Towards a formal model for task allocation via coalition formation
Leila Amgoud

To cite this version:
Leila Amgoud. Towards a formal model for task allocation via coalition formation. 4th international joint conference on Autonomous agents and multiagent systems (AAMAS 2005), ACM SIGAI: Special Interest Group on Artificial Intelligence, Jul 2005, Utrecht, Netherlands. pp.1185–1186, 10.1145/1082473.1082685. hal-04323102

HAL Id: hal-04323102
https://hal.science/hal-04323102
Submitted on 5 Dec 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Towards a Formal Model for Task Allocation via Coalition Formation

Leila Amgoud
IRIT - CNRS
118, route de Narbonne
Toulouse, France
amgoud@irit.fr

ABSTRACT

This paper focuses on the problem of generating coalition structures for task allocation via coalition formation. It provides a unified formal framework for constructing those coalitions structures. The framework takes as input a set of coalitions whose structures are abstract, a conflict relation between the coalitions, and a preference relation between the coalitions and returns the coalitions structures. Three semantics for coalitions structures will be proposed: a basic semantics which will return a unique coalition structure, stable semantics and preferred semantics. These two last may return several coalitions structures at the same time.

Keywords
Coalition formation

1. INTRODUCTION

Generally, to perform complex tasks in multi-agent environments, agents need to form coalitions which are temporary associations between agents in order to carry out joint tasks. As argued in [1, 3, 4, 5, 6], task allocation via coalition formation follows a three steps process: i) generating the coalition structures. The idea here is to form the coalitions such that agents within a coalition should coordinate to achieve a task (or a set of tasks), but those in different coalitions do not. ii) Discussing these structures between the agents in order to select the one which will be adopted. iii) Distributing the gain between the agents of each coalition in the coalitions structure. The way in which the coalitions structures are generated depend broadly on the studied problem. In some applications, for example, it may be required that the tasks are independent, or that a single agent should belong only to one coalition at the same time.

Inspired from work on argumentation theory, particularly the system developed in [2], this paper provides a unified and general formal framework for generating the coalitions structures. That framework is defined in terms of a set of coalitions, a conflict relation between these coalitions and finally a preference relation between the coalitions. The framework returns three semantics of coalition structures: the basic semantics which returns a unique coalitions structure, stable semantics and preferred semantics which are two different refinements of the basic one and may return several coalitions structures at the same time.

2. FORMAL FRAMEWORK

Task allocation via coalition formation can be defined as a finite set \( \mathcal{N} \) of agents who should achieve a finite set \( \mathcal{T} \) of tasks. Each agent aims at maximizing its own satisfaction and also the satisfaction of the whole multi-agent system in which it is a member.

**Definition 1** (Formal framework). A framework for generating coalition structures (FGS) is a triplet \( < \mathcal{C}, \mathcal{R}, \succ > \) where \( \mathcal{C} \) is a set of coalitions, \( \mathcal{R} \) is a binary relation representing a defeat relationship between coalitions, \( \mathcal{R} \subseteq \mathcal{C} \times \mathcal{C} \), and \( \succ \) is a (partial or complete) preordering on \( \mathcal{C} \).

**Definition 2.** Let \( C_1, C_2 \in \mathcal{C} \), and \( S \subseteq \mathcal{C} \). \( C_1 \) attacks \( C_2 \) iff \( C_1 \mathcal{R} C_2 \) and not \( (C_2 \succ C_1) \). \( S \) is conflict-free iff \( \not\exists C_1, C_2 \in S \) s.t \( C_1 \) attacks \( C_2 \). \( S \) defends \( C_1 \) iff for all \( C_2 \) such that \( C_2 \) attacks \( C_1 \), then there is \( C_3 \in \mathcal{C} \) s.t \( C_3 \) attacks \( C_2 \).

Let’s define the basic coalition structure. Intuitively, it is clear that a non-attacked coalitions will belong to the coalition structure. \( C_{Att} \) gathers all such coalitions. This notion is very restrictive since it discards coalitions which appear “good”.

**Definition 3** (Coalitions structure). Let \( < \mathcal{C}, \mathcal{R}, \succ > \) be a finitary FGS. The basic coalitions structure is:

\[
\mathbf{S}_{R, \succ} = \bigcup \{ F^{\geq 0}(0) \} = C_{Att} \cup \bigcup \{ F^{\geq 1}(C_{Att}) \}
\]

where \( F(S) = \{ C \in \mathcal{C} | C \text{ is defended by } S \} \).

In some cases, the set \( \mathbf{S}_{R, \succ} \) may be empty. This is not always desirable in multi-agents applications. In order to palliate this problem, we define the stable structures and the preferred ones.

**Definition 4** (Stable structures). Let \( < \mathcal{C}, \mathcal{R}, \succ > \) be a FGS, and \( S \subseteq \mathcal{C} \). \( S \) is a stable structure iff \( S \) is conflict-free and \( S \) defeats any coalition which is not in \( S \).

A framework FGS may have several stable structures. These stable structures correspond to different ways of achieving the tasks.
**Definition 5 (Preferred structures).** Let \( \langle C, \mathcal{R}, \succ \rangle \) be a FGS, and \( S \subseteq C \). \( S \) is a preferred extension if \( S \) is conflict-free; \( S \) defends all its elements, and \( S \) is maximal (for \( \subseteq \)) among the sets satisfying the 2 above conditions.

Note that each framework FGS has at least one preferred structure.

**Property 1.** Each stable structure is also a preferred one. The reverse is not always true. Moreover, the coalition structure \( \mathcal{S}_{\mathcal{R}, \succ} \) is included in every stable (resp. preferred) structure.

### 3. Illustration

Let’s consider the problem of coalition formation described in [5]. In that work, a multi-agent system is supposed to perform a service that requires several criteria \( \langle c_1, \ldots, c_r \rangle \). Each agent \( a_i \in N \) is supposed to have non-negative capabilities \( B^i = \langle b^i_1, \ldots, b^i_{|b^i|} \rangle \). A capability \( b^i_j \) represents the capacity of the agent \( a_i \) regarding the criterion \( c_j \). To each task \( t \in T \) a vector \( B^t = \langle b^t_1, \ldots, b^t_{|b^t|} \rangle \) of its capabilities is given. An element \( b^t_j \) represents the amount of \( c_j \) necessary for its satisfaction. The function \( \text{Value} \) returns the value of a given coalition. It is assumed that the tasks are independent, an agent cannot belong to more than one coalition at a time, and a coalition can work on a single task at a time.

A coalition should be minimal since each coalition has a cost, and the more the coalition is large, the more costly it is. Moreover, an agent cannot be in a coalition if it is not useful and it cannot help in the achievement of the task. Before giving the formal definition of a coalition, let’s first define formally when a task is achievable.

**Definition 6.** Let \( C \subseteq N \) and \( t \in T \). \( C \) achieves the task \( t \), denoted by \( C \models t \), if \( \forall 1 \leq j \leq r, \sum_{a_i \in C} b^i_j < b^t_j \).

The above definition says that a task is achievable by a group of agents if the capabilities of the agents taken together, are sufficient to what is required by the task.

**Definition 7 (Coalition).** A coalition is a pair \( \langle C, t \rangle \) s.t. \( C \subseteq N \), \( t \in T \), \( C \models t \), \( C \) is minimal for (\( \subseteq \)) among the sets satisfying the above conditions. \( C \) is the support of the coalition, and \( t \) its task. \( C(AS) \) is the set of all the coalitions.

The value of a coalition may be equal to the benefit obtained from the coalition minus the cost of that coalition. For the sake of simplicity, we suppose that this value is given and it is a numerical value. The values of coalitions make it possible to compare them.

**Definition 8.** Let \( \langle C_1, t_1 \rangle, \langle C_2, t_2 \rangle \in C(AS) \). \( \langle C_1, t_1 \rangle \) is more beneficial than \( \langle C_2, t_2 \rangle \), denoted \( \langle C_1, t_1 \rangle > \langle C_2, t_2 \rangle \) \( \iff \text{Value}(\langle C_1, t_1 \rangle) > \text{Value}(\langle C_2, t_2 \rangle) \).

The coalition structures should satisfy the hypothesis already fixed when defining the problem. The first requirement is that an agent cannot belong to more than one coalition at the same time. This kind of conflict will be called “Interferes”.

**Definition 9 (Interfering coalitions).** Let \( \langle C_1, t_1 \rangle, \langle C_2, t_2 \rangle \in C(AS) \). \( \langle C_1, t_1 \rangle \) interferes with \( C_2, t_2 \) \( \iff C_1 \cap C_2 \neq \emptyset \).

The second requirement is that the same task cannot be affected to more than one coalition at the same time. In the coalition structure, it cannot then be the case that two coalitions achieve the same task. This requirement gives raise to another kind of conflict between coalitions. In what follows, this conflict will be called “Competition”.

**Definition 10 (Competing coalitions).** Let \( \langle C_1, t_1 \rangle, \langle C_2, t_2 \rangle \in C(AS) \). \( \langle C_1, t_1 \rangle \) is in competition with \( \langle C_2, t_2 \rangle \) \( \iff t_1 = t_2 \).

The two above relations are brought together in a unique definition of defeat as follows:

**Definition 11.** Let \( \langle C_1, t_1 \rangle, \langle C_2, t_2 \rangle \in C(AS) \). \( \langle C_1, t_1 \rangle \) defeats \( \langle C_2, t_2 \rangle \) \( \iff \langle C_1, t_1 \rangle \) interferes with \( \langle C_2, t_2 \rangle \) or \( \langle C_1, t_1 \rangle \) is in competition with \( \langle C_2, t_2 \rangle \).

The basic coalition structure of this system is: \( \mathcal{S}_{\mathcal{R}, \succ} = \bigcup_{t \in T} \{\mathcal{P}(t)\} \). The following result can be shown:

**Theorem 1.** If the agents do not misrepresent the capabilities of the others, and if they have all the same values for the different coalitions, then their respective frameworks will all return the same coalition structure. Thus, there is no need to the negotiation step.

This result is of great importance since it shows that with such a framework, more work is done by the agents themselves, and consequently this may minimize greatly the communication which is very costly.

### 4. Conclusion

Inspired from works on argumentation theory, we have proposed a unified, general and abstract framework for generating coalition structures in an elegant way. The formal framework has three components: a set of coalitions, a defeasibility relation between the coalitions, and finally a preference relation between the coalitions. In this abstract framework, the notion of coalition remains an abstract entity and its exact definition depends on the studied application. Regarding the notion of defeasibility, it is induced and defined from the constraints of the application. Finally, the preference relation comes from the values that agents can assign to each coalition.

We have proposed three semantics for the coalitions structures. This work is of great importance, since it allows agents to reason about the coalitions, and minimize the negotiation between agents in the second step of the coalition formation process. Moreover, this framework is general enough to capture different propositions made in the literature.

### 5. References


