Bulletin of Electrical Engineering and Informatics
Vol. 10, No. 2, April 2021, pp. 1092~1104
ISSN: 2302-9285, DOI: 10.11591/eei.v10i2.2395

Towards a formal analysis of the multi-robot task allocation problem using set theory

Farouq Zitouni[1], Ramdane Maamri[2], Saad Harous[3]
[1]Department of Computer Science, Kasdi Merbah University-Ouargla, Algeria
[1,2]LIRE Laboratory, Abdelhamid Mehri University-Constantine, Algeria
[2]Department of Computer Science, Abdelhamid Mehri University-Constantine, Algeria
[1,3]Department of Computer Science and Software Engineering UAE University-Abu Dhabi, United Arab Emirates

Article Info

Article history:

Received Mar 17, 2020
Revised Dec 31, 2020
Accepted Feb 6, 2021

Keywords:

Assigning papers to reviewers
Generic solving scheme
Multi-robot systems
Set theory
Task allocation problem
Time complexity

ABSTRACT

Nowadays, the multi-robot task allocation problem is one of the most challenging problems in multi-robot systems. It concerns the optimal assignment of a set of tasks to several robots while optimizing a given criterion subject to some constraints. This problem is very complex, particularly when handling large groups of robots and tasks. We propose a formal analysis of the task allocation problem in a multi-robot system, based on set theory concepts. We believe that this analysis will help researchers understand the nature of the problem, its time complexity, and consequently develop efficient solutions. Also, we used that formal analysis to formulate two well-known taxonomies of multi-robot task allocation problems. Finally, a generic solving scheme of multi-robot task allocation problems is proposed and illustrated on assigning papers to reviewers within a journal.

Corresponding Author:

Farouq Zitouni
Department of Computer Science,
Kasdi Merbah University-Ouargla, Algeria
Email: farouqzitouni@uaeu.ac.ae

1. INTRODUCTION

During the last decade, intensive research on multi-robot systems (MRS) has been conducted to solve several real-world problems. Usually, MRS are fault-tolerant and can efficiently solve complex problems [1-3]. The issue of task allocation is one of the main challenges in MRS. In such a situation, a complex problem is usually divided into elementary and straightforward tasks, where each one is then assigned to a robot or several robots [4]. Solving a multi-robot task allocation (MRTA) problem is to assign tasks to robots while optimizing a given objective function under several constraints [5].

Several centralized (i.e., using a central agent) and decentralized [5-7] approaches have been proposed to solve MRTA problems in the literature. Centralized systems reduce information duplication, and the check of generated solutions' quality is effortless [8]. However, maintaining connectivity between robots and the central agent is difficult, especially in large real-world problems [5]. Also, a load of communications on the central agent can sometimes exceed its tolerable threshold. Besides, centralized approaches are not robust (i.e., if the central agent fails, the system fails too). Decentralized approaches generally overcome the disadvantages of centralized ones, particularly robustness. However, no guarantee [2] is provided on the quality of generated solutions. Currently, we mainly distinguish three taxonomies of MRTA problems, i.e., each one classifies MRTA instances according to some features and criteria [9-11]. Taxonomies [9, 10] are certainly the most adopted. We discuss them in the next paragraphs.

The taxonomy proposed in work [9] counts eight classes: i) [single-task robots, single-robot tasks, instantaneous assignment]; ii) [single-task robots, single-robot tasks, time-extended assignment]; iii) [single-task robots, multi-robot tasks, instantaneous assignment]; iv) [single-task robots, multi-robot tasks, time-extended assignment]; v) [multi-task robots, single-robot tasks, instantaneous assignment]; vi) [multi-task robots, single-robot tasks, time-extended assignment]; vii) [multi-task robots, multi-robot tasks, instantaneous assignment], and viii) [multi-task robots, multi-robot tasks, time-extended assignment]. In the rest of the paper, this taxonomy is called "taxonomy A" for simplicity's sake.

- Single-task robots: This means that each robot can perform a single task at the same time.
- Multi-task robots: This means that some robots can perform multiple tasks simultaneously.
- Single-robot tasks: This means that each task requires exactly one robot for its accomplishment.
- Multi-robot tasks: This means that some tasks require the cooperation of several robots for their accomplishment.
- Instantaneous assignment: This means that the available information only allows instantaneous allocations of tasks to robots, i.e., we do not use time windows, precedence, or synchronization constraints.
- Time-extended assignment: This means that more information is available, i.e., we use time windows, precedence, or synchronization constraints, which allows time-extended allocations of tasks to robots; in other words, the use of planning is unavoidable.

The taxonomy proposed in work [10] distinguishes 21 classes, which are: no dependencies (1 or 2), in-schedule dependencies (2, 5 or 6), cross-schedule dependencies (1, 2, 3, 4, 5, 6, 7 or 8), and complex dependencies (1, 2, 3, 4, 5, 6, 7 or 8). Where the numbers between parentheses represent one of the eight classes defined in work [9]. In this taxonomy, each robot r is supposed to have a schedule of assigned tasks. In the rest of the paper, this taxonomy is called "taxonomy B" for simplicity's sake.

- No dependencies: When allocating a task t to a robot r, we consider t and r.
- In-schedule dependencies: When allocating a task t to a robot r, we consider r and its schedule.
- Cross-schedule dependencies: When allocating a task t to a robot r, we consider r, its schedule, and other robots' schedules. It is worth pointing out that the robots' schedules are **static, i.e., schedules cannot be changed during the allocation process**.
- Complex dependencies: When allocating a task t to a robot r, we considered r, its schedule, and other robots' schedules. It is worth pointing out that the schedules are **dynamic, i.e., schedules can be changed during the allocation process.**

All MRTA problems are firmly *NP*-hard [11]. The computational time to generate optimal solutions for such problems increases exponentially with the problem size (i.e., numbers of robots and tasks). Accordingly, analytical methods are strongly discouraged [12], especially when the problem size is large. In the following, we present some related work.

There exist various common points between the following problems: i) the multi-robot task allocation, ii) the multiple traveling repairmen [13], and iii) the multiple traveling salesmen [14]. The last problems are derived from the well-known traveling salesman problem [15], which has been demonstrated to be *NP*-hard in combinatorial optimization [16]. Therefore, the multi-robot task allocation problem is also *NP*-hard [5]. The goal of the multiple traveling repairman problem is to determine a tour for each repairer while optimizing the average waiting time of customers. The aim of the multiple traveling salesmen problem is to allow each salesperson to visit all the cities and then return to the first one while optimizing their traveling distances. These criteria (i.e., time and distance) are usually the most adopted in MRTA problems. In the last decades, several approaches have been proposed in the literature to solve the different MRTA problems. We mainly distinguish between centralized and distributed methods [17-20].

In centralized approaches, an intermediate robot collects all the required information and allocates tasks to robots. The global objective function is optimized based on complete data [21, 22]. The work presented in [23] combines a map partitioning method with a Markovian Decision Process to assign a set of exploration tasks to a fleet of robots. Centralized approaches have many advantages: the objective function is easily optimized, the simplicity of implementation, and the running time is significantly reduced, to mention but a few. Nevertheless, they have some drawbacks, e.g., i) the scalability in real-life applications is not always reachable: i.e., guarantee that all robots are connected to the central station leads to a heavy traffic burden on it [24]; ii) the coverage area is drastically restricted because the allocations are computed on the central station: i.e., the computational demand on the central robot is too high; and iii) the centralized approaches are not robust: i.e., single failure point. The work described in [25] proposed a methodology to solve a scenario of fire extinguishment that combines quantum genetic algorithms and reinforcement learning.

In distributed approaches, each robot runs the allocation algorithm individually. Usually, distributed methods comprise a consensual phase to guarantee robots' convergence to consistent allocations, despite the network topologies [26]. Consensual stages add supplementary computing charges to the allocation process. Consequently, the makespan of these approaches is increased (a significant amount of time is taken by the robots to converge to a consistent allocation) [27]. Auction-based approaches [28-31] are considered as distributed methods. They are generally employed to solve problems where each robot can execute one task at a time, each task is strictly assigned to one robot, and allocations are time-extended (i.e., ST-SRTA). Auction-based algorithms are shown to produce sub-optimal solutions [9] efficiently. In auction-based approaches, robots offer bids on tasks according to their respective contextual information. The robot with the highest bid wins. The auctioneer might be one of the robots or a dedicated central station [32, 33]. In many ST-SR-TA task allocation scenarios, tasks, and robots are tightly bound [34]. In single-round combinatorial auctions [35], robots consider task bundles when calculating their bids, and the produced allocations are near-optimal. However, the computing time of all possible task bundles is exponentially increased with the number of tasks. In sequential single-item auctions [36], this shortcoming is overcome. Robots place their bids only on unallocated tasks in each round until allocating all tasks. In sequential simultaneous auctions [22], robots place their proposals on unallocated tasks in each round. Only one task is assigned at a time to each robot. An unmissable shortcoming in auction-based algorithms is the network topology, as the robots must place bids on tasks and transmit them to the auctioneer. The approaches proposed in [37-40] presented some auction-based strategies to solve MRTA problems. They used several metaheuristics, such as genetic algorithms, firefly algorithm, ant colony system, quantum genetic algorithms, artificial bee colony optimization. In work [41], the authors examined the MRTA problem using auctions. They analyzed the influence of lossy communications on solutions' quality. In work [42], the authors introduced a method to resolve the MRTA problem. They transformed the situation into a transportation problem and proposed a collision-free task allocation scheme.

In work [43], the authors proposed an algorithm called the "consensus-based bundle algorithm" (CBBA). They combined the consensus-based and the auction-based approaches to take their advantages. The consensus phase is used to agree on the winning bid values. The algorithm proposed in work [34] is derived from CBBA. It is shown to have exceptional performance compared to the CBBA. Other algorithms [44-48] extended from CBBA have also been shown to produce better solutions when solving ST-SR-TA problems with temporal constraints. This family of algorithms is proven to be very robust, despite the network topology [5]. However, they cannot handle the situations where schedules are dynamic. Besides, they produce sub-optimal allocations due to greedy-based strategies in the task inclusion phase [5]. The work in [49] uses CBBA, and allocations are computed using ant colony optimization. The work described in [50] uses the CBBA, and assignments are computed combining ant colony optimization and bat algorithm.

Many algorithms are available in the literature to solve MRTA problems [51, 52]. The authors of the work [53] solved the multi-objective coalition formation problem using the Pareto archived evolution strategy algorithm. The proposed method is centralized and does not consider dynamic scenarios. The approach described in [18] extended the work in [34] to handle online rescheduling. The algorithm is not efficient if much information arrives frequently. The work in [5] presented a solution to the task allocation problems in a distributed MRS, under critical time constraints. The authors of the work [54] proposed a multi-objective optimization of multi-robot task allocation with precedence constraints. The work in [55] discussed a new coordination control for multiple non-holonomic robots for target tracking with limited communications.

We give a mathematical formulation of the MRTA problem using set theory concepts. It allows a formal comprehension of the problem and gives the required tools to develop efficient solutions. At the end of the paper, we provide a generic framework for solving any MRTA problem. The work presented in this paper has three significant contributions. First, we propose a formal analysis of MRTA problems based on set theory concepts. It is not meant to be final or exhaustive, and indeed it has limitations. However, we believe that will help researchers understanding the nature of the problem, its components, and its complexity. Second, we formalize the taxonomies proposed in [9, 10], using our formal analysis. It will help recognize the type of a given problem from its components' properties (i.e., robots, tasks, allocations.). Third, we propose a generic solving scheme (i.e., an algorithm) of MRTA problems. Each step of this scheme is explained, and its time complexity is given. Also, we illustrate the proposed generic solving method in a challenging situation, which is the assignment of papers to reviewers within a journal. In conclusion, the work presented in this paper is not meant to be better than other literature formulations [9-11]; it just complements them.

The rest of the paper is organized as follows. Some standard definitions specific to set theory are given in section 2. Section 3 shows the proposed formal analysis of task allocation in MRS. Sections 4, and 5 provide the formulation of different MRTA classes of taxonomies A [9] and B [10], respectively. Section 6 discusses the proposed generic solving scheme of MRTA problems and highlights its time complexity. Section 7 illustrates the proposed algorithm on a real-world problem, which is the assignment of papers to reviewers within a journal. Finally, a conclusion and some future work are provided in section 8.

2. BACKGROUND ON SET THEORY

In this section, we give some standard definitions specific to the set theory.

Definition 1 *A set is a collection of objects (i.e., set elements). We write $e \in E$ to denote that e is an element of E. For example, the set $E = \{e_1, e_2, e_3\}$ contains three elements. The symbol $|E|$ denotes the cardinality of E. The symbol $\emptyset$ represents the empty set [56]*

Definition 2 *A subset is a set whose elements are all members of another set. The term $F \subseteq E$ is read: F is a subset of E. The term $F \subset E$ is read: F is a proper subset of E [56].*

Definition 3 *Let E be a set. The power set of E, denoted as $\wp(E)$, is the set that contains all possible subsets of E. The cardinality of $\wp(E)$ is $2^{|E|}$ [56]. The power set of E is given by (1).*

$$\wp(E) = \{X | X \subseteq E\} \tag{1}$$

Definition 4 *A collection of subsets F of a set E is called a family of subsets of E. For example, let $E = \{e_1, e_2, e_3, e_4, e_5\}$ be a set, an instance of a family of subsets of E might be given by $F = \{\{e_1, e_2, e_3\}, \{e_4, e_5\}, \{e_1, e_2, e_4\}\}$. It is worth pointing out that any family of subsets of E is a subset of the power set $\wp(E)$ if it has no repeated members [56]. The cardinality of a family of subsets F is equal to the number of subsets it contains. The terms "family of subsets F" and "family F" will be used interchangeably.*

Example 1 *Let $E = \{e_1, e_2, e_3\}$ be a set.*

- *The power set of E is: $\wp(E) = \{\emptyset, \{e_1\}, \{e_2\}, \{e_3\}, \{e_1, e_2\}, \{e_1, e_3\}, \{e_2, e_3\}, \{e_1, e_2, e_3\}\}$.*
- *Collections of subsets $F_1 = \{\{e_1, e_2\}, \{e_2, e_3\}\}$, $F_2 = \{\{e_1\}, \{e_2\}\}$ and $F_3 = \{\{e_1, e_2, e_3\}\}$ are three families of E. The cardinality of F_2 is $|F_2| = 2$.*

Definition 5 *The Cartesian product $E_1 \times \ldots \times E_n$ of sets $E_1, \ldots, E_n$ is defined as the set of all ordered tuples $(e_1, \ldots, e_n)$, with $e_1 \in E_1, \ldots, e_n \in E_n$ as follows [56].*

$$E_1 \times \ldots \times E_n = \{(e_1, \ldots, e_n) | e_1 \in E_1, \ldots, e_n \in E_n\} \tag{2}$$

Definition 6 *Let $E_1, \ldots, E_n$ be n sets. An n-ary relation on these sets is a subset of $E_1 \times \ldots \times E_n$. The sets $E_1, \ldots, E_n$ are called domains of the relation, and n is its degree [56].*

Definition 7 *A k-permutation is an ordered arrangement of k subsets from a family of n subsets ($k \leq n$). If a given family of subsets F has n subsets, then the number of k-permutations generated from F, denoted as $P(n, k)$, is given by (3) [56].*

$$P(n,k) = \frac{n!}{(n-k)!} \tag{3}$$

Example 2 *Let $F = \{\{e_1\}, \{e_2\}, \{e_3\}, \{e_1, e_2\}\}$ be a family. It is evident that F has 24 different 4-permutations, e.g., $\{\{e_1\}, \{e_2\}, \{e_1, e_2\}, \{e_3\}\}$ is one of them.*

Definition 8 *Let E' be a subset of a set E and F' be a subset of a set F. According to work discussed in [10], we can define a utility measure, denoted as $U(E', F')$, for subsets E' and F', (4).*

$$U(E', F') = \begin{cases} V(E', F') & , \quad if\ elements\ of\ E' and\ F' are\ related \\ \infty & , \quad otherwise \end{cases} \tag{4}$$

The term $V(E', F')$ is any function that returns a numerical value according to relationships between elements of subsets E' and F'. For example, if elements of E' and F' are points in the space, $V(E', F')$ may return the average of distances between points of E' and points of F'. In other words, the function $V(E', F')$ is given (5).

$$V(E', F') = \frac{\sum_{i=1}^{|E'|} \sum_{j=1}^{|F'|} \|e_i - f_j\|}{|E'|.|F'|} \tag{5}$$

3. FORMAL ANALYSIS OF MRTA PROBLEMS

In this section, we give the proposed formal analysis of multi-robot task allocation problems.

Definition 9 *A robot is an autonomous (logical or physical) entity that perceives its environment, makes decisions, and performs actions [11]. The set $A = \{a_1, \ldots, a_n\}$ designates a group of n robots.*

Definition 10 *A task is an elementary and independent action accomplished by one or several robots [11]. The set $T = \{t_1, \ldots, t_m\}$ designates a collection of m tasks.*

Definition 11 *A resource refers to any tool (logical or physical) that a task requires for its accomplishment. For example, suppose we want to measure the ambient temperature of someplace (i.e., task). In that case, we need a thermometer (i.e., resource). In MRTA scenarios, robots usually offer resources and tasks use them. The set* $\Omega = \{\omega_1, \dots, \omega_k\}$ *designates a collection of* k *resources.*
Definition 12 *Resources offered by robots is a ternary relation over sets* A*,* Ω*, and* $\mathcal{B} = \{0,1\}$*. It is a set of ordered triples* (a, ω, b) *consisting of elements:* $a \in A$*,* $\omega \in \Omega$*, and* $b \in \mathcal{B}$*. It encodes the relation's information: robot a has b instances of resource ω. This relation is denoted as* $\mathbb{O}$ *(*$\mathbb{O} \subseteq A \times \Omega \times \mathcal{B}$*).*
Definition 13 *Resources required by tasks is a ternary relation over sets* T*,* Ω*, and* $\mathcal{B} = \{0,1\}$*. It is a set of ordered triples* (a, ω, b) *consisting of elements:* $t \in T$*,* $\omega \in \Omega$*, and* $b \in \mathcal{B}$*. It encodes the relation's information: task t requires b instances of resource ω for its accomplishment. This relation is denoted as* $\mathbb{R}$ *(*$\mathbb{R} \subseteq T \times \Omega \times \mathcal{B}$*).*
Definition 14 *Valid assignments over robots and tasks is a quinary relation over sets* A*,* $\mathcal{B} = \{0,1\}$*,* Ω*,* T*, and* $\mathcal{B} = \{0,1\}$*. It is a set of ordered quintuples* (a, b, ω, t, b') *consisting of elements:* $a \in A$*,* $b \in \mathcal{B}$*,* $\omega \in \Omega$*,* $t \in T$*, and* $b' \in \mathcal{B}$*. It encodes the relation: robot* a *has* b *instances of resource* ω *and can be assigned to task* t*, which needs* b' *instances of resource ω. This relation is denoted as* $\mathbb{V}$ *(*$\mathbb{V} \subseteq A \times \mathcal{B} \times \Omega \times \mathrm{T} \times \mathcal{B}$*).*
This relation is given by (6), where $Q_i(a, b, \omega, t, b')$ *are conditions that determine allowed quintuples. For example, suppose the distance separating a robot and a task is larger than a given threshold. In that case, this robot cannot be assigned to this task even if it has the required resources. In other words, these conditions translate the constraints of the considered problem.*

$$\begin{aligned}&\mathbb{V} = \{(a, b, \omega, t, b') | (a, \omega, b) \in \mathbb{O}, (t, \omega, b') \in \mathbb{R}, (b, b') \geq \\ &(1,1), Q_1(a, b, \omega, t, b'), Q_2(a, b, \omega, t, b'), Q_3(a, b, \omega, t, b'), \dots\}\end{aligned} \tag{6}$$

NB: *The set* $\mathcal{B} = \{0,1\}$ *can be replaced by* $\mathbb{N} = \{0,1,2,3, \dots\}$*. However, for simplicity, we use the first set.*
Definition 15 *A coalition is a temporary group of robots working together on some task. Generally, when the task is performed, the coalition is dissolved [11]. If we consider a set* E*, then one can enumerate all possible coalitions using Definition 3. In this context, we should exclude the empty set* $\emptyset$ *since it is not a coalition. Thus, for a given set* E*, we have* $2^{|E|} - 1$ *possible coalitions.*
Definition 16 *Let* E *be a set. An overlapping family* F *of* E *is a family whose subsets are not duplicated and not necessarily pairwise disjoint. An overlapping family* F *should simultaneously verify the following conditions:*
a. $\emptyset \notin F$.
b. $\exists F_1, F_2 \in F | (F_1 \neq F_2) \Rightarrow (F_1 \cap F_2 \neq \emptyset)$.
c. $\bigcup_{i=1}^{|F|} F_i = E$.
Property 1 signifies that the empty subset is not considered (i.e., the empty coalition is not considered). Property 2 implies that a given element can be a member of several subsets simultaneously (i.e., a given robot can belong to several coalitions at the same time). Property 3 denotes that all members of set E should appear in subsets of family F (i.e., all robots are allocated). The symbol $\Theta(E)$ *designates the collection of all possible overlapping families of subsets of a set* E*, the symbol* Θ_i *denotes the ith family in the collection* $\Theta(E)$*, and the symbol* Θ_{ij} *expresses the jth subset in the family* Θ_i*. The number of overlapping families of a set* E *is given by (7) [57].*

$$|\Theta(E)| = \frac{1}{2}\sum_{k=0}^{|E|}(-1)^k \frac{|E|!}{(|E|-k)!k!} 2^{2^{|E|-k}} \tag{7}$$

Definition 17 *Let* E *be a set. A disjoint family* F *of* E *is a family whose subsets are not duplicated and mandatory pairwise disjoint. A disjoint family* F *should simultaneously verify the following conditions:*
a. $\emptyset \notin F$.
b. $\forall F_1, F_2 \in F | (F_1 \neq F_2) \Rightarrow (F_1 \cap F_2 = \emptyset)$.
c. $\bigcup_{i=1}^{|F|} F_i = E$.
Property 1 signifies that the empty subset is not considered (e.g., the empty coalition is not considered). Property 2 implies that a given element cannot be a member of several subsets simultaneously (e.g., any task is allocated one time). Property 3 denotes that all members of set E *should appear in subsets of family* F *(e.g., all tasks are allocated). The symbol* $\Delta(E)$ *designates the collection of all possible disjoint families of subsets of a set* E*, the symbol* Δ_i *denotes the ith family in the collection* $\Delta(E)$*, and the symbol* Δ_{ij} *expresses the jth subset in the family* Δ_i*. The number of disjoint families of a set* E *is given by (8) [58].*

$$|\Delta(E)| = \sum_{k=0}^{|E|}\left(\frac{1}{k!}\sum_{i=0}^{k}(-1)^i \frac{k!}{(k-i)!i!}(k-i)^{|E|}\right) \tag{8}$$

Remark 1 *If for each family $\Delta_i \in \Delta(E)$ we consider its $|\Delta_i|$-permutations, we can define an augmented collection $\Delta'(E)$, where each family $\Delta_i \in \Delta(E)$ is replaced by its $|\Delta_i|$-permutations. The symbol $\Delta_i{}'$ will denote the i^th^ family in the collection $\Delta'(E)$ and the symbol $\Delta_{ij}{}'$ will express the j^th^ subset in the family $\Delta_i{}'$. The number of augmented disjoint families of a set E is given by (9).*

$$|\Delta'(E)| = \sum_{k=0}^{|E|}\left(\left(\frac{1}{k!}\sum_{i=0}^{k}(-1)^i\frac{k!}{(k-i)!\,i!}(k-i)^{|E|}\right)k!\right) \quad (9)$$

Example 3 *Let $E = \{e_1, e_2\}$ be a set.*

- *The collection of all possible overlapping families is:*

$$\Theta(E) = \big\{\{\{e_1\},\{e_2\}\},\{\{e_1,e_2\}\},\{\{e_1\},\{e_1,e_2\}\},\{\{e_2\},\{e_1,e_2\}\},\{\{e_1\},\{e_2\},\{e_1,e_2\}\}\big\}.$$

- *The collection of all possible disjoint families is:*

$$\Delta(E) = \big\{\{\{e_1\},\{e_2\}\},\{\{e_1,e_2\}\}\big\}.$$

- *The collection of all possible augmented disjoint families is:*

$$\Delta'(E) = \big\{\{\{e_1\},\{e_2\}\},\{\{e_2\},\{e_1\}\},\{\{e_1,e_2\}\}\big\}.$$

Definition 18 *Let $A = \{a_1, \dots, a_n\}$ be a set of robots and $T = \{t_1, \dots, t_m\}$ be a set of tasks. Let $\Theta(A) = \{\Theta_1, \dots, \Theta_p\}$ be the collection of overlapping families of A and $\Delta'(T) = \{\Delta'_1, \dots, \Delta_q{}'\}$ be the augmented collection of disjoint families of T. Terms Θ_i and $\Delta_j{}'$ are, respectively, called family of robots' coalitions and family of task' coalitions.*

Partial allocations are a binary relation over collections $\Theta(A)$ and $\Delta'(T)$. It is a set of ordered pairs $(\Theta_i, \Delta_j{}')$ consisting of elements: $\Theta_i \in \Theta(A)$ and $\Delta_j{}' \in \Delta'(T)$. It is a subset of the Cartesian product $\Theta(A) \times \Delta'(T)$. This relation is given by (10) and denoted as Γ.

$$\Gamma = \{(\Theta_i, \Delta_j{}') | \Theta_i \in \Theta(A), \Delta_j{}' \in \Delta'(T), |\Theta_i| = |\Delta_j{}'|, Q_1(\Theta_i), \dots, Q_1{}'(\Delta_j{}'), \dots, Q_1{}''(\Theta_i, \Delta_j{}'), \dots\} \quad (10)$$

Each pair $(\Theta_i, \Delta_j{}') \in \Gamma$ is read as follows: each robots' coalition $\Theta_{ik} \in \Theta_i$ is assigned to tasks' coalition $\Delta_{jk}{}' \in \Delta_j{}'$. For example, let $\Theta_i = \{\{a_1, a_3\},\{a_1, a_2\}\}$ and $\Delta'_j = \{\{t_1, t_2\},\{t_3\}\}$. We read: robots' coalition $\{a_1, a_3\}$ is assigned to tasks' coalition $\{t_1, t_2\}$, and robots' coalition $\{a_1, a_2\}$ is assigned to tasks' coalition $\{t_3\}$. $Q_1(\Theta_i), \dots, Q_1{}'(\Delta_j{}'), \dots, Q_1{}''(\Theta_i, \Delta_j{}'), \dots$ are conditions that determine allowed partial allocations.

Definition 19 *A global allocation Γ' (or merely an allocation) is a sub-relation of Γ whose pairs $(\Theta_i, \Delta_i{}')$ should simultaneously verify the following conditions:*

a. $\bigcup_{i=1}^{|\Gamma'|}\bigcup_{j=1}^{|\Theta_i|}\Theta_{ij} = A.$

b. $\bigcup_{i=1}^{|\Gamma'|}\bigcup_{j=1}^{|\Delta_i{}'|}\Delta_{ij}{}' = T.$

c. $\bigcap_{i=1}^{|\Gamma'|}\bigcap_{j=1}^{|\Delta_i{}'|}\Delta_{ij}{}' = \emptyset.$

d. $\forall(\Theta_i, \Delta_i{}') \in \Gamma' \forall\Theta_{ij} \in \Theta_i \forall\Delta'_{ij} \in \Delta'_i | \{\omega | \omega \in (\Theta_{ij} \times \mathcal{B} \times \Omega \times \Delta'_{ij} \times \mathcal{B})\} \cap \{\omega | \omega \in (\Delta'_{ij} \times \Omega \times \mathcal{B})\} = \{\omega | \omega \in (\Delta'_{ij} \times \Omega \times \mathcal{B})\}.$

Relation $(\Theta_{ij} \times \mathcal{B} \times \Omega \times \Delta'_{ij} \times \mathcal{B}) \subseteq \mathbb{V}$ represents valid assignments over robots in Θ_{ij} and tasks in Δ'_{ij}. Relation $(\Delta'_{ij} \times \Omega \times \mathcal{B}) \subseteq \mathbb{R}$ represents the required resources by tasks in Δ'_{ij}. Condition 1 denotes that all robots are solicited to be assigned to tasks. Condition 2 means that all tasks should be assigned. Condition 3 signifies that a given task is exactly assigned to one robots' coalition. Condition 4 states that each coalition of robots should offer the required resources to the corresponding coalition of tasks. The optimal global allocation is the one that optimizes the utility measure given by (11).

$$\widehat{U} = \sum_{i=1}^{|\Gamma'|}\sum_{j=1}^{|\Theta_i|} U(\Theta_{ij}, \Delta_{ij}{}') \quad (11)$$

Definition 20 *Let $(F_1, F_2) \in \Gamma$ be a partial allocation. Let F_{1i} and F_{2i} be the i^th^ robots' and tasks' coalitions of families F_1 and F_2, respectively. Potential assignments are a sub-relation of the relation $\mathbb{V}$ given by Definition 14 (valid assignments) whose quintuples modeled by (12). This relation is denoted as $\mathbb{P}$.*

$$\mathbb{P} = \{(a, b, \omega, t, b') | (a, b, \omega, t, b') \in \mathbb{V}, a \in F_{1i}, t \in F_{2i}\} \quad (12)$$

Robots in F_{1i} ar related to tasks in F_{2i} (4), if and only if the relation $\mathbb{P}$ is not empty.

Example 4 *In this example, we consider $A = \{a_1, a_2\}$, $T = \{t_1, t_2, t_3\}$, and $\Omega = \{\omega_1, \omega_2\}$. Relations $\mathbb{O}$, $\mathbb{R}$, and $\mathbb{V}$ are given as follow:*

- $\mathbb{O} = \{(a_1, \omega_1, 1), (a_1, \omega_2, 0), (a_2, \omega_1, 1), (a_2, \omega_2, 1)\}$.
- $\mathbb{R} = \{(t_1, \omega_1, 1), (t_1, \omega_2, 0), (t_2, \omega_1, 1), (t_2, \omega_2, 0), (t_3, \omega_1, 0), (t_3, \omega_2, 1)\}$.
- $\mathbb{V} = \{(a_1, 1, \omega_1, t_1, 1), (a_1, 1, \omega_1, t_2, 1), (a_2, 1, \omega_1, t_1, 1), (a_2, 1, \omega_1, t_2, 1), (a_2, 1, \omega_2, t_3, 1)\}$.

Let $F_1 = \{\{a_1, a_2\}, \{a_2\}\}$ be a family of robots' coalitions; and $F_2 = \{\{t_1\}, \{t_2, t_3\}\}$ be a family of tasks' coalitions. If we consider $F_{12} = \{a_2\}$ and $F_{22} = \{t_2, t_3\}$, then $\mathbb{P} = \{(a_2, 1, \omega_1, t_2, 1), (a_2, 1, \omega_2, t_3, 1)\}$.

Definition 21 *Let $\Gamma' = \{(E_1, F_1), \ldots, (E_p, F_p)\}$ be a global allocation. A family of potential assignments is the set $[\mathbb{P}] = \{\mathbb{P}_1, \ldots, \mathbb{P}_p\}$ of potential assignments given by Definition 20 (each term $\mathbb{P}_i \in [\mathbb{P}]$ is computed as in Example 4).*

The concept of a family of potential assignments allows us to build task schedules of robots. The use of this concept will be detailed in Section 5.

4. FORMULATION OF TAXONOMY A

This section gives the formulation of taxonomy A [9], using our proposed formulation.

Definition 22 *Let $\mathbb{O}$ be the relation of resources offered by robots given in Definition 12. All robots of a given problem are said to be Single-Task, if the following condition is satisfied.*

$$\forall a \in A \exists! \, \omega \in \Omega | (a, \omega, 1) \in \mathbb{O} \tag{13}$$

Condition 13 signifies that each robot $a \in A$ offers one and only one resource $\omega \in \Omega$ for accomplishing tasks.

Definition 23 *Let $\mathbb{O}$ be the relation of resources offered by robots given in Definition 12. Some robots of a given problem are said to be Multi-Task, if the following condition is satisfied.*

$$\forall a \in A \exists \omega_1, \ldots, \omega_p \in \Omega | (a, \omega_1, 1) \in \mathbb{O} \wedge \ldots \wedge (a, \omega_p, 1) \in \mathbb{O} \tag{14}$$

Condition 14 signifies that some robots $a \in A$ offer several resources $\omega \in \Omega$ for accomplishing tasks.

Definition 24 *Let $\mathbb{R}$ be the relation of resources required by tasks given in Definition 13. All tasks of a given problem are said to be Single-Robot, if the following condition is satisfied.*

$$\forall t \in T \exists! \, \omega \in \Omega | (t, \omega, 1) \in \mathbb{R} \tag{15}$$

Condition 15 signifies that each task $t \in T$ requires one and only one resource $\omega \in \Omega$ for its accomplishment.

Definition 25 *Let $\mathbb{R}$ be the relation of resources required by tasks given in Definition 13. Some tasks of a given problem are said to be Multi-Robot, if the following condition is satisfied.*

$$\forall t \in T \exists \omega_1, \ldots, \omega_p \in \Omega | (t, \omega_1, 1) \in \mathbb{R} \wedge \ldots \wedge (t, \omega_p, 1) \in \mathbb{R} \tag{16}$$

Condition 16 means that some tasks $t \in T$ require several resources $\omega \in \Omega$ for their accomplishment.

Definition 26 *Let $\Gamma' = \{(F_1, F_1'), \ldots, (F_p, F_p')\}$ be an allocation.*

- *If the execution of tasks in Γ' are associated with any temporal model (e.g., time intervals [59], simple temporal networks [60]). In this case, we say that the allocation of robots to tasks is performed in a time-extended manner.*
- *Conversely, we say that the allocation of robots to tasks is performed in an instantaneous manner (i.e., no temporal model is provided, and tasks are allocated as they come).'*

5. FORMULATION OF TAXONOMY B

In this section, we formulate the taxonomy described in [10], using the proposed analysis. It is worth pointing out that each robot should have a schedule of tasks, which is a critical issue in this context. The following example shows how robots' schedules are built.

Example 5 In this example, *we consider $A = \{a_1, a_2, a_3\}$, $T = \{t_1, t_2, t_3, t_4\}$, and $\Omega = \{\omega_1, \omega_2\}$. Relations $\mathbb{O}$, $\mathbb{R}$, $\mathbb{V}$, and Γ' are given as follow:*

- $\mathbb{O} = \{(\mathrm{a}_1, \omega_1, 1), (\mathrm{a}_1, \omega_2, 1), (\mathrm{a}_2, \omega_1, 1), (\mathrm{a}_2, \omega_2, 1), (\mathrm{a}_3, \omega_1, 1), (\mathrm{a}_3, \omega_2, 1)\}$.

- $\mathbb{R} = \left\{ \begin{matrix} (\mathrm{t}_1, \omega_1, 1), (\mathrm{t}_1, \omega_2, 1), (\mathrm{t}_2, \omega_1, 1), (\mathrm{t}_2, \omega_2, 1), \\ (\mathrm{t}_3, \omega_1, 1), (\mathrm{t}_3, \omega_2, 1), (\mathrm{t}_4, \omega_1, 1), (\mathrm{t}_4, \omega_2, 1) \end{matrix} \right\}.$
- $\mathbb{V} = \left\{ \begin{matrix} (\mathrm{a}_1, 1, \omega_1, \mathrm{t}_1, 1), (\mathrm{a}_1, 1, \omega_1, \mathrm{t}_2, 1), (\mathrm{a}_1, 1, \omega_1, \mathrm{t}_3, 1), (\mathrm{a}_1, 1, \omega_1, \mathrm{t}_4, 1), \\ (\mathrm{a}_1, 1, \omega_2, \mathrm{t}_1, 1), (\mathrm{a}_1, 1, \omega_2, \mathrm{t}_2, 1), (\mathrm{a}_1, 1, \omega_2, \mathrm{t}_3, 1), (\mathrm{a}_1, 1, \omega_2, \mathrm{t}_4, 1), \\ (\mathrm{a}_2, 1, \omega_1, \mathrm{t}_1, 1), (\mathrm{a}_2, 1, \omega_1, \mathrm{t}_2, 1), (\mathrm{a}_2, 1, \omega_1, \mathrm{t}_3, 1), (\mathrm{a}_2, 1, \omega_1, \mathrm{t}_4, 1), \\ (\mathrm{a}_2, 1, \omega_2, \mathrm{t}_1, 1), (\mathrm{a}_2, 1, \omega_2, \mathrm{t}_2, 1), (\mathrm{a}_2, 1, \omega_2, \mathrm{t}_3, 1), (\mathrm{a}_2, 1, \omega_2, \mathrm{t}_4, 1), \\ (\mathrm{a}_3, 1, \omega_1, \mathrm{t}_1, 1), (\mathrm{a}_3, 1, \omega_1, \mathrm{t}_2, 1), (\mathrm{a}_3, 1, \omega_1, \mathrm{t}_3, 1), (\mathrm{a}_3, 1, \omega_1, \mathrm{t}_4, 1), \\ (\mathrm{a}_3, 1, \omega_2, \mathrm{t}_1, 1), (\mathrm{a}_3, 1, \omega_2, \mathrm{t}_2, 1), (\mathrm{a}_3, 1, \omega_2, \mathrm{t}_3, 1), (\mathrm{a}_3, 1, \omega_2, \mathrm{t}_4, 1) \end{matrix} \right\}.$
- $\Gamma' = \{(\{\{a_1\}\}, \{\{t_1\}\}), (\{\{a_1, a_2\}, \{a_1, a_2, a_3\}\}, \{\{t_2\}, \{t_3, t_4\}\})\}.$

Γ' is a global allocation, which means that: robots' coalition $\{a_1\}$ is assigned to tasks' coalition $\{t_1\}$, robots' coalition $\{a_1, a_2\}$ is assigned to tasks' coalition $\{t_2\}$, and robots' coalition $\{a_1, a_2, a_3\}$ is assigned to tasks' coalition $\{t_3, t_4\}$. The family of potential assignments corresponding to Γ' is $[\mathbb{P}] = \{\mathbb{P}_1, \mathbb{P}_2, \mathbb{P}_3\}$, where:

- *Potential assignments $\mathbb{P}_1$ over subsets $\{a_1\}$ and $\{t_1\}$ are:*

$$\mathbb{P}_1 = \{(\mathrm{a}_1, 1, \omega_1, \mathrm{t}_1, 1), (\mathrm{a}_1, 1, \omega_2, \mathrm{t}_1, 1)\}$$

Robots' coalition $\{a_1\}$ is assigned to tasks' coalition $\{t_1\}$. We have only one partial schedule: i.e., robot a_1 offers both resources ω_1 and ω_2 to task t_1.

- *Potential assignments $\mathbb{P}_2$ over subsets $\{a_1, a_2\}$ and $\{t_2\}$ are:*

$$\mathbb{P}_2 = \{(\mathrm{a}_1, 1, \omega_1, t_2, 1), (\mathrm{a}_1, 1, \omega_2, t_2, 1), (\mathrm{a}_2, 1, \omega_1, t_2, 1), (\mathrm{a}_2, 1, \omega_2, t_2, 1)\}$$

Robots' coalition $\{a_1, a_2\}$ is assigned to tasks' coalition $\{t_2\}$. We have four partial schedules: i.e., robot a_1 offers both resources ω_1 and ω_2 to task t_2; robot a_2 offers both resources ω_1 and ω_2 to task t_2; robot a_1 offers resource ω_1 to task t_2 and robot a_2 offers resource ω_2 to task t_2; or robot a_1 offers resource ω_2 to task t_2 and robot a_2 offers resource ω_1 to task t_2.

- *Potential assignments $\mathbb{P}_3$ over subsets $\{a_1, a_2, a_3\}$ and $\{t_3, t_4\}$ are:*

$$\mathbb{P}_3 = \left\{ \begin{matrix} (\mathrm{a}_1, 1, \omega_1, \mathrm{t}_3, 1), (\mathrm{a}_1, 1, \omega_2, \mathrm{t}_3, 1), (\mathrm{a}_1, 1, \omega_1, \mathrm{t}_4, 1), (\mathrm{a}_1, 1, \omega_2, \mathrm{t}_4, 1), \\ (\mathrm{a}_2, 1, \omega_1, \mathrm{t}_3, 1), (\mathrm{a}_2, 1, \omega_2, \mathrm{t}_3, 1), (\mathrm{a}_2, 1, \omega_1, \mathrm{t}_4, 1), (\mathrm{a}_2, 1, \omega_2, \mathrm{t}_4, 1), \\ (\mathrm{a}_3, 1, \omega_1, \mathrm{t}_3, 1), (\mathrm{a}_3, 1, \omega_2, \mathrm{t}_3, 1), (\mathrm{a}_3, 1, \omega_1, \mathrm{t}_4, 1), (\mathrm{a}_3, 1, \omega_2, \mathrm{t}_4, 1) \end{matrix} \right\}$$

Robots' coalition $\{a_1, a_2, a_3\}$ is assigned to tasks' coalition $\{t_3, t_4\}$. We have $3^{2\times 2} = 81$ partial schedules (we only cite one instance): e.g., robot a_1 offers resource ω_1 to task t_3, robot a_1 offers resource ω_2 to task t_3, robot a_2 offers resource ω_1 to task t_4, and robot a_3 offers resource ω_2 to task t_4.

Generally, if we have three sets A, T and Ω of robots, tasks, and resources, respectively. In this case, we should consider $|A|^{(|T|\times|\Omega|)}$ different partial schedules of size $|T| \times |\Omega$. In our example, we should consider $1 \times 4 \times 81$ global schedules, and the schedule of each robot will contain its assigned tasks: e.g., if we take the partial schedules: i) robot a_1 offers both resources ω_1 and ω_2 to task t_1; ii) robot a_1 offers resource ω_1 to task t_2 and robot a_2 offers resource ω_2 to task t_2; and iii) robot a_1 offers resource ω_1 to task t_3, robot a_1 offers resource ω_2 to task t_3, robot a_2 offers resource ω_1 to task t_4 and robot a_3 offers resource ω_2 to task t_4. In this case, i) the schedule of robot a_1 contains tasks $\{(t_1, \omega_1), (t_1, \omega_2), (t_2, \omega_1), (t_3, \omega_1), (t_3, \omega_2)\}$; ii) the schedule of robot a_2 contains tasks $\{(t_2, \omega_2), (t_4, \omega_1)\}$; and iii) the schedule of robot a_3 contains one task $\{(t_4, \omega_2)\}$. It is worth pointing out that the order of execution of tasks of each robot depends on how tasks are allocated to robots, i.e., instantaneous or time-extended assignments.

Definition 27 *Let $\Gamma' = \{(E_1, F_1), \ldots, (E_p, F_p)\}$ be an allocation and $[\mathbb{P}] = \{\mathbb{P}_1, \ldots, \mathbb{P}_p\}$ be its corresponding family of potential assignments. We consider robots $a_1, \ldots, a_n$. We showed, via Example 5, how robots' schedules are built. Let $S_{a_i}(\tau)$ be the schedule of robot a_i at time τ. We define the features of taxonomy B [10] as follows.*

- ***No dependencies:***

  As shown in (17) shows that a given robot's utility measure for a specific task depends on them.

$$\forall t_j \in S_{a_i}(0): U(\{a_i\}, \{t_j\}) = f(a_i, t_j) \quad (17)$$

- ***In-schedule dependencies:***

  As shown in (18) shows that a given robot's utility measure for a specific task depends on its schedule.

$$\forall t_j \in S_{a_i}(0): U(\{a_i\}, \{t_j\}) = f\left(a_i, S_{a_1}(0)\right) \quad (18)$$

- ***Cross-schedule dependencies:***

As shown in (19) shows that a given robot's utility measure for a specific task depends on all robots' schedules, including itself. However, the schedules are static.

$$\forall t_j \in S_{a_i}(0): U(\{a_i\},\{t_j\}) = f\left(a_i, S_{a_1}(0), \dots, S_{a_n}(0)\right) \qquad (19)$$

– ***Complex dependencies:***

As shown in (20) shows that a given robot's utility measure for a specific task depends on all robots' schedules, including itself. However, the schedules are dynamic.

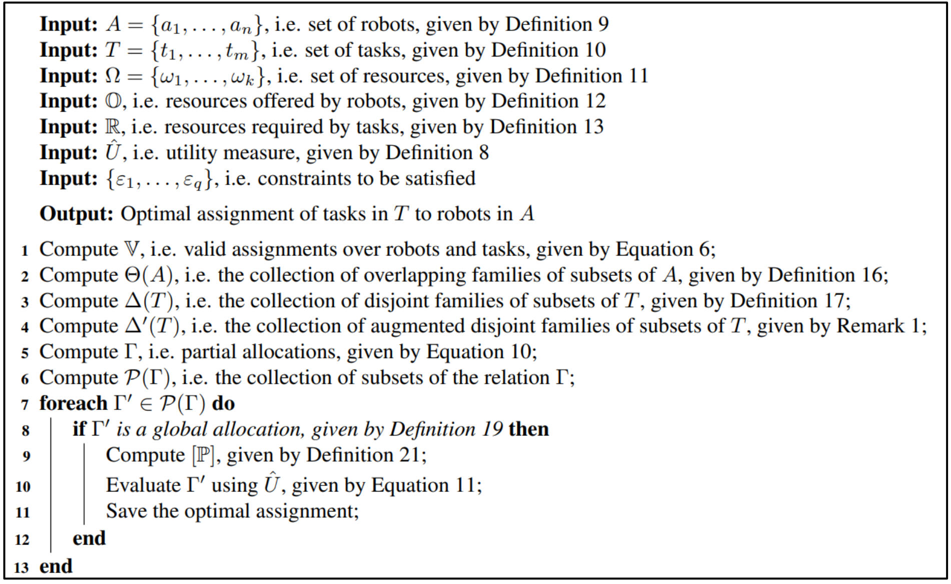
$$\forall t_j \in S_{a_i}(0): U(\{a_i\},\{t_j\}) = f\left(a_i, S_{a_1}(\tau), \dots, S_{a_n}(\tau)\right) \qquad (20)$$

6. SOLVING SCHEME OF MRTA PROBLEMS

This section overviews how the proposed formulation is used in a generic solving scheme and explains its analytical time complexity. Algorithm 1 highlights the main steps in solving a MRTA problem.

- Line 1 computes the relation $\mathbb{V}$, a subset of the Cartesian product of relations $\mathbb{O}$ and $\mathbb{R}$. The relation $\mathbb{V}$ contains all valid assignments between tasks and robots. The time complexity of this step is $\mathcal{O}(max\{|\mathbb{O}|,|\mathbb{R}|\}^2)$.
- Line 2 computes the collection $\Theta(A)$ of overlapping families of subsets of the set A; i.e., a given robot may belong to several coalitions, since it can perform several tasks (simultaneously or sequentially). The time complexity of this step is $\mathcal{O}\left(|\wp(A)| \times 2^{|\wp(A)|}\right)$.
- Line 3 computes the collection $\Delta(T)$ of disjoint families of subsets of the set T; i.e., a given task must belong to only one coalition, since it must be done once. The time complexity of this step is also $\mathcal{O}\left(|\wp(T)| \times 2^{|\wp(T)|}\right)$.
- Line 4 computes the collection $\Delta'(T)$ of all possible permutations of families in $\Delta(T)$. The time complexity of this step is $\mathcal{O}\left(\max_{\Delta_i \in \Delta(T)}\{|\Delta_i|!\}\right)$.
- Line 5 computes the relation Γ, a subset of the Cartesian product $\Theta(A) \times \Delta'(T)$. This relation contains all partial allocations. The time complexity of this step is $\mathcal{O}(\max\{|\Theta(A)|,|\Delta'(T)|\}^2)$.
- Line 6 computes all global allocations. The time complexity of this step is $\mathcal{O}\left(|\wp(\Gamma)| \times 2^{|\wp(\Gamma)|}\right)$.
- Finally, each global allocation is evaluated using the considered utility measure subject to some constraints. The optimal assignment of tasks in T to robots in A is returned.

Agorithm 1. Generic solving of a MRTA problem



```
Input: $A = \{a_1, \dots, a_n\}$, i.e. set of robots, given by Definition 9
Input: $T = \{t_1, \dots, t_m\}$, i.e. set of tasks, given by Definition 10
Input: $\Omega = \{\omega_1, \dots, \omega_k\}$, i.e. set of resources, given by Definition 11
Input: $\mathbb{O}$, i.e. resources offered by robots, given by Definition 12
Input: $\mathbb{R}$, i.e. resources required by tasks, given by Definition 13
Input: $\hat{U}$, i.e. utility measure, given by Definition 8
Input: $\{\varepsilon_1, \dots, \varepsilon_q\}$, i.e. constraints to be satisfied
Output: Optimal assignment of tasks in $T$ to robots in $A$
1  Compute $\mathbb{V}$, i.e. valid assignments over robots and tasks, given by Equation 6;
2  Compute $\Theta(A)$, i.e. the collection of overlapping families of subsets of $A$, given by Definition 16;
3  Compute $\Delta(T)$, i.e. the collection of disjoint families of subsets of $T$, given by Definition 17;
4  Compute $\Delta'(T)$, i.e. the collection of augmented disjoint families of subsets of $T$, given by Remark 1;
5  Compute $\Gamma$, i.e. partial allocations, given by Equation 10;
6  Compute $\mathcal{P}(\Gamma)$, i.e. the collection of subsets of the relation $\Gamma$;
7  foreach $\Gamma' \in \mathcal{P}(\Gamma)$ do
8      if $\Gamma'$ is a global allocation, given by Definition 19 then
9          Compute $[\mathbb{P}]$, given by Definition 21;
10         Evaluate $\Gamma'$ using $\hat{U}$, given by Equation 11;
11         Save the optimal assignment;
12     end
13 end
```

It is worth pointing out that MRTA problems are strongly *NP*-hard, i.e., shown through complexity analysis. Therefore, exact methods are not highly recommended to handle large-scale MRTA problems. Thus, the use of metaheuristics-based techniques is recommended [37, 49].

7. ILLUSTRATIVE EXAMPLE

We give an illustrative example to show how Algorithm 1 is employed to solve a given MRTA problem. We choose the problem of assigning papers to reviewers within a journal. We suppose that the Bulletin of Electrical Engineering and Informatics has received 20 papers to consider for publication on a given day. The editor-in-chief (EIC) has ten potential reviewers. After checking all the papers, the EIC concluded that they are articulated around five topics. Hence, he should assign each paper to some reviewers (e.g., two reviewers) while optimizing a given criterion under some constraints. This situation is indeed a MRTA problem, where: (a) reviewers are the robots (Definition 9), i.e., $|A| = 10$; (b) papers are the tasks (Definition 10), i.e., $|T| = 20$; and (c) topics are the resources (Definition 11), i.e., $|\Omega| = 5$. Topics for which reviewers are experts are summarized in Table 1 (Definition 12), i.e., relation $\mathbb{O}$. Topics covered by papers are summarized in Table 2 (Definition 13), i.e., relation $\mathbb{R}$.

Line 1 of Algorithm 1: We compute $\mathbb{V}$, i.e., valid assignments over robots and tasks, using (6) and Tables 1 and 2 (Definition 14). We suppose that no constraint is considered: i.e., $\{\varepsilon_1, \dots, \varepsilon_q\}$ is empty (all valid assignments are accepted). It is worth pointing out that in the worst case, we should have $|A| \times |T| \times |\Omega|$ valid assignments: i.e., $|\mathbb{V}| = |A| \times |T| \times |\Omega|$. For our example, $|\mathbb{V}| = 10 \times 20 \times 5 = 1000$, which is a large number to consider.

Line 2, 3, and 4 of Algorithm 1: We compute $\Theta(A)$ (Definition 16), $\Delta(T)$ (Definition 17), and $\Delta'(T)$ (Remark 1), respectively. Cardinalities of collections $\Theta(A)$, $\Delta(T)$, and $\Delta'(T)$ are given as follows. There is a tip to reduce the size of collections $\Theta(A)$ and $\Delta'(T)$. For instance, if the journal policy states that each paper is assigned exactly to 2 reviewers; therefore, the EIC might consider: (a) families $\Theta_i \in \Theta(A)$ for which $|\Theta_{ij}| = 2$; and (b) families $\Delta_i{}' \in \Delta'(T)$ for which $|\Delta_{ij}{}'| = 1$.

$$|\Theta(A)| = \tfrac{1}{2}\textstyle\sum_{k=0}^{10}(-1)^k \frac{10!}{(10-k)!k!} 2^{2^{10-k}}$$

$$|\Delta(T)| = \textstyle\sum_{k=0}^{20}\left(\frac{1}{k!}\sum_{i=0}^{k}(-1)^i \frac{k!}{(k-i)!i!}(k-i)^{20}\right)$$

$$|\Delta'(T)| = \textstyle\sum_{k=0}^{20}\left(\left(\frac{1}{k!}\sum_{i=0}^{k}(-1)^i \frac{k!}{(k-i)!i!}(k-i)^{20}\right)k!\right)$$

Line 5 and 6 of Algorithm 1: We compute $\Gamma = \Theta(A) \times \Delta'(T)$, partial allocations, given by (10) and $\wp(\Gamma)$, the powerset of relation Γ. Line 7 to 13 of Algorithm 1: For each subset in $\wp(\Gamma)$, we check if it is a global allocation (Definition 19). If this is the case, we evaluate the global allocations and return the optimal one. For the current example, line 9 is ignored since assignments are instantaneous. For example, if we have:

$$\Gamma' = \left\{\begin{matrix} (\{a_6, a_7\}, \{t_1\}), (\{a_9, a_7\}, \{t_2\}), (\{a_4, a_{10}\}, \{t_3\}), (\{a_4, a_1\}, \{t_4\}), (\{a_7, a_{10}\}, \{t_5\}), \\ (\{a_8, a_3\}, \{t_6\}), (\{a_4, a_8\}, \{t_7\}), (\{a_2, a_9\}, \{t_8\}), (\{a_4, a_{10}\}, \{t_9\}), (\{a_1, a_4\}, \{t_{10}\}), \\ (\{a_4, a_9\}, \{t_{11}\}), (\{a_5, a_9\}, \{t_{12}\}), (\{a_2, a_6\}, \{t_{13}\}), (\{a_6, a_8\}, \{t_{14}\}), (\{a_6, a_2\}, \{t_{15}\}), \\ (\{a_8, a_7\}, \{t_{16}\}), (\{a_5, a_{10}\}, \{t_{17}\}), (\{a_9, a_{10}\}, \{t_{18}\}), (\{a_2, a_6\}, \{t_{19}\}), (\{a_{10}, a_9\}, \{t_{20}\}) \end{matrix}\right\}$$

We evaluate Γ' using the objective function given by (21). We sum portions of topics offered by two reviewers compared to topics needed by a paper. The optimal global assignment is the one that maximizes this sum. Of course, the EIC might use other criteria to evaluate a global allocation: e.g., impose a maximum occurrence of a given reviewer.

$$\widehat{U} = \textstyle\sum_{i=1}^{|\Gamma'|}\sum_{j=1}^{|\Theta_i|}\left(\frac{\left|\left\{\omega|\omega\in\left(\Theta_{ij}\times\mathcal{B}\times\Omega\times\Delta'_{ij}\times\mathcal{B}\right)\right\}\right|}{\left|\left\{\omega|\omega\in\left(\Delta'_{ij}\times\Omega\times\mathcal{B}\right)\right\}\right|}\right) \tag{21}$$

Table 1. Topics of reviewers (relation $\mathbb{O}$)

	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9	a_{10}
ω_1	1	1	0	1	0	1	1	0	1	1
ω_2	1	0	0	0	1	0	1	1	0	0
ω_3	0	0	0	1	0	1	1	0	1	1
ω_4	0	0	1	1	1	1	0	0	0	0
ω_5	1	1	0	0	1	1	0	0	0	1

Table 2. Topics covered by papers (relation $\mathbb{R}$)

	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}	t_{11}	t_{12}	t_{13}	t_{14}	t_{15}	t_{16}	t_{17}	t_{18}	t_{19}	t_{20}
ω_1	1	1	0	1	0	0	1	0	1	0	1	1	1	1	0	1	1	0	1	1
ω_2	1	1	0	1	1	0	1	0	1	1	1	0	1	1	0	0	0	0	1	0
ω_3	0	1	0	0	0	1	0	0	1	1	0	1	1	1	1	0	1	0	1	1
ω_4	0	1	1	0	0	0	1	1	0	1	1	0	1	1	1	1	0	0	1	0
ω_5	0	1	0	1	0	0	0	0	1	0	0	0	0	0	0	1	1	1	0	0

8. CONCLUSION AND FUTURE WORK

This paper proposed a formal analysis of multi-robot task allocation problems using elementary set theory concepts. We formulated two well-known taxonomies of MRTA problems (i.e., the taxonomy of Gerkey and Matarić entitled “A formal analysis and taxonomy of task allocation in multi-robot systems”; and the taxonomy of Korsah, Stentz, and Dias entitled “A comprehensive taxonomy for multi-robot task allocation”), then we developed them using the proposed analysis. A generic solving scheme of MRTA problems is presented, and its time complexity is discussed. Finally, the proposed generic solving method is employed to solve a real-world problem: i.e., the assignment of papers to reviewers within a journal. In conclusion, the work introduced in this paper is not intended to be better than the previous formulations; it merely complements them. As future work, we plan to represent the taxonomy of Ernesto, Marie, Hakim, and Maria entitled “A taxonomy for task allocation problems with temporal and ordering constraints” using our framework.

ACKNOWLEDGEMENTS

This research work is supported by UAEU Grant: 31T102-UPAR-1-2017.

REFERENCES

[1] Y. Cao, W. Yu, W. Ren, and G. Chen, “An overview of recent progress in the study of distributed multiagent coordination,” *IEEE Transactions on Industrial informatics*, vol. 9, no. 1, pp. 427-438, 2012.

[2] A. Khamis, A. Hussein, and A. Elmogy, “Multi-robot task allocation: A review of the state-of-the-art,” in *Cooperative Robots and Sensor Networks 2015*, vol. 604, pp. 31-51, 2015.

[3] A. Torreno, E. Onaindia, A. Komenda, and M. Stolba, “Cooperative multi-agent planning: A survey,” *ACM Computing Surveys (CSUR)*, vol. 50, no. 6, p. 84, 2018.

[4] G. A. Kaminka, “Autonomous agents research in robotics: A report from the trenches,” in *2012 AAAI Spring symposium series*, pp. 1-7, 2012.

[5] X. Chen, P. Zhang, G. Du, and F. Li, “A distributed method for dynamic multi-robot task allocation problems with critical time constraints,” *Robotics and Autonomous Systems*, vol. 118, pp. 31-46, 2019.

[6] L. Fleischer, M. X. Goemans, V. S. Mirrokni, and M. Sviridenko, “Tight approximation algorithms for maximum general assignment problems,” *SODA '06: Proceedings of the seventeenth annual ACM-SIAM symposium on Discrete algorithm*, pp. 611-620, 2006.

[7] D. Turra, L. Pollini, and M. Innocenti, “Fast unmanned vehicles task allocation with moving targets,” 2004 43rd *IEEE Conference on Decision and Control (CDC) (IEEE Cat. No.04CH37601),* Nassau, 2004, pp. 4280-4285, vol. 4.

[8] B. Horling and V. Lesser, “A survey of multi-agent organizational paradigms,” *The Knowledge engineering review*, vol. 19, no. 4, pp. 281-316, 2004.

[9] B. P. Gerkey and M. J. Mataric, “A formal analysis and taxonomy of task allocation in multi-robot systems,” *The International journal of robotics research*, vol. 23, no. 9, pp. 939-954, 2004.

[10] G. A. Korsah, A. Stentz, and M. B. Dias, “A comprehensive taxonomy for multi-robot task allocation,” *The International Journal of Robotics Research*, vol. 32, no. 12, pp. 1495-1512, 2013.

[11] E. Nunes, M. Manner, H. Mitiche, and M. Gini, “A taxonomy for task allocation problems with temporal and ordering constraints,” *Robotics and Autonomous Systems*, vol. 90, pp. 55-70, 2017.

[12] M. B. Dias, R. Zlot, N. Kalra, and A. Stentz, “Market-based multirobot coordination: A survey and analysis,” *Proceedings of the IEEE*, vol. 94, no. 7, pp. 1257-1270, 2006.

[13] J. Fakcharoenphol, C. Harrelson, and S. Rao, “The k-traveling repairmen problem,” *ACM Transactions on Algorithms (TALG)*, vol. 3, no. 4, 2007.

[14] E. C. Brown, C. T. Ragsdale, and A. E. Carter, “A grouping genetic algorithm for the multiple traveling salesperson problem,” *International Journal of Information Technology & Decision Making*, vol. 6, no. 02, pp. 333-347, 2007.

[15] S. Lin, “Computer solutions of the traveling salesman problem,” *The Bell System Technical Journal*, vol. 44, no. 10, pp. 2245-2269, 1965.

[16] G. Dantzig, R. Fulkerson, and S. Johnson, “Solution of a large-scale traveling-salesman problem,” *Journal of the operations research society of America*, vol. 2, no. 4, pp. 393-410, 1954.

[17] Y. Wang, Y. Chen, and Y. Lin, “Memetic algorithm based on sequential variable neighborhood descent for the minmax multiple traveling salesman problem,” *Computers & Industrial Engineering*, vol. 106, pp. 105-122, 2017.

[18] Z. Luo, H. Qin, and A. Lim, “Branch-and-price-and-cut for the multiple traveling repairman problem with distance constraints,” *European Journal of Operational Research*, vol. 234, no. 1, pp. 49-60, 2014.

[19] P. Junjie and W. Dingwei, “An ant colony optimization algorithm for multiple travelling salesman problem,” *First International Conference on Innovative Computing, Information and Control-Volume I (ICICIC'06),* Beijing, 2006, pp. 210-213.

[20] P. Venkatesh and A. Singh, “Two metaheuristic approaches for the multiple traveling salesperson problem,” *Applied Soft Computing*, vol. 26, pp. 74-89, 2015.

[21] J. Chen and D. Sun, “Coalition-based approach to task allocation of multiple robots with resource constraints,” *IEEE Transactions on Automation Science and Engineering*, vol. 9, no. 3, pp. 516-528, 2012.

[22] L. Huang, Y. Ding, M. Zhou, Y. Jin, and K. Hao, “Multiple-solution optimization strategy for multirobot task allocation,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 11, pp. 4283-4294, 2020.

[23] G. Lozenguez, L. Adouane, A. Beynier, P. Martinet, and A.-I. Mouaddib, “Map partitioning to approximate an exploration strategy in mobile robotics,” in *Advances on practical applications of agents and multiagent systems*, vol. 88, pp. 63-72, 2011.

[24] K. Zhang, E. G. Collins Jr, and D. Shi, “Centralized and distributed task allocation in multi-robot teams via a stochastic clustering auction,” *ACM Transactions on Autonomous and Adaptive Systems (TAAS)*, vol. 7, no. 2, pp. 1-22, 2012.

[25] F. Zitouni and R. Maamri, “Cooperative learning-agents for task allocation problem,” in *Interactive Mobile Communication, Technologies and Learning*, vol. 725, pp. 952-968, 2017.

[26] A. Tahbaz-Salehi and A. Jadbabaie, “On consensus over random networks,” in *44th Annual Allerton Conference*, Citeseer, 2006.

[27] M. Alighanbari and J. P. How, “Decentralized task assignment for unmanned aerial vehicles,” *Proceedings of the 44th IEEE Conference on Decision and Control,* Seville, Spain, 2005, pp. 5668-5673.

[28] D. P. Bertsekas, “The auction algorithm for assignment and other network flow problems: A tutorial,” *The Practice of Mathematical Programming*, vol. 20, no. 4, pp. 133-149, 1990.

[29] J. Guerrero and G. Oliver, “Auction and swarm multi-robot task allocation algorithms in real time scenar ios,” *Multi-Robot Systems, Trends and Development*, pp. 437-456, 2011.

[30] C. Wei, K. V. Hindriks, and C. M. Jonker, “Dynamic task allocation for multi-robot search and retrieval tasks,” *Applied Intelligence*, vol. 45, no. 2, pp. 383-401, 2016.

[31] D.-H. Lee, “Resource-based task allocation for multi-robot systems,” *Robotics and Autonomous Systems*, vol. 103, pp. 151-161, 2018.

[32] S. Sariel and T. Balch, “Real time auction-based allocation of tasks for multi-robot exploration problem in dynamic environments,” in *Proceedings of the AAAI-05 Workshop on Integrating Planning into Scheduling*, pp. 27-33, 2005.

[33] A. M. Kwasnica, J. O. Ledyard, D. Porter, and C. DeMartini, “A new and improved design for multiobject iterative auctions,” *Management science*, vol. 51, no. 3, pp. 419-434, 2005.

[34] W. Zhao, Q. Meng, and P. W. Chung, “A heuristic distributed task allocation method for multivehicle multitask problems and its application to search and rescue scenario,” *IEEE transactions on cybernetics*, vol. 46, no. 4, pp. 902-915, 2015.

[35] S. Zaman and D. Grosu, “A combinatorial auction-based mechanism for dynamic vm provisioning and allocation in clouds,” *IEEE Transactions on Cloud Computing*, vol. 1, no. 2, pp. 129-141, 2013.

[36] M. Nanjanath and M. Gini, “Repeated auctions for robust task execution by a robot team,” *Robotics and Autonomous Systems*, vol. 58, no. 7, pp. 900-909, 2010.

[37] F. Zitouni, R. Maamri, and S. Harous, “Fa-qabc-mrta: a solution for solving the multi-robot task allocation problem,” *Intelligent Service Robotics*, vol. 12, no. 4, pp. 407-418, 2019.

[38] F. Zitouni and R. Maamri, “Fa-setpower-mrta: A solution for solving the multi-robot task allocation problem,” in *IFIP International Conference on Computational Intelligence and Its Applications*, vol. 522, pp. 317-328, 2018.

[39] Zitouni and R. Maamri, “An adaptive protocol for dynamic allocation of tasks in a multi-robot system,” *2016 International Conference on Advanced Aspects of Software Engineering (ICAASE),* Constantine, 2016, pp. 128-133.

[40] F. Zitouni, S. Harous, and R. Maamri, “Towards a distributed solution to the multi-robot task allocation problem with energetic and spatiotemporal constraints,” *Computer Science*, vol. 21, no. 1, 2020.

[41] M. Otte, M. J. Kuhlman, and D. Sofge, “Auctions for multi-robot task allocation in communication limited environments,” *Autonomous Robots*, vol. 44, no. 3, pp. 547-584, 2020.

[42] F. Xue, H. Tang, Q. Su, and T. Li, “Task allocation of intelligent warehouse picking system based on multi-robot coalition.” *KSII Transactions on Internet & Information Systems*, vol. 13, no. 7, pp. 3566-3582, 2019.

[43] H.-L. Choi, L. Brunet, and J. P. How, “Consensus-based decentralized auctions for robust task allocation,” *IEEE transactions on robotics*, vol. 25, no. 4, pp. 912-926, 2009.

[44] H.-L. Choi, A. K. Whitten, and J. P. How, “Decentralized task allocation for heterogeneous teams with cooperation constraints,” Proceedings of the 2010 American Control Conference, Baltimore, MD, 2010, pp. 3057-3062.

[45] G. P. Das, T. M. McGinnity, S. A. Coleman, and L. Behera, “A fast distributed auction and consensus process using parallel task allocation and execution,” *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, San Francisco, CA, 2011, pp. 4716-4721.

[46] L. Johnson, S. Ponda, H.-L. Choi, and J. How, “Asynchronous decentralized task allocation for dynamic environments,” *in Infotech@ Aerospace 2011*, 2011, pp. 1441-1452.

[47] E. Kivelevitch, K. Cohen, and M. Kumar, “A market-based solution to the multiple traveling salesmen problem,” *Journal of Intelligent & Robotic Systems*, vol. 72, no. 1, pp. 21-40, 2013.

[48] G. Binetti, D. Naso, and B. Turchiano, “Decentralized task allocation for surveillance systems with critical tasks,” *Robotics and Autonomous Systems*, vol. 61, no. 12, pp. 1653-1664, 2013.

[49] F. Zitouni, S. Harous, and R. Maamri, “A distributed approach to the multi-robot task allocation problem using the consensus-based bundle algorithm and ant colony system,” *IEEE Access*, vol. 8, pp. 27479-27494, 2020.

[50] F. Zitouni, S. Harous, and R. Maamri, “A distributed solution to the multi-robot task allocation problem using ant colony optimization and bat algorithm,” in *Advances in Machine Learning and Computational Intelligence*, 2020, pp. 477-490.

[51] S. Zhang and S. Wang, “Flexible assembly job-shop scheduling with sequence-dependent setup times and part sharing in a dynamic environment: Constraint programming model, mixed-integer programming model, and dispatching rules,” *IEEE Transactions on Engineering Management*, vol. 65, no. 3, pp. 487- 504, 2018.

[52] S. Nguyen, M. Zhang, M. Johnston, and K. C. Tan, “Automatic programming via iterated local search for dynamic job shop scheduling,” *IEEE transactions on cybernetics*, vol. 45, no. 1, pp. 1-14, 2014.

[53] M. Agarwal, N. Kumar, and L. Vig, “Non-additive multi-objective robot coalition formation,” *Expert Systems with Applications*, vol. 41, no. 8, pp. 3736–-3747, 2014.

[54] K. Padmanabhan Panchu, M. Rajmohan, R. Sundar, and R. Baskaran, “Multi-objective optimisation of multi-robot task allocation with precedence constraints,” *Defence Science Journal*, vol. 68, no. 2, pp. 175-182, 2018.

[55] L. Jin, S. Li, H. M. La, X. Zhang, and B. Hu, “Dynamic task allocation in multi-robot coordination for moving target tracking: A distributed approach,” *Automatica*, vol. 100, pp. 75–-81, 2019.

[56] T. Jech, “Set theory,” *Springer Science & Business Media*, 2013.

[57] L. Comtet, “Advanced Combinatorics: The art of finite and infinite expansions,” Springer Science & Business Media, 2012.

[58] W. Asakly, A. Blecher, C. Brennan, A. Knopfmacher, T. Mansour, and S. Wagner, “Set partition asymptotics and a conjecture of gould and quaintance,” *Journal of Mathematical Analysis and Applications*, vol. 416, no. 2, pp. 672-682, 2014.

[59] J. F. Allen, “Maintaining knowledge about temporal intervals,” in *Readings in qualitative reasoning about physical systems*, pp. 361-372, 1990.

[60] R. Dechter, I. Meiri, and J. Pearl, “Temporal constraint networks,” *Artificial intelligence*, vol. 49, no. 1-3, pp. 61-95, 1991.

BIOGRAPHIES OF AUTHORS

Farouq Zitouni obtained his PhD in Computer Science from Abdelhamid Mehri University, Constantine, Algeria. He is currently a Professor at the department of computer science at Kasdi Merbah University - Ouargla and member of LIRE Laboratory, Abdelhamid Mehri University-Constantine, Algeria. He is working on the field of multi-robot task allocation and multi-robot systems. His main research interests include optimization and metaheuristic algorithms. He published some papers in international conferences and journals. Currently he is visiting the UAE University.

Ramdane Maamri is an associate professor of computer science in University of Constantine 2, Algeria. He received his master from University of Minnesota, USA, and his Ph.D in Computer Science from University of Mentouri Constantine, Algeria. He has participated in many national and international research projects and supervised a number of masters and Ph.D students. He is currently the head of software technologies and information systems department. His research interests include software engineering, verification and validation, web services, multi agent systems and ambient systems.

Saad Harous obtained his PhD in computer science from Case Western Reserve University, Cleveland, OH, USA in 1991. He has more than 30 years of experience in teaching and research in three different countries: USA, Oman and UAE. He is currently a Professor at the College of Information Technology, in the United Arab Emirates University. His teaching interests include programming, data structures, design and analysis of algorithms, operating systems and networks. His research interests include parallel and distributed computing, P2P delivery architectures, wireless networks and the use of computers in education and processing Arabic language. He has published more than 200 journal and conference papers and book chapters. He is an IEEE senior member.