
Algorithms for Finding Coalitions Exploiting a New Reciprocity Condition

Guido Boella, *Università di Torino, Italy.*
E-mail: guido@di.unito.it

Luigi Sauro, *Università di Napoli, Italy.*
E-mail: sauro@na.infn.it

Leendert van der Torre, *University of Luxembourg, Luxembourg.*
E-mail: leon.vandertorre@uni.lu

Abstract

We introduce a reciprocity criterion for coalition formation among goal-directed agents, which we call the indecomposable do-ut-des property. It refines an older reciprocity property, called the do-ut-des or *give-to-get* property by considering the fact that agents prefer to form coalitions whose components cannot be formed independently. A formal description of this property is provided as well as an analysis of algorithms and their complexity. We provide an algorithm to decide whether a coalition has the desired property, and we show that the problem to verify whether a single coalition satisfies the property is tractable. Moreover, we provide an algorithm to search all the sub-coalitions of a given coalition satisfying the new property. Even if this problem is not computationally tractable, we show that in several cases, also the complexity of this problem may decrease considerably.

1 Introduction

There are many kinds of coalitions, varying, for example, in the period they exist, the principle that glues them together, or the properties that characterize them. There are long term coalitions like a marriage or NATO, as well as short term coalitions like two agents in a single transaction on eBay. Coalitions can be based on reciprocity, like the eBay transaction, or on altruism, or on a combination like when you help the son of a friend of your neighbour, because the neighbour feeds your pets when you are on holidays. An example of a property distinguishing kinds of coalitions is whether larger or smaller coalitions are the most stable. These dimensions may be related, for example when for long term coalitions larger coalitions are more stable (as coalitional game theory promotes the grand coalition), maybe because on the long term goals are more abstract and fuzzy, whereas on the short term smaller coalitions may be preferred, because they are more easily formed. Consequently, there are also various decision problems, such as coalitional game theory in economics and coalition formation in artificial intelligence and multiagent systems.

2 Algorithms for finding coalitions exploiting a new reciprocity condition

Defining algorithms for the problem of coalition formation and studying their complexity has been addressed in several regards. For example, Sandholm et al. [29] study the problem of finding which partition of the population of agents maximizes a global utility function. Shehory and Kraus have studied in [34] task allocation problems via coalition formation.

In this paper we study the formation of coalitions among goal-directed agents. We start from a theory of social power and dependence introduced by Castelfranchi [11]. In this context a coalition is a group of agents agreeing to cooperate for the achievement of a shared goal or to exchange with each other the achievement of their own goals. The second case is of particular interest as different networks of exchanges can be considered but not all of them are realizable together. This is an important problem because, on the one hand, Castelfranchi's theory is abstract and covers most of the work in multiagent systems, and on the other hand, the number of applications in which groups of agents have to accomplish tasks together is growing steadily.

We assume that the formation of a coalition is supported by unanimous and enforced agreements, i.e. a coalition is effectively formed only when all its members agree to it (unanimity) and they cannot deviate from what established in the agreement, once they decide to enter it (enforcement). Under these assumptions, we develop a reciprocity criterion of admissibility, the indecomposable do-ut-des property (i-dud in the following), establishing which coalitions cannot be formed under the assumption that the agents are self-interested. The i-dud property is a refinement of an earlier reciprocity property, called the do-ut-des or *give-to-get* property [5], saying that an agent *sees to* a goal of another agent only if this enables it to *obtain*, directly or indirectly, the satisfaction of one of its own goals. The new reciprocity property refines the old one by taking into account also the fact that a coalition formation process can itself be costly and usually the costs involved in a coalition formation process increase with the number of agents involved. Furthermore, being a coalition agreed unanimously, the more agents are involved in it, the larger is the risk of defections, jeopardizing the formation of the coalition. Thus, for the kinds of coalitions satisfying the reciprocity property introduced in this paper, agents prefer to form coalitions which are as small as possible.

Even if a coalition satisfies the new property, then this does not imply that it will be formed in practice. Usually there will be many rivaling coalitions satisfying the property, and the theory only eliminates coalitions that will definitely not be formed. If agents consider whether to form coalition, they will also consider all subsets of this coalition. The research problem of this paper is therefore to develop algorithms to find coalition satisfies the new reciprocity condition. This breaks down into the following research questions:

1. How to define the new reciprocity condition?
2. How to decide whether a coalition satisfies the new reciprocity condition?
3. How to find all subcoalitions of a coalition satisfying the new reciprocity condition?
4. What is the complexity of the algorithms?

To find the algorithms, we use a technique familiar from social network analysis: we represent the problem as a graph, and use graph based algorithms.

We illustrate the algorithms not only by abstract examples, but we also show how they can be used in a simplified version of the chips and boxes game described by Grosz et al. [21], describing a general problem of exchange of resources. There are four agents and each of them has two chips and two boxes, both the chips and the boxes can be colored red, green

or blue. An agent has only the goal to fill each of his boxes with a chip of the same color of the box, and therefore agents exchange chips with each other.

With respect to other criteria of admissibility the i-dud property is novel in several aspects. First, at the best of our knowledge i-dud is the first criterion which explicitly takes into account whether a coalition can be decomposed in sub-coalitions that can formed independently. Secondly, in contrast with other criteria which need to compare coalitions with each other, as for example the notion of core in Cooperative Game Theory, in our approach a coalition has all the required information to check whether it satisfies the i-dud property. Finally, the problem to decide whether a coalition satisfies the i-dud property is tractable, whereas core membership is not tractable even in qualitative games [18, 39].

This article is based and extend previous work of the authors. First of all, [7] formalizes the notions of power and dependence which underlie the dependence networks used in this paper. Moreover, it introduces measures for quantifying the importance of an agent. The social importance of agents cannot be reduced to a property of a single agent, but it depends on an entire community in which that agent lives and in particular on its capability and exclusiveness with respect to the other agents to achieve their goals. That paper adapts van den Brink and Gilles' β -measure such that it can be used for these dependence networks. The β -measure is used to rank agents inside a dominance structure, i.e., a weighted directed graph where each node represents an agent and a weighted arc represents the strength of a dominance relation between two agents.

[32] introduces the notion of coalition in dependence networks and the two qualitative criteria for coalitions used in this article. The relation between the two criteria and the game theoretic notion of core is studied in [5]. Algorithms for coalitions in dependence networks have been introduced first in [6]. This work is extended in this article by detailing the algorithms, discussing examples, and contextualizing with respect to other works.

In Section 2 we define the new reciprocity property and provide some examples of this notion. In Section 3, we provide an algorithm to verify if a single coalition satisfies the i-dud property, and we extend it to search all the sub-coalitions of a given coalition satisfying the new property. In Section 4 we consider the complexity of the two problems and in Section 5 we illustrate the algorithms by a detailed chips and boxes example.

2 Reciprocity: from do-ut-des to i-dud coalitions

2.1 Dependence networks

Depending on the application, a coalition formation problem can be represented at different levels of abstraction. For example, if you want to study how agents have to coordinate in order to achieve a goal or how agents should optimally use their resources, then a multiagent system can involve resources, actions, plans [2, 25]. Instead, if you want to study which coalitions are strategically admissible to be formed, you usually do not need such fine-grained descriptions of a multiagent system, coalitions are directly described by means of the consequences that they can obtain collaborating, without any description about which joint plans agents have to perform [29, 39]. Following this idea, we describe potential coalitions abstracting from actions, plans or resources. However, in contrast with these approaches, we do not represent a coalition just indicating the set of goals that it can attain. We want to use the topology of goal exchanges inside a coalition to define our admissibility criterion.

4 Algorithms for finding coalitions exploiting a new reciprocity condition

Thus, inspired by Conte and Sichman [14] we represent a (potential) coalition as a labeled AND-graph of dependencies among agents. A labeled AND-graph consists of a set of nodes \mathcal{V} - which denotes the agents involved in the coalition - and a set \mathcal{E} of labeled AND-arcs. Denoting with Gl the goals exchanged in the coalition, a labeled AND-arc connects an agent a_i to a nonempty set of agents Q and it is labelled with a goal $g \in Gl$, so it can be represented as a triple (a_i, Q, g) . The meaning of such an arc is that the agent a_i desires the goal g and the set of agents Q is committed to the achievement of g . In order to represent a coalition a labeled AND-graph has to satisfy two further conditions. Since a coalition is intended as the result of an agreement process, the first condition is that only those agents that contribute to the achievement of some goals are admitted in this process. The second condition establishes that a coalition formation process does not involve private commitments that do not require any form of collaboration.

Definition 1 A dependence network G is a labeled AND-graph $\langle \mathcal{V}, \mathcal{E} \rangle$, where \mathcal{V} is a finite set of nodes and $\mathcal{E} \subseteq \mathcal{V} \times (2^{\mathcal{V}} \setminus \{\emptyset\}) \times Gl$ is a set of labeled AND-arcs. A coalition C is a subgraph of G satisfying two conditions:

1. for each node $a_j \in \mathcal{V}$, there exists at least an AND-arc (a_i, Q, g) such that $a_j \in Q$ and
2. \mathcal{E} does not contain an AND-arc in the form $(a_i, \{a_i\}, g)$.

With an abuse of notation we mean with $(Q, g) \in C$ that there exists a $(a_i, Q, g) \in \mathcal{E}$. We call (Q, g) a commitment of C . A sub-coalition C' is a subgraph of C where some commitments are suppressed, $(Q, g) \notin C'$, for some $(Q, g) \in C$.

2.2 Do-ut-des or give-to-get

We want to restrict the notion of coalitions assuring, first, reciprocity, and, second, that no sub-coalitions of a coalition can be formed independently. The former property is called do-ut-des or give-to-get [5, 32], and we refine it to match the second requirement called indecomposability.

The do-ut-des property assures that each chain of exchanges involved in a coalition returns something back to each agent involved in it. This property has been characterized in [5, 32] by means of a qualitative preference relation and a notion of dominance similar to those used in Game Theory, such as the notion of core. Here, instead, we propose a version of the do-ut-des property grounded on the topological property of chains of exchanges; the equivalence with the dominance-based approach is shown in [31]. In [32] a goal is meant as a state of affairs that, once attained by a group of agents, benefits one or, in case, more agents. To simplify our formalism, we consider that there do not exist two agents desiring the same goal.

We introduce some preliminary notions. Paths formalize possible *chains of exchanges* among the agents.

Definition 2 A finite sequence of AND-arcs $\mathcal{P} = (a_{i_1}, Q_1, g_1), \dots, (a_{i_n}, Q_n, g_n)$ is a path if for all $2 \leq h \leq n$, $a_{i_h} \in Q_{h-1}$. We denote with $out(a_i)$ the set of labeled AND-arcs outgoing the agent a_i , i.e. $out(a_i) = \{(a_i, Q, g)\}$. Given $\mathcal{O} \subseteq out(a_i)$, we call \mathcal{O}^* the propagation of \mathcal{O} , i.e. the set of AND-arcs containing all the paths $\mathcal{P} = (a_{i_1}, Q_1, g_1), \dots, (a_{i_n}, Q_n, g_n)$ such that (1) $(a_{i_1}, Q_1, g_1) \in \mathcal{O}$ and \mathcal{P} does not contain an arc in $out(a_i) \setminus \mathcal{O}$.

Note that $out^*(a_i)$ contains all (and only) the paths starting from agent a_i .

The do-ut-des property consists of two conditions. The first condition is an efficiency condition: there do not exist two distinct sets of agents in a coalition being committed to the achievement of the same goal. The second condition expresses the notion of reciprocity: given an AND-arc $(a_i, Q, g) \in \mathcal{E}$, each agent a_j involved in Q agrees to provide the goal g to a_i , only in the case this commitment returns the satisfaction of some of its goals by means of a path \mathcal{P} .

Definition 3 A coalition $(\mathcal{V}, \mathcal{E})$ satisfies the do-ut-des property iff

1. there do not exist two commitments $(Q, g), (Q', g) \in C$ such that $Q \neq Q'$ and
2. for all $(a_i, Q, g) \in \mathcal{E}$ and for all $a_j \in Q$, $(a_i, Q, g) \in out^*(a_j)$.

In Figure 1 (a) and (b) a circle represents the agent whose name is written within the circle, and an arrow between agent a_1 and agent a_2 labeled by g_1 represents that agent a_1 depends on agent a_2 for goal g_1 . The other dependencies can be read analogously. Both the coalitions in Figure 1 (a) and (b) satisfy Definition 3.

2.3 Indecomposable do-ut-des or i-dud

The do-ut-des property does not consider the possibility that a coalition can be decomposed in smaller sub-coalitions which can be formed independently. Forming smaller coalitions can be preferred by agents because, e.g., involving less agents they reduce the risk of defections, are easier to monitor, less expensive to form by means of agreements, less trust is required among all the agents, etc.

Example 1 Consider in Figure 1 (a) the sub-coalitions C_1 , involving only agents a_1 and a_2 and the relative arcs, and C_2 , involving only agents a_3 and a_4 and the relative arcs. As the agents involved in C_1 are not interested in the goals achieved in C_2 and vice versa, the two sub-coalitions can be formed independently.

Concerning Figure 1 (b), C denotes the whole or grand coalition, C_1 the coalition consisting of the nodes a_1 and a_2 and the two arcs labelled with the goals g_1 and g_2 . C_2 denotes the coalition consisting of the nodes a_2 and a_3 and the remaining arcs labelled with the goals g_3 and g_4 . The difference is that in Figure 1 (a) both a_1 and a_2 are indifferent between C_1 and the whole coalition, while in Figure 1 (b) a_2 is not indifferent between C_1 and C since in C it receives the goal g_4 not provided in C_1 .

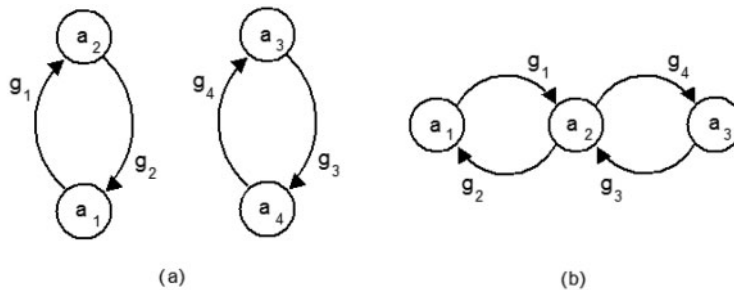


FIG. 1. Two coalitions that satisfy the do-ut-des property but that do not satisfy the i-dud property.

6 Algorithms for finding coalitions exploiting a new reciprocity condition

However, a_1 is indifferent between C_1 and C and a_3 is indifferent between C_2 and C as they receive and have to provide the same goals in both coalitions. When agent a_2 wants to propose to a_1 and a_3 to form coalitions, it has to decide whether to propose C to both agents or C_1 and C_2 separately. Agent a_2 chooses C_1 or C_2 since forming one of them does not affect the possibility for a_2 to form the other sub-coalition, and C_1 and C_2 are individually more reliable to succeed with respect to the whole C - as they involve individually less agents.

The i-dud property consists of three conditions. Given a coalition C , the first condition is condition (1) of the do-ut-des property. The second condition strengthens condition (2) of do-ut-des property by imposing that for each agent a_i in C , $out^*(a_i) = \mathcal{E}$. Thus, a coalition cannot be decomposed in two disconnected subgraphs as in Figure 1 (a). Finally, the third condition takes into account the case shown in Figure 1 (b): there does not exist an agent a_i and a bi-partition $\mathcal{O}_1, \mathcal{O}_2$ of $out(a_i)$ - where we assume that bi-partitions are composed by nonempty sets - such that $\mathcal{O}_1^* \cap \mathcal{O}_2^*$ is empty. The idea underlying this third condition is that if $\mathcal{O}_1^* \cap \mathcal{O}_2^* = \emptyset$, then no agent $a_j \neq a_i$ involved in \mathcal{O}_1^* would be interested in one of the goals achieved in \mathcal{O}_2^* and vice versa. So, a_i can deal separately with the formation of these two sub-coalitions.

Definition 4 A coalition $(\mathcal{V}, \mathcal{E})$ satisfies the i-dud property iff for all the agents $a_i \in \mathcal{V}$,

(Condition 1) there do not exist two commitments $(Q, g), (Q', g) \in C$ such that $Q \neq Q'$,

(Condition 2) $out^*(a_i) = \mathcal{E}$ and

(Condition 3) there does not exist a bi-partition $\mathcal{O}_1, \mathcal{O}_2$ of $out(a_i)$ such that $\mathcal{O}_1^* \cap \mathcal{O}_2^* = \emptyset$.

Considering again the coalitions in Figure 1 (a) and (b), as expected, they both do not satisfy the i-dud property.

Example 2 In Figure 1 (a), the second condition of Definition 4 is not satisfied as, for example, $out^*(a_1) = \{(a_1, \{a_2\}, g_1), (a_2, \{a_1\}, g_2)\} \subset \mathcal{E}$. In Figure 1 (b), the third condition of Definition 4 is not satisfied. Indeed, considering the bi-partition of a_2 composed by $\mathcal{O}_1 = \{(a_2, \{a_1\}, g_2)\}$ and $\mathcal{O}_2 = \{(a_2, \{a_3\}, g_4)\}$, we have that $\mathcal{O}_1^* = \{(a_2, \{a_1\}, g_2), (a_1, \{a_2\}, g_1)\}$ and $\mathcal{O}_2^* = \{(a_2, \{a_3\}, g_4), (a_3, \{a_2\}, g_3)\}$. Thus, $\mathcal{O}_1^* \cap \mathcal{O}_2^*$ is empty.

A labeled AND-graph can represent a potential coalition consisting of all, or a large part of, the opportunities of collaboration in a multiagent system. So we would like to establish not only whether the whole coalition is admissible or not, but also which of its sub-coalitions are admissible to be formed. Figure 2 (a) shows a quite complex coalition that does not satisfy the i-dud property.

Example 3 Indeed, given the bi-partition of $out(a_1)$ $\mathcal{O}_1 = \{(a_1, \{a_4, a_5\}, g_1)\}$ and $\mathcal{O}_2 = \{(a_1, \{a_3\}, g_6)\}$, it can be verified that $\mathcal{O}_1^* \cap \mathcal{O}_2^* = \emptyset$. Figure 2 (b), (c), (d) and (e) show all the sub-coalitions of Figure 2 (a) satisfying the i-dud property. These coalitions clearly satisfy also the do-ut-des property. However, since the do-ut-des property seeks only the reciprocity in a coalition, any composition of coalition (e) with one of the coalitions (b), (c) and (d) satisfies the do-ut-des property as well.

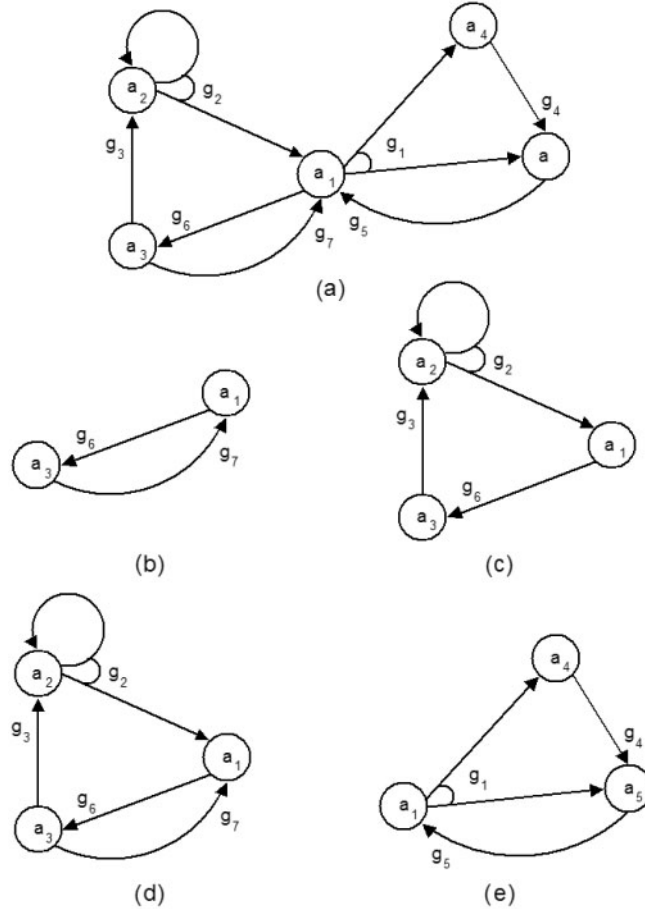


FIG. 2. A complex coalition (a) and the sub-coalitions satisfying the *i-dud* property (b), (c), (d) and (e).

2.4 Condition 2 and Condition 3 as graph-theoretical properties

Given a coalition C , Definition 4 could be used *tout court* in order to design an algorithm finding all the sub-coalitions satisfying the *i-dud* property (C included). However, Definition 4 requires to verify, for each agent a_i involved in C , a condition on the set of bi-partitions of $out(a_i)$. The number of bi-partitions of a set A is equal to the Stirling number $S(n, 2) = 2^{n-1} - 1$, where n is the cardinality of A . Therefore, the problem to verify if just C satisfies the *i-dud* property would increase in complexity exponentially with the cardinality of $out(a_i)$. For this reason we consider an alternative approach in order to make at least the verification of a single coalition tractable.

We reformulate Condition 2 as a property of strong connectivity of a directed graph.

Definition 5 We define a directed graph $\mathbf{G}[C] = \langle \mathbf{V}, \mathbf{E} \rangle$ relative to the coalition $C = \langle \mathcal{V}, \mathcal{E} \rangle$ as follows: the set of nodes \mathbf{V} is equal to the set \mathcal{V} of agents involved in C and $(a_i, a_j) \in \mathbf{E}$ if and only if there exist a goal g and a group of agents Q such that $(a_i, Q, g) \in \mathcal{E}$ and $a_j \in Q$.

8 Algorithms for finding coalitions exploiting a new reciprocity condition

Theorem 1 *The condition that $out^*(a_i) = \mathcal{E}$ is equivalent to say that $\mathbf{G}[C]$ is strongly connected, i.e. for each pair of nodes a_i and a_j there exists a path from a_i to a_j .*

PROOF. *It easy to see that $(a_j, Q, g) \in out^*(a_i)$ if and only if there exists a path in $\mathbf{G}[C]$ from a_i to a_j , and hence to all $a_h \in Q$. Then, the theorem follows from the fact that each agent is involved in the achievement of at least a goal. ■*

Given a generic directed graph \mathbf{G} , we call the strongly connected components of \mathbf{G} the maximal strongly connected sub-graphs containing at least one arc.

Now we consider how to reformulate Condition 3. Under the assumption that $\mathbf{G}[C]$ is strongly connected, Condition 3 is closely related to the notion of biconnectivity for undirected graphs.

Definition 6 *An undirected graph \mathbf{G} is biconnected if and only if it is connected and for all triples of distinct nodes a_i , a_j and a_k , there exists a path \mathbf{p} connecting a_j and a_k such that a_i is not in \mathbf{p} . In the contrary case, a_i is called an articulation node [36]. As for strongly connected components of a directed graph, the biconnected components of an undirected graph \mathbf{G} are the maximal biconnected subgraphs of \mathbf{G} containing at least one arc.*

It is easy to see that two distinct biconnected components share at most one node and, if so, this node is an articulation node.

Definition 7 *Starting from the directed graph $\mathbf{G}[C] = \langle \mathbf{V}, \mathbf{E} \rangle$, we define an undirected graph $\mathbf{G}[C] = \langle \mathbf{V}, \mathbf{E} \rangle$ as follows: $\mathbf{V} = \mathbf{V}$ and, for $a_i \neq a_j$, $\{a_i, a_j\} \in \mathbf{E}$ if and only if (a_i, a_j) or (a_j, a_i) are in \mathbf{E} .*

The following theorem shows that the fact that Condition 3 is not satisfied is indicated by the presence of an articulation node a_i .

Theorem 2 *Let $C = \langle \mathcal{V}, \mathcal{E} \rangle$ be coalition such that $\mathbf{G}[C]$ is strongly connected, if there exists an agent $a_i \in \mathcal{V}$ and a bipartition $\mathcal{O}_1, \mathcal{O}_2$ of $out(a_i)$ such that $\mathcal{O}_1^* \cap \mathcal{O}_2^* = \emptyset$, then a_i is an articulation node of $\mathbf{G}[C]$.*

PROOF. *Assume that there exists an agent $a_i \in \mathcal{V}$ and a bipartition $\mathcal{O}_1, \mathcal{O}_2$ of $out(a_i)$ such that $\mathcal{O}_1^* \cap \mathcal{O}_2^* = \emptyset$ and, per absurdum, a_i is not an articulation node.*

Strong connectivity of $\mathbf{G}[C]$ assures that (assumption1) there exist two agents, say a_1 and a_2 , such that a_i, a_1 and a_2 are distinct and a_1 is involved in \mathcal{O}_1^ and a_2 is involved in \mathcal{O}_2^* , and (assumption2) $\mathcal{O}_1^* \cup \mathcal{O}_2^* = \mathcal{E}$. Since a_i is not an articulation node, there exists an undirected path \mathbf{p} connecting a_1 and a_2 such that a_i is not a node of \mathbf{p} . For (assumption2) and condition (1) of Definition 1, each node in the path is an agent in \mathcal{O}_1^* or \mathcal{O}_2^* . Thus, starting from a_1 it is possible to walk through \mathbf{p} until an agent a_h is in \mathcal{O}_1^* and the successor a_k is in \mathcal{O}_2^* . The presence of an undirected arc connecting a_h to a_k means that one of them is involved in set out of the other one. Without loss of generality we assume that there exists a set of agents Q and a goal g such that $a_h \in Q$ and $(a_k, Q, g) \in \mathcal{E}$. This means that $out(a_h)$ is contained in both \mathcal{O}_1^* and \mathcal{O}_2^* . For strong connectivity of $\mathbf{G}[C]$, we also have that $out^*(a_h) = \mathcal{E}$ and hence $out(a_h)$ is not empty, then $\mathcal{O}_1^* \cap \mathcal{O}_2^* \neq \emptyset$ against the hypothesis. ■*

So, if $\mathbf{G}[C]$ is biconnected (i.e. it does not have articulation points), then it satisfies Condition 3. However, the inverse implication of Theorem 2 does not hold and Condition 3 can be satisfied even if $\mathbf{G}[C]$ has an articulation point a_i . This is due to the fact that the undirected graph $\mathbf{G}[C]$ breaks an AND-arc in several undirected arcs, so the biconnected

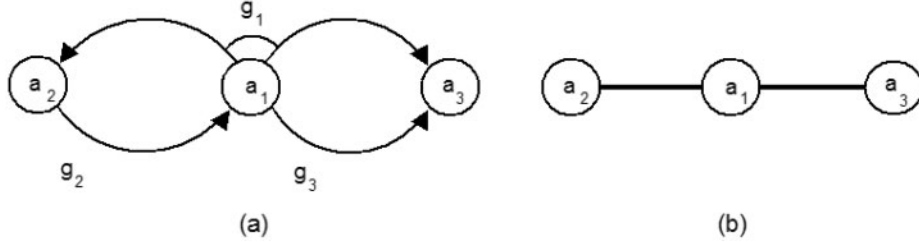


FIG. 3. A labeled AND-graph and the corresponding undirected graph.

components sharing a_i may not correspond to any bi-partition of $out(a_i)$. Figure 3 considers this fact.

Example 4 Figure 3 (b) represents the undirected graph $\mathbf{G}[C]$ of the coalition C in Figure 3 (a). There exist two biconnected components of $\mathbf{G}[C]$, one for each arc, sharing a_1 as articulation node. However, both arcs $\{a_1, a_2\}$ and $\{a_1, a_3\}$ correspond to the AND-arc $(a_1, \{a_2, a_3\}, g_1)$, thus they have to be considered as a single component because $\{(a_1, \{a_2, a_3\}, g_1)\}^*$ contains both of them. Being $out(a_1)$ equal to $\{(a_1, \{a_2, a_3\}, g_1)\}$, there does not exist a bi-partition $\mathcal{O}_1, \mathcal{O}_2$ of $out(a_1)$ such that $\mathcal{O}_1^* \cap \mathcal{O}_2^* = \emptyset$. Thus, C satisfies Condition 3.

This property holds in general, when some biconnected components of $\mathbf{G}[C]$ contain some arcs corresponding to the same AND-arc of C , they are considered as a single component. If this grouping process ends with a single component consisting of the whole undirected graph $\mathbf{G}[C]$, then no bi-partitions of $out(a_i)$ falsify Condition 3.

3 The algorithm for finding coalitions

3.1 Finding coalitions

In this section we design a procedure `FIND-I-DUD` (see Algorithm 1) which finds all the sub-coalitions of a coalition C satisfying the i-dud property (C included). We use the reformulation of Condition 2 and Condition 3 in Definition 4 in terms of strong connectivity of directed graphs and biconnectivity of undirected graphs. By doing so, we also decompose our problem as much as possible in well known problems in graph theory.

The variable `I_DUD` in line 1 stores the set of sub-coalitions of C which satisfy the i-dud property. In line 2 `NO-DUPL-COMMITMENTS` checks in \mathcal{E} the presence of commitments with the same goal but assigned to different sets of agents (Condition 1) and it returns the set NDC of all combinations C' obtained from C by deleting all the duplicated commitments except one. This way, the sub-coalitions in NDC are the maximal sub-coalitions which satisfy Condition 1. Since all their sub-coalitions satisfy this condition as well, we do not need to check recursively this condition on them, i.e. `NO-DUPL-COMMITMENTS` needs to be run only once.

Algorithm 1: FIND-I-DUD

Data: $C = \langle \mathcal{V}, \mathcal{E} \rangle$.
Result: I_DUD , the set of sub-coalitions of C that satisfy the i-dud property

```

1  $I\_DUD \leftarrow \emptyset$ ;
2  $NDC = \text{NO-DUPL-COMMITMENTS}(C)$ ;
3 forall  $C' \in NDC$  do
4    $I\_DUD \leftarrow I\_DUD \cup \text{FIND-2-3}(C')$ ;

```

For each coalition in NDC the procedure FIND-2-3 is run (lines 3-4). FIND-2-3 (Algorithm 2) takes as input a coalition C - which satisfies Condition 1 - and it returns the set of sub-coalitions of C , C included, that satisfy Condition 2 and Condition 3. As we already have checked Condition 1, we can add the results of FIND-2-3 to the set I_DUD .

The variable S stores the subsets of C that satisfy Condition 2 and Condition 3, in line 1 this variable is initialized to the empty set.

In line 2 SC-COMPONENTS calculates the strongly connected components SCC of $\mathbf{G}[C]$ - algorithms for this procedure are well known [15, 36]. Three cases are distinguished.

Case 1: $\mathbf{G}[C]$ has no strongly connected components, that is SCC is empty. Since strong connectivity is a necessary condition for the satisfaction of Condition 2, no sub-coalitions of C satisfy the i-dud property. Therefore, S is empty.

Case 2: $\mathbf{G}[C]$ is not strongly connected, but there exist some strongly connected components ($|SCC| > 1$). In this case only the sub-coalitions of C such that the relative directed graphs are subgraphs of a strongly connected component can satisfy Condition 2. Therefore, in lines 7-10, for each strongly connected component, the maximal labeled AND-graph $\langle \mathcal{V}', \mathcal{E}' \rangle$, *included* in the component, is constructed. The function FIND-2-3 is recursively called on $\langle \mathcal{V}', \mathcal{E}' \rangle$ and its output is added to S .

Case 3: $\mathbf{G}[C]$ is strongly connected ($|SCC| = 1$), therefore, Condition 2 is satisfied. It remains to check Condition 3 and we use the characterization by means of the biconnected components of $\mathbf{G}[C]$ (see Section 2).

In line 12 the set of biconnected components BC and the set of articulation points A_NODES are calculated. In lines 13-15 FIND-2-3 checks, for each articulation node a_i , if there exists an AND-arc (a_i, Q, g) such that the other agents in Q are involved in two, or more, biconnected components, then these biconnected components replaced with their union (see Figure 3). In the case we end with a single component, $|BC| = 1$, C satisfies also Condition 3 and it is added to S . Then, in lines 18-20, the sub-coalitions C' obtained removing a single commitment (Q, g) from C are constructed and FIND-2-3 is recursively called on them.

If $|BC| > 1$, then C does not satisfy Condition 3. Also all the subsets C' of C such that $\mathbf{G}[C']$ is not included in a component of BC cannot satisfy Condition 3, therefore for each component $\langle \mathbf{V}, \mathbf{E} \rangle$ in BC , the maximal subgraph of C *included* in $\langle \mathbf{V}, \mathbf{E} \rangle$ is selected. FIND-2-3 is recursively called on C' and the output is added to S (lines 22-25). Finally, S is returned, line 27.

Figure 4 shows the execution of the algorithm FIND-I-DUD on the tagged AND-graph relative to Figure 2 (a). Each box in Figure 4 represents a recursive call of the algorithm FIND-I-DUD. The flow of the recursive calls is represented by the arrows connecting the boxes.

Algorithm 2: FIND-2-3

Data: $C = \langle \mathcal{V}, \mathcal{E} \rangle$.
Result: S , the set of sub-coalitions of C that satisfy Condition 2 and Condition 3.

```

1  $S \leftarrow \emptyset$ ;
2  $SCC \leftarrow \text{SC-COMPONENTS}(\mathbf{G}[C])$ ;
3 switch do
4   case  $\mathbf{G}[C]$  has no strongly connected components
5      $S \leftarrow \emptyset$ ;
6   case  $\mathbf{G}[C]$  is not strongly connected, but it has some strongly connected components
7     forall  $\langle \mathbf{V}, \mathbf{E} \rangle \in SCC$  do
8        $\mathcal{V}' \leftarrow \mathbf{V}$ ;
9        $\mathcal{E}' \leftarrow \{(a_i, Q, g) \in \mathcal{E} \mid a_i \in \mathbf{V} \wedge Q \subseteq \mathbf{V}\}$ ;
10       $S \leftarrow S \cup \text{FIND-2-3}(\langle \mathcal{V}', \mathcal{E}' \rangle)$ ;
11   case  $\mathbf{G}[C]$  is strongly connected
12      $(BC, A\_NODES) \leftarrow \text{BC-COMPONENTS}(\mathbf{G}[C])$ ;
13     forall  $(a_i, Q, g) \in \mathcal{E}$  s.t.  $a_i \in A\_NODES$  do
14        $BC' \leftarrow \{\langle \mathbf{V}', \mathbf{E}' \rangle \in BC \mid \{a_i, a_j\} \in \mathbf{E}' \text{ with } a_j \in Q\}$ ;
15        $BC \leftarrow [BC \setminus BC'] \cup \{\bigcup BC'\}$ ;
16     if  $|BC| = 1$  then
17        $S \leftarrow \{C\}$ ;
18     forall  $(Q, g) \in C$  do
19        $C' \leftarrow C \setminus \{(Q, g)\}$ ;
20        $S \leftarrow S \cup \text{FIND-2-3}(C')$ ;
21     else
22       forall  $\langle \mathbf{V}, \mathbf{E} \rangle \in BC$  do
23          $\mathcal{E}' \leftarrow \{(a_i, Q, g) \in \mathcal{E} \mid a_i \in \mathbf{V} \wedge Q \subseteq \mathbf{V}\}$ ;
24          $\mathcal{V}' \leftarrow \mathbf{V}$ ;
25          $S \leftarrow S \cup \text{FIND-2-3}(\langle \mathcal{V}', \mathcal{E}' \rangle)$ ;
26
27 return  $S$ ;
```

Each box consists of one, two or three parts, in the first part, on the left side of the box, the input C of the recursive call is represented. For space reason the tags corresponding to the goals and the agents are omitted. The strongly connected components, if there exist any, are represented in the middle part. If the directed graph $\mathbf{G}[C]$ is strongly-connected, the third part of a box represents the biconnected components of the undirected graph $\mathbf{G}[C]$. If only one component results, then C satisfies the indecomposable do-ut-des property. Boxes outlined with a bold line corresponds to the coalitions that satisfy the indecomposable do-ut-des property. The relative inputs are the same represented in Figure 2 (b), (c), (d) and (e).

As you can see the procedure FIND-I-DUD is called twice to the same input, corresponding to the box outlined by a bold dotted line. This leads to an overhead which slackens

12 Algorithms for finding coalitions exploiting a new reciprocity condition

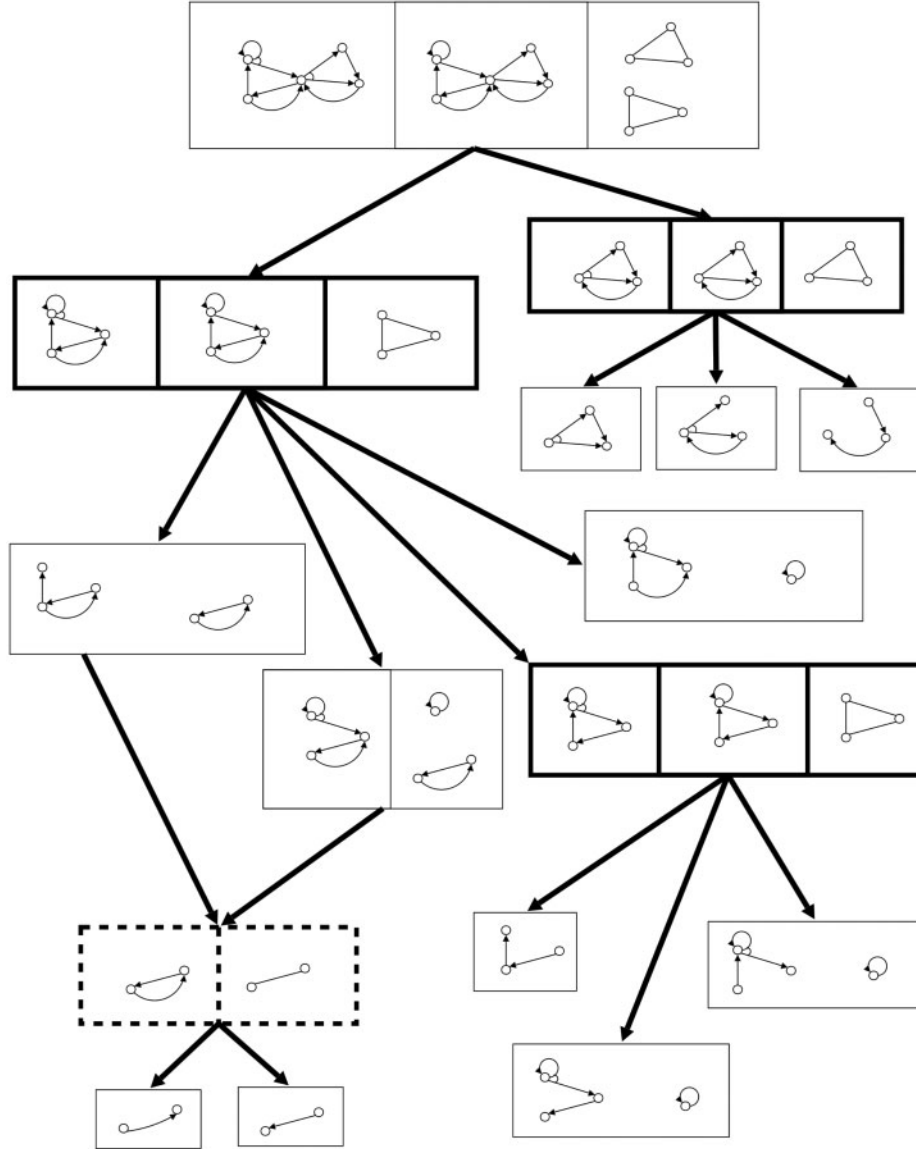


FIG. 4. Execution of the algorithm FIND-I-DUD on the tagged AND-graph in Figure 2 (a).

the algorithm with useless computation. A possible solution consists of standard memoizing technics [15].

3.2 Checking i-dud

Algorithms FIND-I-DUD and FIND-2-3 can be easily modified to simply check if a given coalition satisfies the i-dud property (Algorithm 3).

Algorithm 3: CHECK-I-DUD

Data: $C = \langle \mathcal{V}, \mathcal{E} \rangle$.
Result: A boolean value IS_I_DUD which is true iff C satisfies the idud property.

```

1  $I\_DUD \leftarrow \emptyset$ ;
2  $B = \text{CHECK-NO-DUPL-COMMITMENTS}(C)$ ;
3 if  $B$  then
4    $IS\_I\_DUD = \text{false}$ ;
5 else
6    $IS\_I\_DUD \leftarrow \text{CHECK-2-3}(C)$ ;
```

Algorithm 4: CHECK-2-3

Data: $C = \langle \mathcal{V}, \mathcal{E} \rangle$.
Result: IS_OS a boolean value which is true iff C satisfies Condition 2 and Condition 3.

```

1  $SCC \leftarrow \text{SC-COMPONENTS}(\mathbf{G}[C])$ ;
2 if  $\mathbf{G}[C]$  is not strongly connected then
3    $IS\_OS \leftarrow \text{false}$ ;
4 else
5    $(BC, A\_NODES) \leftarrow \text{BC-COMPONENTS}(\mathbf{G}[C])$ ;
6   forall  $(a_i, Q, g) \in \mathcal{E}$  s.t.  $a_i \in A\_NODES$  do
7      $BC' \leftarrow \{\langle V', E' \rangle \in BC \mid \{a_i, a_j\} \in E' \text{ with } a_j \in Q\}$ ;
8      $BC \leftarrow [BC \setminus BC'] \cup \{\bigcup BC'\}$ ;
9   if  $|BC| = 1$  then
10     $IS\_OS \leftarrow \text{true}$ ;
11  else
12     $IS\_OS \leftarrow \text{false}$ ;
```

CHECK-NO-DUPL-COMMITMENTS checks whether condition 1 is satisfied. Then, the FOR statement in FIND-I-DUD is replaced with an IF-THEN-ELSE statement which returns false if C does not satisfy cond1, it calls CHECK-2-3 on C , otherwise. CHECK-2-3 (Algorithm 4) returns false if $\mathbf{G}[C]$ is not strongly connected. In contrary case, it checks, as in FIND-2-3, whether condition 3 is satisfied and returns accordingly.

4 Complexity of the algorithms

In this section we discuss the complexity of FIND-I-DUD.

4.1 Checking for i-dud

First of all, we show that the problem of checking if a coalition satisfies the i-dud property is tractable.

14 Algorithms for finding coalitions exploiting a new reciprocity condition

Theorem 3 We denote with m the number of agents involved in C and with l the number of arcs in $\mathbf{G}[C]$. $O(l \cdot m)$ is an upper bound for the problem to verify if a coalition satisfies the *i-dud* property.

PROOF. The procedure *SC-COMPONENTS* takes a time proportional to l [15]. In the case $\mathbf{G}[C]$ is not strongly connected, C does not satisfies the *i-dud* property and *CHECK-2-3* returns false. In the contrary case, the procedure *BC-COMPONENTS* is called on the undirected graph $\mathbf{G}[C]$.

Also *BC-COMPONENTS* can be executed in a time that is proportional to $|\mathbf{E}|$ and, since $|\mathbf{E}| \leq l$, so far Algorithm 2 has a complexity that is proportional to l . We have to consider now the complexity of the cycle corresponding to the lines 6-8. The number of iterations of the cycle 6-8 is less than l . The instruction in line 7 has as upper bound m , assuming that, during the execution of *BC-COMPONENTS*, a data structure is stored associating each arc with the biconnected component in which it is included. Since the sets of arcs of two distinct biconnected components are disjoint, also the instruction in line 8 can be performed in time proportional to the set of distinct biconnected components found in line 5, which has an upper bound in m . Therefore, the cycle 6-8 has an upper bound in $O(l \cdot m)$. Since $O(l \cdot m)$ is an upper bound also to check Condition 1, it is an upper bound for the problem to verify if a coalition satisfies the *i-dud* property. ■

4.2 Finding coalitions

Theorem 4 Finding all the sub-coalitions that satisfy the *i-dud* property is in the worst case exponential space in the size of the input.

PROOF. With respect to the problem to find all the sub-coalitions of C that satisfy the *i-dud* property, consider that C satisfies Condition 1 and it contains only *AND*-arcs as $(a_i, \{a_j\}, g)$. In this case we can represent C as a directed graph $\mathbf{G}[C]$, where each arc (u, v) univocally corresponds to a goal. A priori, the set of subgraphs to check is equal to 2^l . Then, assume that $\mathbf{G}[C]$ is a clique, that is for each pair of nodes there exists an arc connecting them, since any simple cycle is a *i-dud* coalition and the number of cycles in a clique is exponential in its dimension, then we need exponential space in the size of the input. ■

However, when the connectivity of the graphs decreases the actual complexity of the problem is drastically reduced. Thus, if C does not satisfy the *i-dud* property, then, either $\mathbf{G}[C]$ is not strongly connected or $\mathbf{G}[C]$ has more than one component as calculated in the lines 12-15. In both cases *FIND-2-3* is called directly on the subgraphs calculated respectively in lines 8-9 and 23-24. So, if there are k of these subgraphs, each of them with l_i arcs, we have that the number of the graph which remain to be verified is $2^{l_1} + \dots + 2^{l_k}$ instead of (approximately from below) $2^{l_1 + \dots + l_k} - 1$. In the worst case $\mathbf{G}[C]$ is not strongly connected and it has one strongly component with $l-1$ arcs. In this case 2^{l-1} graphs remain to be verified instead of $2^l - 1$.

We note that this fact occurs not only once, but every time a sub-coalition of C does not satisfy the *i-dud* property. Moreover, if C is a proper *AND*-graph this phenomenon can be amplified by the fact that when an *AND*-arc is removed in line 8, it may disconnect a strongly connected component of $\mathbf{G}[C]$.

Returning to the coalition C in Figure 2 (a), the number of *AND*-arcs is 7, so a priori $2^7 = 128$ sub-coalitions should be checked by the algorithm *FIND-I-DUD*. However, C does

not satisfy Condition 3 and, after BC results to be greater than 1 in line 16 of FIND-2-3, FIND-2-3 is called on the sub-coalitions in Figure 2 (d) and (e). The first sub-coalition has 4 AND-arcs and the second one has just 3 arcs. So after a single call of FIND-2-3, it remains $2^4 + 2^3 = 24$ sub-coalitions instead of 127. Figure 4 shows that the total number of sub-coalitions checked is equal to 16, i.e. only the 12,5% of the number of all sub-coalitions of C .

5 A chips and boxes example

In the previous sections we have considered examples directly expressed in the high level of abstraction of labeled AND-graphs, here we consider an example described at a more refined level of abstraction. The example consists of simple version of the game described in Grosz et al. [21] and describes a general problem of exchange of resources. This not only tests again how the do-ut-des and the indecomposable do-ut-des properties restrict the set of admissible coalitions, but it also gives us the possibility to show and discuss how labeled AND-graphs describing a possible coalition have to be instantiated starting from a concrete scenario.

There are four agents and each of them has two chips and two boxes, both the chips and the boxes can be colored as follows: red r , green g or blue b . An agent has only one goal, to fill each of his boxes with a chip of the same color of the box, to this end agents can exchange chips with each other. The initial configuration is:

agents	boxes	chips
a_1	$r r$	$r g$
a_2	$r g$	$b b$
a_3	$b g$	$r b$
a_4	$b b$	$r g$

As the agents are not self-sufficient to see to their own goals, they need to exchange chips with each other. The number of possible exchanges is given by the number of partitions of the 8 chips into 4 sets of 2 elements times the permutations of the 4 sets, that is $\binom{8}{2} \cdot 4! = 672$ combinations. Of course a big amount of those combinations does not provide the achievement of any goal, as the one in which all the agents maintain their chips in the initial configuration.

Now we have to translate the given problem in terms of AND-graphs as in Definition 1, so first we have to define the possible commitments that can be sustained. Of course this is a crucial step as an incorrect formalization of which group of agents can reliably provide which goals may lead to counterintuitive results. A first requirement for a group of agents Q to be committed to a certain goal g is that Q has the *power* to see to g . Of course what it means that Q can achieve g may depend on the specific domain, for example, you may consider that a group of agents Q can achieve g when Q has a probability of 0,8 to obtain the satisfaction of g , which seems to be a reasonable guarantee for that goals, or you could use the notion of effectiveness developed in Coalition Logic [27], that is there exists a joint strategy for Q such that it can see to g no matter what the other agents do.

If the commitments reflect the mere capabilities of groups of agents, then some undesired properties may induce to a formalization of the commitments which vanishes the possibility of the dud and i-dud criteria to prune unfeasible coalitions. For example in Coalition Logic the notion of power satisfies the property of coalitional monotonicity, that is if Q has the

16 Algorithms for finding coalitions exploiting a new reciprocity condition

power to achieve g , then every superset Q' of Q as the same power in the plan that Q has to execute in order to achieve g , even if the added agents do not play any role. However, if we consider the commitment (Q', g) , instead of the commitment (Q, g) , then, since the do-ut-des property formalize a notion of reciprocity, all the agents in Q' have to obtain some goals in exchange, but in this case some agents in Q' obtain the achievement of one, or more, of their own goals without doing anything. One possibility to avoid this problem is to consider that a group of agents has the power to achieve a set of goals only if all its agents are necessary or at least useful [7].

Coming back to our problem, the set of agents is $A = \{a_1, a_2, a_3, a_4\}$, the set of the goals Gl is $\{rr, rg, bg, bb\}$, according to the boxes owned by the agents. A group of agents has the power to achieve a goal if they have the corresponding chips. So, for example, the group of agents a_2 and a_3 has the power to achieve the goal bb of the agent a_4 by giving their blue chips to it. We assume that the notion of power requires that if a group of agents has the power to achieve a goal, then all its members are necessary for the achievement of that goal, in this contest an agent in a group either is necessary for the achievement of a goal or it is not.

We show the set of possible commitments by grouping together the group of agents that has the power to achieve the same goal. In the table each row contains in the first column all the group of agents that can achieve the goal indicated in the second column.

Q	g
$\{a_1, a_3\}, \{a_1, a_4\}, \{a_3, a_4\}$	rr
$\{a_1, a_3\}, \{a_1, a_4\}, \{a_1\}, \{a_4\}, \{a_1, a_4\}$	rg
$\{a_1, a_2\}, \{a_1, a_3\}, \{a_2, a_4\}, \{a_3, a_4\}$	bg
$\{a_2\}, \{a_2, a_3\}$	bb

Given a set of commitments, as the goals are known, it univocally corresponds a coalition, therefore in the following we consider as coalitions the sets of commitments. Clearly, when we consider a coalition we want to check the admissibility we have to assure that the commitments in it are compatible, that is there does not exist an agent which is required to provide the same chip in two commitments (Q_1, g_1) and (Q_2, g_2) achieving distinct goals. For example, the set of commitments $(\{a_3, a_4\}, bg)$ and $(\{a_3, a_4\}, rr)$ is compatible because the agents use in the achievement of bg and rr different chips. On the contrary the commitments $(\{a_1\}, rg)$ and $(\{a_1, a_3\}, rr)$ are incompatible as the total amount of red chips at their disposal of a_1 and a_3 is equal to 2 whereas the amount of red chips required to achieve at the same time both the goals rg and rr is equal to 3, this means that the agent a_1 should use at the same time its unique red chip for the achievement of both the goals. We notice that all the commitments involving disjoint groups of agents are compatible.

As a coalition is do-ut-des implies that there do not exist two groups in it achieving the same goal, we can restrict our search to the coalitions which satisfy this condition. We find these coalitions by solving a constraint satisfaction problem as follows: we select a group of agents that can achieve the goal rr , for example $\{a_1, a_3\}$, then we see which groups of agents has the power to achieve the goal rg compatibly with the fact that $\{a_1, a_3\}$ achieves rr , and so on up to the goal bb . The result of this constraint satisfaction problem is shown in Figure 5, every path starting from the top up to the bottom of the graph constitute a

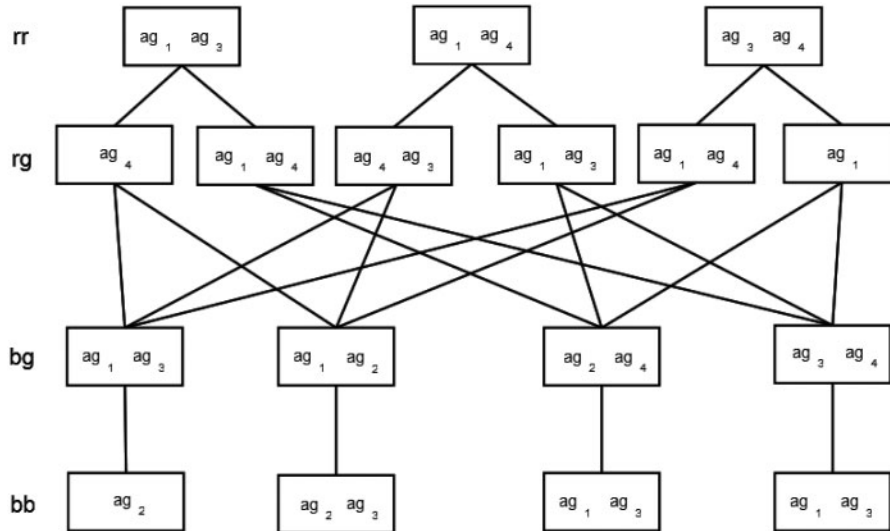


FIG. 5. *The possible coalitions in the chips and boxes example.*

maximal coalition satisfying the condition of no repeated goals. Then for any of them, we can apply the algorithm shown in section 3 to select the i-dud coalitions.

The set of the coalitions to be checked is the sets of all the groups lying in the same path and the corresponding goals - that is 157 coalitions.

All the maximal coalitions are do-ut-des because all the goals of the agents are achieved and also all the chips at the disposal of the agents are used, then all the agents obtain and take part in the satisfaction of a goal. However these are not the only ones, for example, the coalition C_1 composed by the commitments $(\{a_1, a_3\}, rr)$ and $(\{a_1, a_3\}, bg)$ is also do-ut-des. Analogously, another do-ut-des coalition C_2 is composed by the commitments $(\{a_2\}, bb)$ and $(\{a_4\}, rg)$. Applying the definition it can be shown that these are the only coalitions satisfying the do-ut-des property, that is the number of do-ut-des coalitions is equal to 14, i.e. about the 9% of the set of all the ones satisfying the no repeated goals condition. For the indecomposable do-ut-des property the do-ut-des coalition given by the union of C_1 and C_2 does not satisfy the indecomposable do-ut-des property since a_1 and a_3 which are involved in C_1 are not interested in C_2 and vice versa a_2 and a_4 are not interested in C_1 . All the other do-ut-des coalitions satisfy the indecomposable do-ut-des property too.

6 Related work

Several efforts concern the research topic of coalition formation, focussing on different aspects of this problem. Cooperative Game Theory has been the first research area focussed on coalition formation and, nowadays, it remains the historical ground to reason about coalitions [3, 28, 33, 38].

The main aim of Cooperative Game Theory regards the study of admissibility criteria in coalition formation. Several solution criteria have been proposed, as for example the notion

18 *Algorithms for finding coalitions exploiting a new reciprocity condition*

of core, stable sets, nucleus. The notion of core is based, as the Nash equilibrium in Non-cooperative Game Theory, on a dominance relation over the set of possible consequences attainable by the grand coalition. A coalition is dominated if there exists a group of agents able to achieve a consequence which is strictly preferred by all its members.

In this work and in the related ones, we are interested in goal-directed agents, and not in utility-maximizer agents, so our notions of admissibility distinguish in several aspects from the admissibility criteria developed in Cooperative Game Theory. In this paper we consider the do-ut-des property, which is a qualitative criterion of admissibility, with respect to the quantitative approach developed in Cooperative Game Theory. We do not directly compare goals, but simply balance sets of goals by means of inclusion relation. In particular, the do-ut-des property can be used as a qualitative method to restrict the search space of a quantitative criterion of admissibility, we called the q-do-ut-des property in [31], which is even more restrictive than the well-known notion of core. The admissibility criteria developed there require to compare a coalition with all the other potential coalitions in order to verify if it is admissible.

Both the definitions of the do-ut-des property and of the indecomposable do-ut-des property require to compare a coalition, intended as a set of commitments, only with its subsets. This way, a potential coalition has all the required information to check if it satisfies our admissibility criteria as only the comparison with included coalitions is required. For this reason we say that our criteria rather establish if a coalition can be formed *in re* than, as in Game Theory, compare all the possible coalitions with each other in order to seek the most profitable ones. Our criteria are weaker in the sense that, in our representation, a coalition has all the necessary information to verify if it is admissible. So, our criteria can be applied also when we have only a partial view of the multiagent system, and hence of the potential coalitions. Anyhow, despite the differences with Game Theory deriving from fact that Game Theory uses quantitative admissibility criteria on utility maximizer agents and we use qualitative criteria on goal-directed agents, both the approaches are an attempt to formalize the *homo economicus*. We suppose that agents use a quantitative cost-benefit analysis to compare potential coalitions and in [31] we show which class of cost-benefit analysis is compatible with our qualitative reasoning. This provides us the possibility to relate our criteria of admissibility with the quantitative admissibility criteria developed in Cooperative Game Theory.

In the field of Artificial Intelligence the problem of which coalitions should be formed in a multiagent system historically derives from the more general problem of how agents should coordinate with each other. The branch of Artificial Intelligence that deals with this general problem is called Distributed Artificial Intelligence (DAI). One of the major distinctions in DAI is between Distributed Problem Solving (DPS) and Multiagent Systems (MAS) [19, 34, 41]. In DPS, there is some notion of global utility [29] or a set of tasks to be fulfilled [25, 34] and the problem is to find the way the agents have to collaborate in order to maximize the global utility or the number of fulfilled tasks. When an optimal policy is found, it can be imposed to the agents of the system, so a crucial assumption is that agents are not completely autonomous. In MAS, instead, agents are autonomous and their proactive behavior is governed by a private utility function or by private goals (self-interested agents). Therefore, there is not a distinguished agent (the manager) that can impose a particular behavior to the other agents.

DPS is well suited to address, for example, the distributed vehicle routing problem of a single company. A company is responsible of certain deliveries and has a certain number of

vehicles to take care of them. The problem is to find which routes the vehicles have to do in order to accomplish all the deliveries and minimize the global cost of transportation. The MAS approach, instead, can be applied to solve how different delivery companies should outsource to each other their delivery tasks [30].

An interesting bridge between DPS and MAS is given by the mechanism design problem [41], i.e. the problem to find a protocol regulating the negotiation process of autonomous self-interested agents such that the economic behavior of the single agents also leads to an improvement of the social welfare.

A coalition formation process occurs in Distributed Problem Solving when the whole population of agents can be partitioned in groups of agents (coalitions) such that agents within each coalition coordinate their activities, but agents do not coordinate between coalitions.

Sandholm et al. [29] study the problem of finding which partition of the population of agents maximizes a global utility function when super-additivity (agents always do better joining all together) is not guaranteed. This partition is called a coalition structure.

Shehory and Kraus have studied in [34] task allocation problems via coalition formation. There is a set of n agents, $A = \{a_1, \dots, a_n\}$ and a set of m independent tasks $T = \{t_1, \dots, t_m\}$ to be fulfilled, eventually according to a precedence order. Each agent a_i has a vector of real non-negative capabilities $\langle c_1^i, \dots, c_r^i \rangle$. Each capability is a property of an agent that quantifies its ability to perform a specific action or the amount of resources it has at its disposal. For the satisfaction of each task t_l , a vector of capabilities $B_l = \langle b_1^l, \dots, b_r^l \rangle$ is necessary. In order to fulfill the tasks agents can join their capabilities forming coalitions.

The capability vector of a coalition Q , $B_Q = \langle c_1^Q, \dots, c_r^Q \rangle$, is the sum of the capability vectors that the members contribute to Q . A coalition can fulfill a task t_j only if for all $1 \leq k \leq r$, $b_k^j \leq c_k^Q$. For each coalition Q the fulfillment of a specific task provides a certain utility, that may depend on both the task and the coalition. The problem is to find a set of coalitions Q and a task assignment for them such that the sum u_Q of the gained utilities is maximized.

This problem is in general non tractable, so the authors provide a distributed anytime algorithm which finds an approximated solution with ratio bound from the optimal one.

Intractability of reasoning about coalition shows the necessity to find algorithms trying to decrease the complexity of the problem, as we do in this article.

The issues addressed in [29] and [34] have been object of further research. For example, in [16] Dang and Jennings present an anytime algorithm and show experimentally that it improves the performances with respect to the one presented in [29]. Kraus and Plotkin [25] consider a problem of task allocation in the case the set of tasks to be fulfilled are not given in advance, but they arrive according to independent probabilities.

Wooldridge and Jennings [40] study coalition formation as a process composed by four phases: recognition, team formation, plan negotiation and team action.

The recognition problem regards the possibility for an agent to reason about the capabilities of the other agents and the potential for a cooperation in the case it is not self-sufficient in the achievement of a goal. They use Dynamic Logic [22] in order to express sequences of actions, their effects and a primitive, with the meaning that a group of agents is able to execute an action α . Once recognized in a group of agents Q the potential for cooperation for a goal g , an agent attempts, in the team formation phase, to induce in all the members of Q the commitment to achieve g . After a team is formed the agents of that team negotiate about which plan has to be executed for it. Finally, in the team action phase the members of Q create a joint intention for the satisfaction of g by means of a certain plan α .

20 *Algorithms for finding coalitions exploiting a new reciprocity condition*

In these works the goals that an agent can propose to a group of agents Q are only those that are shared with all the members of Q . Otherwise, it should rely on a altruistic predisposition of the other agents. Therefore, cooperative problem solving is somewhat on the boundary of the Distributed Problem Solving and Multiagent Systems approaches, as agents are considered autonomous, but the only social interaction considered is the cooperation for the achievement of a shared goal, which is somewhat similar to the problem to maximize a global utility as in Distributed Problem Solving. Nevertheless, profitable behaviors arises also in the case the agents exchange services in order to satisfy their own goals, or directly exchange goal commitments.

Exploiting the formalism of dependence networks we go beyond this limitation, relax the precondition of the shared goal, and provide a structure of the coalition, guaranteeing that each agent is motivated to participate in the coalition by the satisfaction of some of his own goals.

Dunne and Wooldridge [18, 39] show as several problems of a cooperative game with goal-directed agents, as to verify if a set of goals is in the core of a coalition, are non computationally tractable.

Their representation does not provide any information about the commitments inside a coalition and their possible admissibility, therefore it does not enable to use the do-ut-des property.

Differently from all the above approaches, we base the notion of coalition on social exchanges among agents. The notion of social exchanges has been introduced in the Multi-agent Systems community by Castelfranchi [11, 12] and some formalizations of this notion have been presented in several works [4, 17, 35].

Dependence networks directly stem from this line of research.

Under the hypothesis that agent are self-interested, an agent accepts to help another agent only if it receives some advantages. Thus, a social interaction can occur only in the case of a bilateral dependence.

Dependence networks have been examined at different levels of detail. For example, Sichman and Demazeau [35] introduce not only dependencies on goals like we do in this article, but they also introduce dependence on action. Conte and Sichman [13] consider subjective views on dependence networks and proposes an algorithm to negotiate the construction of shared dependence networks, while here we assume all agents know the real dependencies. In Conte and Sichman [13] a classification of dependency types is proposed, distinguishing dependence on the base of reciprocity and of sharing of beliefs about them.

The social exchanges considered in Sichman and Demazeau [35] have been derived from two kinds of dependencies, mutual and reciprocal dependence. Both mutual and reciprocal dependence involve only two agents, therefore they lead only to dyadic agreements. In some cases, more complex chains of dependencies can be used to help each other. For example, a mother can be dependent on her tall son to replace a light bulb in the kitchen, the son is dependent, in his turn, on his sister to help him in a problem of trigonometry and the sister is dependent on the mother to buy a new skirt. In this example there is no mutual or reciprocal dependence among only two persons, nevertheless a chain of dependencies involving at the same time the mother and her children leads to a worthwhile exchange. This kind of exchanges is called generalized social exchanges.

They also introduce different kinds of dependencies like OR and AND dependencies, like the one used in this article. AMONG-dependence formalizes the notion of reciprocity that we

call the do-ut-des property. Moreover, they consider some cases of composition of coalitions, but they do not define admissibility criteria like we do.

Both the two admissibility criteria of this paper have been defined as a no-dominance condition on some qualitative preferences of the agents. In particular the do-ut-des property is an extension of the notion of reciprocity defined in [14] which takes into account that agents may obtain more than a single goal.

Nevertheless, with respect to their representation of dependence network, we abstract from the representation of the actions needed in order to achieve a goal. For our admissibility criteria we do not need to describe actions, plans, resources or strategies. Also in Cooperative Game Theory games are represented at a level of abstraction which does not describe the actions or the strategies that agents have to perform individually. A game is directly described by means of the consequences that the groups of agents can assure collaborating, without any description about how they can assure those consequences. Moreover we do not make the assumption that an agent can commit a group of agents to achieve a goal g only in the case it is not autonomous for it. Therefore, our AND-arcs do not describe dependencies in the sense developed in [11, 12], but more general potentials for social interaction.

With respect to the approach followed in [14], our approach differs in two aspects. The first aspect regards at which level of abstraction a multiagent system has to be described.

The second aspect from which we differ is more foundational. The notion of reciprocity in [14] is formalized by means of the notion of dependence, i.e. the goals that are exchanged are only those that agents could not achieve on their own. This assumption has the advantage to decrease the space of all possible exchanges: when an agent is self-sufficient in the achievement of one of its own goals, it prefers to rely on its own skills rather than search how to reward other agents that could achieve the same goal. However, in some cases it is worthwhile to reciprocally delegate the achievement of some goals even if the agents involved could achieve these goals on their own. For this reason we relax the condition that agents have to be not self-sufficient in the achievement of the goals exchanged. Therefore, our representation of a multiagent system, the Power Structure, is based only on the representation of the goals desired by the agents and the power that different groups of agents have to achieve that goals.

In recent years several logic-based formalisms have been developed in order to represent some particular notions of power at different levels of abstraction. Pauly [27] provides a modal logic, called Coalition Logic, that axiomatizes a notion of power developed in Game Theory, the α -ability.

Alternating-time Temporal Logic [2, 20, 37], in the following ATL, is one of the most important logic-based formalisms for coordination analysis. It has been developed with a twofold aim. On one hand to describe temporal properties of infinite extensive games with concurrent moves, on the other hand to provide an extension of Computational Tree Logic (CTL) [26] for property verification of modular reactive systems [1].

However these formalisms have not been used to reason about coalition formation, with the recent exception of [10].

Recently, coalition formation has been studied in the framework of boolean games. The two main research works on the issue of boolean games are Harrenstein [23] and Harrenstein et al. [24]. These papers present this new kind of games as to study the strategic interactions between agents. Boolean games are defined in [24] as two players, zero-sum static games where players' utility functions are binary and described by a single propositional

22 *Algorithms for finding coalitions exploiting a new reciprocity condition*

formula, and the strategies available to a player consist of truth assignments to each of a given set of propositional variables controlled by the player. In these works, the authors argue on the strategy profiles for a Boolean as an assignment of binary values to all of its decision variables. The authors allow a strategy profile also to prescribe a value for decision variables that do not occur in the game and assume that the choices made for these "outside" variables do not affect the outcome of the game. It suggests a natural notion of equivalence of Boolean forms: two Boolean forms are said to be equivalent if each strategy profile determines the same outcome in both of them. Defining natural operations on Boolean game forms, they find that, modulo this notion of equivalence, Boolean forms constitute a Boolean algebra. The principal observation of these works are that binary decision variables may be taken as the propositional variables of a propositional language. On this conception, each strategy profile, assigning binary values to all the decision variables, coincides with a valuation for that propositional language. This correspondence between Boolean forms and propositional variables, moreover, proves to determine an isomorphism between the Boolean algebra of Boolean forms and the Lindenbaum algebra of the respective propositional language. These algebraic considerations ensure that Boolean games can straightforwardly be related to classical propositional logic. Equivalence of Boolean forms does not take into account the manipulative powers of the players. The strategic properties of a Boolean game (e.g. a player with a winning strategy or not) depend on the way control over the decision variables over the players as well as on the structure of the underlying Boolean form. The conception of Boolean forms as propositional formulas spawns a number of logical issues concerning distributed control over the propositional variables. Thus, distributed control of the propositional variables becomes a notion amenable to logical analysis.

Another relevant work in on the issue of boolean games is given by Bonzon et al. [9] where they generalize the framework of [24] to n-players games which are not necessarily zero-sum keeping the assumption that each player's preferences are represented by a unique propositional formula, inducing a binary utility function. The authors show how well-known tools from propositional logic can be used to give simple characterizations of the notions of pure strategy Nash equilibria and dominated strategies deriving complexity results for their computation.

These three works are logical based and they do not keep into account the introduction of complex structures such as dependence networks to represent boolean games. Dependency between players and variables in such games naturally induce a dependency relation between players. This view has been proposed by [8].

In contrast with our paper, [8] uses dependence networks to split up a game into a set of interacting smaller games, which can be solved more or less independently. In this paper, instead, we look at characterizing i-dud coalitions in terms of the impossibility to split them in smaller coalitions. Moreover, in [8] simpler graphs are used, undirected and unlabeled, to represent dependencies.

7 Conclusion

In this work we define a criterion of admissibility for coalition formation which is based on the representation of a coalition as a net of exchanges [14]: the i-dud property. This property refines the do-ut-des property [5] by taking into account the fact that two distinct coalitions cannot be considered a whole coalition if they can be formed independently. This condition

arises from the fact that agents prefer to form small coalitions because, as coalitions spring from unanimously agreements, the more are the agents involved in a coalition the more is the risk that one of them gives up joining it.

The i-dud property inherits from the do-ut-des property the fact that it uses only the internal topology of exchanges to check the admissibility of a coalition. Approaches based on Cooperative Game Theory, as [29, 39], abstract from this internal structure, and hence they need to compare a coalition with the other possible coalitions in order to establish its admissibility. This way, also the problem to see if a coalition is admissible, applying for example the notion of core, is intractable.

References

- [1] R. Alur, T. A. Henzinger, F. Y. C. Mang, S. Qadeer, S. K. Rajamani, and S. Tasiran. Mocha: Modularity in model checking. In *Proceedings of the 10th International Conference on Computer Aided Verification, (CAV '98)*, volume 1427 of *Lecture Notes in Computer Science*, pages 521–525, 1998.
- [2] R. Alur, T.A. Henzinger, and O. Kupferman. Alternating-time temporal logic. *Journal of ACM*, 49(5):672–713, 2002.
- [3] R.J. Aumann. The core of a cooperative game without side payments. *Transaction of the American Mathematical Society*, 98:539–552, 1961.
- [4] G. Boella, L. Sauro, and L. van der Torre. Power and dependence relations in groups of agents. In *Procs. of 2004 IEEE/WIC/ACM International Conference on Intelligent Agent Technology (IAT'04)*, Beijin, Cina, 2004.
- [5] G. Boella, L. Sauro, and L. van der Torre. Admissible agreements among goal-directed agents. In *Procs. of 2005 IEEE/WIC/ACM International Conference on Intelligent Agent Technology (IAT'05)*, Paris, France, 2005.
- [6] G. Boella, L. Sauro, and L. van der Torre. Strengthening admissible coalitions. In *Procs. of ECAI'06*, Riva del Garda, 2006.
- [7] G. Boella, L. Sauro, and L. van der Torre. From social power to social importance. *Web Intelligence and Agent Systems Journal*, 5(4):393–404, 2007.
- [8] E. Bonzon, M. Lagasquie-Schiex, and J. Lang. Dependencies between players in boolean games. In *ECSQARU 2007*, pages 743–754, 2007.
- [9] E. Bonzon, M. Lagasquie-Schiex, J. Lang, and B. Zanuttini. Boolean games revisited. In *17th European Conference on Artificial Intelligence (ECAI'06)*, pages 265–269. Springer-Verlag, 2006.
- [10] N. Bulling, C.I. Chesnevar, and J. Dix. Modelling coalitions: Atl + argumentation. In *AAMAS 2008*, pages 681–688, 2008.
- [11] C. Castelfranchi. Founding agents' “autonomy” on dependence theory. In *Proceedings of the 14th European Conference on Artificial Intelligence (ECAI'00)*, pages 353–357, Berlin, Germany, 2000.
- [12] C. Castelfranchi. The Micro-Macro Constitution of Power. *ProtoSociology*, 18-19, 2003.
- [13] R. Conte and J. Sichman. Multi-Agent Dependence by Dependence Graphs. In *Proceedings of The First International Joint Conference on Autonomous Agents & Multiagent Systems (AAMAS'02)*, pages 483–490, Bologna, Italy, 2002.
- [14] R. Conte and J. S. Sichman. Dependence graphs: Dependence within and between groups. *Computational & Mathematical Organization Theory*, 8(2):87–112, 2002.

24 *Algorithms for finding coalitions exploiting a new reciprocity condition*

- [15] T. Cormen and C. Leiserson R. Rivest. *Introduction to Algorithms*. MIT Press, 1990.
- [16] D. Dang and N. Jennings. Generating coalition structures with finite bound from the optimal guarantees. In *Proceedings of The Third International Joint Conference on Autonomous Agents & Multiagent Systems (AAMAS'04)*, pages 564–571, New York, USA, 2004.
- [17] M. d'Inverno and M. Luck. A formal view of social dependence networks. In *Proceedings of First Australian Workshop on Distributed Artificial Intelligence (DAI'95)*, volume 1087 of *Lecture Notes in Computer Science*, Camberra, Australia, 1996.
- [18] P. E. Dunne and M. Wooldridge. Preferences in qualitative coalitional games. In *Proceedings of the sixth Workshop on Game Theoretic and Decision Theoretic Agents (GTDT'04)*, pages 29–38, New York, USA, July 2004.
- [19] E. H. Durfee and J. S. Rosenschein. Distributed problem solving and multi-agent systems: Comparisons and examples. In *The Thirteenth International Distributed Artificial Intelligence Workshop*, pages 94–104, Seattle, Washington, 1994.
- [20] V. Goranko and G. van Drimmelen. Decidability and complete axiomatization of the alternating-time temporal logic. *Theoretical Computer Science*, to appear.
- [21] B. Grosz, S. Kraus, S. Talman, B. Stossel, and M. Havlin. The influence of social dependencies on decision-making: Initial investigation with a new game. In *Proceedings of The Third International Joint Conference on Autonomous Agents & Multiagent Systems (AAMAS'04)*, pages 782–789, New York, USA, 2004.
- [22] D. Harel, D. Kozen, and J. Tiuryn. Dynamic logic. In D. Gabbay and F. Guenther, editors, *Handbook of Philosophical Logic Volume II — Extensions of Classical Logic*, pages 497–604. D. Reidel Publishing Company: Dordrecht, The Netherlands, 1984.
- [23] P. Harrenstein. *Logic in Conflict*. PhD thesis, Utrecht University, 2004.
- [24] P. Harrenstein, W. van der Hoek, J. Meyer, and C. Witteveen. Boolean games. In *TARK 2001*, pages 287–298, 2001.
- [25] S. Kraus and T. Plotkin. Algorithms of distributed task allocation for cooperative agents. *Theoretical Computer Science*, 242(1-2):1–27, 2000.
- [26] K.L. McMillan. *Symbolic Model Checking*. PhD thesis, CMU University, 1992.
- [27] M. Pauly. A Modal Logic for Coalitional Power in Games. *Journal of Logic and Computation*, 12:146–166, 2002.
- [28] R. W. Rosenthal. Cooperative games in effectiveness form. *Journal of Economic Theory*, 5:88–101, 1972.
- [29] T. Sandholm, K. Larson, M. Andersson, O. Shehory, and F. Tohm. Coalition structure generation with worst case guarantees. *Artificial Intelligence*, 111(1-2):209–238, 1999.
- [30] T. Sandholm and V. Lesser. Coalitions among computationally bounded agents. *Artificial Intelligence*, 94(1):99–137, 1997.
- [31] L. Sauro. *Formalizing Admissibility Criteria in Coalition Formation Among Goal-directed Agents*. PhD thesis, 2006.
- [32] L. Sauro. Qualitative criteria of admissibility for enforced agreements. *Computational & Mathematical Organization Theory*, 12(2-3):147–168, October 2006.
- [33] L.S. Shapley and M. Shubik. Pure competition, coalitional power and fair division. *International Economic Review*, 10(3):337–362, 1969.
- [34] O. Shehory and S. Kraus. Methods for Task Allocation via Agent Coalition Formation. *Artificial Intelligence*, 101(1-2):165–200, 1998.
- [35] J. S. Sichman and Y. Demazeau. On social reasoning in multi-agent systems. *Revista Iberoamericana de Inteligencia Artificial*, 13:68–84, 2001.

**NOT FOR
PUBLIC RELEASE**

Algorithms for finding coalitions exploiting a new reciprocity condition 25

- [36] R. Tarjan. Depth-first search and linear graph algorithms. *SIAM Journal of Computation*, 1(2):146–160, 1972.
- [37] W. van der Hoek and M. Wooldridge. Cooperation, knowledge, and time: Alternating-time temporal epistemic logic and its applications. *Studia Logica*, 75(1):125–157, 2003.
- [38] J. von Neumann and O. Morgensten. *Theory of Games and Economic Behavior*. John Wiley and Sons, 1944.
- [39] M. Wooldridge and P.E. Dunne. On the computational complexity of qualitative coalitional games. *Artificial Intelligence*, 158(1):27–73, 2004.
- [40] M. Wooldridge and J. Jennings. Towards a theory of cooperative problem solving. In John W. Perram and Jean-Pierre Müller, editors, *Proceedings of the Sixth European Workshop on Modelling Autonomous Agents in Multi-Agent Worlds (MAAMAW-94)*, volume 1069 of *Lecture Notes in Computer Science*, pages 40–53, Odense, Denmark, 1994. Springer.
- [41] G. Zlotkin and J. S. Rosenschein. Mechanism design for automated negotiation, and its application to task oriented domains. *Artificial Intelligence*, 86(2):195–244, 1996.

Received 17 October 2008