

Multi-task Coalition Formation based on Task-group Structure

Chaofeng Lin¹ Shanli Hu^{1,2} Shengfu Zheng¹ Shexiong Su¹

¹Department of Computer Science and Technology, Fuzhou University, Fuzhou 350002, P. R. China

²Key Laboratory for Computer Science, Chinese Academy of Sciences, Beijing 100080, P. R. China

Abstract

Coalition formation is an important manner of communication among agents in MAS. Some previous works only pay attention to forming a coalition for each task respectively, and the gross income of MAS may be not maximal. Thus, combination task called *task-group* is originally taken into account. In this way, the main problem lies that to search a partition of the given set of tasks called *task-group structure* to maximize the gross income. However, considering all of those possible *task-group structures* is impossible with the increasing number of tasks. Thus, an algorithm is developed to predigest the graph of *task-group structures*, and a multi-task coalition formation algorithm is proposed.

Keywords: Multi-agent system, Coalition formation, Combination task, Task-group, Task-group structure

1. Introduction

In MAS, an agent usually needs to cooperate with others for achieving a task or improving its efficiency due to its limited knowledge and capabilities. Coalition formation is an important manner of communication among agents in MAS, also a key problem in the research of MAS.

In this paper, we originally adopt the main concept of combination auction to study the task-oriented coalition formation problem. It's well known that, bids are usually combined together to auction in order to gain more benefit. Analogously, if some tasks are combined together to assign to a big coalition, the gross income of MAS may be increased to some extent. Thus, to improve some previous works [1]-[3], combination task is novelly taken into account to increase the gross income of MAS as much as possible in this paper. However, considering all of those possible combinations of tasks is impossible with the augmentation of the set of tasks. Thus, our main contributions can be summarized as follows:

- ◆ Given the comparability between auction problem and task-oriented coalition formation problem, we originally take combination task called *task-group* into account to maximize the gross income of MAS.
- ◆ To address this problem, some strategies are developed; e.g. using a graph called *relation-restriction-graph* to describe the partial precedence order between tasks.
- ◆ In order to decrease the number of possible combinations of tasks, we introduce a graph of *task-group structure* where each node is corresponding to a partition of the set of tasks, called *task-group structure*; what's more, according to the *relation-restriction-graph* of tasks, an algorithm is proposed to predigest the graph of *task-group structure*, effectively decreasing the searching space.

We begin by analyzing some related works in section 2. Then, the problem we intend to solve is described in section 3. In section 4, the graph of *task-group structure* and *relation-restriction-graph* are introduced firstly; then an algorithm is developed to predigest the graph of *task-group structure*; finally, a coalition formation algorithm is proposed to address the coalition formation problem. In section 5, the simulation results are given. This paper is ended with a conclusion in section 6.

2. Related work

By far, the coalition formation problem is investigated from three main aspects. Firstly, some scholars solve the coalition formation problem based on coalition structure, considering it as searching for a coalition structure with the greatest gross income in the coalition structure graph. Recently, Dang and Jennings[4] introduced an algorithm that only searches for the coalition structures whose maximal coalition cardinalities are not less than a given value; then they proposed a near-optimal anytime coalition structure generation algorithm by partitioning the space in terms

of a small set of elements, which is the best algorithm for searching a near-optimal solution in the coalition structure graph so far [9]. We also presented a smaller granularity of search (cardinality structure) and developed a novel anytime algorithm which effectively minimizes the searching space [10].

Secondly, some researches of coalition formation problem are also concentrated on choosing a strategy for distributing coalition value so that coalition structure is stable. Its basic theory is N-person games theory, such as *Shapley value*, *nucleolus*, *core* and so on. In the recent years, some scholars study the benefit distribution problem in an open environment. In order to distribute benefit with uncertain information, Kraus and Shehory et al proposed four kinds of distribution strategies, i.e. *equal distribution*, *proportional distribution*, *kernel distribution* and *compromise* [6]. What's more, Makoto Yokoo and Naoki Ohta et al [7]-[8] studied the coalitional games in open anonymous environments. They discussed the limitations of some existing solution concepts (e.g. *Shapley value*, *core*, *least core*, and *nucleolus*), and introduced a new solution concept called *anonymity-proof core* to prevent agents from using multiple identifiers and hiding their skills [6]. Based on the anonymity-proof core, they proposed a compact representation of the outcome function in [7].

What's more, people still pay much attention to the task-oriented coalition formation problem, i.e. choosing some agents in a given set of agents to form a coalition for each task in a given set of tasks so that the gross income of MAS is maximized. To address this problem, Kraus and Shehory, et al developed a protocol that enables agents to negotiate and form coalitions, and provides them with *marginal heuristic* and *expert heuristic* for choosing coalition partners; they solved the coalition formation problem with uncertain heterogeneous information [5]. Chalkiadakis, et al proposed a model that utilizes Bayesian reinforcement learning in a way that enables coalition participants to reduce their uncertainty regarding coalitional values and the capabilities of others [1]. Apart from the above work, we also adopted some methods of swarm intelligence to address the task-oriented coalition formation problem. We proposed a multi-task coalition generation algorithm based on an improved ant colony algorithm in [2]; and originally adopted the binary PSO (Particle Swarm Optimization) to address the coalition formation problem, taking overlapping coalition and parallelizability into account simultaneously [3]. However, all of the above work on the task-oriented coalition formation problem only paid attention to forming a coalition for each task respectively. So the gross income they gain may not be the maximal.

3. Problem description

Suppose there is a set of n agents, $A = \{a_1, a_2, \dots, a_n\}$. Each agent a_i has a vector of real non-negative capabilities, $B_{a_i} = \langle b_1^{a_i}, b_2^{a_i}, \dots, b_l^{a_i} \rangle$. Each capability is a property of an agent that quantifies its ability to perform a specific action. There is also a set of m tasks, $T = \{t_1, t_2, \dots, t_m\}$. For the satisfaction of each task t_i , a vector of capabilities, $B_{t_i} = \langle b_1^{t_i}, b_2^{t_i}, \dots, b_l^{t_i} \rangle$, is necessary. Being different from the assumption in [3], we suppose that there is a partial precedence order between the tasks, $t_1 \preceq t_2 \preceq \dots \preceq t_{k_1}, \dots, t_{k_1} \preceq t_{k_2} \preceq \dots \preceq t_{k_k}$, where $t_i \preceq t_j$ means that t_i has a higher precedence order than t_j or the same precedence order with t_j .

Some definitions are given as follows:

◆ Definition 1. Coalition and Coalition value

A coalition can be defined as a group of agents who have decided to cooperate in order to achieve a common task, and the value $V(C)$ of coalition C can be defined as the net income gained after achieving a task through the cooperation of its members.

◆ Definition 2. Coalition capability vector

A coalition C has a vector of capabilities $B_C = \sum_{a_i \in C} B_{a_i}$, which is the sum of capabilities that agents contribute to the specific coalition. A coalition C can perform a task t only if the vector of capabilities necessary for its fulfillment B_{t_i} satisfies $\forall j, 1 \leq j \leq l, b_j^C \geq b_j^{t_i}$, and in the case of precedence order, t_i has no unsatisfied tasks with a higher precedence order.

◆ Definition 3. Task-group and Task-group structure

A task-group G ($G \neq \phi$) can be defined as a subset of T , and a task-group structure GS as a partition of T ; i.e. any two task-groups of a GS are not intersecting.

Now that some necessary definitions and assumptions have been made above, we can formally present the problem. Given a set of n agents $A = \{a_1, a_2, \dots, a_n\}$ and a set of m tasks $T = \{t_1, t_2, \dots, t_m\}$ with a partial precedence order, the problem we solve is searching an optimal task-group structure GS , and forming a fit coalition for each task-group of GS so that the gross income of MAS is maximal and the precedence order is respected as well.

4. The algorithms

4.1. The graph of task-group structure

In order to search an optimal *task-group structure*, a graph of *task-group structure* is introduced in this section (see Fig. 1, $m=4$). We can see that all of those possible *task-group structures* are included in the graph and each node is corresponding to a *task-group structure*. In layer L_i , there are all of those *task-group structures*, each of which is consisted of i *task-groups*. The arcs downwards represent the combination of two *task-groups*, and the arcs upwards represent the decomposition of two *task-groups*.

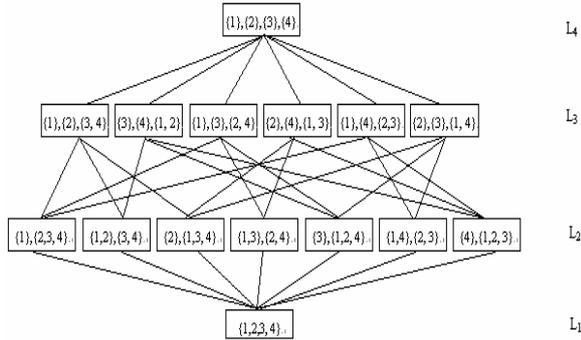


Fig. 1: The graph of *task-group structure* ($m=4$)

The straightforward strategy for searching an optimal *task-group structure* is using the methods of Breadth-First-Search or the Depth-First-Search from up to down or down to up. However, as for m tasks, there are $2^m - 1$ possible *task-groups*, and the number of *task-group structures* is increasing exponentially with the increasing of m . So searching all of the nodes in the graph of *task-group structure* is impossible. Some predigestions of the graph are indeed required.

4.2. Predigestions of the graph of task-group structure

In order to predigest the graph of *task-group structure*, now we introduce the concept of *relation-restriction-graph* used to describe the precedence order between tasks in this section. Similarly, we consider the situation that the number of tasks is 4, and then assume that the precedence order is as follows: $t_1 \preceq t_3 \preceq t_4$ and $t_2 \preceq t_4$. The corresponding *relation-restriction-graph* is given in Fig. 2.

In Fig.2, node t_0 is imaginary; the arc ' \rightarrow ' describes a restriction relation. ' $t_i \rightarrow t_j$ ' represents that t_j is directly restricted by t_i ; we call t_i a direct preceding task of t_j , and call t_j a direct subsequent task of t_i . ' $t_i \rightarrow t_j \rightarrow t_k$ ' represents t_k is indirectly restricted by t_i ; we call t_i a indirect preceding task of t_k , and t_k a indirect subsequent task of t_i . What's more, we assume

that all of the direct preceding tasks and indirect preceding tasks of t_i are the indirect preceding tasks of t_k , and that all of the direct subsequent tasks and indirect subsequent tasks of t_k are the indirect subsequent tasks of t_i .

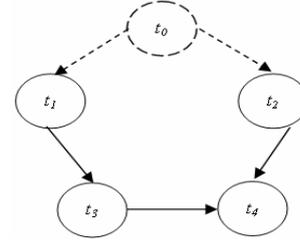


Fig. 2: Relation-restriction-graph ($m=4$)

Now some other definitions are made as follows:

◆ Definition 4. Root task

A root task is defined as a direct subsequent task of t_0 .

◆ Definition 5. Illegal *task-group*

An illegal *task-group* G^* is defined as a special *task-group*, which satisfies that there are at least two tasks in G^* , between which there exists a direct or indirect restriction relation.

◆ Definition 6. Illegal *task-group structure*

An illegal *task-group structure* GS^* can be defined as a special *task-group structure* which satisfies that there is at least one element of GS^* so that a subset of this element is an illegal *task-group*.

Based on the definitions and the assumptions above, some remarks can be easily educed as follows:

- ◆ Remark 1. The indirect restriction relation is a partial order relation.
- ◆ Remark 2. If t_i and t_j are root tasks, $\{t_i, t_j\}$ is not an illegal *task-group*.
- ◆ Remark 3. $\{t_i, t_k\}$ is an illegal *task-group* if $\{t_i, t_j\}$ is an illegal *task-group*, t_i is a direct or indirect preceding task of t_j , and t_k is a direct or indirect subsequent task of t_j .
- ◆ Remark 4. If $\{t_i, t_j\}$ is an illegal *task-group*, so do the *task-groups* one of whose subset is $\{t_i, t_j\}$.

As referred above, it may be impossible to search all of the nodes in a graph of *task-group structure*. In fact, those tasks among which there doesn't exist a direct or indirect restriction relation should not be combined; or else, during the process of performing tasks, some coalition members will do nothing other than waiting for the achievements of all of the direct or indirect preceding tasks of their corresponding task,

so that the costs of performing tasks are increased and the gross income of MAS is decreased. So according to the partial precedence order between tasks, we develop an algorithm to predigest the graph of *task-group structure* (see algorithm 1).

The predigesting algorithm proceeds as follows:

- ◆ **Step 1.** Convert the partial precedence order between tasks to a *relation-restriction-graph*.
- ◆ **Step 2.** According to the *relation-restriction-graph*, obtain the set of all of the illegal *task-group structures* based on definition 6, remark 2 and remark 3.
- ◆ **Step 3.** Search downwards the graph of *task-group structure* from layer L_{m-1} using the method of Breadth-First-Search; for each layer L_i , find out all of the nodes (the set of the nodes is marked as N) corresponding to the illegal *task-group structures*, and then delete some arcs and nodes as follows:
 - ① the arcs connecting a node in N with a node in layer L_{i+1}
 - ② the nodes in layer $L_{i-1}, L_{i-2}, \dots, L_1$ gained after combining two *task-groups* of a node in N (the set of these nodes is marked as N^*)
 - ③ the arcs related to the nodes in N^*
 - ④ the nodes in N
- ◆ **Step 4.** Return the predigested graph of *task-group structure*.

Algorithm 1: The algorithm for predigesting the graph of *task-group structure*

Now we can use algorithm 1 to predigest the graph of *task-group structure* showed in Fig.1. The result is given in Fig.3. Comparing Fig.1 with Fig.3, we can see that the number of the nodes need to search is decreasing sharply.

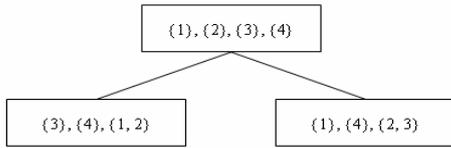


Fig. 3: The predigested graph of *task-group structure* ($m=4$)

4.3. The coalition formation algorithm

Obviously, according to definition 3, a *task-group* can be considered as a special task, whose capability is the summation of all the capabilities of its elements. Similarly, a *task-group structure* can be considered as

a set of some special tasks. In our previous work [2]-[3], two multi-task coalition formation algorithms have been developed. Thus, based on the analysis above, an universal coalition formation algorithm which takes *task-group* and *task-group structure* into account can be summarized (see algorithm 2).

The coalition formation algorithm proceeds as follows:

- ◆ **Step 1.** If time is permitted, search the graph of *task-group structure* after predigesting using algorithm 1 from up to down using the method of Breadth-First-Search; for each node k , proceed as follows:
 - ① Form a coalition for each *task-group* in node k using the coalition formation algorithm developed in [2] or [3].
 - ② Count the gross income of MAS related to node k ; update the optimal *task-group structure*, the corresponding coalitions and the maximal gross income of MAS gained so far.
- ◆ **Step 2.** Return the optimal *task-group structure*, the corresponding coalitions and the maximal gross income of MAS gained so far.

Algorithm 2: The coalition formation algorithm

In algorithm 2, the predigested graph of *task-group structure* is searched from up to down, so all the nodes whose elements are all consisted of single task are considered firstly during the process of coalition formation. Thus, whenever algorithm 2 is ended, the gross income of MAS gained by algorithm 2 is not less than that gained by any of the previous algorithms.

5. Simulations

5.1. The predigesting algorithm

As referred above, there is a partial precedence order between the tasks, $t_1 \preceq t_2 \preceq \dots \preceq t_{n_1}, \dots, t_{k_1} \preceq t_{k_2} \preceq \dots \preceq t_{k_k}$. We also assume that $n^* = \max\{n_1, n_2, \dots, n_k\}$.

In this section, we study how the parameters n^* and k_1 influence the result of predigesting. So we perform several groups of experiments to examine it. Each group of experiments are performed with the same number of tasks, but different n^* or different k_1 . Fig.4 and Fig.5 show the results of two experiments we have performed.

In Fig. 4 and Fig. 5, the number of nodes in the predigested graph of *task-group structure* is decreasing as the value of n^* increases, and increasing as the value of k_1 increases. So it can be concluded

that n^* and k_1 represent the intensity of restriction relations among tasks to some extent; that is to say, there may be more and more restriction relations among tasks with the increasing of n^* or the decreasing of k_1 , which also means that there may be more and more illegal *task-group structures* in the graph of *task-group structure* with the increasing of n^* or the decreasing of k_1 .

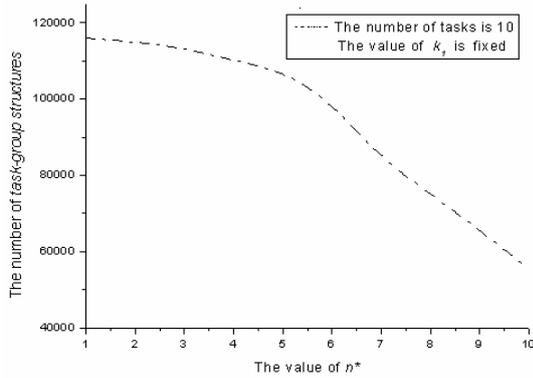


Fig. 4: The result of the group of experiments, where the number of tasks is 10, the value of k_1 is fixed and the value of n^* is varying from 1 to 10

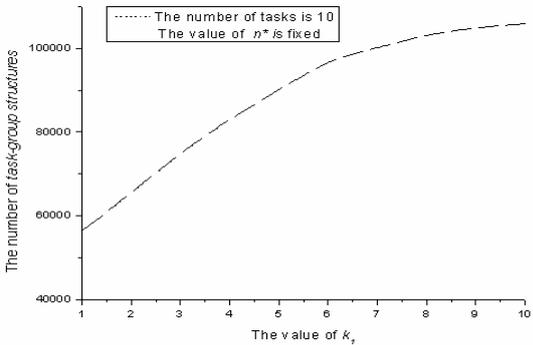


Fig. 5: The result of the group of experiments where the number of tasks is 10, the value of n^* is fixed and the value of k_1 is varying from 1 to 10

5.2. The gross income of MAS

According to definition 1, the value $V(C)$ of coalition C can be defined as the net income gained after achieving a task through the cooperation of its members. In the experiments, $V(C)$ is computed by using the following equation: $V(C) = P(t) - R(C) - Q(C)$, where $P(t)$ is the income due to the achievement of task t ; $Q(C)$ is the extra costs of coalition formation, generally representing the cost of communication among agents; $R(C)$ is the sum of converted costs of the members' capabilities contributing to coalition C . In these experiments, we

use the algorithm developed in [2] to form a coalition for each *task-group*. What's more, there are eight tasks with a partial precedence order, $t_1 \preceq t_2 \preceq t_4$, $t_3 \preceq t_5 \preceq t_6$ and $t_7 \preceq t_8$.

	G_1	G_2	G_3	G_4	G_5	Gross income
1	$\{t_1, t_3, t_4\}$	$\{t_2, t_5\}$	$\{t_6\}$	$\{t_7, t_8\}$	-	605.9
2	$\{t_1, t_3\}$	$\{t_4\}$	$\{t_5, t_6\}$	$\{t_2, t_7\}$	$\{t_8\}$	636.8
3	$\{t_1, t_3\}$	$\{t_4, t_5\}$	$\{t_6\}$	$\{t_2, t_7\}$	$\{t_8\}$	685.1

Table 1. The gross income of MAS of different *task-group structures*

In Table 1, the gross income of MAS gained in experiment 3 (in row 3) is higher than the other two experiments because there doesn't exist any restriction relations among tasks of each *task-group* in row 3. As referred in section 4, if the tasks among which there doesn't exist a direct or indirect restriction relation are combined together, the costs of performing tasks may be increased, and the gross income of MAS may be decreased simultaneously. The results of experiments prove our words.

6. Conclusions

In this paper, we originally adopt the main concept of combination auction to address the task-oriented coalition formation problem. Combination task called *task-group* is taken into account during the process of coalition formation for the first time, guaranteeing that the gross income of MAS gained is not less than that of previous work. In order to make it possible to search an optimal *task-group structure* in the graph of *task-group structure*, we develop an algorithm to predigest the graph of *task-group structure*, effectively decreasing the searching space. Based on the work above, we finally propose a multi-task coalition formation algorithm, which is proved effective and feasible by the simulations.

In the future, how to further decrease the searching space is still a problem need to solve. What's more, we will also pay attention to introducing combination task into the coalition formation problem in an open environment

Acknowledgement

This work is partially supported by National Nature Science Foundation of China (Grant No. 60573076).

References

- [1] G. Chalkiadakis and C. Boutilier, Bayesian reinforcement learning for coalition formation under uncertainty. *Prof. of the 3rd International Joint Conference on Autonomous Agents and Multiagent Systems*, pp.1090-1097, 2004.
- [2] C. F. Lin, S. L. Hu, S. F. Zheng and S. X. Su, Multi-task Coalition Generation Algorithm Based on an Improved Ant Colony Algorithm. *Journal of Computer Research & Development*, 43(Suppl.): 176-181, 2006.
- [3] C. F. Lin and S. L. Hu, Multi-task Overlapping Coalition Parallel Formation Algorithm. *Prof. of the 6th International Joint Conference on Autonomous Agents and Multiagent Systems*, pp.1252-1254, 2007.
- [4] V. D. Dang and N .R. Jennings, Generating Coalition Structures with Finite Bound from the Optimal Guarantees. *Prof. of the 3rd International Joint Conference on Autonomous Agents and Multiagent Systems*, pp.564-571, 2004.
- [5] S. Kraus, O. Shehory and G. Taase, Coalition formation with uncertain heterogeneous information. *Prof. of the 2nd International Joint Conference on Autonomous Agents and Multiagent Systems*, pp.1-8, 2003.
- [6] S. Kraus, O. Shehory and G. Taase, The Advantages of Compromising in Coalition Formation with Incomplete Information. *Prof. of the 3rd International Joint Conference on Autonomous Agents and Multiagent Systems*, pp. 588-595, 2004.
- [7] M. Yokoo, V. Conitzer, T. Sandholm, Naoki Ohta, and Atsushi Iwasaki, Coalitional Games in Open Anonymous Environments. *Prof. of the 20th National Conference on Artificial Intelligence*, pp.509-515, 2005.
- [8] N. Ohta, A. Iwasaki, M. Yokoo, Kohki Maruono, V. Conitzer and T. Sandholm, A Compact Representation Scheme for Coalitional Games in Open Anonymous Environments. *Prof. of the 21st National Conference on Artificial Intelligence*, pp.509-514, 2006.
- [9] T. Rahwan, S. D. Ramchurn, V. D. Dang and N. R. Jennings, Near-optimal anytime coalition structure generation. *Prof. of the 20th International Joint Conference on Artificial Intelligence*, pp.2365-2371, 2007.
- [10] S.X. Su, S. L. Hu and C. Y. Shi, Coalition Structure Generation with Worst Case Guarantees based on Cardinality Structure. *Prof. of the 6th International Joint Conference on*

Autonomous Agents and Multiagent Systems, pp.1182-1184, 2007.