAC}} \ge b$

```
begin
σ^GRAMAC ← 0;
for i ← 0 to m−1 do
    for j ← 0 to n−1 do
        σ^GRAMAC ← σ^GRAMAC + Q[i, j] × T′[i, j];
for i ← 0 to m−1 do
    for i′ ← 0 to m−1 do
        for j ← 0 to n−1 do
            for j′ ← 0 to n −1 do
                σ^GRAMAC ← σ^GRAMAC − ε ×
                A^c′[i, j, i′, j′] × T′[i, j] × T′[i′, j′];
if σ^GRAMAC ≥ b then
    return True;
return False;
end
```

Theorem 2: The GRAMAC problem is $\mathcal{NP}$-*complete.*

Proof: To prove that the GRAMAC problem is $\mathcal{NP}$-*complete*, the demonstration primarily consists of the following two steps: 1) prove that the GRAMAC problem is in the $\mathcal{NP}$ (Nondeterministic Polynomial time) class and 2) prove that an already proven $\mathcal{NP}$-*complete* problem can be polynomially reduced to it.

1) GRAMAC Problem Is in the $\mathcal{NP}$ Class: In the complexity theory, as outlined in [33], a problem is classified as $\mathcal{NP}$ class if a solution can be conjectured using a nondeterministic algorithm and its correctness can be verified in polynomial time against the problem's constraints.

To rigorously prove (1), we consider a derived subproblem of GRAMAC, denoted as Π. Given a provisional assignment matrix T' for the subproblem Π, the objective of Π is to determine if the condition $\sigma^{\text{GRAMAC}} \ge b$ (where $\sigma^{\text{GRAMAC}} = \sum_{i=0}^{m-1}\sum_{j=0}^{n-1} Q[i, j] \times T'[i, j] - \sum_{i=0}^{m-1}\sum_{i'=0}^{m-1}\sum_{j=0}^{n-1}\sum_{j'=0}^{n-1} \epsilon \times A^{c\prime}[i, j, i', j'] \times T'[i, j] \times T'[i', j']$, and b is a constant) holds true. The emphasis is not on finding the maximized value of σ^{GRAMAC}, but rather on evaluating this specific inequality.

Here, we can design a nondeterministic algorithm to verify if both $\sigma^{\text{GRAMAC}} \ge b$ and the constraints (2)–(4) are satisfied for the updated assignment matrix T'. In this context, verifying (2)–(4) are straightforward. As for the expression $\sigma^{\text{GRAMAC}} \ge b$, the nondeterministic algorithm is (in pseudocode-like format) in Algorithm 1.

The time complexity of the provided algorithm is polynomial, specifically $O(m^2 n^2)$. Even with the inclusion of additional constraints for assessment, the overall time complexity remains $O(m^2 n^2)$. Therefore, we can affirm that Π resides within the $\mathcal{NP}$ class. This also confirms that the GRAMAC problem resides in the $\mathcal{NP}$ class.

2) GRAMAC Problem Is $\mathcal{NP}$-Complete: To prove that the GRAMAC problem is $\mathcal{NP}$-*complete*, it suffices to show that an already proven $\mathcal{NP}$-*complete* problem, i.e., 0-1 Max-QAP (see Definition 16), can be polynomially reduced to the GRAMAC problem.

Definition 16: The 0-1 Max-QAP [31], [32] is to

$$\max \sigma^{0-1\mathrm{Max-QAP}} = \sum_{e,\, f=0}^{w-1} B[e, f] \times Z[e, f] + \sum_{\substack{e,\, g=0 \\ e \neq f}}^{w-1} \sum_{\substack{f,\, h=0 \\ g \neq h}}^{w-1} C[e, f, g, h] \times Z[e, f] \times Z[g, h]$$

subject to

$$Z[e, f], B[e, f], C[e, f, g, h] \in \{0, 1\} \quad (0 \le e, f, g, h < w),$$
$$\sum_{e=0}^{w-1} Z[e, f] = 1 (0 \le f < w),$$
$$\sum_{f=0}^{w-1} Z[e, f] = 1 \ (0 \le e < w)$$

where $w \in \mathbb{N}^+$, the matrix C is an $(w \times w) \times (w \times w)$ constant matrix, B is a $w \times w$ constant matrix, and Z is a $w \times w$ variable matrix.

Theorem 2 is proved if we can transfer the GRAMAC problem to the 0-1 Max-QAP by restrictions and equivalent transformations. Therefore, we list the following restrictions for the GRAMAC problem.

Restriction 1: Let $m = n = w$.

Restriction 2: Let $i \neq i'$ and $j \neq j'$.

Restriction 3: Let "$\le$" be restricted to "=" in (4).

Restriction 4: Let $Q[i, j] \in [0, 1]$ be restricted to $Q[i, j] \in \{0, 1\}$.

Restriction 5: Let $\epsilon = -1$.

Restriction 1 ensures that the matrix T of control variables in the GRAMAC problem is square, and Restriction 2 establishes an one-to-one mapping between variables i, j, i', j' and variables e, f, g, h. Restriction 3 ensures an one-to-one $(1-1)$ assignment of roles to agents. With Restriction 4, let $B[\mathrm{e}, f] = Q[i, j]$ $(0 \le e, f, i, j < w)$, and the corresponding time complexity is $O(w^2)$. With Restriction 5, let $C[\mathrm{e}, f, g, h] = A^{c'}[i, j, i', j']$ $(0 \le e, f, g, h < w, 0 \le i, j, i', j' < w)$, and the corresponding time complexity is $O(w^4)$.

Now, the GRAMAC problem becomes a 0-1 Max-QAP. With the above analysis, the total time cost to transform the GRAMAC problem to the 0-1 Max-QAP is within polynomial time, i.e., $O(w^4)$. Since the GRAMAC problem becomes the 0-1 Max-QAP by restriction, the former is more complex than the latter which is strongly $\mathcal{NP}$-*complete*. Therefore, the GRAMAC problem is proven to be $\mathcal{NP}$-*complete*. ■

B. Sufficient and Necessary Condition

In the study by Zhu presented in [3], they note that GRACAG stands as a subproblem of the x-ILP problem, an $\mathcal{NP}$-*complete* issue. Meanwhile, in Theorem 2, we have validated that the GRAMAC challenge is also of $\mathcal{NP}$-*complete* nature. This drives us to probe deeper into the interrelation between the GRACAG and GRAMAC models. To elucidate this connection, we incorporate the subsequent new notations, theorem, and corresponding corollaries.

1) K_j represents a set of nonconflict optimal solution results of role j in the TAP, i.e., $K_j = \{<i, j> | T[i, j] = 1, 0 \le i < m$, and $0 \le j < n\}$. If $K_j \neq \emptyset$, we can deduce $|K_j| = L[j]$.
2) S_j^{GRAMAC} collects all the feasible K_js $(0 \le j < n)$ for the GRAMAC problem.
3) S_j^{GRACAG} is a collection of feasible K_js $(0 \le j < n)$ for the GRACAG problem.

Theorem 3: $\forall\, j \in \{0, 1, \ldots, n-1\}$, *if* $K_j \neq \emptyset$, then $S_j^{\mathrm{GRAMAC}} = S_j^{\mathrm{GRACAG}}$.

Proof: $\forall\, j \in \{0, 1, \ldots, n-1\}$, if $K_j \neq \emptyset$, then there is no conflict among agents throughout the entire group. That is, $c = 0$ and $\sum_{i=0}^{m-1} \sum_{i'=0}^{m-1} \sum_{j=0}^{n-1} \sum_{j'=0}^{n-1} A^c[i, i'] \times T[i, j] \times T[i', j'] = 0$. Thereby, to prove Theorem 3, it suffices to demonstrate the equivalence of expression $\sum_{i=0}^{m-1} \sum_{i'=0}^{m-1} \sum_{j=0}^{n-1} \sum_{j'=0}^{n-1} A^c[i, i'] \times T[i, j] \times T[i', j'] = 0$ and $A^c[i, i'] \times (T[i, j] + T[i', j']) \le 1 (0 \le i, i' < m, 0 \le j, j' < n)$. Since $A^c[i, i'] \in \{0, 1\}$, $T[i, j] \in \{0, 1\}$, $T[i', j'] \in \{0, 1\}$, and $\sum_{i=0}^{m-1} \sum_{i'=0}^{m-1} \sum_{j=0}^{n-1} \sum_{j'=0}^{n-1} A^c[i, i'] \times T[i, j] \times T[i', j'] = 0$, we can deduce that $\forall (0 \le i, i' < m, 0 \le j, j' < n)$ $A^c[i, i'] \times (T[i, j] + T[i', j']) \le 1$. In this case, the GRAMAC problem is transformed into the GRACAG problem.

Based on the above deduction, we can prove that $\forall\, j \in \{0, 1, \ldots, n-1\}$, if $K_j \neq \emptyset$, then $S_j^{\mathrm{GRAMAC}} = S_j^{\mathrm{GRACAG}}$. Proof concluded. ■

Corollary 1: The necessary and sufficient condition for a group to implement an agent conflict-free assignment strategy is the existence of a feasible solution to the GRACAG model.

Proof: In light of Theorem 3, we can deduce that when an agent nonconflict assignment strategy exists within a group, the GRACAG model has a solution, and it is identical to that of the GRAMAC model. Meanwhile, based on the constraints in Definition 8, it can be determined that when the GRACAG model possesses a feasible solution, this solution is an agent conflict-free scheme for the group. This leads to the implication that $\sum_{i=0}^{m-1} \sum_{i'=0}^{m-1} \sum_{j=0}^{n-1} \sum_{j'=0}^{n-1} A^c[i, i'] \times T[i, j] \times T[i', j'] = c = 0$. Hence, we can conclude the proof. ■

Corollary 2: The GRACAG problem is a subproblem of the GRAMAC problem.

Proof: The proof for Corollary 2 is direct and intuitive. Drawing from Theorem 3 and Corollary 1, we ascertain that the necessary and sufficient condition for the GRAMAC problem to have a nonconflict solution is the solvability of the GRACAG problem. As a logical progression, the GRACAG problem emerges as a subproblem of the GRAMAC problem. This completes our demonstration. ■

Note: Corollaries 1 and 2 elucidate not only the mathematical relationship between the two models but also bear significant engineering implications. As the number of agents, roles, and interagent conflicts escalates, the dimensionality of the matrix $\widehat{T}$ and searching space in GRAMAC surges, resulting in a heightened time complexity for its resolution. Nevertheless, the solvability of the GRACAG model can be employed to ascertain whether an agent conflict-free

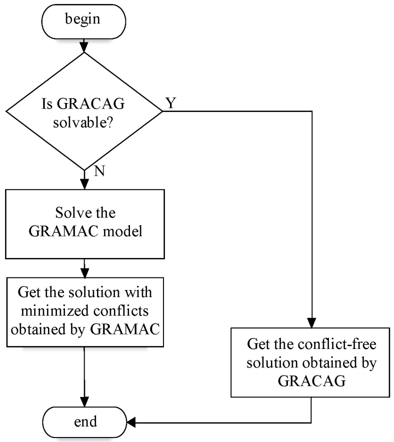


Fig. 3. Accelerated GRAMAC model via Corollaries 1 and 2.

assignment strategy exists for the current system, see Fig. 3. If a solution for the GRACAG model is present, then a non-conflict assignment strategy exists for the GRAMAC problem, aligning seamlessly with the solution of the GRACAG model. Such a strategy substantially streamlines the resolution process of GRAMAC in terms of time. Our subsequent simulation experiments will further corroborate this approach.

VI. Simulation Experiments

A. Experimental Setup

To demonstrate the viability and efficiency of our proposed method, we perform large-scale random simulation experiments on a MacOS-14.14 laptop outfitted with an Apple M2 Ultra processor and 64 GB LPDDR5 RAM. The simulations were implemented using Python 3.9.16 in visual studio code. In all subsequent comparative experiments, we utilize the Gurobi optimizer [25] to address all related models. The rationale behind adopting Gurobi stems from its cost-free access for researchers and its unrestricted support for constraints of any scale. In the experimental section, we continue to leverage the TAP, as introduced in Section II, as the backdrop for our comparison results.

The dimensionality of the TAP is chiefly determined by several parameters: the number of required positions (i.e., roles) within the new BG, denoted as n; the total number of candidates (i.e., agents), represented as m; the requisite number of candidates for each specific position, denoted as L; and the cardinality of the conflictual relationship among the candidates, represented by n_c ($n_c = \sum_{i=0}^{m-1}\sum_{i'=0}^{m-1} A^c[i, i']$).

Given that newly established regional branches typically consist of small to medium-sized teams, frequently numbering up to 100 individuals as cited in reference [14], [34], we designate the range for n to be 1–10 and for m to span from n to 100. Considering the TAP takes into account the situation where agent resources are limited, we strive to assign as many candidates as possible to their respective positions. Hence, we have set that $n_a = \sum_{j=0}^{n-1} L[j] \in \{n, n+1, n+2, \ldots, m\}$. Finally, candidates from different BGs have rare opportunities to have worked together in the past. Therefore, we empirically set the maximal value of n_c as the cardinality of agents. That is, $n_c \in \{0, 1, 2, \ldots, m\}$. For all subsequent simulation experiments, we randomly generated data based on the previously mentioned parameter ranges, unless stated otherwise. The results of each experiment are presented as the average values derived from hundreds of random trials.

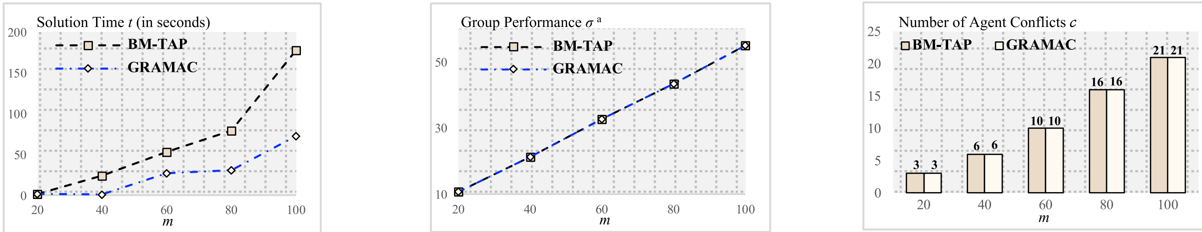
B. Performance Analysis of GRAMAC and BM-TAP

Initially, we contrast the performance of the bilevel optimization model BM-TAP designed in Section III and the GRAMAC model under different scales of TAPs. As illustrated in Fig. 4(a), it is evident that the computational time required for BM-TAP escalates sharply with an increase in the TAP scale, owing to the model's redundant search in the $\widehat{T}$-matrix space. This result accentuates the computational efficiency advantage of the GRAMAC model's single-level optimization. Concurrently, Fig. 4(b) and (c) demonstrates that both models yield consistent group performance and conflict quantities, thereby empirically validating the effectiveness of Theorem 1, which posits the equivalence of the two models.

C. Performance Analysis of Comparative Models Under Agent Conflict-Free Scenarios

Then, we seek to experimentally verify whether, in the presence of a conflict-free assignment scheme within the TAP, the group performance σ of solving the TAP using the GRACAG model aligns with that of the GRAMAC model. To facilitate a comprehensive comparison of different models, we set the value of n to 20, establish the range for m from n to 200, and assign the value of n_c to m. Concurrently, we set $n_a = 0.5m$ because GRACAG may not have a solution when the n_a is too close to m. We also introduce an ideal team performance measure for comparison, denoted as σ^{GRA}. It is calculated as $\sum_{i=0}^{m-1}\sum_{j=0}^{n-1} Q[i, j] \times T[i, j]$, subject to the conditions specified in Expressions (2) through (4). This measure signifies the ideal optimal assignment scheme when agent conflicts are not taken into consideration.

Table IV shows the group performance σ, solution time t, and the number of existing agent conflicts c for each model in TAP of varying scales. From Table IV, we observe that both the GRACAG and GRAMAC models can completely eliminate agent conflicts at the expense of sacrificing approximately 1% of group performance. Furthermore, we notice that the optimal solutions obtained by the GRAMAC model align consistently with those of the GRACAG model across all scales of TAPs. This observation also validates the correctness of Theorem 3, introduced earlier. Moreover, since the GRACAG problem is a subproblem of GRAMAC, its solution speed is faster than that of the GRAMAC model. Therefore, according to Corollary 2, as the scale of the problem under consideration increases, one can first employ the GRACAG model for solving. If a solution is found, it serves as the optimal solution for the GRAMAC model. This approach effectively accelerates the solving process for the GRAMAC problem.



(a) (b) (c)

Fig. 4. Performance analysis of the GRAMAC model and the BM-TAP model. [a]Note: To accelerate the solution time for the BM-TAP model, we set a tolerance level such that the difference between the obtained solution and the optimal solution is within two conflict numbers. (a) Solution times. (b) Group performance. (c) Number of agent conflicts.

TABLE IV
COMPARATIVE ANALYSIS OF MODEL SOLUTIONS FOR TAP WITH NONCONFLICTING OUTCOMES

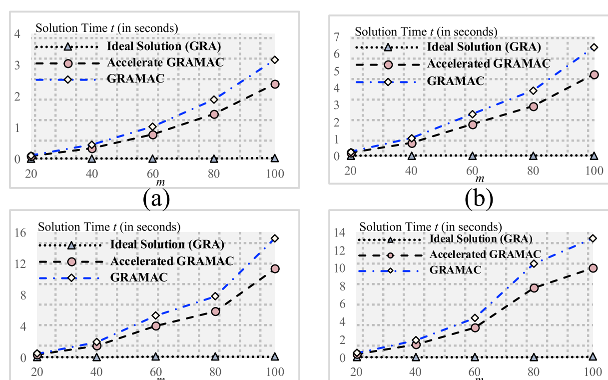
Scale (m_n)	GRA [a] (Ideal Solution)					GRACAG					GRAMAC				
	σ	t			c	σ	t			c	σ	t			c
		Max.	Min.	Ave. [b]			Max.	Min.	Ave.			Max.	Min.	Ave.	
20_10	9.62	0.02s	0.01s	0.01s	5	9.43	0.13s	0.10s	0.10s	0	9.43	0.51s	0.49s	0.50s	0
40_10	19.18	0.01s	0.01s	0.01s	9	18.69	0.41s	0.37s	0.38s	0	18.69	2.17s	2.03s	2.08s	0
60_10	28.32	0.02s	0.01s	0.02s	15	27.47	0.89s	0.84s	0.87s	0	27.47	4.84s	4.69s	4.74s	0
80_10	38.47	0.02s	0.02s	0.02s	21	37.46	1.55s	1.50s	1.52s	0	37.46	8.74s	8.54s	8.62s	0
100_10	**47.87**	0.03s	0.02s	0.03s	**25**	**46.51**	2.41s	2.35s	2.38s	**0**	**46.51**	14.13s	13.71s	13.87s	**0**
40_20	19.68	0.05s	0.01s	0.02s	9	19.48	1.54s	1.48s	1.51s	0	19.48	8.60s	8.24s	8.44s	0
80_20	39.49	0.04s	0.03s	0.03s	22	38.89	6.03s	5.89s	5.96s	0	38.89	36.60s	36.08s	36.35s	0
120_20	59.37	0.05s	0.04s	0.05s	31	58.74	13.70s	13.35s	13.54s	0	58.74	85.25s	83.81s	84.54s	0
160_20	78.72	0.07s	0.05s	0.06s	42	77.72	24.73s	23.17s	23.81s	0	77.72	155.95s	148.33s	150.87s	0
200_20	**98.53**	0.08s	0.06s	0.07s	**50**	**97.37**	36.89s	36.04s	36.46s	**0**	**97.37**	233.45s	231.07s	232.26s	**0**

[a] GRA represents the optimal assignment where agent conflicts are not considered. [b] Average. Symbol σ denotes the group performance, symbol t stands for the time required to find a solution, and symbol c signifies the number of existing agent conflicts obtained for each model.

D. Performance Analysis of Comparative Models Under Scenarios in the Presence of Agent Conflicts

Next, we investigate the performance of the accelerated GRAMAC model (see Fig. 3) when TAP may have a conflict-free solution. The presence or absence of a conflict-free solution for TAP is mainly influenced by two factors: 1) the total number of agents needed for the roles (i.e., n_a) and 2) the number of conflicts in the matrix A^c (i.e., n_c). We use a controlled variables approach to investigate the effects of these factors on the TAP. Initially, we generate a random count of conflicts spanning a range from n to m. Subsequently, we assess the variations in the solution performance of various models with the escalation of n_a.

Figs. 5–7 depict the average computational time t, the average group performance σ, and the average number of agent conflicts c for the models being compared as the ratio of (n_a/m) increases. Fig. 5 depicts that, with the scale of TAP increases, the acceleration effect of the GRAMAC model using the flowchart depicted in Fig. 3 becomes increasingly pronounced. Figs. 6 and 7 also indicate that as the scale increases, the number of agent conflicts obtained by the GRA model also increases. In contrast, the GRAMAC model sacrifices a certain degree of group performance to minimize agent conflicts compared to the GRA model. Note that we have not included the performance of the GRACAR model, as



(a) (b) (c) (d)

Fig. 5. Computational time of compared models under different scales. (a) $n = 4$. (b) $n = 6$. (c) $n = 8$. (d) $n = 10$.

this model may become infeasible when the ratio of (n_a/m) increases.

Fig. 8 reveals the specific proportion at which the count of n_a approaches or equates the total number of agents m, making a conflict-free solution for the TAP largely improbable. A clear observation from Fig. 8 indicates that when the ratio (n_a/m) exceeds 70%, the system predominantly lacks a conflict-free solution. At this point, the GRACAG model has no solution,

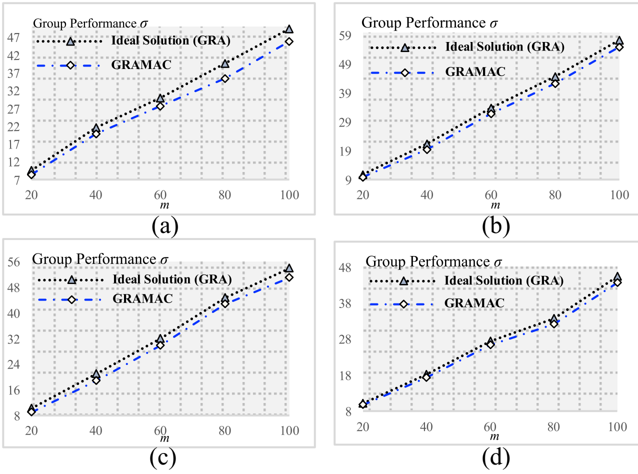


Fig. 6. Group performance of compared models under different scales. (a) $n = 4$. (b) $n = 6$. (c) $n = 8$. (d) $n = 10$.

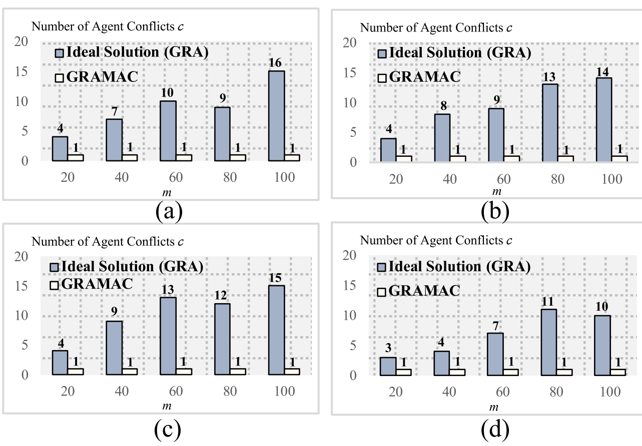


Fig. 7. Number of agent conflicts obtained by the compared models under different scales. (a) $n = 4$. (b) $n = 6$. (c) $n = 8$. (d) $n = 10$. Note: Given that the GRACAG model can be infeasible, the number of conflicts arising in this model is not presented.

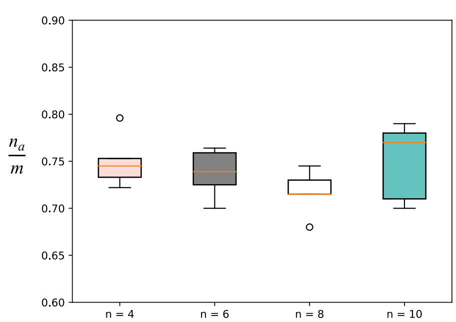


Fig. 8. Boxplot of (n_a/m) ratios in cases where TAP lacks agent conflict-free solutions.

rendering the acceleration strategy in Fig. 3 for GRAMAC ineffective.

Then, we investigate the relationship between the number of agent conflicts n_c in the matrix A^c and the existence of agent conflict-free solutions for the TAP. Similarly, we employ the method of controlled variables to investigate the performance of various models in solving the TAP. We specifically set the

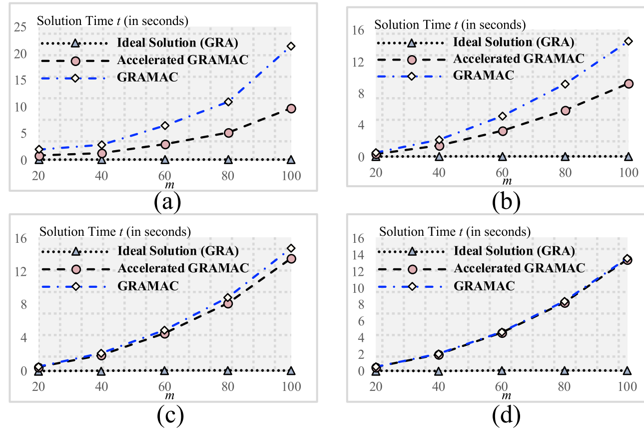


Fig. 9. Computational time of compared models under different (n_a/m) ratios. (a) $(n_a/m) = 0.4$. (b) $(n_a/m) = 0.6$. (c) $(n_a/m) = 0.8$. (d) $(n_a/m) = 1.0$.

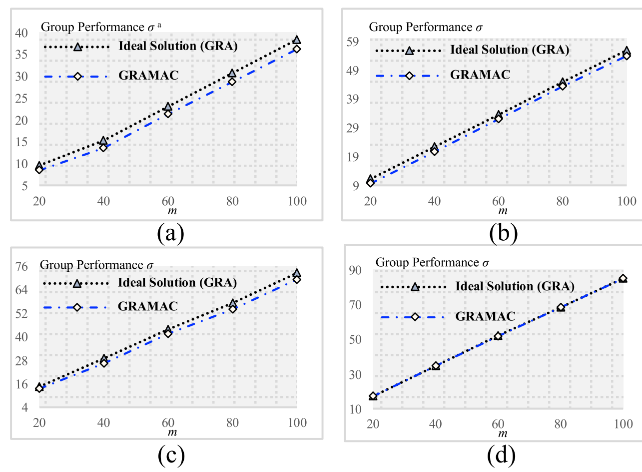


Fig. 10. Group performance of compared models under different (n_a/m) ratios. (a) $(n_a/m) = 0.4$. (b) $(n_a/m) = 0.6$. (c) $(n_a/m) = 0.8$. (d) $(n_a/m) = 1.0$.

ratio of the required number of agents to the total number of agents (n_a/m) at predetermined values of 0.4, 0.6, 0.8, and 1.0. Meanwhile, we set $n = 10$ and m ranges from n to 100. Within these settings, we incrementally increase the number of agent conflicts n_c to assess the performance of the compared models in resolving TAP.

Figs. 9–11 illustrate the variations in the average solution time, the average group performance, and the average number of agent conflicts for different models as the agent conflict number n_c increases under various (n_a/m). Fig. 9 reveals that as (n_a/m) increases, the solvability rate of the GRACAG model gradually decreases. In fact, when $(n_a/m) = 1$, the GRACAG model fails to find a solution, rendering it entirely ineffective when the number of required agents n_a approaches the total number of agents m.

Consistent with the variations observed in the (n_a/m) experiments, Figs. 10 and 11 illustrate that the GRAMAC model manages to significantly reduce the number of agent conflicts c compared to the GRA model, albeit at the expense of a slight loss in group performance σ. Notably, as the ratio of (n_a/m) increases, the minimal compromise in group

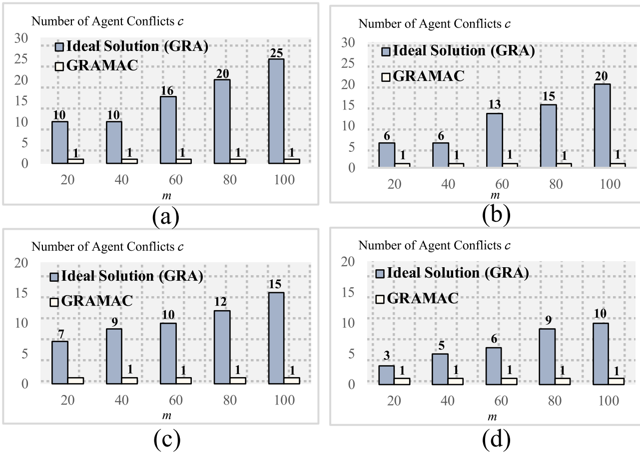


Fig. 11. Number of agent conflicts obtained by the compared models under different (n_a/m) ratios. (a) $(n_a/m) = 0.4$. (b) $(n_a/m) = 0.6$. (c) $(n_a/m) = 0.8$. (d) $(n_a/m) = 1.0$.

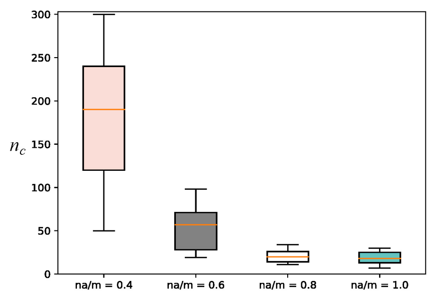


Fig. 12. Boxplot illustrating the distribution of n_c in cases where TAP lacks agent conflict-free solutions.

performance is approximately 0.1%. This also implies that our proposed GRAMAC model is capable of optimizing group performance while minimizing agent conflicts, particularly under conditions of limited agent resources.

Fig. 12 shows the threshold values of n_c for different (n_a/m) ratios, beyond which the TAP lacks conflict-free agent solutions. The findings indicate that as the required number of agents (n_a) increases relative to the total number of agents (m), the threshold value for n_c diminishes. This suggests that when n_a approaches m, the number of tolerable agent conflicts for obtaining a conflict-free solution significantly decreases. In the most extreme case, where (n_a/m) is 1, an agent conflict-free solution for TAP becomes unattainable when n_c reaches 8. This also explains why the solution time curve of the accelerated GRAMAC model in Fig. 9(d) almost coincides with the solution time curve of the GRAMAC model.

E. Performance Evaluation of Comparative Models Exclusively in Agent Conflict Scenarios

Finally, we assess the performance of the GRAMAC model when there are no agent conflict-free solutions to the TAP. Building on the findings from our previous experiments, we set the ratio of (n_a/m) to range from 0.8 to 1 and the overall number of agent conflicts n_c as m. Fig. 13(a)–(c) displays the average computation time, the average group performance, and the average obtained number of agent conflicts for the GRAMAC model under these conditions. Consistent with our previous findings, the GRAMAC model effectively maximizes group performance within an acceptable time frame while minimizing agent conflicts, with an average reduction of approximately 30%.

Based on the experiments conducted, we can conclude that as follows.

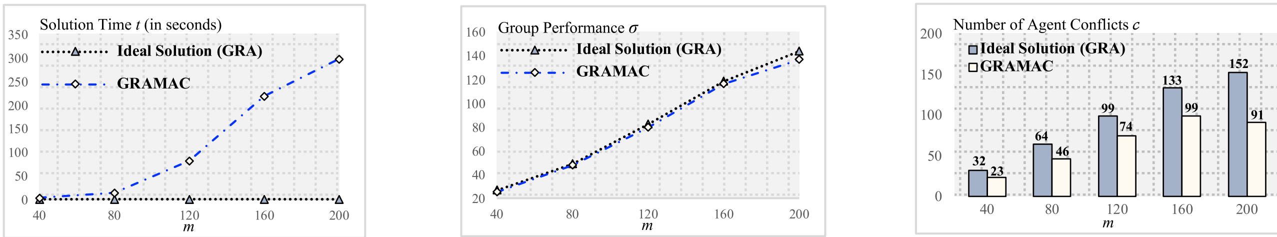
1) GRAMAC serves as an effective approach for solving the TAP outlined in Definition 9, which is characterized as a nonlinear bi-level integer optimization problem.
2) When a conflict-free solution exists for TAP, the solutions obtained by the GRAMAC model are consistent with those from the GRACAG model. This further validates that the GRACAG problem is a subproblem of the GRAMAC problem.
3) Since the solvability of GRACAG is both a necessary and sufficient condition for the GRAMAC model to have a solution without agent conflicts, we can utilize the flowchart from Fig. 3 to expedite the solution process for the GRAMAC model.
4) When agent resources are scarce, leading to the absence of agent conflict-free solutions to the TAP, the GRACAG model becomes ineffective.
5) Compared to the existing GRA+ model, the proposed model achieves optimal assignment schemes while minimizing the number of agent conflicts. This also addresses the limitations of the GRA+ model in dealing with TAPs where conflicts are unavoidable.
6) In the scheduling of complex real-world collaborative systems, scalability is a critical concern as an increase in the number of agents (e.g., humans, machines, and objects) and interactions can lead to a rise in conflicts and potential deadlocks, causing system downtime. The proposed GRAMAC model effectively reduces agent conflicts by optimizing resource allocation, demonstrating its ability to maintain system stability even as the complexity of the system scales up.

VII. Related Work

In task assignment problem [1], [35], [36], [37], role assignment influenced by agent conflicts [35], [38] is pivotal and presents considerable challenges in diverse fields, such as organizational performance, system construction, and system management [3], [39]. Therefore, in the realm of task assignment problems, avoiding conflicts between agents has garnered significant scholarly attention. Numerous researchers have devised models or algorithms applicable to various scenarios to mitigate or eliminate such conflicts [35], [36], [37], [40], [41].

Yang and Hu [36] design a distributed approach for resolving access control conflicts in multidomain environments. Their real-time algorithms effectively manage role inheritance violations and separation of duty violations, providing a scalable solution for large-scale systems.

Bai et al. [37] presented a decentralized auction algorithm that not only optimizes task assignments for dispersed robots but also constructs conflict-free target assignments. Contrary to common intuition, this article demonstrates that a longer



(a) (b) (c)

Fig. 13. Performance of the GRAMAC model in TAP without agent conflict-free solutions. (a) Solution times. (b) Group performance. (c) Number of agent conflicts.

communication range does not necessarily improve algorithmic performance.

Deng et al. [41] proposed a two-phase architecture for coordinated planning of heterogeneous Earth-observation resources, focusing on mitigating task conflicts. The task conflict heuristic allocation (TCHA) method is specifically designed to allocate subtasks to different planning centers efficiently. The approach has been validated through extensive simulations, demonstrating improved efficiency in coordinated task planning.

The aforementioned studies have achieved significant results in their respective research domains and scenarios. This article focuses on avoiding agent conflicts in the TAP, and due to varying scenario constraints, it is difficult to directly compare the effectiveness of different models and algorithms. Still, these studies offer interesting insights and could serve as a basis for evaluating the performance and complexity tradeoffs in our proposed solutions.

Recently, some scholars have leveraged the GRA+ models cooperative with the RBC methodology [2], [3], [4], [14], [16], to properly construct a collaborative system that considers conflict relationships between agents.

For example, Zhu [3] designed the GRACAG model, which considers the conflict between team members, which can help team leaders form a conflict-free team. In real-world scenarios, it is common to encounter situations where agent resources are limited. When the demand for these resources increases, finding a conflict-free team may become impossible. Under such conditions, the GRACAG model is unable to provide a solution, as it is designed to only identify the optimal conflict-free team.

Another approach to resolving member conflicts within the TAP is the use of the GRA with cooperation and conflict factors (GRACCF) model [14], [16]. A crucial metric for applying this model is the preselection of a C^{cf} matrix, which quantifies the degree of conflict and collaboration among team members and serves as the basis for optimal team assignment. However, quantifying this metric is challenging in real-world scenarios. Moreover, the model does not guarantee a minimization of team member conflicts.

The aforementioned limitation highlights the necessity for further research to extend the existing GRA+ model to accommodate both resource-unconstrained and resource-constrained scenarios, especially when conflict-free teams are not attainable. The GRAMAC model introduced in this article emerges as a solution to address these challenges.

VIII. Conclusion and Future Work

GRAMAC is a complex problem because it is intuitively a nonlinear, bilevel integer optimization problem. Initially, we rigorously formalize the problem constraints and objectives. Subsequently, we present the GRAMAC model as a viable solution approach. Our analysis confirms that solving the GRAMAC problem falls within the realm of $\mathcal{NP}$-*complete* problems. Furthermore, we establish that the pre-existing GRACAR model serves as a specialized subproblem of the GRAMAC framework, acting as both a necessary and sufficient condition for realizing an agent conflict-free solution. Finally, relevant simulation experiments are conducted to verify the efficiency and practicality of the proposed methods.

In the future, we will do further research to make the GRAMAC model adaptable to take into account cooperative relationships between agents. Moreover, we will also enhance the applicability of the GRAMAC model in extreme conditions. Here, are the directions for future enhancements.

1) When the number of conflict relations in the matrix A^c dramatically increases, we need to identify either sufficient or necessary conditions that could speed up the solution process for the GRAMAC model.
2) Considering that minimizing agent conflicts may lead to higher computational costs, we can consider using the number of agent conflicts and group expressiveness as dual objectives for multiobjective optimization. We can solve this using the latest BGRA++ model and the GRA-NSGA-II algorithm [42], with the corresponding solutions serving as a strategy for optimizing the GRAMAC model.
3) When solving a larger-scale GRAMAC problem, we consider combining the GRAMAC model with reinforcement learning algorithms to achieve the optimal solution in a reasonable time.

Acknowledgment

The authors would like to thank the anonymous reviewers and the senior editors for their insightful comments that improved the presentation of this article.

References

[1] H. Zhu, *E-CARGO and Role-Based Collaboration: Modeling and Solving Problems in the Complex World*. New York, NY, USA: Wiley/IEEE Press, 2022.

[2] 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.
[3] H. Zhu, "Avoiding conflicts by group role assignment," *IEEE Trans. Syst. Man, Cybern., Syst.*, vol. 46, no. 4, pp. 535–547, Apr. 2016.
[4] 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.
[5] 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.
[6] 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.*, vol. 52, no. 11, pp. 6968–6982, Nov. 2022.
[7] 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. Comput. Soc. Syst.*, vol. 8, no. 3, pp. 668–681, Jun. 2021.
[8] Q. Jiang, D. Liu, H. Zhu, Y. Qiao, and B. Huang, "Quasi group role assignment with role awareness in self-service spatiotemporal crowdsourcing," *IEEE Trans. Comput. Soc. Syst.*, vol. 9, no. 5, pp. 1456–1468, Oct. 2022.
[9] W. Zhang, E. Bai, and J. Li, "Speeding up the schedulability analysis and priority assignment of sporadic tasks under uniprocessor FPNS," *IEEE Trans. Ind. Informat.*, vol. 16, no. 10, pp. 6382–6392, Oct. 2020.
[10] B. Wang, W. Chen, B. Zhang, Y. Zhao, and P. Shi, "Cooperative control-based task assignments for multiagent systems with intermittent communication," *IEEE Trans. Ind. Informat.*, vol. 17, no. 10, pp. 6697–6708, Oct. 2021.
[11] D. Liu, Q. Jiang, H. Zhu, and B. Huang, "Distributing UAVs as wireless repeaters in disaster relief via group role assignment," *Int. J. Coop. Inf. Syst.*, vol. 29, nos. 1–2, 2020, Art. no. 2040002.
[12] 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.
[13] Q. Jiang, H. Zhu, Y. Qiao, D. Liu, and B. Huang, "Refugee resettlement by extending group multirole assignment," *IEEE Trans. Comput. Soc. Syst.*, vol. 10, no. 1, pp. 36–47, Feb. 2023.
[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. Zhou, and R. Alkins, "Group role assignment via a Kuhn–Munkres algorithm-based solution," *IEEE Trans. Syst. Man, Cybern. Part A, Syst. Humans*, vol. 42, no. 3, pp. 739–750, May 2012.
[16] Q. Jiang, H. Zhu, Y. Qiao, D. Liu, and B. Huang, "Extending group role assignment with cooperation and conflict factors via KD45 logic," *IEEE Trans. Comput. Soc. Syst.*, vol. 10, no. 1, pp. 178–191, Feb. 2023.
[17] H. Zhu, "Group role assignment with constraints (GRA+): A new category of assignment problems," *IEEE Trans. Syst. Man, Cybern., Syst.*, vol. 53, no. 3, pp. 1670–1683, Mar. 2023.
[18] H. Zhu and M. Zhou, "Role transfer problems and algorithms," *IEEE Trans. Syst. Man, Cybern. Part A, Syst. Humans*, vol. 38, no. 6, pp. 1442–1450, Nov. 2008.
[19] H. Zhu and M. Zhou, "Efficient Role Transfer Based on Kuhn–Munkres Algorithm," *IEEE Trans. Syst. Man, Cybern. Part A, Syst. Humans*, vol. 42, no. 2, pp. 491–496, Mar. 2012.
[20] C. Tomlin, G. J. Pappas, and S. Sastry, "Conflict resolution for air traffic management: A study in multiagent hybrid systems," *IEEE Trans. Automat. Control*, vol. 43, no. 4, pp. 509–521, Apr. 1998.
[21] H. Zhu, "Group multi-role assignment with conflicting roles and agents," *IEEE/CAA J. Automatica Sinica*, vol. 7, no. 6, pp. 1498–1510, Nov. 2020.
[22] E. Brownlee and H. Zhu, "Group role assignment with busyness degree and cooperation and conflict factors," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. (SMC)*, 2020, pp. 1479–1484.
[23] D. Y. Gao, "Canonical duality theory and algorithm for solving bilevel knapsack problems with applications," *IEEE Trans. Syst. Man, Cybern., Syst.*, vol. 51, no. 2, pp. 893–904, Feb. 2021.
[24] (IBM, Armonk, NY, USA). *IBM ILOG CPLEX Optimization Package*. (2023). [Online]. Available: http://www-01.ibm.com/software/integration/optimization/cplex-optimization-studio/
[25] (Gurobi Optim., Beaverton, OR, USA). *Gurobi Optimizer*. (2023). [Online]. Available: https://www.gurobi.com/faqs/gurobi-optimizer/
[26] R. Wachowiak-Smolíková and H. Zhu, "Data analytics and visualization of adaptive collaboration simulations," *IEEE Trans. Comput. Soc. Syst.*, vol. 10, no. 1, pp. 84–93, Feb. 2023.
[27] H. Zhu, Z. Yu, and Y. Gningue, "Solving the exam scheduling problem with GRA+," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. (SMC)*, 2020, pp. 1485–1490.
[28] H. Zhu, D. Liu, S. Zhang, Y. Zhu, L. Teng, and S. 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.
[29] 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.
[30] C. H. Papadimitriou, "On the complexity of integer programming," *J. ACM*, vol. 28, no. 4, pp. 765–768, Oct. 1981.
[31] S. Cai and K. Knight, "Smatch: An evaluation metric for semantic feature structures," in *Proc. 41st Annu. Meeting Assoc. Comput. Linguist. (ACL)*, 2013, pp. 748–752.
[32] V. Nagarajan and M. Sviridenko, "On the maximum quadratic assignment problem," *Math. Oper. Res.*, vol. 34, no. 4, pp. 859–868, 2009.
[33] M. R. Garey and D. S. Johnson, *Computers and Intractability; A Guide to the Theory of NP-Completeness*. New York, NY, USA: W. H. Freeman & Co., 1990.
[34] Q. Jiang, D. Liu, H. Zhu, Y. Qiao, and B. Huang, "Equilibrium means equity? An E-CARGO perspective on the golden mean principle," *IEEE Trans. Comput. Soc. Syst.*, vol. 10, no. 4, pp. 1443–1454, Aug. 2023.
[35] H. Kuang, M. A. Bashar, K. W. Hipel, and D. M. Kilgour, "Grey-based preference in a graph model for conflict resolution with multiple decision makers," *IEEE Trans. Syst. Man, Cybern., Syst.*, vol. 45, no. 9, pp. 1254–1267, Sep. 2015.
[36] B. Yang and H. Hu, "Secure conflicts avoidance in multidomain environments: A distributed approach," *IEEE Trans. Syst. Man, Cybern., Syst.*, vol. 51, no. 9, pp. 5478–5489, Sep. 2021.
[37] X. Bai, W. Yan, and S. S. Ge, "Distributed task assignment for multiple robots under limited communication range," *IEEE Trans. Syst. Man, Cybern., Syst.*, vol. 52, no. 7, pp. 4259–4271, Jul. 2022.
[38] S. Osborn, R. Sandhu, and Q. Munawer, "Configuring role-based access control to enforce mandatory and discretionary access control policies," *ACM Trans. Inf. Syst. Secur.*, vol. 3, no. 2, pp. 85–106, May 2000.
[39] M. Dastani, V. Dignum, and F. Dignum, "Role-assignment in open agent societies," in *Proc. 2nd Int. Joint Conf. Auton. Agents Multiagent Syst.*, 2003, pp. 489–496.
[40] H. Kim and M. Zhu, "Optimal bilevel lottery design for multiagent systems," *IEEE Trans. Syst. Man, Cybern., Syst.*, vol. 53, no. 11, pp. 7177–7187, Nov. 2023.
[41] M. Deng et al., "A two-phase coordinated planning approach for heterogeneous earth-observation resources to monitor area targets," *IEEE Trans. Syst. Man, Cybern., Syst.*, vol. 51, no. 10, pp. 6388–6403, Oct. 2021.
[42] Q. Jiang et al., "Iterative role negotiation via the bilevel GRA++ with decision tolerance," *IEEE Trans. Comput. Soc. Syst.*, vol. 11, no. 6, pp. 7484–7499, Dec. 2024, doi: 10.1109/TCSS.2024.3409893.

Qian Jiang received the Ph.D. degree in intelligent science and system form Macau Institute of Systems Engineering, Macau University of Science and Technology, Macau, China.

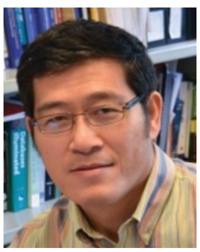
He has published seven regular papers on IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: SYSTEMS, IEEE TRANSACTIONS ON COMPUTATIONAL SOCIAL SYSTEMS, and *International Journal of Cooperative Information Systems*. His research interests include human–machine Systems, computational social simulation, and scheduling and optimization. His research page is: https://jiangqian1997.github.io.

Dr. Jiang has served as a reviewer for IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING and IEEE TRANSACTIONS ON COMPUTATIONAL SOCIAL SYSTEMS.

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

He has been a Full Professor with Guangdong University of Technology, Guangzhou, since 2009, where he is the Vice Dean of the School of Computer Science and Technology, responsible for teaching artificial intelligent logic and discrete math. He is engaged in education and technology transfer on collaborative computing. He has published more than 50 papers on computer magazines and international conferences.

Dr. Liu is a reviewer of IEEE Transactions on Systems, Man, and Cybernetics, IEEE Transactions on Computational Social Systems, and IEEE Transactions on Vehicular Technology. He is serving as the TC Member for the Technical Committee of Distributed Intelligent Systems, a senior member for the CCF Society, and the Vice Secretary-General and a Standing Committee Member of the Technical Committee on Cooperative Computing of the China Computer Federation. He is a member of the IEEE SMC Society.



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

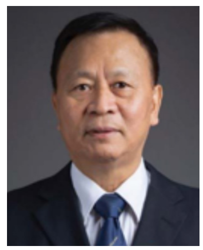
He is a Full Professor and the Coordinator of the Computer Science Program, the Founding Director of the Collaborative Systems Laboratory, Nipissing University, North Bay, ON, Canada. He is also an Affiliate Professor with Concordia University, Montreal, QC, Canada, and an Adjunct Professor with Laurentian University, Sudbury, ON. He was the Chair of the Department of Computer Science and Mathematics, Nipissing University from 2019 to 2021, 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 is the Founding Researcher of *Role-Based Collaboration and Adaptive Collaboration* and the Creator of *the E-CARGO model.* He has accomplished (published or in press) over 230 research works, including 30+ IEEE Transactions articles, six books, five book chapters, four journal issues, and four conference proceedings. His research monograph *E-CARGO and Role-Based Collaboration* can be found at https://www.amazon.com/CARGO-Role-Based-Collaboration-Modeling-Problems/dp/1119693063.

Prof. Zhu is serving as a Member-at-Large for the Board of Governors and a Co-Chair for the Technical Committee of *Distributed Intelligent Systems* of IEEE Systems, Man and Cybernetics (SMC) Society (SMCS), the Editor-in-Chief for *IEEE SMC Magazine*, an Associate Editor for IEEE Transactions on SMC: Systems, IEEE Transactions on Computational Social Systems, *Frontiers of Computer Science*, and IEEE Canada Review. He was the Associate Vice President of Systems Science and Engineering in 2021 and IEEE SMCS, a program (co-)chair of many international conferences, and a PC member of 130+ academic conferences. He is a member of the University Budget Plan Committee and the Arts and Science Executive Committee of Nipissing University.

Baoying Huang 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, and the Ph.D. degree in intelligent science and system from the Institute of Systems Engineering, Macau University of Science and Technology, Macau, China, in 2024.

She has published seven regular papers in IEEE Transactions on Systems, Man, and Cybernetics: Systems, IEEE Transactions on Automation Science and Engineering, and IEEE Transactions on Computational Social Systems, respectively. Her research interests include scheduling and optimization, human–machine systems, computational social simulation, and semiconductor manufacturing systems.

Dr. Huang has served as a reviewer for a number of journals and conferences.



Naiqi Wu (Fellow, IEEE) received the B.S. degree in electrical engineering from Anhui University of Technology, Huainan, China, in 1982, and the M.S. and Ph.D. degrees in systems engineering from Xi'an Jiaotong University, Xi'an, China, in 1985 and 1988, respectively.

From 1988 to 1995, he was with Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang, China, and from 1995 to 1998, he was with Shantou University, Shantou, China. He moved to Guangdong University of Technology, Guangzhou, China, in 1998. He joined Macau University of Science and Technology, Macau, China, in 2013, where he is currently a Professor with the Institute of Systems Engineering. His research interests include production planning and scheduling, manufacturing system modeling and control, discrete event systems, Petri net theory and applications, intelligent transportation systems, and energy systems.

Yan Qiao (Senior Member, IEEE) received the B.S. and Ph.D. degrees in industrial engineering and mechanical engineering from 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 Postdoctoral Research Associate with the Institute of Systems Engineering, Macau University of Science and Technology, Macau, China, where he is currently an Associate Professor with the Institute of Systems Engineering. He has over 120 publications, including one book chapter and 50+ regular papers in IEEE Transactions.

Dr. Qiao was a recipient of the 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.