UNIVERSITA' DEGLI STUDI DI TORINO
DIPARTIMENTO DI INFORMATICA

DOTTORATO DI RICERCA IN INFORMATICA

CICLO XVIII

Formalizing Admissibility Criteria in Coalition Formation among Goal-directed Agents

PRESENTATA DA: Luigi Sauro

TUTORS: prof. G. Boella
prof. L. van der Torre

COORDINATORE DEL DOTTORATO: prof. P. Torasso

ANNI ACCADEMICI: 2002/2003 - 2003/2004 - 2004/2005

SETTORE SCIENTIFICO DISCIPLINARE DI AFFERENZA: INF/01

Abstract

This work studies how goal-directed agents can form profitable coalitions. A coalition is formed when some agents agree to cooperate for the achievement of a shared goal or to exchange with each other the achievement of their own goals.

We define two criteria of admissibility that establish which coalitions can be formed under the assumption that agents are self-interested. The first admissibility criterion, the do-ut-des property, formalizes a notion of reciprocity that, informally, can be described as "I give something only if I obtain something else in exchange". The second admissibility criterion, the composition property, requires that if two coalition formation processes do not effect each other, then they will managed separately. So, roughly, the composition property describes the preference of the agents to form small coalitions. This attitude can be justified by considering that a coalition formation process usually becomes more costly and it has less chance to succeed when the number of the agents involved in it increases.

We compare the do-ut-des property, which is a qualitative criterion of admissibility, with the quantitative approach developed in Cooperative Game Theory. In particular, the do-ut-des property can be used as a qualitative methods to restrict the search space of a quantitative criterion of admissibility, we called the q-do-ut-des property, which is even more restrictive than the well-known notion of core.

Finally, we define a modal logic to reason about which goals a group of agents can assure if they collaborate. With respect to similar modal logics, as Alternating Time Temporal Logic or Coalition Logic of Propositional Control, our logic enables to explicitly refer to the actions that the agents have to perform in order to achieve a certain state of affairs.

Table of Contents

Chapter 1

Introduction

1.1 Introduction

In recent years network technologies make the perspective of several computing units operating in the same environment more and more plausible. This trend is testified by the relevance of the *service-oriented* computing paradigm, in particular web services, where monolithic infrastructures and applications owned and managed by one actor are giving way to networks of coordinating applications (*services*) owned and managed by many partners [CKM+03, AGPS04]. Another example consists of the large diffusion of peer-to-peer networks, or the Grid, enabling several individuals to exchange not only files, but also direct access to computers, software, data and other kinds of resources and skills [Fos02, MGS05]. Artificial agent technology attempts to develop software agents able to operate on behalf of their owners in open contexts. For example some web services, supporting the booking of flights, hotels etc., can cooperate to organize a vacation and interact with a personal digital assistant operating on behalf of a customer. Or, file-sharing peers could be software agents which, based on the tastes of their owner, autonomously seek in a peer-to-peer network the files that could be desired.

Abstracting from a particular application domain, in this thesis we consider self-interested agents whose proactive behaviors may interfere with each other. In this context, an agent has to recognize the presence of the other agents as potentially beneficial or obstructing and eventually to seek profitable collaborations. As we consider self-interested agents, if a group of agents decide to collaborate, then each member has to profit from this collaboration. Another consequence deriving from the hypothesis of self-interested agents is that a group of agents may collaborate regardless from the effects, beneficial or obstructing, on the agents which are not member of the group. For this reason we say that they form a coalition. Usually coalition formation processes are complex and costly social behaviors. For this reason, agents have to recognize which coalitions are not strategically admissible and, in order to economize the coalition formation process, restrict their proposals only to the admissible ones.

The problem of what a coalition is and which coalitions should be formed has been studied by sociologists and economists [Eme62, Gou93, YC93, vNM44, Aum61, Pel98]. Generally, social and economic science has provided an apparatus of concepts that can be used to frame different aspects of Multiagent Systems and useful suggestions about which direction in artificial multiagent technology would be more promising. Some examples regard the study on strategic negotiation [KS03, ZR89], representation of contracts [BvdT04], norms and normative systems [BvdT03]. Vice versa, agent theory has been used as a methodological instrument in the analysis of social [Cas01]

and economic issues [SLA+99, SK98]. Also in the case of coalition formation we borrow from sociologists and economists theories that can be useful to describe and reason about which coalitions are admissible to be formed.

Naturally, in order to formalize social interactions among the agents, it is necessary first to describe what an agent is. An central issue in our study of agents and coalition formation is about modeling the judgement process of an agent with respect to the possible coalitions it can join.

We agree with the idea of Simon [Sim55] that the analysis of the *right* behavior of an agent can follow two different directions. The first one regards the study of the *environment of choices* and the criteria to calculate the *rational* choice depending on a quantitative representation of the reliability and profitability of the outcomes. The second approach studies models of the *choosing organisms* which take into account possible limits in their cognitive processes. This first approach is more idealized and historically has been studied in the field of decision making. The second approach is more directed to the development of concrete models of planning, desires elicitation, conflicts resolution. This approach takes into account possible limitations of an agent, such as the computational resources or the amount of information available, for this reason it has been the one most followed by Artificial Intelligence.

Even if several works have attempted to compare the two approaches and find possible ways to integrate them [vdTDB05, BDH99, Bou94], here we adopt the second direction in the Simon's dichotomy and assume that agents are goal directed.

1.2 Objectives

In this section we introduce the objectives of this thesis, framing them in the research area of coalition formation. Coalition formation leads to extend the cognitive models designed in a single agent perspective, as the BDI model [RG95], in order to enable collaboration among the agents. In particular, agents have to be able:

1. To reason about the effects of possible group behaviors and, in particular, to find out which groups of agents have the power to achieve which goals.

2. To select those group behaviors that are admissible to be proposed in a coalition formation process.

3. To make a proposal and follow an agreement protocol.

4. In the case an agreement is reached, to effectively form a coalition according to the commitments described in the agreement.

Each of these tasks is nowadays a research area in the field of Multiagent Systems and Distributed Artificial Intelligence. Generally, the first task takes as input the capabilities of agents, as for example the actions they are able to execute, and a causal model of the environment in which the agents *live*, then it requires to reason about which group behaviors have to be engaged in order to achieve one or more goals. A large amount of research is focussed on this issue (see [Fer99] for an overview), in Chapter 2 we describe more in details a logic-based approach, Alternating Time Temporal Logic [AHK02], which in recent years has been object of a large interest in the Multiagent Systems community. The second task leads to develop agents which act *economically*, i.e. by proposing a coalition (or by evaluating a proposed coalition) each agent tries to maximize its own profit as much as the other members of the coalition cannot agree to a better proposal. This research area has been historically developed in Cooperative Game Theory [vNM44, Aum61] and more recently also by the Multiagent System community [SD01, SL97, WD04, BS99, KS96, CKB04a, BSvdT04a]. The third task concerns the problem of how to design a protocol for the coalition formation process. Depending on the application domain, several types of protocols can be considered. For example, Boella et al. [BvdTar] and Sandholm et al. [SL02] focusses on the stipulation of contracts, whereas Zlotkin et al. [ZR89, ZR96b] consider a negotiation protocol with repeated offers and counteroffers. Shapley et al. propose a division protocol where the distribution of resources or commitments is established without any form of collusion or negotiation among the agents, but it is imposed following some conditions of fairness [SS69]. Finally, somewhat in middle between negotiation and fair division protocols are auctions, where a offer/counteroffer protocol coexists with a fairness condition, which rules the outcome of an auction, imposed *a priori* by the auctioneer (see Wooldridge [Woo02] for an overview).

The fourth task regards all the cognitive processes required to actually form a coalition when an agreement is stipulated. It studies, for example, the capability of the agents to reach the common knowledge that each agent involved in the agreement has the intention to act accordingly and to warn the other agents when a commitment cannot be fulfilled anymore. These issues has been formalized, for example, by Cohen et al. [CL90] and by Wooldridge et al. [WJ94], whereas the implementation of a concrete system, STEAM, has been developed by Tambe et al. [Tam97].

In this work we are interested in the formalization of the first two tasks mentioned above and we use the *theory of social power and dependence*[1] pioneered by Castelfranchi as starting point [Cas03, Cas00, SD01].

The theory of social power and dependence is an attempt to transfer and refine theories developed initially in the field of sociology [Eme62, Gou93, YC93] to the field of Multiagent Systems. This theory models the potential interactions among the agents which lead to the achievement of a shared goal (*cooperation*) or the reciprocal satisfaction of their own goals (*social exchange*).

The theory of social power and dependence enables to introduce some qualitative notions regarding the profitability for the agents to undertake social exchanges. An agent not only should be able to recognize the potential of a group of agents to help it in the satisfaction of a goal, but also it has to reason about how it can reward them. This involves the development of a social reasoning mechanism [SD01] that analyzes the possibility to profit from mutual-dependencies (two agents depend on each other for the satisfaction of a shared goal) or reciprocal-dependencies (two agents depend on each other for the satisfaction of two different goals). Assume, for example, that an agent ag_2 can achieve a goal g of another agent ag_1. The agent ag_1 may rely on ag_2 in the case it cannot achieve g by itself or it considers more profitable to order this goal to ag_2. Since we are assuming that the agents are self-interested, ag_1 has to provide ag_2 with some reason for the achievement of g. In other words, while ag_2 has the power to help ag_1 by achieving the goal g, ag_1 needs a power of influencing ag_2 to actually achieve g. In human interactions different forms of power of influencing can be considered. ag_1 may try to convince ag_2 that the achievement of g is desirable also for ag_2: different forms of propaganda fall back in this kind of power of influencing. Or, it could try to modify the self-interested nature of ag_2 and convince it to be altruist: several moral and religious precepts are of this kind. Finally ag_1 has a power of influencing ag_2, because it has the power to achieve a goal g' desired by ag_2 and hence it can propose to ag_2 to exchange the achievement of g with the achievement of g'. Social exchanges focus only on the last kind of power of influencing, considering the net of inter-dependencies as the glue for coalition formation. In this sense, coalition formation can be seen as a balancing process [Eme62] in which this inequality between ag_1 and ag_2 is tried to be balanced.

[1] This locution is invented here. However Sichman et al. [SD01] use the locution *dependence theory*, whereas Castelfranchi focusses in some of its works on the notion of power and social functions in general [Cas03, Cas01].

Initially the theory of social power and dependence considered only the case of diadic exchanges, i.e. exchanges involving only two agents [SCCD94, Cas03, SD01]. Nevertheless agents should be able to reason about more complex nets of exchanges, as in cooperative game theory possible coalitions may involve several agents coordinating to maximize their utilities. The following question arises: which groups of agents could form a coalition and which goals could they exchange with each other under the assumption that they are self-interested?

Nets of exchanges, called by Yamagishi et al. generalized exchanges [YC93], have been studied in the field of Multiagent Systems by Conte and Sichman in [CS02b, CS02a]. In particular, they define a notion of admissibility for coalition formation based on the condition of reciprocity, i.e. an agent takes part in the achievement of a goal only if it receives the achievement of some of its own goals in exchange. We call this property, using a circumlocution in Latin, the *do ut des* property (literarily give to get). However, they consider only a particular class of nets of exchanges where each agent has only one goal. The main objective of this thesis is to develop and study some notions of admissibility in the general case where agents can have more than a single goal.

In Chapter 3 we provide a definition of the do-ut-des property in the case of agents with multiple goals. However, the do-ut-des property does not take into account the fact that the more the agents involved in a coalition are, the less the coalition formation process can be managed. For this reason we also consider a more restrictive criterion of admissibility, the composition property, that does not consider admissible a net of exchanges when it composed by two subparts which could be formed independently.

Finally, as each of the four steps in a coalition formation process is preparatory for the next one, the possibility to employ our admissibility criteria depends on the possibility to formalize and recognize the power of groups of agents. For this reason we develop in Chapter 5 a modal logic [Che80] to formalize the notion of power. This logic is closely related to the ones already developed by Pauly [Pau02a], Alur et al. [AHK02] and van der Hoek et al. [vdHW05]. However, it differs from the previous approaches because it enables to describe explicitly not only which groups of agents can assure the achievement of a certain goal, but also which shared plan they have to execute. This way, our approach is more in compliance with several frameworks regarding task allocation problems [Fer99, SK98], cooperative problem solving [WJ94] or social reasoning mechanism [SD01]. In these frameworks an agent, which desires a goal g, does not first consider a possible group of agents and then verify if it can assure the achievement of g. On the contrary it initially generates a plan achieving g, and then it looks for those agents

that are able to execute that plan.

1.3 Assumptions

In this section we discuss two crucial assumptions we made in the study of admissibility criteria for coalition formation processes.

The first assumption distinguishes somewhat our work from the previous work regarding social exchanges. Conceptually, social exchanges are formalized on top of the theory of social power and dependence as follows: the basic notion is the notion of social power. Intuitively[2], an agent, or a group of agents, has the power to achieve a state of affairs if two conditions are satisfied. First, it has all the required capabilities to bring it about that state of affairs. This condition alone is not sufficient to correctly describe the notion of social power. There could be, indeed, lots of states of affairs that an agent can bring it about, but if these states of affairs do not satisfy any of the goals of the agent itself or of another agent, then this ability cannot be used profitably neither directly nor by means of an exchange. Thus, it does not provide any power. For this reason, the notion of social power requires also that the state of affairs an agent is able to achieve satisfies "the possible goal of some agent" [Cas03]. Starting from the notion of power the next step in the formalization of social exchanges is to define the notion of dependence. An agent ag_1 depends on another agent ag_2 for a goal g if it is not self-sufficient in the achievement of g, whereas ag_2 has the power, or is a member of a group that has the power, to achieve g. So, the notion of dependence can be represented, as in [CS02b, CS02a, SCCD94], by means of a graph, called dependence graph, where the nodes represent the agents in a multiagent system and the arcs represent the dependencies among the agents. Considering a generalized exchange as a sub-graph of a dependence graph, the do-ut-des property is defined in [CS02b] as an admissibility criterion for generalized exchanges.

Thus, the do-ut-des property is formalized starting from the dependencies among the agents, and hence it requires for each generalized exchange that the agents involved are not self-sufficient in the achievement of the goals that they obtain in the exchange. However, this condition could be too restrictive in some cases. It could be the case that two agents, even if they are self-sufficient in the achievement of their own goals, may find it profitable to help reciprocally or exchange their goals when the costs that each of them have

[2]In Chapter 2 we will provide a more detailed description of the theory of social power and dependence as well as of the notion of generalized exchanges and the notion of reciprocity.

to sustain collaborating are less than the costs they would sustain when they act individually.

For this reason we relax the hypothesis that an agent has to be not self-sufficient in the achievement of goal to admit a collaboration. Thus, if a group of agents Q has the power to achieve a goal g, every agent that desires g may rely on Q for its achievement, independently from the fact that it can or cannot achieve g by its own. From a formal point of view, we formalize admissible criteria directly on a structure representing the power of groups of agents.

The second assumption regards which kind of protocol is followed in the coalition formation process. Even if our work does not focus on the study of agreement protocols, it is necessary to make some assumptions on it because, generally, protocols influence the ways the agents can behave and the relative outcomes. For example, the social reasoning mechanism developed by Sichman et al. in [SD01] the authors distinguish between cooperation, when two agents mutually depend for the achievement of a shared goal, and social exchange, two agents reciprocally depend for the achievement of their own goals. The authors argue that an agent, in general, should prefer cooperation to social exchange because, by sharing the same goal in cooperation, it should not fear an absence of reciprocation from the other agents involved in the social interaction. In other words, when two agents agree to make an exchange, one of them could be dishonest not achieving, once obtained its own goal, the goal of the other agent[3].

We do not make this distinction considering that the protocol used enables to form trustfully coalitions both in the case of cooperation as well as in the case of exchange. In particular, we restrict the domain of interest to the cases in which the protocol used in the coalition formation processes consists of a collusive behavior supported by unanimous and enforced agreements. With collusive behaviors we mean that the agents can decide to join a coalition without imposition by anyone. With unanimity we mean that an agreement is stipulated, and hence it is effective, when all the members involved in the agreement sign it. Finally, an agreement is said to be enforced when, once an agreement is stipulated, the cooperations or social exchanges described in it become commitments for the parties and hence they cannot deviate their behavior from what established in the agreement. In real cases the enforcement of an agreement is assured by the presence of a normative system that monitors the behavior of the agents and punishes possible deviation

[3]Somewhat surprisingly Yamagishi et al. [YC93] provide some experimental evidence in the field of sociology that, mutual trust being equal, generalized social exchanges (called there network generalized exchanges) promotes a higher level of collaboration with respect to the case of cooperation (called group-generalized exchanges).

from the agreement. The institutional fact that constitutes and represents an unanimous and enforced agreement is a contract. So, at the end, we can say that the protocol the agents use to form a coalition is a stipulation of a contract. The possibility for the agent to reason on the real efficiency of the normative system and eventually violate a contract (see for example Boella et al. [BL02]) is out of the scope of this work.

Finally, we notice that the assumption of enforced agreements is not a necessary condition for the occurrence of exchanges, but, as suggested in [YC93], a way to overcome the potential lack of mutual trust among the agents.

1.4 Methodology

In this section we present the methodology we have used to define admissibility criteria in coalition formation. The crucial methodological issues regard:

- The typology of agents considered in the study of coalition formation, i.e. goal-directed agents, and its relevance in our context.
- The level of abstraction we have chosen to formalize our notions of admissible coalitions.
- Which kind of evidence we have adopted to find out and justify these notions.
- How to relate the level of abstraction chosen with a more detailed representation of a multiagent system.

As said in Section 1.1, we consider multiagent systems populated by goal directed agents. Traditionally a goal is meant as a condition that distinguishes the possible outcomes in two disjoint subsets, nominally satisfied - not satisfied. The idea is that an agent does not simply sequentially process the set of possible choices looking for the best one, but it *constructs*, by means of some cognitive process, its choice and, in order to manage the complexity of this cognitive process, need to simplify its judgement criterion [Sim55]. However, different aspects may play a role in the judgement of the possible outcomes. In these cases the proactive behavior of an agent cannot easily described by means of the satisfaction of a single condition.

For this reason we assume that agents may have multiple judgement criteria as found in the theory of decision with multiple objectives [KR76]. There are several example that illustrate this typology of judgements. We consider the following [KR76]:

A mayor must decide whether to approve a major new electric power generation station. There is needed for more electricity, but a new station would worsen the city's air quality, particularly in terms of air pollutants such as sulfur dioxide [...]. The mayor should be concerned with the effects that his actions will have on (1) the health of residents (2) the economic condition of the residents (3) the psychological state of the residents (4) Economy of the city and the state. [...]

In this example a decision leads to an analysis of its effects under different and not easily comparable aspects such as the health of the residents or the economy of the city. This entails that at first level of analysis the judgement on a possible decision has to be split on multiple, independent objectives. Each objective can be associated with a condition describing when that objective can be considered satisfied. Each pair objective-condition constitutes a different goal. Since the objectives are not comparable also the goals are not, or even if the *owner* can compare them by means of some cost-benefit analysis, probably the other agents consider them as not comparable because, in general, it is much more easy to recognize the goals of the other agents with respect to know how they compare them. For a compelling explanation of this issue we remand, without repeating it here, to the story *The Lady or the tiger?* in [New80].

Even if goals are not directly comparable, the achievement of sets of goals could be. So, in the example of the electric power generation station, assuring a good economic condition and health for the resident is better than assuring only a good economic condition, Newell in [New80] calls to this principle as the *preference of joint goal satisfaction*. On the other hand, it is reasonable in a coalition formation process that an agent, for the same set of achieved goals, prefers to be involved in the achievement of the goals of the other agents as little as possible. We use these kinds of reasoning to formalize qualitative preferences of the agents and to define our criteria of admissibility.

The second methodological issue regards the level of abstraction we have used to define our admissibility criteria. We said in Section 1.3 that we define our criteria on a structure representing the power of groups of agents, so some comments about the notion of power are required. Several works attempted to formalize the notion of power [Cas03, AHK02, Pel98, Pau02a, BSvdT04b, Por70]. However, a general characterization of this notion still misses. For example the fact that a group of agents Q has the power to achieve a certain goal g could mean that the agents in Q know a shared plan assuring the achievement of g and they have all the abilities and resources needed to execute that plan. This meaning of power has been studied in [AHK02, vdHW05] and it is additive in the sense that adding new agents to a group does not waste its power. So, for example, if an hacker is able

to jeopardize a site, every group of agents in which he is involved will be dangerous for that site.

However, in several contexts this interpretation cannot be satisfactory used. Saying that a fencer has the power to win the world cup, does not means that he has *a priori* a plan to win, on the contrary, while fencing, he must attack and defend reacting to what the challenger does. Therefore, in this context the notion of power refers generically to his skills and, furthermore, it does not seem to mean that the fencer can certainly obtain the cup, but, for example, that he has good chances to win compared with the other athletes.

When we consider further contexts other relevant differences may arise. For example, an arbiter has the power to delay a match for few minutes or a judge has the power to sentence a suspect. In these contexts the specified power derives from the role the arbiter and the judge are playing, and hence, it could be misleading to claim that the group composed by the arbiter and the goalkeepers has the power to delay a match for few minutes or that a group of individuals involving a judge has the power to sentence a suspect. Therefore in these contexts the notion of power is typically not additive. Finally, another notion of power concerns the reliability to demand from a group of agents a certain goal. For example, a surgery team has the power to operate an appendicitis. However, this power is not additive in the sense that we are not inclined to pay even more persons for this operation. In other words, we consider that all the member of a group Q have to be necessary or at least useful for the achievement of a goal g, only in this case the group Q can be ordered for g and hence it has the power to achieve g. Also these meanings of power are not additive.

So, how to deal with all these different notions of power? A first solution is to refer to only one, or few, of them and develop our theory on it. In this perspective we develop a logic enabling to formalize different notions of power and in particular the ones described in the last example where all the agents involved in the achievement of a goal are required to be necessary or at least useful. However, referring to a particular notion of power has the drawback to limit the generality of our work.

The second solution is to bypass this issue, asking if it is really necessary in order to define our admissibility criteria to precisely characterize how to *calculate* the power of group of agents. A suggestion in this sense comes from Cooperative Game Theory. A cooperative game is defined as a function *att* associating each group of agents with the consequences they have the power to obtain. No particular assumptions is made about neither how they can actually attain them (due to plans, skills, roles, etc.) nor what *attaining a consequence* really means. On condition that a notion of power is homogeneously used to define *att*, what really matters in the study of which

coalitions can or should be formed is the distribution of power among the groups. However, according to the particular notion of power used, *att* may satisfy some particular properties as, for example, additivity.

Analogously to Cooperative Game Theory, we use a level of abstraction, the Power Structure, in which the power of groups agents is already provided assuming that the problem to calculate the power of groups of agents is addressed in a more detailed description of the multiagent system. This level of abstraction has been already used in the field of Multiagent Systems by Pauly [Pau02a] to study the axiomatization of a particular notion of power, called α-ability.

The third methodological issue regards how we characterize an admissibility criterion. As an admissibility criterion has to provide an *economical norm* on the behavior of the agents, we borrow from Game Theory the method to define it as a no-dominance condition, as for example the Nash equilibrium or the notion of core. A no-dominance condition is based on the preferences of the agents involved in a coalition. In contrast with Game Theory, our preferences are qualitative, as we do not directly compare goals, but simply balance sets of goals by means of inclusion relation. Another important difference with Game Theory is that the admissibility criteria developed there require to compare a coalition with all the other potential coalition in order to verify if it is admissible. 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 they 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*. So, due to the large amount of efforts in the field of Game Theory, a comparison of our results with game theoretical results would provide us some further indication of the evidence of our admissibility criteria. For this reason, we suppose that agents use a quantitative cost-benefit analysis to compare potential coalitions and 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.

Finally, the applicability of our admissibility criteria depends on the possibility to constructively formalize the notion of power from a more detailed representation of a multiagent system. We develop a framework by means of modal logic technics to describe some particular notions of power. In this framework a multiagent system consists of an environment with states and

transitions and a set of agents, each of them characterized by the actions it is able to execute. A logical approach buy us the possibility to rely on robust results which have been recently developed as Coalition Logic [Pau02a], Alternating Temporal Logic [AHK02] or CL-PC [vdHW05]. With respect to these frameworks our approach has the advantage to be modular, i.e. we first define a logic to describe the environment and the effect of executing some concurrent actions (represented as boolean formulas on individual actions) on its states. Then, we define a logic to describe the actions the agents can execute individually as well as the concurrent actions that groups of agents can execute by coordinating. Finally, we put the two modules together and define which states of the environment a group of agents can assure by executing certain concurrent actions. As in CL-PC we consider that a concurrent action may specify both that some actions have to be executed as well as that some other actions have not to be execute. This involves an asymmetry in the groups of agents that are able to execute a certain concurrent action β because if β prescribes that an action a has to be executed, then a group of agents able to execute β needs at least an agent able to execute a. On the other hand, if β prescribes that an action a have not to be executed, then all the agents able to execute a can obstruct the execution β, therefore a group of agents able to execute β has to include all the agents able to execute a.

1.5 Structure of the thesis

This thesis is structured as follows. Chapter 2 surveys some relevant results related with our objectives. First, a general overview is given on Coalition Formation in Multiagent Systems and, more in general, in Distributed Artificial Intelligence. Then, the theory of social power and dependence is shown, giving a special emphasis to its impact on Coalition Formation. Finally, Alternating-time Temporal Logic is outlined as a witness of those modal logics which formalize the power of a group of agents to attain a certain state of affairs.

Chapter 3 is the core of this thesis. We define a representation of a multiagent system, the Power Structure, which abstracts from the characteristics of the single agents and directly describes which groups of agents has the power to achieve which sets of goals. If a group can see to the achievement of one or more goals, then it can potentially commit itself with this respect. So, a proposal for the formation of a coalition consists of a set of these (potential) commitments. Once defined possible coalitions, we define two admissible criteria, the do-ut-des and the composition property, which prune those coalitions that cannot be strategically formed. We characterize our cri-

teria first like game theoretical no-dominance criteria, then, for a particular class of Power Structures, we called singleton Power Structures, we characterize them by means of the chains of exchanges each agent is interested in. Finally we discuss an algorithm to find the singleton Power Structures that satisfy the composition property.

In Chapter 4 we relate our approach with some results in the field of Cooperative Game Theory. First, we provide a brief overview on Cooperative Game Theory. Then, we investigate the relation among our do-ut-des property and the notion of core.

The Power Structure defined in Chapter 3 describes, roughly speaking, the power distributed among the agents *as such*. Therefore, it represents a multiagent system at a high level of abstraction which, generally, is not the one we have as input. Thus, Chapter 5 is devoted to the formalization of a modal logic which enables to formalize some notions of power that cannot naturally be formalized in similar logics, as Alternating-Time Temporal Logic. In particular, a complete axiomatization of our logic is presented.

Chapter 2

Related Works

2.1 Introduction

Several works concern the research topic of coalition formation, focussing on different aspects of this problem. Moreover, as with all the research topics, its boundaries are somewhat fuzzy and overlap with other research topics in the field of Multiagent Systems and, in general, Artificial Intelligence. In this section we try to provide an overview of the work on coalition formation and to relate it to other closely related research areas.

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) [ZR96b, DR94, SK98]. In DPS, there is some notion of global utility [SLA+99] or a set of tasks to be fulfilled [KP00, SK98] 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 [SL97].

An interesting bridge between DPS and MAS is given by the mechanism design problem [ZR96b], 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.

In Section 2.2 and Section 2.3 we provide an overview on some relevant results concerning coalition formation respectively in the field of Distributed Problem Solving and Multiagent Systems. In the remaining sections we give a more detailed overview of the theories and formalisms which are closely related to the present work. In particular Section 2.4 describes the theory of social power and dependence and a derived model of coalition formation

developed by Sichman et al. [SD01]. In Section 2.5 we describe the work of Conte et al. where generalized exchanges are formalized and a definition of reciprocity for generalized exchanges is provided. Finally, we briefly present Alternating Temporal Logic [AHK02], a logic-based framework used to describe coordination policies and a particular notion of power. Describing ATL gives us the possibility to show some properties which a notion of power can satisfy and to emphasize the limits and advantages of our approach to the coordination issue.

2.2 Coalition formation in Distributed Problem Solving

A coalition formation process occurs in DPS 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. [SLA+99] study the problem of finding which partition of the population of agents maximizes a global utility function when superadditivity (agents always do better joining all together) is not guaranteed. This partition is called a coalition structure. The problem has been formalized at a high level of abstraction associating to each coalition the utility value that it can assure collaborating. This representation has been already used in the field of Cooperative Game Theory and is called the characteristic form of a game.

A trivial solution to this problem consists of searching the set of all possible partition for the optimal one. However, the complexity of this algorithm in the worst case is highly intractable as the number of possible partitions of a population of agents is $O(a^a)$ and $\omega(a^{\frac{a}{2}})$, where a is the cardinality of the population. So, an algorithm is provided that, searching *only* 2^{a-1} possible partitions, finds a ratio bound to the optimal value equal to a. This algorithm simply limits its search to the set of partitions composed by the grand coalition, i.e. the coalition composed by all the agents, and all the partitions with cardinality equal to two. It is proved that:

- The best value found b is greater than $\frac{u^*}{a}$, where u^* is the optimal value.
- As the number of agents grows, the fraction of coalition structures needed to be searched approaches zero.
- No other search algorithm can establish any bound k while searching only 2^a or fewer partitions.

They also provide an anytime algorithm that improves the bound found from the previous algorithm. Informally, this algorithm receives as input the bound b and starts a *for* loop. Starting from a up to 3, each iteration of the loop consists of searching a level l of partitions, where a level is understood as the set of all partitions with cardinality l. Level one and level two as been already searched in the previous algorithm, this is the reason the loop ends when $l = 3$. Once terminated, it returns the partition with the greatest value found so far. It is proved that when the search of a level l is completed, the bound to the optimal value is equal to $\lceil \frac{a}{h} \rceil$ if $a \equiv h - 1(\mod h)$ and $a \equiv l(\mod 2)$, $\lfloor \frac{a}{h} \rfloor$ otherwise, where:

$$h = \left\lfloor \frac{a-l}{2} \right\rfloor + 2$$

Therefore, since the level $l = a$ consists of a single partition composed by singleton groups of agents, searching only this partition, the bound is decreased from a (which is the bound found by the previous algorithm) to $\frac{a}{2}$. However, in general to lower the bound from $\frac{a}{n}$ to $\frac{a}{n+1}$ it is necessary to search two levels more, due to the factor $\frac{1}{2}$ in the definition of h, but this is exponentially costly.

Kraus et al. in [SK98] have studied task allocation problems via coalition formation. There is a set of n agents, $Ag = \{ag_1, \ldots, ag_n\}$ and a set of m independent tasks $T = \{t_1, \ldots, t_m\}$ to be fulfilled, eventually according to a precedence order. Each agent ag_i has a vector of real non-negative capabilities $\langle c_1^i, \ldots, c_r^i \rangle$. Each capability is a property of an agent that quantifies its ability to perform a specific action or the amount of resource it has at its disposal. For the satisfaction of each task t_l, a vector of capabilities $B_t = \langle b_1^l, \ldots, b_r^l \rangle$ is necessary. In order to fulfill the tasks agents can join their capabilities forming coalitions. Two possibilities are taken into account, in the first case an agent can join at most one coalition and its contribution to the coalition is its capability vector, in the second case an agent can join more than one coalition and it has to decide how to share its capability vector among the joined coalitions. The capability vector of a coalition Q, $B_Q = \langle c_1^Q, \ldots, c_r^Q \rangle$, 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 $\mathcal{Q}$ and a task assignment for them such that the sum $u_{\mathcal{Q}}$ 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. First of all a $b \leq n$ is fixed and it is assumed that any coalition can be composed by at most b agents. The algorithm consists of two main stages:

- In the first stage, the values of all the possible coalitions are computed. Informally, for all the tasks that a coalition Q can fulfill the value of that task for Q is calculated by subtracting the costs the coalition has to sustain in order to fulfill that task to the benefit associated to that task. The value of Q is the maximum of these values. The calculation of the coalition values is done distributively, where each agent commits itself to calculate the values L_i of a subset of the coalitions in which it is involved.

- The second stage of the algorithm consists of an iterative distributed greedy procedure in which two sub-stages occur:

  - The coalitional values are re-calculated. As different coalitions may be capable to fulfill the same task, when a task is assigned to a coalition Q, it no longer affects the value of other coalitions. Moreover, in the case agents may join only one coalition all the coalition involving some agent in Q cannot be formed anymore, whereas if agents can join more than a single coalition, the residual capability vectors of the agents in Q change affecting the set of tasks that the other coalitions containing some agent in Q can fulfill.
  - The agents decide upon the preferred coalitions and form them. In particular, each agent ag_i proposes the greatest value in L_i. Then, the greatest of the proposed values is chosen and the corresponding coalition Q is formed to achieve the corresponding task. In the case of partitioning of the agents into coalitions, all the agents ag_i delete in their L_i all the values corresponding to a coalition involving an agent in Q. In the case agents may join more than one coalition, the capability vector of the agents in Q have to be updated according to their contribution to the task execution.

  The iteration continues until all the tasks are assigned or no coalition can be formed to achieve any of the remaining tasks.

The algorithm provides a task assignment for a set of coalitions $\mathcal{Q}$ whose ratio bound is given by the formula:

$$\rho = \frac{u^*}{u_{\mathcal{Q}}} = \sum_{i=1}^{M} \frac{1}{i}$$

where u^* correspond to the maximal utility attainable and $M = \max_{Q \in \mathcal{Q}} |Q|$. Moreover it shown that the complexity of the algorithm is $O(n^b|T|^5)$ per agent in the case some precedence order have to be respected, whereas it is $O(n^b|T|)$ in the case no precedence order exists.

The issues addressed in [SLA+99] and [SK98] have been object of further research. For example, in [CKB04b] the authors present an anytime algorithm and show experimentally that it improves the performances with respect to the one presented in [SLA+99]. Kraus et al. [KP00] 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.

2.3 Coalition Formation in Multiagent Systems

Wooldridge et al. [WJ94] 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 [HKT84] in order to express sequences of actions and their effects and provide, as primitive, a predicate $Agts(\alpha, Q)$ with the meaning that Q is the group of agents able to execute the action α. Moreover each agent has its own beliefs about the capabilities of the agents and the effects of performing an action. In this setting the authors represent the fact that an agent recognizes the potential for cooperation in the group of agents Q when it desires a goal g, it is not self-sufficient in the achievement of g and it believes that there exists a plan for g that can be executed by Q.

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 and the common belief that also the other members in Q has the same commitment.

After a team is formed with the purpose to achieve a goal g, the agents of that team negotiate about which plan has to be executed for it. Roughly, the negotiation is modeled as follows: first those proposals that are commonly believed to achieve the goal are selected to be objects of negotiation. Then, the negotiation may proceed with proposal and counter proposals until a plan that is considered admissible by all the agents is found (eventually without reaching an agreement).

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. The formalism adopted to represent joint intentions derives from the work of joint intentionality of Cohen et al [CL90]. In particular the joint intention for a group of agents Q to execute a plan in order to achieve a state of affairs implies that each agent in Q desires to execute that plan α and if it believes that α has been executed or it cannot be executed or the goal cannot be achieved anymore, then it desires to warn all the other agents about these situations.

Sandholm et al. in [SL97] present a model of coalition formation among bounded rational agents, where the representation of the joint capabilities of groups of agents are represented as the characteristic form of a game as in [SLA+99]. This means that a *monetary* value is associated to each group of agents which corresponds to the maximal utility that a group can gain forming a coalition. In the case a group form a coalition they have to agree about how to divide this utility.

Calculating the value of a coalition, i.e. the best way they can coordinate, can be NP-hard, as for example in the distributed vehicle routing problem (see Section 2.1). The bounded rationality of the agents affects the measure of the value of a group of agents by considering that the computation of this value has itself a cost. So, agents want to allocate only a fixed amount of computational resources and, by using it, find an approximation of the value of a coalition. Usually, computational resources consist of CPU cycles, so the authors consider that an anytime algorithm, specific to the particular domain, can be used to find the best value of a coalition until the assigned CPU cycles are finished. Another approach is to consider a design-to-time algorithm, i.e., provided a given amount of CPU cycles, a specific algorithm is designed to find an approximated solution using that amount. Unfortunately the authors do not seem to take into account that also this design problem may have some costs, in particular in term of computational resources.

Once determined the value of each group of agents, the authors use the notion of core (see Definition 34) as a solution criteria to determinate which coalitions can be formed, and how inside each coalition the value is divided among its members.

Zlotkin et al. [ZR89, ZR96a, ZR96b] propose a model of negotiation among two self-interested agents to solve a task allocation problem. In this model each agent ag_i has a set of tasks T_i to be fulfilled and each task can be performed by both agents. A task allocation consists of a partition $\mathcal{T} = (T_1', T_2')$ of $T_1 \cup T_2$, where T_i' corresponds to the tasks allocated to the agent ag_i. Each set of tasks involves a certain cost, so the agents negotiate among the possible allocations. In several cases, the presence of other agents does not lead conflicts or competitions, but it is a chance to reduce the cost relative to the task fulfillments. Consider, for example, a version of

the distributed vehicle routing problem, where each agent corresponds to a single vehicle (postman problem). Two sets of addressed letters, T_1 and T_2, are assigned respectively to the agents ag_1 and ag_2. They have to deliver their letters to corresponding mailboxes and come back to the post office.

If the agent ag_1 was alone, then it would find the route r_1 that fulfills all assigned deliveries and minimizes the transportation cost. If c_1 is the transportation cost of r_1, the presence of another agent ag_2 cannot make worse c_1, as ag_1 could still perform r_1 refusing any kind of collaboration with ag_2. Nevertheless, the two agents can re-assign each other their deliveries in such a manner to decrease respectively c_1 and c_2. Therefore, since there is a chance for the agents to do better acting together, the negotiation process can be viewed as a coalition formation process. The profitability $u_i(\mathcal{T})$ of a task allocation $\mathcal{T} = (T_1', T_2')$ for an agent ag_i, can be defined as the difference between the cost c_i' of best route to fulfill T_i' and c_i [ZR89]. The set NS of task allocations $\{T_1', T_2'\}$ that can be proposed in a negotiation, satisfies the following two properties:

- individual rationality: for each agent, (T_1', T_2') is more profitable of the task allocation (T_1, T_2) corresponding to the case they do not negotiate.
- Pareto optimality: There does not exist a task allocation (T_1'', T_2'') that is better than (T_1', T_2') for at least one agent and no worse for the other.

The negotiation protocol is iterative: at each step both agents simultaneously propose a task allocation in NS. In each step at least an agent have to make a concession, otherwise they reach a conflict. This entails that if $\mathcal{T}_i(t)$ is the task allocation proposed by the agent ag_i in the step t, then $u_i(\mathcal{T}_i(t+1)) \leq u_i(\mathcal{T}_i(t))$. A negotiation can end with a conflict or an agreement. A conflict occurs when both the two agents do not make any concession, i.e. when for all $i \in \{1, 2\}$, $u_i(\mathcal{T}_i(t+1)) = u_i(\mathcal{T}_i(t))$. In this case the agents do not cooperate and their utilities are equal to zero, i.e. they do not gain any advantage with respect to the case they would be alone. An agreement occurs in a certain step t in the case one of the agent makes an offer that is more profitable for the other agent than its own offer. Formally an agreement occurs if and only if there exists $i \neq j$, $u_i(\mathcal{T}_j(t)) \geq u_i(\mathcal{T}_i(t))$.

Nash [Nas50] has proposed as admissible agreements those allocations (T_1', T_2') which maximize the product of the agent utilities. In [ZR89] the authors report a strategy, called the Zeuthen strategy, for the two agents that assures to reach an admissible agreement. The authors consider also a case of incomplete information. In particular, in the postman problem they assume that the road map is common knowledge, therefore the agents know the addresses of the letters and the relative distances, however each of them

may have a partial information about the letters that the other agent has to deliver. This way, an agent can strategically lie about the set of letters it has to deliver. To better describe this possibility we consider a qualitative example. The agent ag_1 has to deliver two letters respectively to the address a and b, whereas the agent ag_2 has to deliver only one letter to the address d. Now the agent ag_1 may tell to the other agent ag_2 that it has to deliver three letters respectively to the addresses a and b and c, where the third letter is a *phantom* letter. Then, ag_1 could propose to deliver the letter of ag_2 to the address d, that is on the c's way, and, in exchange, ag_2 is responsible to deliver the letters a and b. Being the address d on the c's way, it is cheaper to deliver a letter to the address d than to the address c. So, since ag_2 has to deliver only one letter to the address d and then come back to the post-office, it is not individually rational for ag_2 to deliver the phantom letter of ag_1. This means that ag_2 will never counter-propose a task allocation where it is responsible to deliver the phantom letter to the address c and hence the agent ag_1 can safely lie about it. If the addresses a and b are closer for ag_2 than the address c, the task allocation proposed by ag_1 could satisfy the admissibility criterion proposed by Nash and hence the agent ag_2 could agree to it. Nevertheless, ag_1 has to deliver only one letter to the address d, as no letter has actually to be delivered to the address c, therefore its proposal, which is actually *unbalanced* on behalf of ag_1, could be not admissible.

In the case agents may lie about their initial sets of tasks, it is shown that with the negotiation protocol described above it could be beneficial and safe for an agent to lie. Nevertheless, in the case of mixed proposals, i.e. proposals that associate a probability p to a task allocation $\mathcal{T} = (T_1', T_2')$ and a probability $1 - p$ to the task allocation (T_2', T_1') it never beneficial to lie.

Several other works are focussed on coalition formation in Multiagent Systems. Tambe et al. [Tam97] have implemented a system, STEAM, that is based on a model of teamwork closely related to the one shown in [WJ94]. The main difference is in the negotiation process, where they assume the presence of a team leader that establishes the common plan. In [CKB04a], a market-based model is used for a multiagent system, where the agents are heterogenous self-interested traders offering and bidding their resources in the market in order to fulfill their tasks. The work suggests that diversity, in the offered resources, is an important issue and allows the agents to obtain the needed resources by local interactions alone. Wooldridge et al. [WD04, DW04] shows 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 (see Section 4.3), are non computationally tractable. Brainov et al. [BS99] redefine the notion of dependence, as developed by Castelfranchi, for utility-maximizer agents and show some relationship among dependencies and classical game

theoretical notion as Pareto optimality.

2.4 The theory of social power and dependence

As argued by Alonso [Alo98], in the model developed in [WJ94] 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 DPS and MAS 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 DPS. Nevertheless, profitable behaviors arises also in the case the agents exchange services in order to satisfies their own goals, or directly exchange goal commitments. The notion of social exchanges has been introduced in the Multiagent Systems community by Castelfranchi et al. [Cas00, Cas03] and some formalizations of this notion have been presented in several works [dL96, SD01, BSvdT04b].

In this section we introduce the theory of social power and dependence referring in particular to the formalization of Sichman et al. [SD01]. The authors use the theory of social power and dependence to develop a social reasoning mechanism enabling agents to recognize the possibility for helpful social interactions with the other agents. Agents are considered goal-directed and heterogeneous, i.e. each agent may have different goals and capabilities. Agents may cooperate for the achievement of a common goal as well as to exchange services in order to achieve their own goals.

2.4.1 External description

A multiagent system consists of a population of agents where each agent is characterized by a personal description of itself and of the other agents. This description is called *external description* and it represents an agent in terms of its own goals and capabilities. It is assumed that each agent has a complete and correct description of itself (introspection fidelity).

The authors distinguish two types of capabilities that an agent ascribes to itself or to the other agents: the ability to perform a certain action or the possibility to use a certain resource. This distinction is made because the execution of an action is considered more costly than the release of a resource. The release of a resource may let free the agent to engage itself in something else, instead an action takes the agent up in its execution.

Even if in several cases this distinction is valid, we think that it cannot be claimed as a general law. For example, it could be that the resource of an agent consists of a complex device and giving it to another agent involves to explain how to use it, which seems to be an action as it takes the agent up in its execution. Moreover, depending on the domain and the temporal horizon we are considering, some actions can be compatibly performed. For example, reasoning about the social interactions that I can engage along the day, I could compatibly buy a medicine for my mother and help my sister with her homework. Finally, the distinction made between actions and resources seems to not affect remarkably the definitions in [SD01]. For all these reasons we do not propose different types of capabilities, but we only speak about actions.

The external description of an agent consists of: a set of agents Ag populating the multagent system. A set of goals Gl of all the agents. A set of actions Ac the agents can execute. A set of plans Pl. A function $w(g)$ associates each goal with a real value indicating the importance of that goal for ag_i. A function $c(a)$ associates each action a with a real value indicating the cost of that action.

Some predicates establish the capabilities and goals of the agents. These predicates use typed variables to refer to the elements of a given set. ag_i,ag_j and ag_k are variables denoting agents, g, g' denote goals, a is a variable denoting an action and p denotes a plan.

The temporal aspects of a plan, regarding for example the parallel or sequential execution of actions, are not represented. Plans are specified just indicating the actions they need to be executed and the goals they achieve if executed. The predicate $uses_a(p, a)$ indicates that the action a is needed in the execution of the plan p. The goals achieved by a plan are represented by means of the predicate $achieves(p, g)$, whereas the fact that two goals are different is represented by the predicate $diff(g, g')$.

The characteristics of an agent are defined by means of the following predicates. The predicate $is_g(g, ag_i)$ denotes that the agent ag_i desires the goal g. Each agent has different actions and plans that it can use in order to achieve some goals. The predicates $is_a(a, ag_i)$, and $is_p(p, ag_i)$ denote that respectively the action a or the plan p are at the disposal of the agent ag_i.

Definition 1 (External description) *An external description is a tuple:*

$$\langle Ag, Gl, Ac, Pl, w : Gl \rightarrow \mathbb{R}, c : Ac \rightarrow \mathbb{R}, uses_a, achieves, diff, is_g, is_a, is_p \rangle$$

where:

- *Ag is a set of agents.*

- *Gl is a set of goals.*
- *Ac is a set of actions.*
- *Pl is a set of plans.*
- *w is a function associating each goal with a real value indicating the importance of that goal.*
- *c is a function associating each action with a real value inidicating the cost of that action.*
- *$uses_a$ is a predicate whose denotation is a set of pairs $(p, a) \in Pl \times Ac$ indicating that the action a is needed in the execution of the plan p.*
- *achieves is a predicate whose denotation is the set of pairs $(p, g) \in Pl \times Gl$ indicating that the plan p, if executed, achieves the goal g.*
- *diff is a predicate whose denotation is the set of pairs $(g, g') \in Gl \times Gl$ such that $g \neq g'$.*
- *is_g is a predicate whose denotation is the set of pairs $(g, ag_i) \in Gl \times Ag$ indicating that the goal g is desired by the agent ag_i.*
- *is_a and is_p are predicates whose denotations are respectively the sets of pairs $(a, ag_i) \in Ac \times Ag$ and $(p, ag_i) \in Pl \times Ag$ indicating that the action a or the plan p are at the disposal of the agent ag_i.*

2.4.2 The formalization of dependence

Starting from the basic predicates in Definition 1, in this section we see how to formalize the fact that an agent depends on another agent for the achievement of one of its own goals. First of all, we represent the capabilities of the agents as follows. The predicate $is\_plan(ag_i, g, p)$ states that ag_i has at its disposal the plan p in order to achieve the goal g. The predicate $has\_plan(ag_i, g)$ states that there exists a plan p such that $is\_plan(ag_i, g, p)$. $needs_a(ag_i, p, a)$ states that the agent ag_i lacks the action a in order to execute the plan p. $has\_all_a(ag_i, p)$ states that the agent ag_i has at its disposal all the actions required in the execution of the plan p. Finally, $autplan_a(ag_i, g, p, ag_k)$ states that the agent ag_k has at its disposal a plan p to achieve the goal g and all the actions needed to execute p are at the disposal of ag_i.

1. $is\_plan(ag_i, g, p) \Leftrightarrow is_p(ag_i, p) \wedge achieves(p, g)$
2. $has\_plan(ag_i, g) \Leftrightarrow \exists p\ is\_plan(ag_i, g, p)$

3. $needs_a(ag_i, p, a) \Leftrightarrow uses_a(p, a) \wedge \neg is_a(ag_i, a)$

4. $has\_all_a(ag_i, p) \Leftrightarrow \forall a\ (uses_a(p, a) \Rightarrow is_a(ag_i, a))$

5. $aut\_plan_a(ag_i, g, p, ag_k) \Leftrightarrow is\_plan(ag_k, g, p) \wedge has\_all_a(ag_i, p)$

Notice that the predicate $autplan_a(ag_i, g, p, ag_k)$ does not completely formalize that an agent ag_i is autonomous, by means of the plan p, in the achievement of the goal g. Indeed, the predicate says that the agent ag_k is the *owner* of the plan p, and hence, in the case $k \neq i$, the agent ag_i would have all the *physical* capabilities in order to achieve the goal g, but it could not know how to adoperate them. In this case it would be dependent for the achievement of g on the know-how, represented by the plan p, of ag_k.

The previous predicates are the building blocks to define the notions of dependence and autonomy of agents. First of all we define some preliminary notions of dependence. $basic\_dep_a(ag_i, ag_j, g, p, a)$ states that the agent ag_i, in order to achieve the goal g by means of the plan p, depends on the agent ag_j for the action a. $dep\_plan_a(ag_i, ag_j, g, p, ag_k)$ states that the agent ag_k has at its disposal a plan p for the achievement of the goal g such that the agent ag_i to execute this plan depends on the agent ag_j for an action.

6. $basic\_dep_a(ag_i, ag_j, g, p, a) \Leftrightarrow$
   $achieves(p, g) \wedge needs_a(ag_i, p, a) \wedge is_a(ag_j, a)$

7. $dep\_plan_a(ag_i, ag_j, g, p, ag_k) \Leftrightarrow$
   $is_p(ag_k, p) \wedge \exists a\ basic\_dep_a(ag_i, ag_j, g, p, a)$

The notion of dependence of an agent ag_i on another agent ag_j is a subjective notion, i.e. it reflects the point of view of a third agent ag_k according to the plans that are at its disposal. Informally, an agent ag_k considers that an agent ag_i depends on another agent ag_j in the achievement of a goal g if two conditions occur: the first is that the agent ag_i cannot achieve g on its own (or, in other words, it is dependent for g), the second is that ag_k knows a plan p that achieves g and such that ag_i needs the help of ag_j for its execution. The second condition is formalized by means of the predicate $dep\_plan_a(ag_i, ag_j, g, p, ag_k)$ defined in the item 7, the first condition is formalized by the predicate $dep_a(ag_i, g, ag_k)$. An agent ag_i is dependent for a goal g from the point of view of an agent ag_k in the case that:

- ag_i desires the goal g.
- ag_k thinks that g is achievable, i.e. ag_k has at its disposal at least one plan p to achieve g.

- For all the plans p at the disposal of ag_k which achieve g, ag_i does not have all the needed actions to execute p.

The formal definition of $dep_a(ag_i, g, ag_k)$ is the following:

8. $dep_a(ag_i, g, ag_k) \Leftrightarrow$
   $is_g(ag_i, g) \wedge has\_plan(ag_k, g) \wedge \neg\exists p\ aut\_plan_a(ag_i, g, p, ag_k)$

The notion of dependence is, then, formalized as follows:

9. $dep\_on_a(ag_i, ag_j, g, ag_k) \Leftrightarrow$
   $dep_a(i, g, k) \wedge \exists p\ dep\_plan_a(ag_i, ag_j, g, p, ag_k)$

The previous definition of dependence formalizes the potential of a social interaction. Nevertheless, under the hypothesis that agent are self-interested, an agent accepts to help other agent only if it receives some advantages. Thus, a social interaction can occur only in the case of a bilateral dependence. Two types of bilateral dependence are considered, the first is called mutual dependence and the second reciprocal dependence. From the point of view of an agent ag_k, two agents ag_i and ag_j are mutually dependent if there exists a goal g they both desire and there exists a plan p at the disposal of ag_k such that ag_i and ag_j depends on each other for the execution of p. Two agents ag_i and ag_j reciprocally depends on each other from the point of view of the agent ag_k if they depend on each other for the satisfaction of distinct goals g and g'.

10. $MD(ag_i, ag_j, g, ag_k) \Leftrightarrow$
    $dep\_on_a(ag_i, ag_j, g, ag_k) \wedge dep\_on_a(ag_j, ag_i, g, ag_k)$

11. $RD(ag_i, ag_j, g, g', ag_k) \Leftrightarrow$
    $dep\_on_a(ag_i, ag_j, g, ag_k) \wedge dep\_on_a(ag_j, ag_i, g', ag_k) \wedge diff(g, g')$

2.4.3 Social reasoning mechanism

In this section we describe the reasoning mechanism developed in [SD01] that enables agents to use the information about mutual and reciprocal dependencies to socially interact. The first step in the reasoning mechanism is to establish in which relation an agent and a goal are. Given an agent ag_i and a goal g, it could be that: g is not a goal of ag_i, $NG(ag_i, g)$. g is a goal of ag_i, but it has not a plan to achieve it, $NP(ag_i, g)$. g is a goal of ag_i and ag_i is also autonomous in its achievement, $AUT(ag_i, g)$. ag_i desires g and it has also a plan to achieve g, but it is not autonomous in the execution of this plan, $DEP(ag_i, g)$.

1. $NG(ag_i, g) \Leftrightarrow \neg is_g(ag_i, g)$
2. $NP(ag_i, g) \Leftrightarrow is_g(ag_i, g) \wedge \neg has\_plan(ag_i, g)$
3. $AUT(ag_i, g) \Leftrightarrow is_g(ag_i, g) \wedge \exists p\ aut\_plan_a(ag_i, g, p, ag_i)$
4. $DEP(ag_i, g) \Leftrightarrow dep_a(ag_i, g, ag_i)$

In the case $NG(ag_i, g)$ holds, the agent ag_i is not directly motivated in the achievement of the goal g, but it could be the case, as something to be exploited, that g is exchanged with one of its own goal. In the case $NP(ag_i, g)$ holds, then the agent ag_i cannot achieve the goal g but it also does not know how achieve it and hence on whom it might depend. In the case $AUT(ag_i, g)$ holds the agent ag_i is able to achieve the goal g without the help of the other agents. The authors assume in this case that if ag_i intends to achieve g, it will rely on its own, without engaging any social interaction. The fourth case, $DEP(ag_i, g)$, induces ag_i to look for the help of the others and hence to consider which type of dependence situation occurs with the other agents.

Being dependent for the achievement of the goal g, the agent ag_i *calculates* which is the dependence situation with another agent ag_j with respect to g. It is assumed that an agent may use either its own plans or those of the others in order to reason about his autonomy or dependence for a certain goal. Nevertheless to reduce the computational complexity ag_i calculates its dependence situation starting from its own plans. This task is performed by the following three steps.

1. For all $ag_j \neq ag_i$, ag_i verifies if it depends on ag_j using its own plans.
2. When detected a dependence relation on another agent ag_j, ag_i calculates if ag_j depends on ag_i for one of its own goals. As in the step before, ag_i uses only its own plans.
3. If either a mutual or reciprocal dependence is detected in the previous step, ag_i tries to verify whether this same conclusion could be inferred by using the plans of ag_j.

Performed these three steps, six possible situations of dependence on another agent ag_j are considered:
independence: ag_i does not depend on ag_j in the achievement of the goal g, $IND(ag_i, ag_j, g)$.
locally believed mutual dependence: ag_i infers, using its own plan, a mutual dependence with ag_j for g, but it does not deduce the same using the plans of ag_j, $LBMD(ag_i, ag_j, g)$.

mutually believed mutual dependence: ag_i infers a mutual dependence with ag_j for g both using its own plans and the plans of ag_j, $MBMD(ag_i, ag_j, g)$.
locally believed reciprocal dependence: Using its own plan, ag_i infers a reciprocal dependence with ag_j for the goals g and g', but it cannot deduce the same fact using the plans of ag_j, $LBRD(ag_i, ag_j, g, g')$.
mutually believed reciprocal dependence: Using its own plan, ag_i infers a reciprocal dependence with ag_j for the goals g and g', moreover it can deduce the same using the plans of ag_j, $MBRD(ag_i, ag_j, g, g')$.
unilateral dependence: Using its own plan, ag_i infers a dependence on ag_j for the goals g, but, the latter does not depend on it for any of its goals, $UD(ag_i, ag_j, g)$.

These six dependence situations are formalized as follows:

1. $IND(ag_i, ag_j, g) \Leftrightarrow DEP(ag_i, g) \wedge \neg dep\_on_a(ag_i, ag_j, g, ag_i)$
2. $LBMD(ag_i, ag_j, g) \Leftrightarrow MD(ag_i, ag_j, g, ag_i) \wedge \neg MD(ag_i, ag_j, g, ag_j)$
3. $MBMD(ag_i, ag_j, g) \Leftrightarrow MD(ag_i, ag_j, g, ag_i) \wedge MD(ag_i, ag_j, g, ag_j)$
4. $LBRD(ag_i, ag_j, g, g') \Leftrightarrow RD(ag_i, ag_j, g, g', ag_i) \wedge \neg RD(ag_i, ag_j, g, g', ag_j)$
5. $MBRD(ag_i, ag_j, g, g') \Leftrightarrow RD(ag_i, ag_j, g, g', ag_i) \wedge RD(ag_i, ag_j, g, g', ag_j)$
6. $UD(ag_i, ag_j, g) \Leftrightarrow dep\_on_a(ag_i, ag_j, g, ag_i) \wedge \neg \exists g' dep\_on_a(ag_j, ag_i, g, ag_i)$

Once ag_i has calculated its dependence situations with respect to the other agents it can reason about which coalition to form, as we see in the next section.

2.4.4 Coalition formation

In the previous section we defined a social reasoning mechanism enabling an agent ag_i to infer the set of possible social interactions. This mechanism can be used by ag_i to choose a goal to achieve and a plan to execute.

First of all, an agent has to restrict its decision process only to the goals that are achievable. A goal g is achievable if there exists at least one plan p that achieves g and for all the actions involved in p there exists at least one agent having that action at its disposal. Thus, the notion of achievable goal is formalized by the following predicates:

1. $available_a(a) \Leftrightarrow \exists ag_i \ \ is_a(ag_i, a)$
2. $feasible_a(p) \Leftrightarrow \forall a \ (uses_a(p, a) \Rightarrow available_a(a))$

3. $achievable(g,p) \Leftrightarrow achieves(p,g) \wedge feasible_a(p)$

It is assumed that an agent ag_i relies only on its own plans to verify if a desired goal is achievable. Formally, ag_i infers that the goal g is achievable in the following way:

4. $achievable(ag_i,g) \Leftrightarrow$
$$AUT(ag_i,g) \vee (DEP(ag_i,g) \wedge \exists p\,(is_p(ag_i,p) \wedge achievable(g,p))$$

When the agent ag_i derives which of the desired goals are achievable, it decides to pursue the goal g that maximizes the importance function w. Decided which goal g to pursue, ag_i has to choose, when more than a single plan achieves g, which plan to use. In this case it chooses the plan, among those which are feasible, with the minimum cost. In a first approximation the cost of a plan is supposed to be the sum of the costs of all the actions needed for its execution.

$$c(p) = \sum_{a \in \{a | uses_a(p,a)\}} c_a(a)$$

Chosen which plan p to execute, ag_i could be not self-sufficient in its execution. In this case ag_i individuates, by means of the predicate $basic\_dep_a$, the missing actions and the agents that can provide them. If more than one agent can provide the same action, then ag_i has to choose to whom of them it should propose a coalition formation. By calculating its dependence situation with respect to the potential partners, two criteria has been adopted for the partner selection.

The first criterion distinguishes locally believed from mutually believed dependencies. Mutually believed dependence situations are preferred to locally believed situations because the proponent will not need to convince the addressee of their dependence situation. If we associate a cost function to this communication, mutually believed dependencies (*MBRD* and *MBMD*) are cheaper and hence preferred to locally believed dependencies (*LBRD* and *LBMD*).

The second criterion distinguishes reciprocal from mutual dependencies. Mutual dependencies are preferred to reciprocal dependencies because sharing the proponent and the addressee the same goal, each of them should not fear an absence of reciprocation from the other one. Combining the two criteria we obtain a partial preference relation such that *MBMD* is preferred to *LBMD* and to *MBRD*, and these last two dependence situations are preferred to *LBRD*.

2.5 Coalition formation with generalized dependencies

In this section we consider an extension, due to Conte and Sichman [CS02b, CS02a], of the social reasoning mechanism formalized in [SD01] addressing the problem of generalized social exchanges [YC93] as possible interactions. The social exchanges considered in [SD01] 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.

To take into account generalized social exchanges in the social reasoning mechanism the authors make two relevant assumptions:

- A single perspective of the multiagent system is considered. In [SD01] each agent had a data structure, called external description, representing beliefs about goals, plans, capabilities and resources of itself and the other agents. In [CS02b] only one external description of the multiagent system is considered corresponding to the external description of a single agent, the observer.

- Each agent is not autonomous only for one goal and there exists a single plan that achieves that goal. This way, an agent does not have to address the problem to choose among several goals the most profitable one.

2.5.1 Graph representation

Since we are assuming a single external description of a multiagent system, the dependencies among agents can be formalized by means of the notion of $basic\_dep_a$ defined in Section 2.4.2. Moreover, as we have assumed that for each goal g, there exists only a plan p achieving it, the relation $basic\_dep_a$ does not need in this context to explicitly refer to a particular plan.

In this section we represent the set of basic dependencies among the agents as a tagged directed graph DG. The set of nodes in DG is composed by the set of agents Ag and the set of actions Ac. The set of arcs E consists of two types: the first type of arcs is used to represent the actions that an agent needs in order to achieve a goal (under the assumption that there exists only a plan p achieving it). An arc of this type consists of tagged directed arc connecting an agent node to an action node, where the tag corresponds to a goal of the agent. The second type of arcs represents the agents that can execute an action, therefore it consists of directed arcs connecting the action nodes to the agent nodes. Since $(ag_i, a, g) \in E$ implies that the agent ag_i lacks the action a for the goal g, it cannot be the case that (a, ag_i) is in E.

Definition 2 (Dependence Graph) [1] *A Dependence Graph DG is a tuple* $\langle Ag, Ac, Gl, E \subseteq [Ag \times Ac \times Gl] \cup [Ac \times Ag] \rangle$, *where Ag is a set of agents* ag_i, *Ac is the set of actions* a_i *the agents can execute, Gl is the set of goals g desired by the agents. E is a set of tagged arcs such that if* $(ag_i, a, g) \in E$, *then* $(a, ag_i) \notin E$.

An arc $(ag_i, a, g) \in E$ *represents that g is a goal of* ag_i, *a is needed in the execution of the plan achieving g and* ag_i *is not able to execute a.* $(a, ag_j) \in E$ *represents the case* ag_j *is able to execute the action a.*

Given a Dependence Graph a basic dependence corresponds to a path from an agent ag_i to an agent ag_j.

Definition 3 (Basic dependence) *Let* $DG = \langle Ag, Ac, Gl, E \rangle$ *be a Dependence Graph, an agent* ag_i *basic depends on an agent* ag_j *for the action a, needed to achieve the goal g,* $basic\_dep(ag_i, ag_j, g, a)$, *iff* $(ag_i, a, g) \in E$ *and* $(a, ag_j) \in E$.

An example of basic dependence is shown in Figure 2.1 (a), in particular $basic\_dep(ag_1, ag_2, g, a)$.

Given an arc $(ag_i, a, g) \in E$, it could be the case that more than one agent is able to execute the action a. If Q is a group of agents that can execute the action a we say that the agent ag_i OR-depends on Q. OR-dependencies describe the situation in which an agent can rely on more than a single agent for the execution of a needed action.

[1] In [CS02b] the authors provide a definition of Dependence Graph in the general case where more than one plan can be used to achieve a goal. Nevertheless, the assumptions in Section 2.5 are crucial for the definition of generalized dependencies that will be defined in Section 2.5.2, therefore we prefer to directly define Dependence Graph under these assumptions.

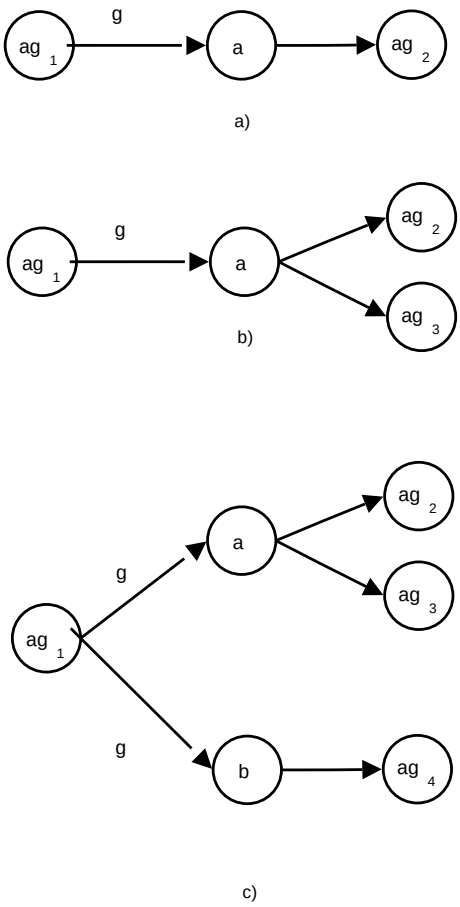

Figure 2.1: *Graph representation of the basic dependence, OR-dependence and AND-dependence relations.*

Definition 4 (OR-dependence) *Given a Dependence Graph DG, an agent ag_i OR-depends on the group of agents Q for the action a needed for the goal g, $OR\_dep(ag_i, Q, g, a)$, iff $|Q| > 1$ and*[2]

$$Q \subseteq \{ag_j \in Ag \mid basic\_dep(ag_i, ag_j, g, a)\}$$

In Figure 2.1 (b) we show an example of OR-dependence, in particular it holds that $OR\_dep(ag_1, \{ag_2, ag_3\}, g, a)$.

In general it could be the case that an agent needs more than an action to achieve a goal. This means that it needs the help of several agents or of the

[2]In [CS02b] the authors have defined $Q = \{ag_j \in Ag \mid basic\_dep(ag_i, ag_j, g, a)\}$, however, in order to study the relation between OR-dependence and AMONG-dependence in the next section, we guess that this definition is more simple.

same agent but for several actions, the authors call this situation a case of AND-dependence. Since for each needed action a there could be more than one agent able to execute it, the AND-dependence relation considers the set of all agents that can execute a.

Definition 5 (AND-dependence) *Given a Dependence Graph DG, an agent ag_i AND-depends on the groups of agents $\mathbf{Q} \subseteq [2^{Ag} \setminus \emptyset]$ for the set of actions $A \subseteq Ac$ needed to achieve its goal g, $AND\_dep(ag_i, \mathbf{Q}, g, A)$, iff $|A| > 1$ and the followings hold:*

1. $A = \{a \in Ac \mid (ag_i, a, g) \in E\}$
2. $Q \in \mathbf{Q}$ *iff there exists an action* $a \in A$ *such that* $Q \subseteq \{ag_j \in Ag \mid (a, ag_j) \in E\}$

An example of AND-dependence is shown in Figure 2.1 (c), in particular $AND\_dep(ag_1, \{\{ag_2, ag_3\}, \{ag_4\}\}, g, \{a, b\})$.

2.5.2 AMONG and GROUP dependencies

Once defined the notions of OR-dependence and AND-dependence the authors in [CS02b] consider a generalized notion of reciprocal dependence, the AMONG-dependence. AMONG-dependence formalizes, under the assumption that each agent has only one goal, the notion of reciprocity that we will call in Chapter 3 the do-ut-des property. More precisely an AMONG-dependence exists for a group of agents Q when each $ag_i \in Q$ basic depends on at lest another agent $ag_j \in Q$, and there exists al least an agent $ag_h (\neq ag_j) \in Q$ that basic depends on ag_i.

Definition 6 *Let DG be a Dependence Graph, an AMONG-dependence exists for a group of agents Q iff for all $ag_i \in Q$, the following holds:*

1. *there exists at least an agent $ag_j \in Q$, an action a and a goal g such that $basic\_dep(ag_i, ag_j, g, a)$.*
2. *there exists at least an agent $ag_h (\neq ag_j) \in Q$, a goal g' and an action a' such that $basic\_dep(ag_h, ag_i, g', a')$*

A particular case of AMONG-dependence is given by a cycle of dependencies as in Figure 2.2 (a). This corresponds to the more intuitive generalization of the dyadic reciprocal dependence. However, more complex situations may occur, and some of them require some further comments.

It could be the case that Q is a group of AMONG-depending agents and an agent $ag_i \in Q$ may OR-depend on a group of agents $Q' \subseteq Q$ (see

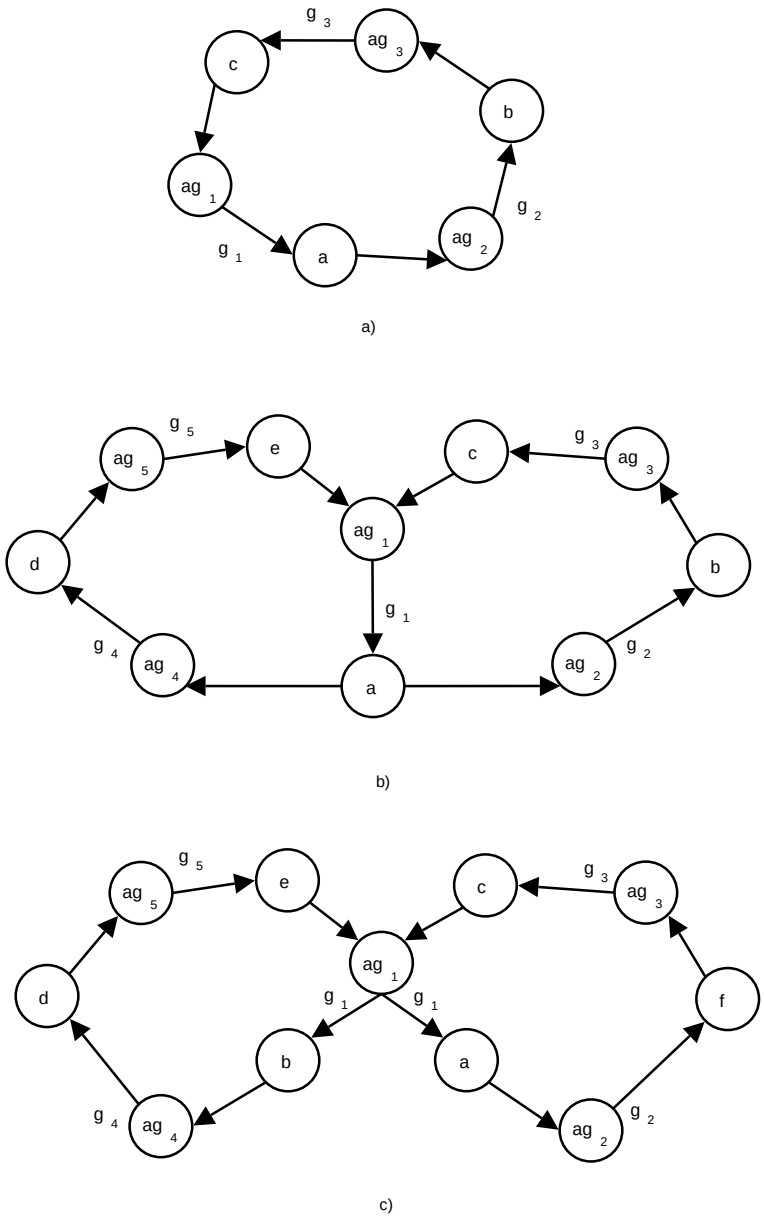

Figure 2.2: *Examples of AMONG-dependence: a cycle (a), including an OR-dependence (b), including an AND-dependence (c).*

Figure 2.2 (b)), in this case the AMONG-dependence for Q is composed by several AMONG-dependencies each of them involving only one agent in Q'. In Figure 2.2 (b) the agent ag_1 OR-depends on the agents ag_2 and ag_4, $OR\_dep(ag_1, \{ag_2, ag_4\}, g_1, a)$. Thus, the whole AMONG-dependence consists of two AMONG-dependencies, the first involving the group of agents ag_1, ag_2 and ag_3 and the second involving the group of agents ag_1, ag_4 and ag_5. Furthermore, in the process of coalition formation, it is unnecessary that the agent ag_2 and the agent ag_4 contribute the same action, therefore only one of the two AMONG-dependencies $\{ag_1, ag_2, ag_3\}$ and $\{ag_1, ag_4, ag_5\}$ will actually be formed.

Another possibility is that Q is a group of AMONG-depending agents and $ag_i \in Q$ AND-depends on a group of agents $Q' \subseteq Q$. In Figure 2.2 (c), for example, the agent ag_1 AND-depends on both the agents ag_3 and ag_4, $AND\_dep(ag_1, \{\{ag_2\}, \{ag_4\}\}, g_1, \{a, b\})$. In this case both the groups of agents $\{ag_1, ag_2, ag_3\}$ and $\{ag_1, ag_4, ag_5\}$ are AMONG-depending. In contrast with the case of OR-dependence, these two AMONG-dependencies are not only compatible with each other but, since the agent ag_1 needs both the actions a and b for the achievement of its goal, if a coalition is formed, then it involves both the groups $\{ag_1, ag_2, ag_3\}$ and $\{ag_1, ag_4, ag_5\}$.

2.6 Alternating-time Temporal Logic

The theory of social power and dependence, as formalized in Section 2.4, enables to reason about coordination among agents, and in particular to represent which agents can achieve a given goal. However, the expressivity of this formalism is limited for two reasons. First, plans do not have any particular structure but they are represented at a high level of abstraction as sets of actions. Secondly, plans specify only that a certain action has to be executed, but they do not consider the possibility that an action does not have to be executed. So, agents have only the possibility to help each other, but not to obstruct.

In recent years several logic-based formalisms have been developed in order to represent some particular notions of power at different level of abstractions. Pauly [Pau02a] provides a modal logic, called Coalition Logic, that axiomatizes a notion of power developed in Game Theory, the α-ability (see Section 4.2). Informally, a group of agents Q is α-able for a propositional formula ϕ if and only if there exists a joint strategy of Q which, no matter what the other agents do, assures that the multiagent system results in a state that satisfies ϕ. Another example is CL-PC [vdHW05]. In CL-PC each agent can decide the truth value of a set of boolean variables under

its control. This way, the notions of α-ability can be formalized as a standard multi-modal Kripke model and the usual semantic for box and diamond operators can be used.

Alternating-time Temporal Logic [AHK02, GvDar, vdHW03], 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) [McM92] for property verification of modular reactive systems [AHM+98]. In particular with respect to the first aim, ATL gives the possibility to formalize two different notion of power developed in Game Theory, the α-ability and β-ability.

2.6.1 ATL models, syntax and semantics

A model in ATL describes the concurrent moves that agents can execute in a game. A game is described by means of the number of agents k, a set S of states, set Π of propositional variables and a labeling function mapping each state s to the set of propositional variables true in that state. In each state an agent has a set of possible moves at his disposal. A function δ describes the transitions of the game according to the agents' moves.

Definition 7 (Concurrent game structure) *A concurrent game structure is a tuple $GS = \langle k, S, \Pi, \pi, d, \delta \rangle$ with the following components.*

- *A set of $k \geq 1$ agents. We identify the agents with $1, \ldots, k$.*
- *A finite set S of states.*
- *A finite set Π of propositions.*
- *$\pi : S \rightarrow 2^{\Pi}$ is a function associating each state s with the propositions true in s.*
- *A strictly positive valued function $d : \{1, \ldots, k\} \times S \rightarrow \mathbb{N}$ describing the number of moves available to an agent in a given state. We identify the moves of an agent $1 \leq ag \leq k$ at state s with the numbers $1, \ldots, d(ag, s)$. For each state s a move vector is a tuple $\langle j_1, \ldots, j_k \rangle$ such that for each agent $1 \leq ag \leq k$, $1 \leq j_{ag} \leq d(ag, s)$. Given a state s, $D(s)$ denotes the set $\{1, \ldots, d(1, s)\} \times \ldots \times \{1, \ldots, d(k, s)\}$ of move vectors.*

- *For all the states $s \in S$ and for all the move vectors $\langle j_1, \ldots, j_k \rangle \in D(s)$, $\delta(s, j_1, \ldots, j_k) \in S$ is the state resulting from the state s and the move vector $\langle j_1, \ldots, j_k \rangle$. The function δ is called transition function.*

Alternating-time Temporal Logic is designed to express the group capabilities to satisfy certain properties on possible traces of a concurrent game. Similar to CTL, these properties are expressed by means path formulas $\bigcirc\phi$, $\Box\phi$ and $\phi_1 \mathcal{U} \phi$, having respectively the meaning of "in the next time ϕ", "always in the future ϕ" and "ϕ_1 until ϕ_2". With respect to CTL, that can express only eventuality or the necessity of path formulas, in ATL a path formula is preceded by a group operators $\langle\langle Q \rangle\rangle$, with Q a group of agents, with the meaning "Q can see to it that ϕ".

Definition 8 (ATL Syntax) *Given a concurrent game structure $GS = \langle k, Q, \Pi, \pi, d, \delta \rangle$, the set of ATL formulas is defined by the following grammar:*

$$p \mid \neg\phi \mid \phi_1 \vee \phi_2 \mid \langle\langle Q \rangle\rangle \bigcirc \phi \mid \langle\langle Q \rangle\rangle \Box\phi \mid \langle\langle Q \rangle\rangle \phi_1 \mathcal{U} \phi_2$$

where $p \in \Pi$, $Q \subseteq \{1, \ldots, k\}$ and ϕ, ϕ_1 and ϕ_2 are ATL formulas.

Notation *Given an ATL formula ϕ, we write $\langle\langle Q \rangle\rangle \Diamond \phi$ for $\langle\langle Q \rangle\rangle \top \mathcal{U} \phi$ and $[\![Q]\!]\phi$ for $\neg\langle\langle Q \rangle\rangle \neg\phi$.*

A strategy of an agent ag is formalized in ATL as a function f_{ag} that maps each finite sequence $s_0, \ldots, s_h$ of states in a move that is feasible for ag in the state s_h. This definition of a strategy reflects the assumption that the agents can observe the evolution of the game state after state with perfect information and total recall. More restrictive assumptions can be formalized enforcing some invariance conditions on the function f_{ag}. For example blind strategies are formalized considering that f_{ag} depends for each sequence of states $s_0, \ldots, s_h$ only on s_0 and h, whereas bounded recall is described by a strategy f_{ag} which depends only on the last b states of $s_0, \ldots, s_h$. A strategy profile of a group Q is defined as a function that associates for each member ag of Q a strategy f_{ag}.

Definition 9 (Strategies) *We denote with S^+ the set of all nonempty finite state sequences. A strategy for an agent ag is the function f_{ag} that maps each sequence of states $\lambda = s_0, \ldots, s_h \in S^+$ to a natural number $f_{ag}(\lambda)$ such that $f_{ag}(\lambda) \leq d(ag, s_h)$. Given a group of agents $Q \subseteq \{1, \ldots, k\}$, a strategy profile F_Q is a function that maps each player $ag \in Q$ to one of its strategies f_{ag}.*

An infinite sequence of states is called an output. If a group of agents adopt a strategy profile F_Q then, the space of possible outputs of a game are restricted to those determined by F_Q. In the following definition we formalize the outputs determined by F_Q.

Definition 10 (Outputs) *We denote with S^ω the set of all infinite state sequences and, given $\lambda = s_0, s_1, s_2, \ldots \in S^\omega$, $\lambda[i]$ is the element in the position i in λ, whereas $\lambda[i, j]$, with $i \leq j$, is the subsequence of λ from the element in the position i up to the element in the position j. Given a state s and a strategy profile F_Q, the output of F_Q in the state s, $out(s, F_Q)$, is the set of state sequences $\lambda = s_0, s_1, s_2, \ldots \in S^*$ such that $s_0 = s$ and for all $i \geq 0$ there exists a move vector $\langle j_1, \ldots, j_k \rangle \in D(s_i)$ such that*

- *For all $ag \in Q$, $j_{ag} = [F_Q(ag)](\lambda[0, i])$.*
- *$\delta(s_i, j_1, \ldots, j_k) = s_{i+1}$.*

An ATL formula $\langle\langle Q \rangle\rangle \phi$ means that there exists a strategy profile F_Q such that for all outputs λ of F_Q, λ satisfies the formula ϕ. Therefore, the group operator $\langle\langle\rangle\rangle$ formalizes the so-called α-ability, see Definition 30, i.e. the group of agents Q can see to the satisfaction of the formula ϕ independently from which strategy profile the other agents choose. Consequently, the ATL formula $[\![\{1, \ldots, k\} \setminus Q]\!] \phi$ tells us that the group of agents that are not in Q do not have a strategy profile that assures $\neg\phi$ independently from which strategy profile Q choose. This means that for all the strategy profiles $F_{\{1,\ldots,k\}\setminus Q}$, there exists a strategy profile F_Q such that ϕ is satisfied, this condition is called β-ability.

Definition 11 (ATL semantics) *Given concurrent game structure $GS = \langle k, S, \Pi, \pi, d, \delta \rangle$, a state $s \in S$ and an ATL formula ϕ, we write $GS, s \models \phi$ to indicate that GS satisfies the formula ϕ in the state s. The satisfaction relation is defined inductively as follows:*

- *$GS, s \models p$, for $p \in \Pi$, iff $p \in \pi(s)$.*
- *$GS, s \models \neg\phi$ iff $GS, s \not\models \phi$.*
- *$GS, s \models \phi_1 \vee \phi_2$ iff $GS, s \models \phi_1$ or $GS, s \models \phi_2$.*
- *$GS, s \models \langle\langle Q \rangle\rangle \bigcirc \phi$ iff there exists a strategy profile F_Q such that for all $\lambda \in out(s, F_Q)$, $GS, \lambda[1] \models \phi$.*
- *$GS, s \models \langle\langle Q \rangle\rangle \Box \phi$ iff there exists a group strategy F_Q such that for all $\lambda \in out(s, F_Q)$ and $i \geq 0$, $GS, \lambda[i] \models \phi$.*

- $GS, s \models \langle\langle Q \rangle\rangle \phi_1 \mathcal{U} \phi_2$ *iff there exists a group strategy* F_Q *such that for all* $\lambda \in out(s, F_Q)$*, there exists a position* $i \geq 0$ *such that* $GS, \lambda[i] \models \phi_2$ *and for all* $0 \leq j < i$*,* $GS, \lambda[j] \models \phi_1$*.*

It can be seen that $GS, s \models \langle\langle Q \rangle\rangle \phi$ implies $GS, s \models [\![\{1, \ldots, k\} \setminus Q]\!] \phi$. So, the notion of α-ability is more restrictive that the notion of β-ability. We also notice that the CTL quantifiers *eventually* E and *always* A correspond respectively the operators $\langle\langle \{1, \ldots, k\} \rangle\rangle$ and $\langle\langle \emptyset \rangle\rangle$. In [AHK02] also the model checking problem is studied and it is found that it can be solved in time $O(m \cdot l)$ for a concurrent game structure with m transitions and an ATL formula of length l.

2.6.2 Axiomatization of ATL

In [GvDar] a sound and complete axiomatization is provided of the ATL semantic. This axiomatization is provided in the following definition.

Definition 12 (ATL axiomatic system) *The axiomatic system for ATL consists of the following rules of inference and axioms, where* Q, Q_1, Q_2 *are groups of agents and* ϕ*,*ψ *and* τ *are ATL formulas.*
Rules of inference:

Modus ponens: ϕ *and* $\phi \rightarrow \psi$ *entails* ψ*.*

$\langle\langle Q \rangle\rangle \bigcirc$**-monotonicity:** $\phi \rightarrow \psi$ *entails* $\langle\langle Q \rangle\rangle \bigcirc \phi \rightarrow \langle\langle Q \rangle\rangle \bigcirc \psi$

$\langle\langle \emptyset \rangle\rangle \Box$**-necessitation:** ϕ *entails* $\langle\langle \emptyset \rangle\rangle \Box \phi$

Axioms:

1. ϕ*, where* ϕ *is any propositional tautology.*
2. $\neg \langle\langle Q \rangle\rangle \bigcirc \bot$
3. $\langle\langle Q \rangle\rangle \bigcirc \top$
4. $\neg \langle\langle \emptyset \rangle\rangle \bigcirc \neg \phi \rightarrow \langle\langle Ag \rangle\rangle \bigcirc \phi$
5. $\langle\langle Q_1 \rangle\rangle \bigcirc \phi \wedge \langle\langle Q_2 \rangle\rangle \bigcirc \psi \rightarrow \langle\langle Q_1 \cup Q_2 \rangle\rangle \bigcirc (\phi \wedge \psi)$*, where* $Q_1 \cap Q_2 = \emptyset$*.*
6. $\langle\langle Q \rangle\rangle \Box \phi \leftrightarrow \phi \wedge \langle\langle Q \rangle\rangle \bigcirc \langle\langle Q \rangle\rangle \Box \phi$
7. $\langle\langle \emptyset \rangle\rangle (\psi \rightarrow (\phi \wedge \langle\langle Q \rangle\rangle \bigcirc \psi)) \rightarrow \langle\langle \emptyset \rangle\rangle (\psi \rightarrow \langle\langle Q \rangle\rangle \bigcirc \phi)$
8. $\langle\langle Q \rangle\rangle \phi \mathcal{U} \psi \leftrightarrow \psi \vee (\phi \wedge \langle\langle Q \rangle\rangle \bigcirc \langle\langle Q \rangle\rangle \phi \mathcal{U} \psi)$

9. $\langle\langle\emptyset\rangle\rangle\Box((\psi \vee (\phi \wedge \langle\langle Q\rangle\rangle \bigcirc \tau)) \rightarrow \tau) \rightarrow \langle\langle\emptyset\rangle\rangle\Box(\langle\langle Q\rangle\rangle\phi\mathcal{U}\psi \rightarrow \tau)$

From the axiomatization in Definition 12 it derives that the $\langle\langle\rangle\rangle$ operator satisfies some properties already studied when the notion of α-ability has been introduced in Cooperative Game Theory. In particular, these properties are super-additivity, coalition monotonicity and outcome monotonicity:
super-additivity: If two disjoint groups of agents Q_1 and Q_2 can see respectively to the formulas ϕ and ψ, then their union can see to $\phi \wedge \psi$. Formally, $\vdash (\langle\langle Q_1\rangle\rangle\phi \wedge \langle\langle Q_2\rangle\rangle\psi) \rightarrow \langle\langle Q_1 \cup Q_2\rangle\rangle\phi \wedge \psi$, where ϕ and ψ are ATL formulas and $Q_1, Q_2 \subseteq Ag$ are such that $Q_1 \cap Q_2 = \emptyset$.
coalition monotonicity: If a group of agents Q can see to ϕ, then all the groups containing Q can see to ϕ. Formally, $\vdash \langle\langle Q\rangle\rangle\phi \rightarrow \langle\langle Q'\rangle\rangle\phi$, where ϕ is an ATL formulas and $Q \subseteq Q' \subseteq Ag$.
outcome monotonicity: If a group of agents can see to the formula ϕ, then it can see to any logical consequence of ϕ. Formally, if $\vdash \phi \rightarrow \psi$, then $\vdash \langle\langle Q\rangle\rangle\phi \rightarrow \langle\langle Q\rangle\rangle\psi$, where ϕ and ψ are ATL formulas and $Q \subseteq Ag$.

It can be shown that the previous conditions are not independent with each other, in particular is outcome monotonicity and super-additivity together implies coalitional monotonicity.

2.7 Summarizing

In this chapter we have described some relevant works focussed on Coalition Formation in the field of Distributed Artificial Intelligence. The first three sections have been devoted to provide a general overview on this research area.

Section 2.4 and Section 2.5 describe the theory of social power and dependence. In particular we followed the works of Sichman et al. [SD01] and Conte et al. [CS02b] as they stress the relevance of this theory in the area of Coalition Formation. The theory of social power and dependence has been the starting point in the definition of our admissibility criteria.

In Section 2.6, we have described a logic-based formalism, ATL [AHK02, GvDar], to reason about group behaviors and the effects that they can assure. This logic falls in an area of modal logics [Pau02a, vdHW05, vdHW03], which formalize, at different level of abstraction and with different expressive power, how agents can collaborate in order to force a certain state of affairs. We have chosen to focus on ATL as it is one of the more expressive and, perhaps, the most popular. In Chapter 5, we also develop a logic on the same issue, so we will use ATL for comparisons.

Finally we notice that coalition formation processes and in particular admissibility criteria has been the research issue of Cooperative Game Theory.

As Chapter 4 is focussed on the relationship between our approach and Cooperative Game Theory, we preferred to report there a brief overview on it.

Chapter 3

Coalitions and admissibility criteria

3.1 Introduction

This chapter is devoted to the formalization of two qualitative criteria of admissibility in coalition formation processes, the do-ut-des property and the composition property. The do-ut-des property describes a condition of reciprocity: an agent *gives* a goal only if this fact enables it to *obtain*, directly or indirectly, the satisfaction of one of its own goals. The composition property is a refinement of the do-ut-des property which takes into account the fact that a coalition formation process can itself be costly and, as we consider unanimous agreements, the more agents are involved in a coalition, the larger is the risk that one of them decides to not join the coalition. Therefore, agents prefer to form a coalition that is as small as possible.

First of all, we need to represent a multiagent system in a suitable way, i.e. without describing details which are not relevant for the definition of our criteria and, furthermore, could restrict their applicability. In Section 2.5 we have shown how Conte et al. [CS02b] have used a graph representation to describe both the actions that an agent needs in order to achieve a goal and the agents that can provide it with the execution of these actions. These graphs, called Dependence Graphs, enable to represent the inter-dependencies occurring among agents and define some admissibility criteria to select some of all the possible generalized exchanges. In particular, the notion of AMONG-dependence describes a condition of reciprocity under the assumption that each agent involved in an exchange obtains from it the satisfaction of only a single goal.

With respect to the approach followed in [CS02b], our approach differs in two aspects. The first aspect regards at which level of abstraction a multiagent system has to be described. The notion of reciprocity defined in [CS02b] does not take into account which particular actions are involved in a generalized exchange, but only the goals that each agent obtains and provides. Also in Cooperative Game Theory, which is focussed on coalition formation processes and their admissibility, 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 group of agents can assure collaborating, without any description about how they can assure them. As we will see, also for our admissibility criteria we do not need to describe actions, plans, resources or strategies.

The second aspect from which we differ is more foundational and we have already introduced it in Section 1.3. The notion of reciprocity in [CS02b] 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, as already discussed in Section 1.3, in some case 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 Section 3.2 we formalize the Power Structures and provide several examples which motivates our formalization with respect to others. In Section 3.3 we describe some relevant properties that, with respect to different notion of power, the Power Structures may satisfy and we show their relationships. Starting from the representation of a multiagent system as a Power Structure, in Section 3.4 we define the set of the possible coalitions that can be formed, we call these potential coalitions Power Frames. In Section 3.5 we formalize our first admissibility criterion, the do-ut-des property, and we show some examples and conditions which emphasize its difference from the game theoretical admissibility criteria. In Section 3.6 we show further conditions which may be relevant, especially when it holds that the more are the goals that a group of agents have to achieve, the more are the costs it has to sustain. In Section 3.7 we introduce a particular class of Power Frames, the singleton Power Frames. Singleton Power Frames roughly corresponds to the generalized exchanges studied in [CS02b] and we characterize the do-ut-des property for this class of Power Frames. In Section 3.8 we define the second admissibility criterion, the composition property, and we show with some examples its differences from the do-ut-des property. As done for the do-ut-des property, in Section 3.9 we characterize the composition property for singleton Power Frames. Finally, in Section 3.10, we describe an algorithm to find the Power Frames which satisfy the composition property and we discuss its computational complexity.

3.2 Power Structures: abstracting goal-directed group behaviors

A multiagent system, as we are concerned, can be modeled as follows: first you define the environment in which the agents operate and what an agent

is in terms, for example, of the actions it can execute or the goals it desires to achieve. Then, a multiagent system consists of a collection of agents populating the same environment. Compared to this kind of representation, we consider here a more abstract conceptualization of a multiagent system: the Power Structures.

Power Structures do not describe the particular characteristics of the single agents, they directly characterize which groups of agents, collaborating, have the power to achieve which goals without representing how they can achieve that goals.

A Power Structure can describe the mere capabilities of a group of agents Q to achieve one or more goals simultaneously. In this case Q has the power to achieve a set of goals G if it can coordinate in such a manner to assure the achievement of all the goals in G. However, depending on the application domain, the expression "the group of agents Q can assure the achievement of all the goals in G" may have different meanings. For example, as seen in Section 2.6, it can mean that the agents in Q have a joint plan which achieves all the goals in G, no matter what the other agents do. Another interpretation is that the joint plan has a probability to achieve all the goals in G that is greater than, for example, 0.8. Other meanings may not involve the notion of plan at all, for example, you can assume that the group of agents Q can achieve all the goals in G by means of reactive coordination [Fer99] or due to the fact that in the past Q already did it once, several times, or the most of the times. Finally the agents in Q can assure the achievement of all the goals in G due to the roles or the jobs they have in a society, for example, a surgery team can operate an appendicitis.

Moreover, at least in problems of coalition formation, a better description of the reliability for the achievement of a set of goals may lead to consider further conditions than the mere capabilities of groups of agents. So, in some cases it could be required that all the agents in Q are necessary or at least useful for the satisfaction of the goals in G, or it could be useful to distinguish if a group of agents can achieve precisely a set of goals G or at least a set of goals G.

If a group of agents Q has the power to achieve a set of goals, then these goals can be committed to it in an agreement. Thus, we call a potential commitment, or simply a *commitment*, a pair consisting of a group of agents and a set of goals it has the power to achieve.[1] A Power Structure abstracts

[1]Ferber in [Fer99] describes a more general notion of commitment and in particular he formalized a commitment as a triple describing not only *who* commits itself to do *what*, but also *to whom* this commitment is addressed. In this context we do not feel the need to explicitly take this issue into account. As we do not consider the hypothesis that a group of agents do not respect a commitment, we do not need to represent who would complain

from which particular notion of power has been employed to establish the set of possible commitments. As we will see in Definition 13, a Power Structure simply describes this set.

A Power Structure is composed of a set of agents Ag, a set of goals Gl, a function *goals* that maps each agent to the subset of Gl representing its own goals, a relation $pow \subseteq 2^{Ag} \times 2^{Gl}$ that associates a group of agents Q with the sets of goals they have the power to achieve and a subset *comp* of 2^{pow} that describes which sets of commitments are compatible, i.e., considered the commitments $(Q_1, G_1), \ldots, (Q_n, G_n) \in pow$, each group of agents Q_i can achieve respectively the set of goals G_i without rising conflicts if and only if $\{(Q_1, G_1), \ldots, (Q_n, G_n)\} \in comp$. *comp* satisfies the properties that the empty set is in *comp* and each commitment is compatible with itself. It is also assumed that if a set of commitments P does not arise conflicts, then each subset P' of P does not arise conflicts as well. Finally, since $(Q, G) \in pow$ involves that Q can achieve all the goals in G, and the other agents cannot obstruct it, *comp* satisfies the property that if $Q_1, \ldots, Q_n$ are all mutually disjoint, then they can compatibly achieve respectively $G_1, \ldots, G_n$.

Definition 13 (Power Structure) *A Power Structure, PS, is a tuple*

$$\langle Ag, Gl, goals\colon Ag \rightarrow 2^{Gl}, pow \subseteq 2^{Ag} \times 2^{Gl}, comp \subseteq 2^{pow} \rangle$$

where Ag is a set of agents. Gl is a set of goals. goals is a function that maps each agent in Ag to the set of goals it desires to achieve. pow is a relation that associates each group of agents with the sets of goals that the group has the power to achieve. comp is a subset of 2^{pow} *such that* $(Q_1, G_1), \ldots, (Q_n, G_n) \in$ *pow can be compatibly committed just in the case* $\{(Q_1, G_1), \ldots, (Q_n, G_n)\} \in$ *comp. The relations pow and comp satisfy the following conditions:*

1. $\emptyset \in comp$
2. *for all* $(Q, G) \in pow$, $\{(Q, G)\} \in comp$
3. *if* $P \in comp$, *then for all subsets* $P' \subseteq P$, $P' \in comp$
4. *if* $(Q_1, G_1), \ldots, (Q_n, G_n) \in pow$ *and for all* $1 \leq i, j \leq n$ *with* $i \neq j$, $Q_i \cap Q_j = \emptyset$, *then* $\{(Q_1, G_1), \ldots, (Q_n, G_n)\} \in comp$.

Both the relations *pow* and the relation *comp* concern the possibility that groups of agents achieve different goals. It is an issue if they are really necessary as defined. One possibility would be not to include the set *comp*

in case it is not accomplished .

and define Power Structures as the qualitative cooperative games defined in Wooldridge et al. [WD04]. Doing so, you can consider that, for example, two groups Q_1 and Q_2 of agents can compatibly achieve respectively the set of goals G_1 and G_2 if and only if $G_1 \cup G_2$ is in the power of $Q_1 \cup Q_2$. In this case the notion of compatibility is embed in the relation *pow*.

However, the following example shows a case where without the set *comp* it is possible to describe only the overall set G of goals a group Q of agents can achieve if they collaborate, but it is not possible to correctly describe a case where different sets of goals can independently be committed to different groups, i.e. without the necessity of further coordination among them to overcome possible conflicts. This aspect is relevant in the coalition formation process since, in general, the need for coordination involves some further activities that can be costly in terms of resources.

Example 1 *Let us consider a database that stores information about the students of a given University. In particular, it contains two relations, a relation* Students *that stores the personal data (registration number, name, surname, address, etc.) of the students and a relation* Exams *that stores the data relative to the exams sustained in that University (date of the exams, name of the exam, registration number, mark, etc.). The relation* Students *has the field* registration number *as its primary key and, to assure the referential integrity, each value of the field* registration number *in the relation* Exams *has to occur in the relation* Students*. The database is used only by two employees, employee*$_1$ *and employee*$_2$*, operating concurrently. Possible conflicts between the employees are managed by a database management system, DBMS, by conceding writing and reading rights to the employees. The employee employee*$_1$ *has the goal* g_1 *to add to the relation* Exams *a tuple with the data relative to an exam sustained by a student with registration number* X*. The employee employee*$_2$ *has the goal* g_2 *to change the registration number of that student from* X *to* Y*. Note that if the two employees simultaneously have access to the database and update the database as desired, then the referential integrity among the relations* Exams *and* Students *could be lost, i.e. there would be in* Exams *a tuple describing the exam of a student with registration number* X*, where no student in the relation* Students *would correspond to this registration number, as the value of* X *is changed to* Y*. Therefore, in this case the goal* g_1 *would not be properly satisfied in the sense that the employee employee*$_1$ *would want to see the updated registration number from* X *to* Y *also applied to the tuple he is going to add. In order to avoid this problem the agents have to coordinate as follows. The DBMS assigns the writing right first only to employee*$_1$*, whereas employee*$_2$ *is waiting. When employee*$_1$ *has finished updating the relation* Exams*, the*

DBMS provides employee$_2$ with the writing right which can therefore modify the value of the registration number X.

Formally, $Gl = \{g_1, g_2\}$ and goals(employee$_1$) $= \{g_1\}$, goals(employee$_2$) $= \{g_2\}$ and goals(DBMS) $= \emptyset$. The relation pow is composed as follows: each employee alone does not have the power to update the database since it needs the writing right from DBMS, therefore it alone does not have the power to achieve any goal. Also the DBMS cannot update the database as required, therefore it cannot achieve the goal g_1 or g_2 alone. If DBMS provides only employee$_1$ with the writing right and, then, employee$_1$ effectively updates the relation Exams*, then the goal g_1 is satisfied, so $(\{DBMS, employee_1\}, \{g_1\}) \in$ pow, whereas for g_2 the employee employee$_2$ is required, therefore:*

$$(\{DBMS, employee_1\}, \{g_2\}) \notin pow \text{ and } (\{DBMS, employee_1\}, \{g_1, g_2\}) \notin pow$$

Analogously, the commitment $(\{DBMS, employee_2\}, \{g_2\})$ is in pow, whereas both $(\{DBMS, employee_2\}, \{g_1\})$ and $(\{DBMS, employee_2\}, \{g_1, g_2\})$ are not. Since if the Database Management System provides both the employees, at the same time, with the writing rights, then the referential integrity among the relation Exams *and the relation* Students *is lost, then g_1 and g_2 cannot compatibly be achieved respectively by $\{DBMS, employee_1\}$ and $\{DBMS, employee_2\}$, i.e.:*

$$\{(\{DBMS, employee_1\}, \{g_1\}), (\{DBMS, employee_2\}, \{g_2\})\} \notin comp$$

However, if the three agents coordinate as described above, then they can avoid conflicts and hence they can achieve both the goals g_1 and g_2. Therefore, $(\{DBMS, employee_1, employee_2\}, \{g_1, g_2\}) \in pow$.

As shown in Example 1, it could be the case that two distinct groups of agents, Q_1 and Q_2, can respectively achieve the goals g_1 and g_2, and also, when joined together, they can overcome internal conflicts and achieve both the goals. Defining a Power Structure without *comp* this would mean that Q_1 and Q_2 could compatibly achieve respectively g_1 and g_2, but this was not the case in that example. In Example 1, indeed, one of the employees has to wait that the other has finished to update the database to start its activity. This involves a delay in the achievement of its goal.

Another approach to define Power Structures could be to consider the relation *comp* as a subset of 2^{pow}, but to define the relation *pow* as a subset of $2^{Ag} \times Gl$. In this case the relation *comp* establishes if a group of agents is able to achieve different goals simultaneously.

This approach cannot represent the cases in which a group Q of agents has only the possibility to achieve an entire set of goals, say $\{g_1, g_2\}$ and

cannot achieve only a part of them. This can occur, for example, in the case that the achievement of the goal g_1 entails the goal g_2 or vice versa. Also this information is relevant in the coalition formation process where the stipulation of an agreement is a public announcement. In these cases if Q cannot achieve g_1 and g_2 independently, but only together, then they also cannot negotiate independently their commitments.

The following example shows a Power Structure that cannot be correctly represented using this approach.

Example 2 *Let PS be a Power Structure consisting of the population of agents* $Ag = \{ag_1, ag_2, ag_3\}$*, the set of all goals* $Gl = \{g_1, g_2, g_3, g_4\}$*, a function goals such that* $goals(ag_1) = \{g_1\}$*,* $goals(ag_2) = \{g_2, g_4\}$ *and* $goals(ag_3) = \{g_3\}$*, a relation* $pow = \{(\{ag_1\}, \{g_2\}), (\{ag_2\}, \{g_1, g_3\}), (\{ag_3\}, \{g_4\})\}$ *and finally* $comp = 2^{pow}$*.*

In Example 2, the agent ag_2 is able to achieve the goals g_1 and g_3 together, but it is not able to achieve them separately, indeed $(\{ag_2\}, \{g_1\})$ and $(\{ag_2\}, \{g_3\})$ are not in pow. This means that if it agrees for example with ag_1 to achieve g_1, then it cannot prevent the achievement also of g_3, and, in case the agreement is a public announcement, ag_3 does not feel the need to negotiate the achievement of the goal g_4 in exchange of the goal g_3.

On the contrary, if we had represented the relation power as a subset of $2^{Ag} \times Gl$, requiring that the relation $comp$ represents when a group of agents can achieve more than one goal, then we would not have been able to represent Example 2 correctly. Indeed stating that ag_2 has the power to achieve $\{g_1, g_2\}$, we should say in this approach that $(\{ag_2\}, g_1) \in pow$ and $(\{ag_2\}, g_2) \in pow$ and $\{(\{ag_2\}, g_1), (\{ag_2\}, g_2)\} \in comp$. But this way we are saying, for example that ag_2 has the power to achieve g_1 which is not true in the model expressed by PS.

Summarizing if on one hand we had defined the Power Structure without the relation $comp$, representing it implicitly with the relation pow, we would have not been able to distinguish between a Power Structure PS_1 with $(Q_1, G_1) \in pow_1$, $(Q_2, G_2) \in pow_1$, $(Q_1 \cup Q_2, G_1 \cup G_2) \in pow_1$ and $\{(Q_1, G_1), (Q_2, G_2)\} \in comp_1$ from a Power Structure PS_2 with $(Q_1, G_1) \in pow_2$, $(Q_2, G_2) \in pow_2$, $(Q_1 \cup Q_2, G_1 \cup G_1) \in pow_2$ and $\{(Q_1, G_1), (Q_2, G_2)\} \notin comp_2$. However, this distinction can be relevant because the fact that $\{(Q_1, G_1), (Q_2, G_2)\} \in comp_1$ in PS_1 means that Q_1 and Q_2 does not need further coordination efforts or a different joint plan in order to achieve both G_1 and G_2. Instead, in PS_2, Q_1 and Q_2 cannot independently achieve G_1 and G_2 and hence some different joint plan has to be considered in order to achieve both of them.

On the other hand, if we had represented the relation power as a subset of $2^{Ag} \times Gl$, then we would not have been able to distinguish between a Power Structure PS_1 with $(Q, \{g_1, g_2\}) \in pow_1$, $(Q, \{g_1\}) \in pow_1$ and $(Q, \{g_2\}) \in pow_1$ from a Power Structure PS_2 where $(Q, \{g_1, g_2\}) \in pow_2$, $(Q, \{g_1\}) \notin pow_2$ and $(Q, \{g_2\}) \notin pow_2$. But this can be relevant in coalition formation processes because in the first case the group of agents Q can deal with the achievement of g_1 and g_2 in different agreements, whereas in the second case this could be not possible.

3.3 Properties of Power Structures

Power Structures have been formalized to describe the possible commitments that agents cooperating can fulfill. As emphasized in Section 3.2, in different contexts the fact that a group of agents is reliable for the fulfilment of a commitment may have different meanings. For this reason no particular constraints have been expressed in Definition 13 apart from those providing to *pow* and *comp* the meaning respectively that a group of agents can achieve of a set of goals (and other agents cannot obstruct it) and that a set of commitments does not conflict.

However, depending on the application domain and how a Power Structure has been derived from a more detailed description of a multiagent system, additional properties may hold. For example, if the notion of power requires that all the agents in a group are necessary in the achievement of a set of goals, then the following property holds: if a group of agents Q has the power to achieve a set of goals G, then the supersets of Q cannot have the power to achieve G. Another property involving both *pow* or *comp* is that if a group of agents Q has the power to achieve separately the sets of goals G_1 and G_2, i.e. $(Q, G_1), (Q, G_2) \in pow$ and the achievement of this two sets of goals does not cause any incompatibility, i.e. $\{(Q, G_1), (Q, G_2)\} \in comp$, then Q has the power to achieve both of them, i.e. $(Q, G_1 \cup G_2) \in pow$. However, as we will see in Section 3.6, the group of agents Q may not consider the possibility to be committed to $G_1 \cup G_2$ if it can achieve them separately. Intuitively, the group of agents Q can profit from the commitments (Q, G_1) and (Q, G_2) in different coalition formation processes asking for each of them something in exchange. This does not occur if they consider the commitment $(Q, G_1 \cup G_2)$. Thus, it can be reasonable to consider that this property in these cases does not hold.

In this section we consider in particular the properties deriving from the analysis of the notion of power developed in Game Theory as well as in cooperative problem solving and theory of social power and dependence

[Aum61, Pau02a, AHK02, BSvdT04a]. These properties are:
coalitional monotonicity: Adding new members to a group does not decrease its power, i.e. if a group of agents Q has the power to achieve a set of goals G, then all the groups that contain Q have the power to achieve G.
super-additivity: Merging together two disjoint groups of agents allows them to continue to operate without interfering with each other, i.e. if two groups of agents Q_1 and Q_2 are disjoint and Q_1 has the power to achieve the set of goals G_1 and Q_2 has the power to achieve the set of goals G_2, then the union of Q_1 and Q_2 has the power to achieve the union of G_1 and G_2.
goal monotonicity: A group of agents Q has the power to achieve a set of goals G means that it has the power to achieve at least all the goals in G, i.e. if a group of agents Q has the power to achieve a set of goals G, then for all subsets of goals $G' \subseteq G$, Q has the power to achieve G'.
group minimality: If a set of agents Q has the power to achieve a set of goals G, then all the agents in Q are necessary for the achievement of G. So, if $(Q, G) \in pow$, then there does not exist a $Q' \subset Q$ and a set of goals $G' \supseteq G$ such that $(Q', G') \in pow$.
g-super-additivity: If two sets of agents Q_1 and Q_2 have the power to compatibly achieve respectively the sets of goals G_1 and G_2, then there exists a subset of all the agents $Q_1 \cup Q_2$, with the power to achieve all the goals $G_1 \cup G_2$.
seriality: Usual notion of seriality, i.e. each group of agents has the power to achieve at least a set of goals (possibly the empty set).
property-1:[2] Each group of agents has the power to achieve the empty set of goals. In other words, it is not the case that however a group of agents behave, it achieves some goals.
property-2: Each group of agents $Q \neq Ag$ has in case only the power to achieve the empty set of goals and if this is the case, then also all its supersets have the power to achieve the empty set of goals.

In the following definition we formalize the previous properties.

Definition 14 *Given a Power Structure* $PS = \langle Ag, Gl, goals, pow, comp \rangle$, *we define the following properties it can satisfy:*

coalitional monotonicity: *if* $(Q, G) \in pow$, *then for all* $Q' \supset Q$, $(Q', G) \in pow$.

[2]The last two properties have been introduced in order to relate the other ones, so they do not have any particular name. In particular property-1 relates super-additivity with coalitional monotonicity, whereas property-2 derives from the fact that coalition monotonicity and group minimality, which would seem to be in contradiction, can be in some particular cases consistent. These cases are characterized by property-2.

super-additivity: *if* $(Q_1, G_1) \in pow$, $(Q_2, G_2) \in pow$ *and* $Q_1 \cap Q_2 = \emptyset$, *then* $(Q_1 \cup Q_2, G_1 \cup G_2) \in pow$.

goal monotonicity: *if* $(Q, G) \in pow$, *then for all* $G' \subset G$, $(Q, G') \in pow$.

group minimality: *if* $(Q, G) \in pow$ *and* $G \neq \emptyset$, *then for all* $G' \supseteq G$ *and for all* $Q' \subset Q$, $(Q', G') \notin pow$.

g-super-additivity: *if* $(Q_1, G_1) \in pow$, $(Q_2, G_2) \in pow$ *and* $\{(Q_1, G_1), (Q_2, G_2)\} \in comp$, *then there exists a* $Q \subseteq Q_1 \cup Q_2$ *such that* $(Q, G_1 \cup G_2) \in pow$.

seriality: *for all* $Q \subseteq Ag$, *there exists a set of goals* G *such that* $(Q, G) \in pow$.

property-1: *for all* $Q \subseteq Ag$, $(Q, \emptyset) \in pow$.

property-2: *for all* $Q \subset Ag$ *and for all* $G \neq \emptyset \subseteq Gl$, $(Q, G) \notin pow$ *and if* $(Q, \emptyset) \in pow$, *then for all groups of agents* $Q' \supset Q$, $(Q', \emptyset) \in pow$.

The properties in Definition 14 are not independent with each other and the following theorem shows some of their relationships.

Theorem 1 *Let* $PS = \langle Ag, Gl, goals, pow, comp \rangle$ *be a Power Structure. The following hold:*

1. *if pow is coalitional monotonic and g-super-additive, then it is super-additive.*
2. *if pow is goal monotonic and serial, then it satisfies property-1.*
3. *if pow is super-additive and satisfies property-1, then it is coalitional monotonic.*
4. *pow is coalitional monotonic and group minimal iff it satisfies property-2.*

proof: For Proposition 1, assume that Q_1 and Q_2 are disjoint and that $(Q_1, G_1) \in pow$ and $(Q_2, G_2) \in pow$. Due to Definition 13, $\{(Q_1, G_1), (Q_2, G_2)\}$ is in *comp* and, since *pow* is g-super-additive, there exists a $Q \subseteq Q_1 \cup Q_2$ such that $(Q, G_1 \cup G_2) \in pow$. Since *pow* is also coalition monotonic, then $(Q_1 \cup Q_2, G_1 \cup G_2) \in pow$.

With respect to Proposition 2, since *pow* is serial, for all $Q \in Ag$ there exists a set of goals G such that $(Q, G) \in pow$. Since *pow* is goal monotonic, for all $G' \subseteq G$, $(Q, G') \in pow$, and in particular $(Q, \emptyset) \in pow$.

With respect to Proposition 3, assume that $(Q, G) \in pow$ and that $Q \subset Q'$. Let $\hat{Q} = Q' \setminus Q$, since pow satisfies property-1, then $(\hat{Q}, \emptyset) \in pow$, but hence for the super-additivity $(Q \cup \hat{Q}, G \cup \emptyset) \in pow$, i.e. $(Q', G) \in pow$.

With respect to the left-right implication of the Proposition 4, assume that pow does not satisfies property-2. Then, one of the two cases holds. First, a $Q \subset Ag$ and a set of goals $G \neq \emptyset$ exist such that $(Q, G) \in pow$, but in this case for the coalition monotonicity $(Ag, G) \in pow$, whereas for the group minimality $(Ag, G) \notin pow$. The second case is that there exists a group of agents $Q \subset Ag$ and a superset $Q' \supset Q$ such that $(Q, \emptyset) \in pow$ and $(Q', \emptyset) \notin pow$, but this is against the coalition monotonicity. In the right-left implication of Proposition 4 group minimality immediately follows from the fact that for all $Q \subset Ag$ the only set of goals in the power of Q is the empty set. The coalition monotonicity derives from the fact that if for a group of agents $Q \subseteq Ag$, $(Q, \emptyset) \in pow$, then for all $Q' \supset Q$, $(Q', \emptyset) \in pow$. Whereas for the other sets of goals such that $(Ag, G) \in pow$, it is also trivially satisfied since sets of agents that strictly contain the whole Ag do not exist. □

3.4 Power Frames: potential proposals for coalition formation

We say that a coalition is formed when a group of agents reaches an agreement in which they engage each other to the achievement of some goals. Given a group of agents Q, it is necessary to formalize which agreements the members of Q can stipulate. The power relation pow of a Power Structure PS describes all the possible commitments. Nevertheless, as the coalition formation process is a collusive behavior, the agents in Q can reach an agreement only on the commitments involving themselves. Therefore, if there exists a $(Q', G) \in pow$ such that $Q' \subseteq Q$, then this commitment can be part of the agreement among the agents in Q.

Thus, an agreement of a group of agents Q could be formalized as a subset P of pow such that for all $(Q', G') \in P$, $Q' \subseteq Q$. Nevertheless, as $comp$ in PS prescribes, not all of the subsets of pow can be compatibly carried out. This means that the agreements that can be proposed are only those subsets of pow in which all the elements are compatible with each other, i.e. only those subsets of pow that are elements of $comp$. Moreover, being $pow \subseteq 2^{Ag} \times 2^{Gl}$, it could be that for a given set of goals G, $(\emptyset, G) \in pow$. Such a pair could be used to represent that the goals in G are achieved independently from the behaviors of the agents in the multiagent system. In any case, considering a commitment $(\emptyset, G)$ to be part of an agreement does not make sense as

it does not involve the collusion of any agent. Therefore, we assume that an agreement does not contain any commitment $(\emptyset, G)$. Furthermore, for Definition 13, the empty set is in *comp*, in this case we consider the empty set as the null agreement, i.e. the absence or the failure of an agreement among some agents.

In the following we call a Power Frame an element of *comp* corresponding to a proposal for an agreement.

Definition 15 (Power Frame) *Let* $PS = \langle Ag, Gl, goals, pow, comp\rangle$ *be a Power Structure,* pfr *is a Power Frame of* PS *iff* $pfr \in comp$ *and for all* $(Q, G) \in pfr$, $Q \neq \emptyset$.

A Power Frame is said to be maximal if there does not exist another Power Frame that contains it.

Notation (Maximal Power Frame) *Let* $PS = \langle Ag, Gl, goals, pow, comp\rangle$ *be a Power Structure, we say that a Power Frame* pfr *of* PS *is maximal iff there does not exist a Power Frame* pfr' *of* PS *such that* $pfr \subset pfr'$.

3.5 The first admissibility criterion: the do-ut-des property

In the previous section we have seen that the relation *comp* of a Power Structure *PS* describes the set of all Power Frames that can be used as a proposal for an agreement. However, some Power Frames are not supposed to actually lead to an agreement among the agents involved. For example, a Power Frame in which an agent is committed to achieve a goal for another agent that does not provide any goal in exchange, is not admissible as an agreement, because there is no reciprocity among the agents.

In this section we formalize this notion of reciprocity by defining a property named *do ut des* (literally *give in order to receive*). This property can be used as an *admissibility* criterion that establishes which Power Frames can actually form a coalition.

The do-ut-des property is defined, similarly as several game theoretical admissibility criteria, as a no-dominance condition. In particular, our dominance relation is considered only among a Power Frame and its subsets. This way, a Power Frame contains all the required information to establish if it satisfies the do-ut-des property. The reason we have called this property do-ut-des will be clear in the next section where we consider a particular class of Power Frames called singleton Power Frames.

Given a Power Structure $PS = \langle Ag, Gl, goals, pow, comp\rangle$, we define a qualitative preference relation, the do-ut-des preference relation, between two of its Power Frames and we use it in order to define a dominance relation between a Power Frame *pfr* and one of its subsets. The qualitative preference relation is defined by means of the functions *adv* and *obl* that represent, for each agent in *PS*, respectively the advantages and the obligations deriving from *pfr*. Since we are considering goal-directed agents, the advantages that an agent ag_i receives in the case that a coalition is formed according to a Power Frame *pfr* consist of the goals achieved in *pfr* that are desired by ag_i.

Definition 16 (adv function) *Given a Power Structure PS and a Power Frame pfr of PS, $adv[pfr](ag_i)$ associates each agent $ag_i \in Ag$ with the set of goals achieved in pfr that are goals of ag_i. Formally:*

$$adv[pfr](ag_i) = \{g \in goals(ag_i) \mid \exists (Q, G) \in pfr \text{ s.t. } g \in G\}$$

In a Power Frame *pfr* the advantages of an agent consist of the set of its goals that are satisfied in *pfr*, no matter who achieves it. On the contrary the obligations of an agent in a Power Frame derive from the goals in which achievement it is involved. Nevertheless, it could be different to achieve a goal alone from achieving it with the help of other agents, as the costs, of different nature, can be shared with the other. This means that it is relevant to consider not only the goals in which achievement an agent is involved, but also with which other agents it cooperates for these achievements. Therefore, given a Power Frame *pfr*, the function *obl* maps each agent ag_i to the subset of *pfr* consisting of the commitments that involve ag_i.

Definition 17 (obl function) *Given a Power Structure PS and a Power Frame pfr of PS, $obl[pfr](ag_i)$ associates each agent $ag_i \in Ag$ with the set of $(Q, G) \in pfr$ ag_i is involved in. Formally:*

$$obl[pfr](ag_i) = \{(Q, G) \in pfr \mid ag_i \in Q\}$$

Now we are able to define the do-ut-des preference relation, $\leq_i$, of an agent ag_i. Informally, given two Power Frames pfr_1 and pfr_2, it is the case that $pfr_1 \leq_i pfr_2$ if the set of goals that ag_i receives from pfr_1 is contained in the one it receives in pfr_2, whereas the obligations it has to sustain in pfr_2, $obl[pfr_2](ag_i)$, are contained in the ones relative to pfr_1.

Definition 18 (Do-ut-des preference relation) *Let PS be a Power Structure and pfr_1 and pfr_2 be two Power Frames of PS, the agent ag_i prefers pfr_1 to pfr_2, $pfr_2 \leq_i pfr_1$, iff $adv[pfr_2](ag_i) \subseteq adv[pfr_1](ag_i)$ and $obl[pfr_1](ag_i) \subseteq obl[pfr_2](ag_i)$.*

Notation (Strict preferences) *We say that ag_i strictly prefers pfr_1 to pfr_2, $pfr_2 <_i pfr_1$, if $pfr_2 \leq_i pfr_1$ and $pfr_1 \not\leq_i pfr_2$.*

We notice that if $pfr' \subset pfr$, then $obl[pfr'](ag_i) \subseteq obl[pfr](ag_i)$. Therefore, $adv[pfr](ag_i) = adv[pfr'](ag_i)$ if and only if $pfr \leq_i pfr'$.

In contrast with the usual preference relations defined in game theory, the do-ut-des preference relation is a qualitative method that compares two Power Frames since it just balances the advantages and obligations with a containment relation without considering a deeper, and quantitative, analysis of a Power Frame. Also from a formal point of view, the usual preference relations defined in game theory are total, reflexive and transitive (total pre-order) [OR94], whereas, for any agent ag_i, the do-ut-des preference relation defines a partial pre-order between Power Frames of the Power Structure *PS*. Therefore it could be the case that, given an agent ag_i and two Power Frames pfr_1 and pfr_2, $pfr_1 \not\leq_i pfr_2$ and $pfr_2 \not\leq_i pfr_1$. Another important difference between the preference relations used in game theory and the do-ut-des preference relation is that the former ones extensively represent, for each agent, a personal judgement criterion based on, for example, (1) how it calculates the benefits deriving from the goals achieved in a Power Frame, (2) how it calculates the costs relative to one of its obligations, and (3) how it balances the benefits with the costs. The do-ut-des preference relation, on the contrary is defined by means of a process that, known the goals of the agents, is homogeneous for all the agents. Of course, the first approach is more general, nevertheless it requires to know the unconstrained personal judgement criteria for each agent. The do-ut-des preference relation provides a way to estimate the profitability of a Power Frame with respect to another that can be applied in the case only the goals of the agents are known.

By means of Definition 18, we provide a qualitative criterion, the *do-ut-des* property, to establish the admissibility of a Power Frame. In order to define the do-ut-des property we introduce some notations.

Notation (Agents involved in a Power Frame) *Given a Power Structure PS and a Power Frame pfr of PS, we say that an agent ag_i is involved in pfr (or simply that it is in pfr) iff there exists a pair $(Q, G) \in pfr$ such that $ag_i \in Q$. We write $ag_i \ltimes pfr$ to indicate that ag_i is involved in pfr.*

The do-ut-des property is defined by means of a dominance relation we call do-ut-des dominance. Informally, given two Power Frames pfr and pfr', pfr' do-ut-des dominates pfr if there exists an agent ag_i in the domain of pfr which strictly prefers pfr' to pfr and all the agents involved in pfr' prefers pfr' at least as pfr. In this case, for ag_i it is more advantageous to not agree to pfr and to propose pfr' to the agents involved in it. If they had agreed to

pfr, then, as *pfr* is no more available due to the rejection of ag_i, they would harmlessly agree to *pfr*′.

Definition 19 (Do-ut-des dominance) *Give two Power Frames pfr*′ *and pfr of a Power Structure PS, we say that pfr*′ *do-ut-des dominates pfr iff the following conditions hold:*

1. *there exists an* $ag_i \ltimes pfr$ *such that* $pfr <_i pfr'$.
2. *for all* $ag_j \ltimes pfr'$, $pfr \leq_j pfr'$.

The following theorem shows that if a Power Frame *pfr*′ do-ut-des dominates a Power Frame *pfr*, then *pfr*′ is strictly contained in *pfr*.

Theorem 2 *Given two Power Frames pfr*′ *and pfr of a Power Structure PS, if pfr*′ *do-ut-des dominates pfr, then* $pfr' \subset pfr$.

proof: From Definition 19 if *pfr*′ do-ut-des dominates *pfr*, then for all $ag_j \ltimes pfr'$, $pfr \leq_j pfr'$. From Definition 18 this entails that for all $ag_j \ltimes pfr'$, $obl[pfr'](ag_j) \subseteq obl[pfr](ag_j)$. Assume *per absurdum* that $pfr' \not\subseteq pfr$ and let $(\bar{Q}, \bar{G})$ be such that $(\bar{Q}, \bar{G}) \in pfr'$ and $(\bar{Q}, \bar{G}) \notin pfr$. But, if so, being $\bar{Q} \neq \emptyset$, there exists an agent $ag_{\bar{\jmath}} \in pfr'$ such that $(\bar{Q}, \bar{G}) \in obl[pfr'](ag_{\bar{\jmath}})$ and $(\bar{Q}, \bar{G}) \notin obl[pfr](ag_{\bar{\jmath}})$. But this contradicts the fact that $obl[pfr'](ag_{\bar{\jmath}}) \subseteq obl[pfr](ag_{\bar{\jmath}})$. So $pfr' \subseteq pfr$, but since *pfr* cannot do-ut-des dominate itself, as the first condition of Definition 19 is not satisfied, then $pfr' \subset pfr$. □

Once defined the do-ut-des dominance relation we say that a Power Frame satisfies the do-ut-des property, or simply it is do-ut-des, in the case that it is not dominated by any of its subsets.

Definition 20 (Do-ut-des Power Frames) *Given a Power Structure PS, a Power Frame pfr of PS is do-ut-des if and only if there does not exist a Power Frame pfr*′ *that do-ut-des dominates pfr.*

We notice that the do-ut-des property is based on the fact that, given a Power Frame *pfr*, all the commitments $(Q, G) \in pfr$ are considered reliable by all the agents involved in *pfr*, i.e. they share the same notion of power. In contrary case, some of the agent could not accept to form a coalition based on the proposal *pfr*, because it consider that one, or more, of the commitments (Q, G) are incorrect, i.e. the group Q has not the power to achieve G. So, in [SD01], each agent has different plans at its disposal and this fact leads to the two cases of mutually believed dependence and locally believed dependence we discussed in Section 2.4.4. In a more general context of the notion of power several other cases can be considered. For example, an

agent may consider that a group of agents Q has the power to achieve a set of goals G because Q has a probability of 0.8 to achieve G, which seems to be a reasonable guarantee for those goals, however another agent may consider 0.8 not enough to consider Q reliable for the achievement of G, therefore for the latter agent Q has not the power to achieve G.

Moreover, some counterintuitive results can arise in the case that some particular properties of a Power Structure hold. For example, a Power Structure could describe the mere capabilities of groups of agents as in the notion of α-ability formalized in Alternating Temporal Logic. If so, from the fact that a group of agents Q has the power to achieve a set of goals G, it can be derived that, adding other agents to Q, the resulting group of agents Q' has the power to achieve G as well (coalitional monotonicity), even if the added agents do not play any role, for example, in the plan that Q has to execute in order to achieve G. Nevertheless, if we consider a Power Frame *pfr* with 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 at least useful for this end.

The previous two issues emphasize as a coalition formation process can fail because (1) each of the agents involved represents the multiagent system with a different Power Structure or because (2) they use a notion of power, in particular the coalitional monotonic ones, that makes the do-ut-des property ineffective. However, as in [CS02b], we focus only on a single Power Structure, which can be considered as an external, *objective*, view of a multiagent system or as the internal description of an agent reasoning about which agreements are admissible to be proposed. In this perspective we do not address the first issue. With respect to the second issue we simply consider that the notion of do ut des can provide useful results according to the input on which it is applied. If this input, i.e. a certain Power Frame, does not accurately describe which agents actually take part in a commitment, then also the results provided by the do ut des property will be scarcely significant.

The following example shows a simple case of a do-ut-des Power Frame corresponding to a reciprocal dependence in the theory of social power and dependence.

Example 3 *A Power Structure is such that* $Ag = \{ag_1, ag_2\}$, $goals(ag_1) = \{g_1\}$, $goals(ag_2) = \{g_2\}$ *and comp contains a Power Frame pfr given by the following table (where each row corresponds to a pair* $(Q, G) \in pfr$*):*

	Q	G
1	$\{ag_1\}$	$\{g_2\}$
2	$\{ag_2\}$	$\{g_1\}$

We show that pfr is do-ut-des. The Power Frames strictly contained in pfr are $pfr_1 = \{(\{ag_1\}, \{g_2\})\}$, $pfr_2 = \{(\{ag_2\}, \{g_1\})\}$ *and the empty set. To prove that pfr is do-ut-des we have to show that no one of* pfr_1, pfr_2 *and the empty set do-ut-des dominates pfr. Since it is not the case that* $pfr \leq_1 pfr_1$, *the second condition of Definition 19 does not hold and hence* pfr_1 *does not do-ut-des dominates pfr. Analogously, it is not the case that* $pfr \leq_2 pfr_2$ *and hence* pfr_2 *does not do-ut-des dominates pfr. Finally, since both* ag_1 *and* ag_2 *receive a goal from pfr, for the empty set neither* $pfr <_1 \emptyset$ *nor* $pfr <_2 \emptyset$. *But this means that the first condition of Definition 19 is not satisfied and hence the empty set does not do-ut-des dominate pfr.*

Example 3 shows that the assumption of enforced agreements is crucial for the motivation of the do-ut-des property. Indeed, *pow* is admissible for the two agents only if each of them is certain that the other will meet its commitment. If, for example, ag_2 did not meet its commitment, then *pow* would be equivalent to pfr_1 which is not do-ut-des, as it is do-ut-des dominated by the empty set.

It is possible that, given a Power Structure *PS*, there exist several Power Frames that satisfy the do-ut-des property, or there could not be any, apart from the empty set. The following example shows these facts.

Example 4 *Let PS be the following Power Structure:* $Ag = \{ag_1, ag_2\}$, $goals(ag_1) = \{g_1, g_3\}$, $goals(ag_2) = \{g_2, g_4\}$, *pow is given by the following table:*

	Q	G
1	$\{ag_1\}$	$\{g_2\}$
2	$\{ag_1\}$	$\{g_4\}$
3	$\{ag_2\}$	$\{g_1\}$
4	$\{ag_2\}$	$\{g_3\}$

and $comp = 2^{pow}$. *Following Example 3, it is possible to see that the do-ut-des Power Frames are only and all the subsets of pow containing at least one of the first two rows and one of the second two rows in the table. Therefore, if we consider a Power Structure* PS' *such that* pow' *consists only of the first two rows of the table, then the only do-ut-des Power Frame is the empty set.*

The following theorem shows that in a do-ut-des Power Frame *pfr* each commitment $(Q, G) \in pfr$ has to contain at least a goal that is not achieved in other commitments of *pfr*. In other words, a do-ut-des Power Frame *pfr* does not admit useless commitments that, if removed, do not harm the set of goals which are achieved in *pfr*.

Theorem 3 *Let pfr be a Power Frame of a Power Structure PS, if there exists a* $(\hat{Q}, \hat{G}) \in pfr$ *such that*

$$\bigcup_{(Q,G)\in pfr} G = \bigcup_{(Q,G)\in[pfr\setminus\{(\hat{Q},\hat{G})\}]} G$$

then pfr is not do-ut-des.

proof: Since the set of goals achieved in *pfr* and in $pfr \setminus \{(\hat{Q}, \hat{G})\}$ are the same, then for each agent $ag_j \ltimes pfr$, $adv[pfr](ag_j) = adv[pfr \setminus \{(\hat{Q}, \hat{G})\}](ag_j)$. Therefore for all $ag_j \in [pfr \setminus \{(\hat{Q}, \hat{G})\}]$, $pfr \leq_j [pfr \setminus \{(\hat{Q}, \hat{G})\}]$.

For an agent $ag_i \in \hat{Q}$, it is also the case that $obl[pfr \setminus \{(\hat{Q}, \hat{G})\}](ag_i) \subset obl[pfr](ag_i)$ and hence $pfr <_i [pfr \setminus \{(\hat{Q}, \hat{G})\}]$. But this means that $pfr \setminus \{(\hat{Q}, \hat{G})\}$ do-ut-des dominates *pfr*. □

3.6 Do-ut-des property and costly goals

In this section we discuss a further discriminant in the admissibility of a Power Frame that can be applied in cases when achieving a set of goals is more costly than to achieve only a part of it. For example, we said in Section 3.3 that a Power Structure can satisfy the following property: if Q has the power to achieve G_1 and G_2 separately (i.e. $(Q, G_1) \in pow$ and $(Q, G_2) \in pow$) and it can do that compatibly (i.e. $\{(Q, G_1), (Q, G_2)\} \in comp$), then it has also the power to achieve them together (i.e. $(Q, G_1 \cup G_2) \in pow$). This property may derive from the fact that the agents in Q have two different plans to achieve respectively the goals in G_1 and in G_2 and, these plans being compatible, they can be executed together achieving the goals in $G_1 \cup G_2$. If we consider that each plan has a not null cost, executing both the plans is more costly than to execute only one of them, and hence, if the agents in Q do not have alternative plans to achieve $G_1 \cup G_2$, achieving $G_1 \cup G_2$ is more costly than achieving G_1 or G_2 individually. In this case there are two Power Frames that can be checked for the do-ut-des property, a Power Frame pfr_1 that contains the commitment $(Q, G_1 \cup G_2)$ and another Power Frame pfr_2 that contains, instead of $(Q, G_1 \cup G_2)$, the commitments (Q, G_1) and (Q, G_2). What we prove in this section is that if pfr_2 is do-ut-des, then

also pfr_1 is do-ut-des, but the inverse is not true, we give an example where pfr_1 is do-ut-des whereas pfr_2 is not.

Example 5 *Let PS be the following Power Structure: $Ag = \{ag_1, ag_2, ag_3\}$, each ag_i desires only one goal and no two agents desires the same goal. We denote with g_i the goal desired by the agent ag_i. The relation pow is entirely given by the following table:*

	Q	G
1	$\{ag_1\}$	$\{g_2\}$
2	$\{ag_1\}$	$\{g_3\}$
3	$\{ag_1\}$	$\{g_2, g_3\}$
4	$\{ag_2\}$	$\{g_1\}$

Finally $comp = 2^{pow}$. It could be verified that the Power Frame pfr_1 consisting of the rows 3 and 4 is do-ut-des. Nevertheless, the agent ag_1 could split the set of goals $\{g_2, g_3\}$ it has to provide in pfr_1 in the two different sets of goals $\{g_2\}$ and $\{g_3\}$ and see that the Power Frame pfr_2 composed by the rows 1, 2 and 4 is not do-ut-des, as the subset composed by the rows 1 and 4 do-ut-des dominates it.

The previous example shows that sometime a Power Frame pfr_1 can be do-ut-des even if another Power Frame pfr_2 obtained from pfr_1 selecting a pair $(Q, G) \in pfr_1$ and splitting G in two subsets could not be do-ut-des. If in Example 5 for the agent ag_1 is more costly to provide the set of goals $\{g_2, g_3\}$ with respect to provide only one of them, then, even if pfr_1 is do-ut-des, it should not agree to pfr_1, since pfr_2 is do-ut-des dominated by a Power Frame in which ag_1 provides only the goal g_2. Therefore, under the assumption that for each group of agents to achieve a set of goals G is more costly than to achieve a subset $G' \subset G$, this example shows that agents should split the achievement of some goals in as much commitments as possible.

With respect to Example 5 we call pfr_2 a refinement of pfr_1 and provide a formal definition of it.

Definition 21 (Power Frame Refinement) *Let PS be a Power Structure and $pfr_1, pfr_2 \in comp$ two Power Frames of PS. pfr_2 is a simple refinement of pfr_1 iff there exists a group of agents Q and two nonempty sets of goals G_1 and G_2 such that $pfr_1 \setminus pfr_2 = \{(Q, G_1 \cup G_2)\}$ and $pfr_2 \setminus pfr_1 = \{(Q, G_1), (Q, G_2)\}$. pfr_2 is a refinement of pfr_1 if there exists a sequence of Power Frames $\overline{pfr_1}(= pfr_1), \ldots, \overline{pfr_n}(= pfr_2)$ such that for all $2 \leq i \leq n$, $\overline{pfr_i}$ is a simple refinement of $\overline{pfr_{i-1}}$.*

In Example 5 we have shown that a Power Frame can be do-ut-des even if a refinement of it is not, for this reason we argued that the refinement of a Power Frame *pfr* provides a more detailed analysis of the do-ut-des property and should be taken into account instead of *pfr*. If so, it should never be the case that a Power Frame *pfr* is not do-ut-des whereas one of its refinement is and hence we wonder if the fact that a refinement of *pfr* is do-ut-des implies that *pfr* is do-ut-des as well. In Theorem 4 and in Corollary 5 we prove that this conjecture is true.

Theorem 4 *Let $PS = \langle Ag, Gl, goals, pow, comp\rangle$ be a Power Structure and $pfr_1, pfr_2 \in comp$ two Power Frames of PS such that pfr_2 is a simple refinement of pfr_1. If pfr_2 is do-ut-des, then pfr_1 is do-ut-des.*

proof: First of all the set of goals provided in pfr_1 and pfr_2 are the same, so for each agent ag_i, $adv[pfr_1](ag_i) = adv[pfr_2](ag_i)$.

Assume that pfr_1 is not do-ut-des, we prove that neither pfr_2 is do-ut-des. If pfr_1 is not do-ut-des, then there exists a $pfr \subset pfr_1$ such that *pfr* do-ut-des dominates pfr_1. Now two possibilities can occur: either $(Q, G_1 \cup G_2) \notin pfr$, or $(Q, G_1 \cup G_2) \in pfr$. In the first case $pfr \subset pfr_2$ and it is straightforward to see that *pfr* do-ut-des dominates pfr_2. In the second case consider the Power Frame $pfr' = [pfr \setminus \{(Q, G_1 \cup G_2)\}] \cup \{(Q, G_1), (Q, G_2)\}$. Since both *pfr* and pfr' achieves the same set of goals and involve the same set of agents, it holds that for all $ag_i \ltimes pfr'$, $adv[pfr'](ag_i) = adv[pfr](ag_i)$. Now, *pfr* do-ut-des dominates pfr_1 and hence $adv[pfr](ag_i) = adv[pfr_1](ag_i)$ and, by definition of simple refinement, $adv[pfr_1](ag_i) = adv[pfr_2](ag_i)$. So, for all $ag_i \ltimes pfr'$, $pfr_2 \leq_i pfr'$.

It remains to prove that there exists an agent in pfr_2 that strictly do-ut-des prefers pfr' to pfr_2. Since *pfr* do-ut-des dominates pfr_1, then there exists an agent $ag_i \ltimes pfr_1$ such that $adv[pfr](ag_i) = adv[pfr_1](ag_i)$ and $obl[pfr](ag_i) \subset obl[pfr_1](ag_i)$. As before, it holds that $adv[pfr'](ag_i) = adv[pfr_2](ag_i)$. Furthermore, if a commitment (Q, G) is in $[obl[pfr_1](ag_i) \setminus obl[pfr](ag_i)]$, then $(Q, G) \in pfr_1$ and $(Q, G) \notin pfr$. Then, (Q, G) cannot be the commitment $(Q, G_1 \cup G_2)$. Since the only difference between pfr_1 and pfr_2 as well as between *pfr* and pfr' is the replacement of $(Q, G_1 \cup G_2)$ with the commitments (Q, G_1) and (Q, G_2), it holds that $(Q, G) \in pfr_2$ and $(Q, G) \notin pfr'$. Therefore, $obl[pfr'](ag_i) \subset obl[pfr_2](ag_i)$ and hence $pfr_2 <_i pfr'$. □

Given Definition 21 and Theorem 4, it is straightforward to prove the following corollary.

Corollary 5 *Let $PS = \langle Ag, Gl, goals, pow, comp\rangle$ be a Power Structure and $pfr_1, pfr_2 \in comp$ two Power Frames of PS such that pfr_2 is a refinement of pfr_1. If pfr_2 is do-ut-des, then pfr_1 is do-ut-des.*

In the general case, Corollary 5 could be used in the search of all the do-ut-des Power Frames of a Power Structure as a method to decrease the number of Power Frames to check. Moreover, in the case we know that the more goals are achieved, the more costly their achievement is, we can simply to ignore the Power Frames pfr_1 which are refined by other Power Frames pfr_2.

3.7 Do-ut-des property and singleton Power Frames

In this section we consider a particular class of Power Frames of a Power Structure, called singleton Power Frames, and we characterize the do-ut-des property for this class. Singleton Power Frames are more closely related to the notion of dependence as defined in the theory of social power and dependence [Cas03, SD01, CS02b] as they consist only of commitments achieving singleton sets of goals, like $(Q, \{g\})$. Moreover, singleton Power Frames enable to clarify why the property in Definition 20 is called do-ut-des, i.e. how it expresses the reciprocity condition.

Definition 22 (Singleton Power Frames) *Let PS be a Power Structure and $pfr \in$ comp a Power Frame of PS. pfr is a singleton Power Frame iff for all $(Q, G) \in pfr$, G is a singleton, i.e. $G = \{g\}$. With an abuse of notation we write (Q, g) instead of $(Q, \{g\})$.*

In order to characterize the do-ut-des property in a singleton Power Frame *pfr* we consider, for each agent ag_i involved in *pfr*, of the subset of *pfr* which is of some interest for ag_i, we call this subset the useful Power Frame of ag_i relative to *pfr*. Informally, the useful Power Frame of an agent consists of the set of all commitments that are included in a *chain* of exchanges providing to ag_i at least one goal. More formally, it is defined by considering, first, the set of commitments that are directly useful for an agent ag_i, i.e. the set of commitments providing the achievement of a goal desired by ag_i. Then, as an exchange may involve a chain longer than the dyadic exchange, for each agent ag_j involved in a commitment useful for ag_i we consider all the commitments that are directly useful for ag_j. We continue until no new commitment is included. This way, all the commitments that could directly or indirectly play a role in an exchange profitable for ag_i are taken into account.

Definition 23 (Directly useful Power Frames) *Given a Power Structure PS and a singleton Power Frame pfr of PS. The directly useful Power*

Frame of pfr for an agent ag_i, $duf(ag_i, pfr)$, is:

$$duf(ag_i, pfr) = \{(Q, g) \in pfr \mid g \in goals(ag_i)\}$$

To formalize the chains of commitments useful for an agent, we say that the Power Frame $pfr' \subseteq pfr$ is the useful Power Frame of pfr for an agent ag_i, $uf(ag_i, pfr)$, if and only if pfr' is the minimal set that contains the directly useful Power Frame of ag_i and such that if $ag_j \ltimes uf(ag_i, pfr)$, then $duf(ag_j, pfr)$ is contained in $uf(ag_i, pfr)$. We constructively formalize this set as a least fix point.

Definition 24 (Useful Power Frames) *Let PS be a Power Structure and pfr a singleton Power Frame of PS. The useful Power Frame of pfr for an agent ag_i, $uf(ag_i, pfr)$, is the least fix point of a functional τ on 2^{pfr}:*

$$\tau(Y) = \begin{cases} duf(ag_i, pfr) \text{ if } Y = \emptyset \\ \\ duf(ag_i, pfr) \cup Y \cup \bigcup_{ag_j \ltimes Y} duf(ag_i, pfr) \text{ otherwise} \end{cases}$$

It is straightforward to see that the functional τ is monotonic and continuous with respect to the inclusion relation and hence for the Tarski-Knaster theorem there exists a least fix-point F_l for τ and it is equal to $\bigcup_{i \geq 0} \tau^i(\emptyset)$, where τ^i denotes the composition $\tau \circ \cdots \circ \tau$ for i times [McM92, Tar55]. Therefore, $uf(ag_i, pfr)$ satisfies the properties that $duf(ag_i, pfr) \subseteq uf(ag_i, pfr)$ and if $ag_j \ltimes uf(ag_i, pfr)$, then $duf(ag_j, pfr) \subseteq uf(ag_i, pfr)$.

The following theorem shows the relationship between the do-ut-des property and useful Power Frames: pfr is a do-ut-des singleton Power Frame if and only if two different groups of agents are not committed in the achievement of the same goal and there does not exists an agent ag_i and a commitment $(Q, g) \in pfr$ such that $ag_i \in Q$ and (Q, g) is not $uf(ag_i, pfr)$.

Theorem 6 *Let $PS = \langle Ag, Gl, goals, pow, comp \rangle$ be a Power Structure and $pfr \in comp$ be a singleton Power Frame of PS. pfr is do-ut-des iff the following holds:*

1. *there do not exist two distinct groups of agents Q_1 and Q_2 and a goal g such that both (Q_1, g) and (Q_2, g) are in pfr.*
2. *for all $(Q, g) \in pfr$ and for all $ag_i \in Q$, $(Q, g) \in uf(ag_i, pfr)$.*

proof:
$\Longrightarrow$

The first condition immediately follows from Theorem 3. For the second condition, assume that there exists an agent $ag_i \ltimes pfr$ and a $(Q, g) \in pfr$ such

that $(Q, g) \in obl[pfr](ag_i)$ and $(Q, g) \notin uf(ag_i, pfr)$. As $obl[uf(ag_i, pfr)](ag_i) \subseteq uf(ag_i, pfr)$, $(Q, g) \notin obl[uf(ag_i, pfr)](ag_i)$. Since $obl[uf(ag_i, pfr)](ag_i)$ is contained in $obl[pfr](ag_i)$, it holds that $obl[uf(ag_i, pfr)](ag_i) \subset obl[pfr](ag_i)$.

We also have that:

$$adv[uf(ag_i, pfr)](ag_i) = adv[duf(ag_i, pfr)](ag_i) = adv[pfr](ag_i)$$

So, being $adv[uf(ag_i, pfr)](ag_i) = adv[pfr](ag_i)$ and $obl[uf(ag_i, pfr)](ag_i) \subset obl[pfr](ag_i)$, $pfr <_i uf(ag_i, pfr)$.

We also have that for all the agents $ag_j \ltimes uf(ag_i, pfr)$, $duf(ag_j, pfr) \subseteq uf(ag_i, pfr)$ and hence $adv[uf(ag_i, pfr)](ag_j) = adv[pfr](ag_j)$. But this means that for all $ag_j \ltimes uf(ag_i, pfr)$, $pfr \leq_j uf(ag_j, pfr)$.

Thus, $pfr <_i uf(ag_i, pfr)$ and for all $ag_j \ltimes uf(ag_i, pfr)$, $pfr \leq_j uf(ag_j, pfr)$, but this means that $uf(ag_i, pfr)$ do-ut-des dominates pfr against the hypothesis.

$\Longleftarrow$

Assume that pfr is not do-ut-des, this entails that there exists a Power Frame $pfr' \subset pfr$ and an agent $ag_i \ltimes pfr$ such that $pfr <_i pfr'$. By definition (see Definition 18) this means that:

1. $obl[pfr'](ag_i) \subset obl[pfr](ag_i)$
2. $adv[pfr'](ag_i) = adv[pfr](ag_i)$

Let $(Q, g) \in [obl[pfr](ag_i) \setminus obl[pfr'](ag_i)]$. Due to Definition 17, $(Q, g) \notin pfr'$ and hence, being $uf(ag_i, pfr') \subseteq pfr'$, $(Q, g) \notin uf(ag_i, pfr')$. As $adv[pfr'](ag_i) = adv[pfr](ag_i)$ and for any goal there exists at most one set of agents that achieves it, it holds that $duf(ag_i, pfr)$ is equal to $duf(ag_i, pfr')$.

From the hypothesis that pfr is not do-ut-des it is also the case that for all the agents $ag_j \ltimes pfr'$, $pfr \leq_j pfr'$. This means that for all the agents $ag_j \ltimes pfr'$, $adv[pfr'](ag_j) = adv[pfr](ag_j)$ and hence, as for ag_i, for all $ag_j \ltimes pfr'$, $duf(ag_j, pfr)$ is equal to $duf(ag_j, pfr')$. Now the two facts: (1) $duf(ag_i, pfr) = duf(ag_i, pfr')$ and (2) for all $ag_j \ltimes pfr'$, $duf(ag_j, pfr) = duf(ag_j, pfr')$, together with Definition 24, entail that $uf(ag_i, pfr') = uf(ag_i, pfr)$, but this means that $(Q, g) \notin uf(ag_i, pfr)$ against the hypothesis. □

Theorem 6 explains the term do-ut-des: first, when a group of agents Q commits itself in the achievement of a goal g, it releases the other groups from the achievement of the same goal. Second, an agent ag_i participates to a commitment (Q, g) only if (Q, g) is part of a chain of commitments that provides ag_i with a goal in exchange.

A singleton Power Frame pfr of a Power Structure PS can be represented by means of a tagged AND-graph as follows: the agents of PS are the nodes

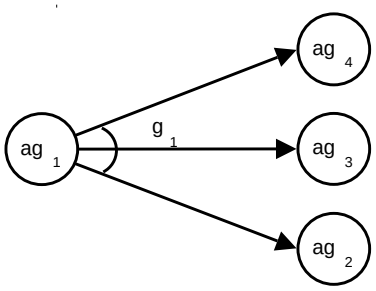

Figure 3.1: *An tagged AND-arcs of a singleton Power Frame.*

of the graph. If $(Q, g) \in pfr$, then for all the agents ag_i that desire g we draw an AND-arc from ag_i to Q tagged with g. For example, Figure 3.1 shows the tagged AND-arc representing that $Q = \{ag_2, ag_3, ag_4\}$ has the commitment to achieve the goal g_1 that is desired by the agent ag_1. This way, $adv[pfr](ag_i)$ is the set of the tags associated with the AND-arcs starting from ag_i. Similarly $obl[pfr](ag_i)$ is the set of $(Q, g) \in pfr$ corresponding to an AND-arcs that reach ag_i, and eventually also other agents.

Definition 25 (Power Frame graph representation) *Given a singleton Power Frame pfr of a Power Structure PS. The graph representation of pfr, $\mathcal{G}[pfr]$, is a tagged AND-graph $\langle \mathcal{V}, \mathcal{E} \rangle$, such that $\mathcal{V} = Ag$ and $\mathcal{E} \subseteq \mathcal{V} \times 2^{\mathcal{V}} \times Gl$ is a set of triples defined as follows:*

$$\mathcal{E} = \{(ag_i, Q, g) \mid g \in goals(ag_i) \ \wedge \ (Q, g) \in pfr\}$$

A Power Frame graph representation can be viewed as the analogous of the Dependence graphs developed in [CS02b]. Nevertheless, with respect to the Dependence Graph, we abstract from the representation of the actions needed in order to achieve a goal and 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 [Cas03, Cas00], but more general potentials for social interaction.

To emphasize the topological characterization of Theorem 6, we provide an interpretation of it by means of the Power Frame graph representation

$\mathcal{G}[pfr] = \langle \mathcal{V}, \mathcal{E} \rangle$. A sequence of AND-arcs $\mathcal{P} = (ag_{i_1}, Q_1, g_1), \ldots, (ag_{i_n}, Q_n, g_n)$ is a path if for all $2 \leq h \leq n$, $ag_{i_{h-1}} \in Q_h$. We say that $\mathcal{P}$ starts from ag_{i_1} and reaches Q_n with g_n. It is straightforward to see that $(Q, g) \in uf(ag_i, pfr)$ if and only if there exists a path in $\mathcal{G}[pfr]$ starting from ag_i that reaches Q with g. Therefore, the second condition of Theorem 6 can be reformulated saying that in a do-ut-des singleton Power Frame *pfr*, given a commitment $(Q, g) \in pfr$, for each agent ag_i involved in Q there exists a path in $\mathcal{G}[pfr]$ starting from ag_i and reaching Q with g.

In the following example we show, using this characterization, a singleton Power Frame that is not do-ut-des.

Example 6 *Let PS be a Power Structure such that* $Ag = \{ag_1, ag_2, ag_3, ag_4\}$, $Gl = \{g_1, g_2, g_3, g_4, g_5\}$ *and goals is such that* $goals(ag_1) = \{g_1, g_5\}$ *and for all the other* ag_i, $goals(ag_i) = \{g_i\}$. *comp contains a Power Frame pfr given by the following table:*

	Q	G
1	$\{ag_1\}$	$\{g_2\}$
2	$\{ag_2\}$	$\{g_1\}$
3	$\{ag_3\}$	$\{g_4\}$
4	$\{ag_4\}$	$\{g_3\}$
5	$\{ag_4\}$	$\{g_5\}$

The Power Frame graph representation of pfr is shown in Figure 3.2. Considering the commitment $(\{ag_4\}, g_5)$ *there does not exist a path in Figure 3.2 starting from* ag_4 *that reaches* $\{ag_4\}$ *with* g_5. *Therefore, pfr is not do-ut-des.*

We notice that pfr satisfies the condition that each agent involved obtains at least a goal and provides at least a goal. This condition has been used in [CS02b] to define the notion of reciprocity in the case each agent desires at most one goal. This example shows that, when an agent desires multiple goals in a singleton Power Frame, this condition is a necessary but not sufficient condition to describe the notion of reciprocity.

Roughly speaking, the do-ut-des property, concerning the reciprocity of goal exchanges, is somewhat related with the notion of strong connectivity of directed graph [CR90]. Nevertheless, dealing with tagged AND-graph is somewhat more tricky. In fact if in Figure 3.2 we replace the tag g_3 with the tag g_5, then the resulting Power Frame is do-ut-des.

In the following examples we consider only singleton Power Frames and we provide in figures the graphical interpretation of them.

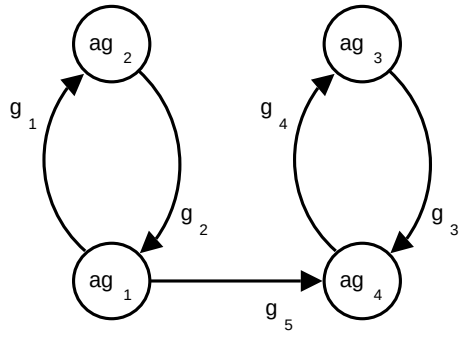

Figure 3.2: *A singleton Power Frame that does not satisfy the do-ut-des property.*

3.8 The second admissibility criterion: the composition property

We have seen in Section 3.5 that the do-ut-des property establishes the admissibility of a Power Frame *pfr* by considering, for each agent involved in it, the advantages and the obligations of that agent in the case *pfr* is agreed. Doing so, the do-ut-des property does not take into account the costs, or the risks, inherent in the agreement process. However, it is reasonable that some aspects regarding the agreement process may affect the valuation of a Power Frame. For example, the more the number of the agents involved in an agreement process, the more this agreement process can be costly in terms of the duration it takes to be reached. Moreover, the more the number of the agents involved in an agreement, the greater the risk that one of the involved agents can cause the agreement to fail.

To take into account this aspect, in this section we define an admissibility criterion we call the composition property. The general idea underlying the composition property is that, given a Power Frame *pfr*, the agents involved prefer to cut it in subsets if these subsets have more chances to reach an agreement. Consider the following example:

Example 7 *Let PS be a Power Structure such that* $Ag = \{ag_1, ag_2, ag_3, ag_4\}$*, each* ag_i *desires only one goal and no two agents desire the same goal. We denote with* g_i *the goal desired by the agent* ag_i*. The relation pow is given by the following table:*

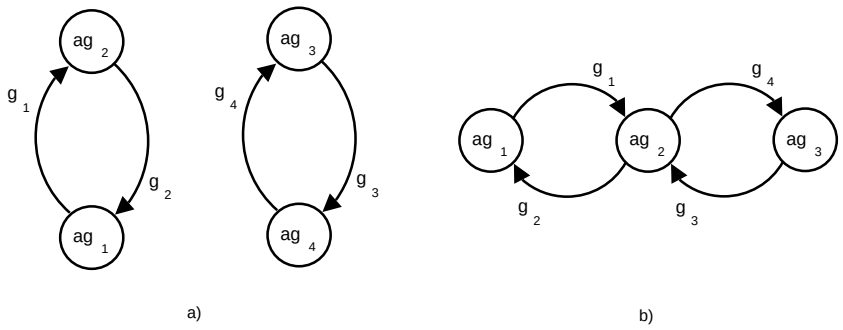

Figure 3.3: *Example of Power Frames that do not satisfy the composition property.*

	Q	G
1	$\{ag_1\}$	$\{g_2\}$
2	$\{ag_2\}$	$\{g_1\}$
3	$\{ag_3\}$	$\{g_4\}$
4	$\{ag_4\}$	$\{g_3\}$

and the relation $comp = 2^{pow}$*. The graphical representation of pow is shown in Figure 3.3 (a). It is straightforward to prove that the do-ut-des Power Frames of PS are, apart from the empty Power Frame, pow and the Power Frames composed respectively by the first and second rows, and by the third and fourth rows.*

- $pfr_1 = \{(\{ag_1\}, \{g_2\}), (\{ag_2\}, \{g_1\})\}$
- $pfr_2 = \{(\{ag_3\}, \{g_4\}), (\{ag_4\}, \{g_3\})\}$

In the previous example ag_1 and ag_2 are both indifferent between *pow* and pfr_1, indeed their advantages and obligations are the same. This means that if they agreed to *pow*, then they would agree also to pfr_1. Therefore, being ag_1 and ag_2 the only agent involved in pfr_1, if an agreement is reached with respect to *pow*, then also an agreement would be reached with respect to pfr_1. The contrary would not be true, as ag_3 or ag_4 could not agree to *pow*. This means that for ag_1 and ag_2 it is more reliable to form a coalition according to pfr_1 than to *pow*, and hence they may prefer pfr_1 to *pow* in order to minimize the risk of a failure in the coalition formation process.

Analogously, the same occurs for ag_3 and ag_4 with respect to pfr_2 and *pow*, therefore *pow* can be decomposed in two subsets pfr_1 and pfr_2 such that

the coalition formation processes of these two Power Frames do not interfere with each other.

In the previous example, a Power Frame has been partitioned into subsets pfr_1 and pfr_2 such that the agents involved in pfr_1 are not interested in pfr_2 and vice versa. However, this could be not the only case where a Power Frame can be decomposed into independent and more reliable subsets, as the following example shows.

Example 8 *Let PS be the following Power Structure:* $Ag = \{ag_1, ag_2, ag_3\}$. $Gl = \{g_1, g_2, g_3, g_4\}$. *The function goals associates to* ag_1 *the single goal* g_1, *to* ag_3 *the single goal* g_3 *and two* ag_2 *the goals* g_2 *and* g_4. *The relation pow is given by the following table:*

	Q	G
1	$\{ag_1\}$	$\{g_2\}$
2	$\{ag_2\}$	$\{g_1\}$
3	$\{ag_2\}$	$\{g_3\}$
4	$\{ag_3\}$	$\{g_4\}$

and the relation $comp = 2^{pow}$. *The graphical representation of pow is shown in Figure 3.3 (b). It is straightforward to prove that the do-ut-des Power Frames of PS are the empty set, pow and the Power Frames composed respectively by the first and second rows, and by the third and fourth rows.*

- $pfr_1 = \{(\{ag_1\}, \{g_2\}), (\{ag_2\}, \{g_1\})\}$
- $pfr_2 = \{(\{ag_2\}, \{g_3\}), (\{ag_3\}, \{g_4\})\}$

The previous example shows a situation slightly different with respect to Example 7. The difference is that in Example 7 both ag_1 and ag_2 were indifferent between pfr_1 and pow according to the preference relation in Definition 18. In Example 8, instead, ag_2 is not indifferent between pfr_1 and pow since in pow it receives the goal g_4 that is not mentioned in pfr_1.

However, ag_1 is indifferent between pfr_1 and pow and ag_3 is indifferent between pfr_2 and pow, hence if they agree to pow, then they would agree respectively to pfr_1 and pfr_2. Then, assume that pow is proposed to ag_2 and ag_2 does not consider, for example, pfr_1 worthwhile. In this case, instead of agreeing to pow, ag_2 would propose pfr_2 to ag_3 (being sure that it is indifferent among pow and pfr_2). Therefore, also for ag_2, if it agrees to pow, then it would agree also to pfr_1. So if an agreement on pow is reached, then also an agreement of pfr_1 would be reached. The contrary is false since it

could be the case that ag_3 does not consider pow, and hence pfr_2, worthwhile. Symmetrically, the same holds also for pfr_2 and pow. Therefore, pfr_1 and pfr_2 are individually more reliable to succeed with respect to the whole pow. Notice that what is important here is that pfr_1 does not affect the possibility for ag_2 to reach an agreement with respect to the rest of pow, i.e. pfr_2 (we will find this condition explicitly formalized in definition of the composition property).

The composition property is a refinement of the do-ut-des property and hence if a Power Frame satisfies the composition property it is also do-ut-des. As the do-ut-des property, the composition property is a condition of no-dominance between a Power Frame and its subsets, where the dominance relation is the same as in Definition 19. What marks a difference between the composition property and the do-ut-des property is the qualitative preference relation. In particular, we refine the do-ut-des preference relation in Definition 18 in order to take into account the attitude of the agents to cut a Power Frame in subsets if they can be dealt independently, we call this refinement c-p preference relation. Informally, an agent ag_i c-p prefers a subset $pfr' \subset pfr$ to a Power Frame pfr if one of the following two cases occurs: first, it do-ut-des prefers pfr' at least as much as pfr (as a particular case $pfr' =_i pfr$, i.e. ag_i has no preference for either pfr' or pfr). In the second case, ag_i is interested in pfr', i.e. it obtains the achievement of some of its goals from pfr', and all the other agents involved in $pfr \setminus pfr'$ do-ut-des prefer $pfr \setminus pfr'$ at least as much as pfr. Notice that it is important, in the second case, to assume that $adv[pfr'](ag_i) \neq \emptyset$, otherwise it is always the case that an agent c-p prefers the empty Power Frame to a not-empty Power Frame pfr and hence only the empty Power Frame would satisfy the composition property.

Definition 26 (C-p preference relation) *Let PS be a Power Structure and pfr and pfr' be two Power Frames of PS . We say that the agent ag_i strictly c-p prefers pfr' to pfr, $pfr \lessdot_i pfr'$, iff $pfr' \subset pfr$ and one of the following conditions is satisfied:*

1. $pfr \leq_i pfr'$
2. $adv[pfr'](ag_i) \neq \emptyset$ *and for all* $ag_j (\neq ag_i) \ltimes [pfr \setminus pfr']$, $pfr \leq_j [pfr \setminus pfr']$

Defined the preference relation for the composition problem, the dominance relation is defined as in Definition 19 substituting the $\leq_i$ relation with the $\lessdot_i$ relation.

Definition 27 (C-p dominance relation) *Given a Power Frame pfr of a Power Structure PS, we say that $pfr' \subset pfr$ c-p dominates pfr iff the following conditions hold:*

1. *there exists* $ag_i \ltimes pfr$ *such that* $pfr \lessdot_i pfr'$
2. *for all* $ag_j \ltimes pfr'$, $pfr \lessdot_j pfr'$

In the previous definition the existence of an agent $ag_i \ltimes pfr$ such that $pfr \lessdot_i pfr'$ is necessary in the case we compare pfr with the empty Power Frame (being the second condition automatically satisfied). In all the other cases the existence of at least an agent involved in pfr' guarantees that if the second condition is satisfied also the first one is. Moreover, we notice that if pfr' do-ut-des dominates pfr, then it is also the case that pfr' c-p dominates pfr.

The composition property is defined analogously to the do-ut-des property as a no-dominance criterion.

Definition 28 (C-p Power Frames) *A Power Frame pfr of a Power Structure PS satisfies the composition property, shortly it is a c-p Power Frame, if and only if there does not exist a Power Frame* $pfr' \subset pfr$ *that c-p dominates pfr.*

Since the do-ut-des dominance relation implies the c-p dominance relation, it holds that if a Power Frame satisfies the composition property, then it satisfies also the do-ut-des property.

Considering again Examples 7 and 8 we show that, as expected, the power function *pow* does not satisfy the composition property in both cases.

Example 9 *Let consider the Power Structure PS of Example 7, we show that the Power Frame* $pfr_1 = \{(\{ag_1\}, \{g_2\}), (\{ag_2\}, \{g_1\})\}$ *c-p dominates pow. First of all* $pfr_1 \subset pow$, *moreover for* ag_1 *and* ag_2 *it is the case that respectively* $pfr_1 =_1 pow$ *and* $pfr_1 =_2 pow$, *but this means for Definition 26 that* $pow \lessdot_1 pfr_1$ *and* $pow \lessdot_2 pfr_1$. *Therefore, all the agents involved in* pfr_1 *c-p prefer* pfr_1 *to pow. Since* pfr_1 *is not empty, there also exists an agent in pow that c-p prefers* pfr_1 *to pow. But this means that* pfr_1 *c-p dominates pow.*

Consider now the Power Structure PS of Example 8, we show that the Power Frame $pfr_1 = \{(\{ag_1\}, \{g_2\}), (\{ag_2\}, \{g_1\})\}$ *c-p dominates pow. As in the previous case* $pfr_1 \subset pow$ *and* $pfr_1 =_1 pow$, *therefore* $pow \lessdot_1 pfr_1$. *For* ag_2 *it is the case that*

- *It is interested in* pfr_1: $adv[pfr_1](ag_2) = \{g_2\}$
- ag_3 *has no preference for either pow or* $pow \setminus pfr_1$

Thus, the second condition of Definition 26 is satisfied, hence $pow \lessdot_2 pfr_1$ *holds. But this means that* pfr_1 *c-p dominates pow and hence pow does not satisfy the composition property.*

Finally, we prove that the Power Structure pfr_1 satisfies the composition property. Considering the Power Frame $pfr'_1 = \{(\{ag_1\}, \{g_2\})\}$ it is not true that $pfr_1 \leq_1 pfr'_1$ and hence the first condition of Definition 26 does not hold. Furthermore, being $adv[pfr'_1](ag_1) = \emptyset$ also the second condition of Definition 26 does not hold. Therefore, it is not the case that $pfr_1 \lessdot_1 pfr'_1$ and hence pfr'_1 does not c-p dominate pfr_1. Analogously, it can be proved that $pfr''_1 = \{(\{ag_2\}, \{g_1\})\}$ does not c-p dominate pfr_1. To prove that pfr_1 satisfies the composition property, it remains to show that the empty set does not c-p dominates pfr_1. For ag_1 it is not the case that $pfr_1 \leq_1 \emptyset$, therefore the first condition of Definition 26 does not hold. Moreover, being $adv[\emptyset](ag_1) = \emptyset$, also the second condition of Definition 26 does not hold, therefore it is not the case that $pfr_1 \lessdot_1 \emptyset$. Analogously, it can be proved that $pfr_1 \lessdot_2 \emptyset$. Thus, the first condition of Definition 27 does not hold and hence the empty set does not c-p dominate pfr_1.

3.9 Composition property and singleton Power Frames

As done for the do-ut-des property in Section 3.7, in this section we consider the composition property in the case of singleton Power Frames and we characterize it in terms of the useful Power Frames of the agents involved (Definition 24). First of all, since a Power Frame that satisfies the composition property satisfies the do-ut-des property as well, it holds that, for all $(Q, g) \in pfr$ and for all $ag_i \in Q$, $(Q, g) \in uf(ag_i, pfr)$ (see Theorem 6).

However, this property is not sufficient to characterize the composition property. In particular, two other properties hold. The first is that for all the agents $ag_i \ltimes pfr$, $uf(ag_i, pfr) = pfr$, this means that the Power Frame cannot be partitioned in two subsets such that all the agents involved in one are indifferent to the other. The second condition takes into account the case shown in Example 8. Assume that the directly useful Power Frame of an agent ag_i, $duf(ag_i, pfr)$, can be partitioned in two non empty subsets, say D_1 and D_2 (in the following we call such kinds of partitions bi-partitions). Consider now the useful Power Frames pfr_1 and pfr_2 calculated by replacing in Definition 24 the directly useful Power Frame of ag_i, $duf(ag_i, pfr)$, respectively with D_1 and D_2. In the following we call pfr_1 and pfr_2 the useful Power Frames of ag_i restricted to respectively to D_1 and D_2. If $pfr_1 \cap pfr_2 = \emptyset$, then no agent $ag_j \neq ag_i$ involved in pfr_1 would be interested in one of the goals achieved in pfr_2 and vice versa. But this means that ag_i can deal separately with the formation of the two coalitions, or, more formally, that both

pfr_1 and pfr_2 c-p dominates pfr. The following theorem shows that also the inverse implication holds.

Theorem 7 *Let $PS = \langle Ag, Gl, goals, pow, comp\rangle$ be a Power Structure and $pfr \in comp$ a singleton Power Frame of PS. pfr satisfies the composition property iff the following conditions hold:*

1. *there do not exist two distinct groups of agents Q_1 and Q_2 and a goal g such that both (Q_1, g) and (Q_2, g) are in pfr.*
2. *for all $ag_i \ltimes pfr$, $uf(ag_i, pfr) = pfr$.*
3. *for all $ag_i \ltimes pfr$, if $duf(ag_i, pfr)$ can be partitioned in two non empty subsets D_1 and D_2, then the Power Frames pfr_1 and pfr_2, obtained from Definition 24 by substituting $duf(ag_i, pfr)$ respectively with D_1 and D_2, are not disjoint.*

proof:
$\Longrightarrow$

If *pfr* satisfies the composition property, then it is also do-ut-des. Therefore, due to the Theorem 6, there do not exist two distinct groups of agents Q_1 and Q_2 and a goal g such that both (Q_1, g) and (Q_2, g) are in *pfr*.

Assume *per absurdum* that there exists an agent $ag_i \ltimes pfr$ such that at least one of the following holds:

1. $uf(ag_i, pfr) \subset pfr$
2. $duf(ag_i, pfr)$ can be partitioned in two non empty subsets D_1 and D_2 such that the Power Frames pfr_1 and pfr_2, obtained from Definition 24 by substituting $duf(ag_i, pfr)$ respectively with D_1 and D_2, are disjoint, i.e. $pfr_1 \cap pfr_2 = \emptyset$.

In the first case, by construction, for all $ag_j \ltimes uf(ag_i, pfr)$, $duf(ag_j, pfr) \subseteq uf(ag_i, pfr)$ and hence $adv[pfr](ag_j) = adv[uf(ag_i, pfr)](ag_j)$. But this means that $pfr \leq_j uf(ag_i, pfr)$ and hence, being $uf(ag_i, pfr) \subset pfr$, $pfr \lessdot_j uf(ag_i, pfr)$. Thus, we have that the second condition of Definition 27 is satisfied. If $uf(ag_i, pfr) \neq \emptyset$, then it also holds that there exists an $ag_j \ltimes pfr$ such that $pfr \lessdot_j uf(ag_i, pfr)$. Therefore, both the conditions of Definition 27 are satisfied, hence $uf(ag_i, pfr)$ c-p dominates *pfr* against the hypothesis. Also in the case $uf(ag_i, pfr) = \emptyset$ it c-p dominates *pfr*. Indeed, the second condition of Definition 27 is satisfied because $uf(ag_i, pfr)$ is empty and, since $adv[pfr](ag_i) = \emptyset$, $pfr \leq_i uf(ag_i, pfr)$ and hence $pfr \lessdot_i uf(ag_i, pfr)$. Therefore, also the first condition of Definition 27 is satisfied.

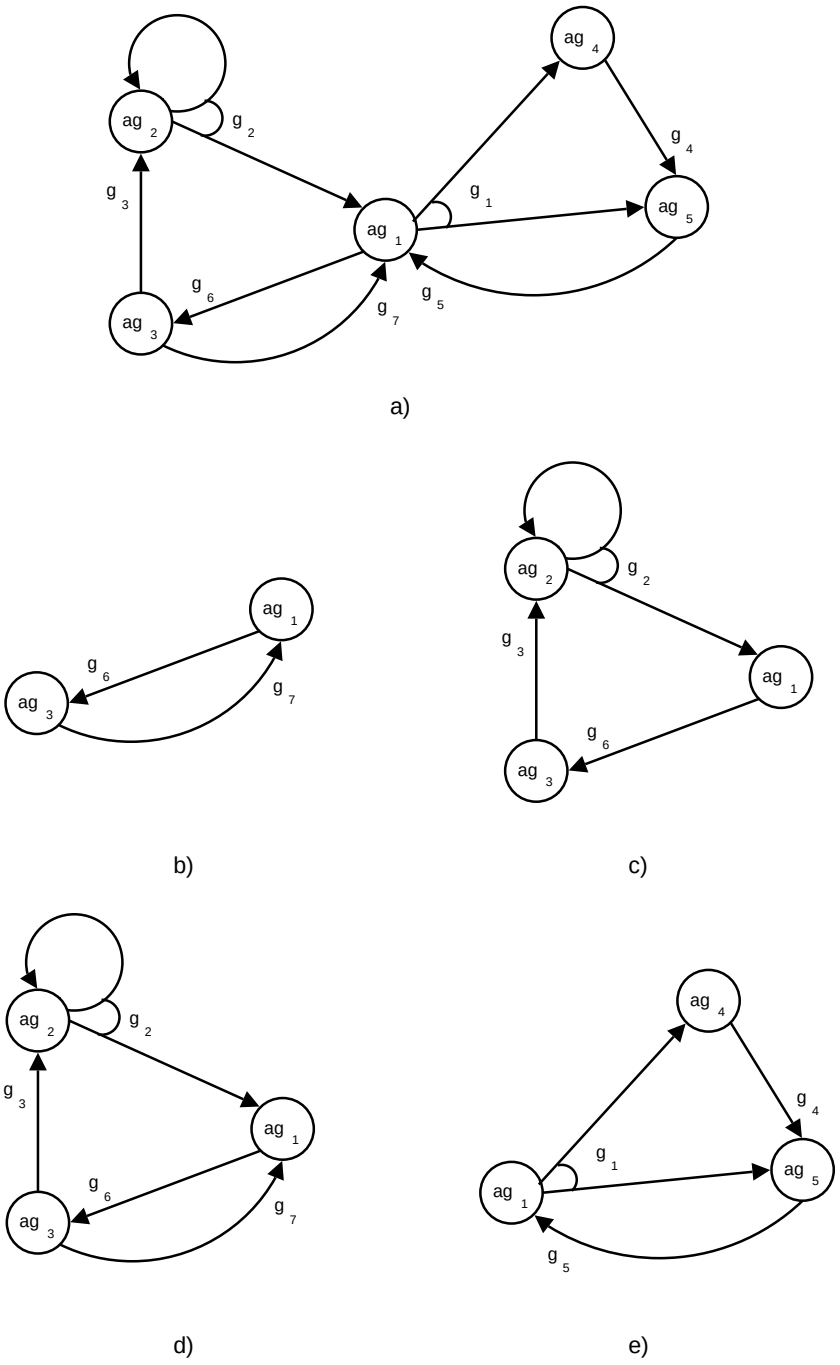

Figure 3.4: *A Complex Case of Power Frames satisfying the composition property.*

In the second case we show that pfr_1 c-p dominates pfr. For all $ag_h(\neq ag_i) \ltimes pfr_1$, as for the previous case, $duf(ag_h, pfr) \subseteq pfr_1$. This means that $pfr \leq_h pfr_1$ and hence, since $pfr_1 \subset pfr$, $pfr \lessdot_h pfr_1$. We still have to prove that also for ag_i, $pfr \lessdot_i pfr_1$. It is straightforward that $pfr_1 \cup pfr_2 = uf(ag_i, pfr)$. Since from the previous case we know that $uf(ag_i, pfr) = pfr$ and we have stated that $pfr_1 \cap pfr_2 = \emptyset$, we have that $pfr_2 = pfr \setminus pfr_1$. Moreover, since $D_1 \neq \emptyset$, $adv[pfr_1](ag_i) \neq \emptyset$. By construction, for all $ag_j(\neq ag_i) \ltimes pfr_2$, $duf(ag_j, pfr) \subseteq pfr_2$. Then, $pfr \leq_j pfr_2$. But this means that for ag_i the second condition of Definition 26 is satisfied and hence $pfr \lessdot_i pfr_1$. Therefore also in this case pfr_1 c-p dominates pfr against the hypothesis.

$\Longleftarrow$

Assume *per absurdum* that there exists a pfr' that c-p dominates pfr. By definition we have that $pfr' \subset pfr$ and:

1. there exists an $ag_i \ltimes pfr$ such that $pfr \lessdot_i pfr'$.
2. for all $ag_j \ltimes pfr'$, $pfr \lessdot_j pfr'$.

We distinguish two cases. In the first case $pfr' = \emptyset$. Since for all $ag_i \ltimes pfr$, $adv[pfr'](ag_i) = \emptyset$, then the second item of Definition 26 cannot be satisfied. Therefore for the agent $ag_i \ltimes pfr$ that c-p prefers pfr' to pfr it is the case that $pfr <_i \emptyset$. But this means that $duf(ag_i, pfr) = \emptyset$ and hence also $uf(ag_i, pfr) = \emptyset$ against the hypothesis that $uf(ag_i, pfr) = pfr$.

In the second case $pfr' \neq \emptyset$. Assume that for all the agents $ag_j \ltimes pfr'$, $pfr \leq_j pfr'$. We have that $adv[pfr](ag_j) \subseteq adv[pfr'](ag_j)$ and hence, as goals are achieved by at most a group of agents, for all $ag_j \ltimes pfr'$, $duf(ag_j, pfr) \subseteq duf(ag_j, pfr')$. Then, $uf(ag_j, pfr) \subseteq uf(ag_j, pfr')$, but, being $uf(ag_j, pfr') \subseteq pfr'$ and $pfr' \subset pfr$, also $uf(ag_j, pfr) \subset pfr$ holds against the hypothesis. Therefore there exists an agent $ag_{\bar{\jmath}} \ltimes pfr'$ such that $pfr \not\leq_{\bar{\jmath}} pfr'$.

Since we have assumed that $pfr \lessdot_{\bar{\jmath}} pfr'$, then we have that the second condition of Definition 26 holds for $ag_{\bar{\jmath}}$. So, it holds that:

1. $adv[pfr'](ag_{\bar{\jmath}}) \neq \emptyset$.
2. for all $ag_h(\neq ag_{\bar{\jmath}}) \ltimes [pfr \setminus pfr']$, $pfr \leq_h [pfr \setminus pfr']$.

Let $D_1 = duf(ag_{\bar{\jmath}}, pfr) \cap pfr'$ and $D_2 = duf(ag_{\bar{\jmath}}, pfr) \cap [pfr \setminus pfr']$. By construction $D_1 \cap D_2 = \emptyset$ and $D_1 \cup D_2 = duf(ag_{\bar{\jmath}}, pfr)$, moreover, since $adv[pfr'](ag_{\bar{\jmath}}) \neq \emptyset$, then $D_1 \neq \emptyset$ and, since $pfr \not\leq_{\bar{\jmath}} pfr'$, then $adv[pfr \setminus pfr'](ag_{\bar{\jmath}}) \neq \emptyset$ and hence $D_2 \neq \emptyset$. But this means, by hypothesis, that the Power Frames pfr_1 and pfr_2 obtained substituting in Definition 24 $duf(ag_{\bar{\jmath}}, pfr)$ respectively with D_1 and D_2 are such that $pfr_1 \cap pfr_2 \neq \emptyset$. Since for all $ag_h(\neq ag_{\bar{\jmath}}) \ltimes [pfr \setminus pfr']$, $pfr \leq_h [pfr \setminus pfr']$, then $pfr_2 \subseteq [pfr \setminus pfr']$. Then, being $pfr_1 \cap pfr_2 \neq \emptyset$, there

exists another agent $ag_{\hat{\jmath}}(\neq ag_{\bar{\jmath}}) \ltimes pfr'$ such that $duf(ag_{\hat{\jmath}}, pfr) \not\subseteq pfr'$ and hence $pfr \not\preceq_{\hat{\jmath}} pfr'$.

Since *per absurdum* we have supposed that $pfr \preccurlyeq_{\hat{\jmath}} pfr'$, then the second condition of Definition 26 holds also for $ag_{\hat{\jmath}}$. Thus, $ag_{\bar{\jmath}}$ is not involved in $pfr \setminus pfr'$ because, in contrary case, it should be $pfr \leq_{\bar{\jmath}} [pfr \setminus pfr']$, against the fact that $adv[pfr'](ag_{\bar{\jmath}}) \neq \emptyset$. Finally, we know that for all $ag_h(\neq ag_{\bar{\jmath}}) \ltimes [pfr \setminus pfr']$, $pfr \leq_h [pfr \setminus pfr']$. Since $ag_{\bar{\jmath}}$ is not involved in $pfr \setminus pfr$, then for all agents $ag_h \ltimes [pfr \setminus pfr']$ it is the case that $pfr \leq_h [pfr \setminus pfr']$, but this cannot be because, as proved before, in this case for each agent $ag_h \ltimes [pfr \setminus pfr']$, $uf(ag_h, pfr) \subseteq [pfr \setminus pfr'] \subset pfr$ against the hypothesis. □

The following example describes a complex Power Structure and the set of Power Frames that satisfy the composition property.

Example 10 *A Power Structure is such that* $Ag = \{ag_1, ag_2, ag_3, ag_4, ag_5\}$. $Gl = \{g_1, g_2, g_3, g_4, g_5, g_6, g_7\}$. *The function goals associates* ag_1 *with the goals* g_1 *and* g_6*;* ag_2 *with the single goal* g_2*;* ag_3 *with the goals* g_3 *and* g_7*,* ag_4 *with the goal* g_4 *and* ag_5 *with the goal* g_5*. The relation pow is given by the following table:*

	Q	G
1	$\{ag_4, ag_5\}$	$\{g_1\}$
2	$\{ag_1, ag_2\}$	$\{g_2\}$
3	$\{ag_2\}$	$\{g_3\}$
4	$\{ag_5\}$	$\{g_4\}$
5	$\{ag_1\}$	$\{g_5\}$
6	$\{ag_3\}$	$\{g_6\}$
7	$\{ag_1\}$	$\{g_7\}$

and the relation $comp = 2^{pow}$*. The graphical representation of pow is shown in Figure 3.4 (a). Using Theorem 7, it can be verified that the Power Frames that satisfies the composition properties, apart from the empty set, are the following:*

- $pfr_1 = \{(\{ag_1\}, \{g_7\}), (\{ag_3\}, \{g_6\})\}$
- $pfr_2 = \{(\{ag_1, ag_2\}, \{g_2\}), (\{ag_3\}, \{g_6\}), (\{ag_2\}, \{g_3\})\}$
- $pfr_3 = \{(\{ag_1\}, \{g_7\}), (\{ag_1, ag_2\}, \{g_2\}), (\{ag_3\}, \{g_6\}), (\{ag_2\}, \{g_3\})\}$
- $pfr_4 = \{(\{ag_1\}, \{g_5\}), (\{ag_4, ag_5\}, \{g_1\}), (\{ag_5\}, \{g_4\})\}$

Where the graphical representations of pfr_1, pfr_2, pfr_3 and pfr_4 are shown in Figure 3.4, respectively (b), (c), (d) and (e). pow, for example, does not satisfy the composition property because the unique bi-partition of $duf(ag_1, pow) = \{(\{ag_3\}, g_6), (\{ag_4, ag_5\}, g_1)\}$ in $D_1 = \{(\{ag_3\}, g_6)\}$ and $D_2 = (\{ag_4, ag_5\}, g_1)$ is such that the useful Power Frames pfr_1 and pfr_2 restricted to respectively to D_1 and D_2, represented in Figure 3.4 (d) and (e), are disjoint.

Since the cardinality of pow is equal to 7 and all the subsets of pow are in comp, the set of all singleton Power Frames is equal to $2^7 = 128$. However only four singleton Power Frames satisfy the composition property, that approximatively corresponds to the 3% of all the singleton Power Frames.

3.10 An algorithm for the composition property in singleton Power Frames

In this section we design a procedure `FIND` which finds all the subsets of a singleton Power Frame *pfr* satisfying the composition property (*pfr* included).

Theorem 7 provides a characterization of the composition property for singleton Power Frames that could be used *tout court* in order to design `FIND`. However, Theorem 7 requires to verify, for each $ag_i \ltimes pfr$, a condition on the set of bi-partitions of $duf(ag_i, pfr)$. 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 a singleton Power Frame satisfies the composition problem would increase in complexity exponentially with the cardinality of $duf(ag_i, pfr)$. For this reason we consider an alternative approach in order to make at least the verification problem tractable.

We use graph theory to design an algorithm (Algorithm 1) which decomposes our problem as much as possible in well known problems in this area. To simplify the problem we make three assumptions on the singleton Power Frame. The following theorems implicitly refer to these assumptions. The first assumption is that every goal is assigned to at most one group of agents. The second assumption is that for each committed goal g there exists at least an agent $ag_i \ltimes pfr$ that desires g. The third assumption is that there do not exist commitments like $(\{ag_i\}, g)$, where ag_i is the only agent that desires g.

It is straightforward to show that if a singleton Power Frame *pfr* does not satisfy one of these three assumptions, then it does not satisfy the composition property. The following considerations provide a guideline to design a sub-procedure of `FIND` that, before executing Algorithm 1, deals with the singleton Power Frames which do not satisfy the given assumptions:

- To verify that a singleton Power Frame *pfr* satisfies the three assump-

tion is computationally tractable. We do not prove formally this assertion, however it easy to see how, for example, representing the set of all the goals Gl by means of red-black trees [CR90] and associating to each goal the set of agents which desire it, the set of commitments $(\{ag_i\}, g)$ that do not satisfy the third assumption can be individuated in a time that is upper bounded by $n \cdot \log(j)$, where n is the cardinality of pfr and j the cardinality of Gl.

- In the case the first assumption does not hold the problem can be reformulated by selecting all the combinations of commitments in pfr that satisfy this assumption and calling the algorithm `FIND` for each of them. In the case the second assumption does not hold, the *accused* commitments, i.e. the commitments that make this condition false, can be removed from pfr, as every subset of pfr containing one of them does not satisfy the composition property. In the case the third assumption is not satisfied, each of the *accused* commitments $(\{ag_i\}, g)$ constitutes a subset of pfr satisfying the composition property and they can be removed from pfr as all the other subsets of pfr containing one of them do not satisfy the composition property.

In Theorem 7, we have shown that the composition property for a singleton Power Frame pfr is equivalent to the satisfaction of three conditions. The first condition is that there do not exist two groups of agents that are committed in pfr to the same goal. We have considered this condition as an assumption. The second condition is that for all the agents ag_i involved in pfr, $uf(ag_i, pfr) = pfr$. The third condition is that for all the bi-partitions D_1 and D_2 of a $duf(ag_i, pfr)$, the Power Frame pfr_1 and pfr_2, obtained from Definition 24 replacing $duf(ag_i, pfr)$ respectively with D_1 and D_2, are not disjoint. The algorithm `FIND` verifies the second and third conditions of Theorem 7 sequentially.

We characterize the second condition as a property of strong connectivity of a directed graph. We define a direct graph $\mathbf{G}[pfr] = \langle \mathbf{V}, \mathbf{E} \rangle$ relative to the singleton Power Frame pfr as follows: the set of nodes $\mathbf{V}$ is equal to the set of agents involved in pfr and $(ag_i, ag_j) \in \mathbf{E}$ if and only if there exist a goal g and a group of agents Q such that $(Q, g) \in pfr$, $g \in goals(ag_i)$ and $ag_j \in Q$. As suggested in Figure 3.5, the directed graph $\mathbf{G}[pfr]$ is obtained by considering tagged AND-arcs of $\mathcal{G}[pfr]$ and *breaking* them. If there exists a path in $\mathcal{G}[pfr]$, as for example in Figure 3.5 (a), starting from $ag_1 \ltimes pfr$ and reaching a set of agents containing the agent ag_4, then there exists a path in $\mathbf{G}[pfr]$, Figure 3.5 (b), that starts from ag_1 and reaches ag_4. The following theorem shows that the condition that for all the agents ag_i $uf(ag_i, pfr) = pfr$, is equivalent

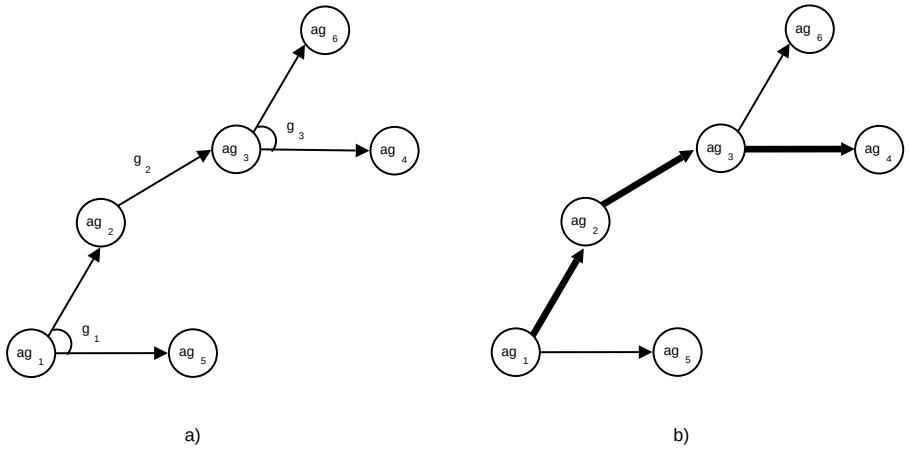

Figure 3.5: *A tagged AND-graph and the corresponding directed graph.*

to say that $\mathbf{G}[pfr]$ is strongly connected, i.e. for each to nodes ag_i and ag_j there exists a path from ag_i to ag_j.

Theorem 8 *Given a Power Frame pfr, for all* $ag_i \ltimes pfr$ $uf(ag_i, pfr) = pfr$ *iff* $\mathbf{G}[pfr]$ *is strongly connected.*

proof:
$\Longrightarrow$

Given two agents ag_i and ag_j involved in pfr, we have that there exists a $(Q, g) \in pfr$ with $ag_j \in Q$. Since $(Q, g) \in uf(ag_i, pfr)$, as seen in Section 3.7, there exists a path in $\mathcal{G}[pfr]$ which starts from ag_i and reaches Q with g. So, by construction, there exists a path in $\mathbf{G}[pfr]$ from ag_i to ag_j. But this means that $\mathbf{G}[pfr]$ is strongly connected.
$\Longleftarrow$

For all $(Q, g) \in pfr$ there exists, for the second assumption we made, at least an agent $ag_h \ltimes pfr$ such that $(Q, g) \in duf(ag_h, pfr)$. Since $\mathbf{G}[pfr]$ is strongly connected, there exists a path from ag_i to ag_h. Therefore, ag_h is involved in $uf(ag_i, pfr)$ and, being $(Q, g) \in duf(ag_h, pfr)$, this means that $(Q, g) \in uf(ag_i, pfr)$. □

The fact that the condition $uf(ag_i, pfr) = pfr$ can be translated in the condition of strong connectivity of the directed graph $\mathbf{G}[pfr]$ has the advantage that there exist well known and optimized algorithms to calculate the strongly connected components of a directed graph, i.e. the maximal strongly connected subgraphs which contain at least an arc.

Assumed that $\mathbf{G}[pfr]$ is strongly connected, now we see how to translate the third condition of Theorem 7 in a graph property. This condition is

closely related with the notion of biconnectivity for undirected graphs. An undirected graph G is biconnected if and only if for all triples of distinct nodes ag_i, ag_j and ag_k, there exists a path connecting ag_j and ag_k such that ag_i is not in the path. If, on the contrary, there exists a triple of distinct nodes ag_i, ag_j and ag_k such that ag_i is in all the paths connecting ag_j and ag_k, ag_i is called an articulation node [Tar72]. As for strongly connected components of a direct graph, given an undirected graph G, the biconnected components are the maximal biconnected subgraphs of G which contain at least an 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.

Starting from the directed graph $\mathbf{G}[pfr] = \langle \mathbf{V}, \mathbf{E} \rangle$, it is possible to define a undirected graph $\mathsf{G}[pfr] = \langle \mathsf{V}, \mathsf{E} \rangle$ as follows: $\mathsf{V} = \mathbf{V}$ and, for $ag_i \neq ag_j$, $\{ag_i, ag_j\} \in \mathsf{E}$ if and only if (ag_i, ag_j) or (ag_j, ag_i) are in $\mathbf{E}$.

The following theorem shows that, under the assumption that $\mathbf{G}[pfr]$ is strongly connected, biconnectivity of $\mathsf{G}[pfr]$ guarantees that the third condition is satisfied.

Theorem 9 *Let pfr be a singleton Power Frame such that* $\mathbf{G}[pfr]$ *is strongly connected, if there exists an agent* $ag_i \ltimes pfr$ *and a bipartition* D_1 *and* D_2 *of* $duf(ag_i, pfr)$ *such that the corresponding restricted useful Power frames* pfr_1 *and* pfr_2*, are disjoint, then* ag_i *is an articulation node of* $\mathsf{G}[pfr]$*.*

proof:

The third assumption we made, i.e. there do not exist commitments like $(\{ag_i\}, g) \in pfr$ such that ag_i is the only agent that desire g, and the strong connectivity of $\mathbf{G}[pfr]$ assure that if pfr_1 and pfr_2 are disjoint, then there exists two agents, say ag_1 and ag_2, such that ag_i, ag_1 and ag_2 are distinct and ag_1 is involved in pfr_1 and ag_2 is involved in pfr_2. Assume *per absurdum* that ag_i is not an articulation node. Since $\mathbf{G}[pfr]$ is strongly connected G is connected, so there exists an undirected path p connecting ag_1 and ag_2. Since ag_i is not an articulation node, ag_i is not a node of p. Each node in the path is an agent in pfr_1 or pfr_2 as we have that $pfr_1 \cup pfr_2 = uf(ag_i, pfr) = pfr$. Now starting from ag_1 it is possible to walk through the path p until an agent $ag_h \ltimes pfr_1$ and the successor $ag_k \ltimes pfr_2$. The presence of an undirected arc connecting ag_h to ag_k means that one of them is in the directly useful Power Frame of the other one. Without lost of generality we assume that $ag_h \in duf(ag_k, pfr)$. This means, due to Definition 24, that $ag_h \ltimes pfr_2$ and hence $duf(ag_h, pfr)$ is contained in both pfr_1 and pfr_2. From the fact that $uf(ag_h, pfr) = pfr$ we have that $duf(a_h, pfr)$ is not empty, then $pfr_1 \cap pfr_2 \neq \emptyset$ against the hypothesis. □

A consequence of this theorem is that if $\mathsf{G}[pfr]$ is biconnected (i.e. it does not have articulation points), then it satisfies the third condition.

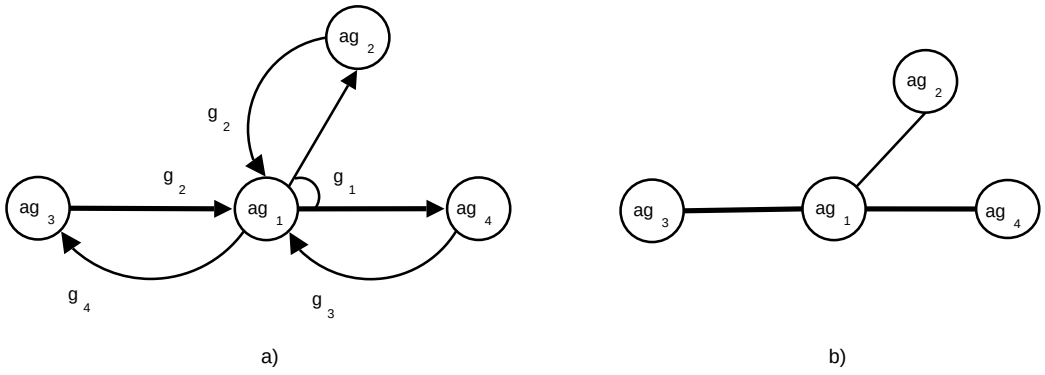

Figure 3.6: *A tagged AND-graph and the corresponding undirected graph.*

Strong connectivity of the directed graph $\mathbf{G}[pfr]$ is a necessary condition for the composition property, and adding the biconnectivity of the undirected graph $\mathsf{G}[pfr]$ we have a sufficient condition for the composition property. It remains to consider the case in which $\mathbf{G}$ is strongly connected and G is not biconnected. Due to Theorem 9, the possibility that the third condition is not satisfied is indicated by the presence of an articulation node ag_i in the undirected graph G. Following the proof of Theorem 9, it can be proved that the biconnected components $\mathsf{G}_1, \ldots, \mathsf{G}_N$ sharing ag_i indicates the ways to find a bipartition of $duf(ag_i, pfr)$ that could make the third condition false. In particular, if a bi-partition D_1 and D_2 of $duf(ag_i, pfr)$ is such that the restricted useful Power Frames pfr_1 and pfr_2 are disjoint, then there exists a $1 \leq k < N$ such that $\mathsf{G}[D_1] \subseteq \bigcup_{l \leq k} \mathsf{G}_l$ and $\mathsf{G}[D_2] \subseteq \bigcup_{k<l \leq N} \mathsf{G}_l$.

However, two facts have to be taken into account. First, some bi-partitions of the biconnected components may not correspond to any bi-partition of $duf(ag_i, pfr)$ as the undirected graph $\mathsf{G}[pfr]$ *breaks* an AND-arc in several undirected arcs. Second, it could be that a commitment $(\{ag_i\}, g) \in pfr$ is desired by two, or more, agents ag_h that are involved in different G_j. However, $(\{ag_i\}, g)$ is in the intersection of the Power Frames corresponding to the G_j and hence these components have to be considered as a single component. Figure 3.6 considers both these two facts. Figure 3.6 (b) represents the undirected graph $\mathsf{G}[pfr]$ of a Power Frame pfr represented by the tagged AND-graph $\mathcal{G}[pfr]$ in Figure 3.6 (a). There exist three biconnected components of G, one for each arc, sharing ag_1 as articulation node. Nevertheless, both the arcs $\{ag_1, ag_2\}$ and $\{ag_1, ag_4\}$ corresponds to the AND-arc tagged with g_1. Therefore, since there does not exist a bi-partition of $duf(ag_1, pfr)$ that separates these two components, these have to be considered a single one.

Even if the arcs $\{ag_1, ag_2\}$ and $\{ag_1, ag_4\}$ have to be considered part of the same component, $\mathsf{G}[pfr]$ can still be decomposed in two components sharing ag_1 as articulation node, the first contains $\{ag_1, ag_2\}$ and $\{ag_1, ag_4\}$, the second contains only $\{ag_1, ag_3\}$. These two components correspond to the restricted useful Power Frames of ag_1, pfr_1 and pfr_2, obtained propagating the bi-partition $D_1 = \{(\{ag_3\}, g_3)\}$ and $D_2 = \{(\{ag_2, ag_4\}, g_1)\}$. In the tagged AND-graph in Figure 3.6 (a), both ag_3 and ag_2 depends on ag_1 for the same goal g_2, this means that the commitment $(\{ag_1\}, g_2)$ is both in the Power Frames pfr_1 and pfr_2, and hence the two components have to be considered as a single component. Since we ended with a single component consisting of the whole undirected graph $\mathsf{G}[pfr]$ this means that no bi-partitions of $duf(ag_1, pfr)$ falsify the third condition and hence pfr satisfies the composition property.

3.10.1 Description of the algorithm

In this section we provide an algorithm for the procedure `FIND` that uses the result shown in the previous section. We assume that the procedure `FIND`, before executing Algorithm 1, calls a sub-procedure which manages, as informally described in Section 3.10, the three assumptions we made. However, for simplicity, we do not formalize this sub-procedure here. Algorithm 1 has as input a representation of a Power Frame pfr as a tagged AND-graph $\mathcal{G}[pfr]$ and provides as output the subsets of pfr, pfr included, satisfying the composition property.

This algorithm calls two procedures `STRONGLY-CONNECTED-COMPONENTS` and `BICONNECTED-COMPONENTS`, the first calculates the strongly connected components of the directed $\mathbf{G}[pfr]$ and the second calculates the biconnected components of the undirected graph $\mathsf{G}[pfr]$. Algorithms for these procedures are well studied in graph theory [CR90, Gib85, Tar72], so we do not describe them in detail. For the sake of simplicity we also omit some manipulations that have been defined in the previous section, as the construction of $\mathbf{G}[pfr]$ and $\mathsf{G}[pfr]$ starting from $\mathcal{G}[pfr]$. However, it is not difficult to figure out procedures `STRONGLY-CONNECTED-COMPONENTS` and `BICONNECTED-COMPONENTS` that, without any particular overhead, take as input $\mathcal{G}[pfr]$ and deal with it respectively as $\mathbf{G}[pfr]$ and $\mathsf{G}[pfr]$.

The variable $COMP\_PFR$ stores the subsets of pfr that satisfy the composition property, in line 1 this variable is initialized to the empty set. In line 2, the strongly connected components SCC of $\mathbf{G}[pfr]$ are calculated. Three cases are distinguished.

Case 1: $\mathbf{G}[pfr]$ is strongly connected, therefore $uf(ag_i, pfr) = pfr$ is satisfied for all $ag_i \ltimes pfr$. In this case the set of biconnected components BC

Algorithm 1:

Data: $\mathcal{G}[pfr] = \langle \mathcal{V}, \mathcal{E} \rangle$, the tagged AND-graph relative to the Power Frame *pfr*.

Result: *COMP_PFR*, the set of subsets of *pfr* that satisfy the composition problem.

1 $COMP\_PFR \leftarrow \emptyset$;
2 $SCC \leftarrow$ `STRONGLY-CONNECTED-COMPONENTS` $(\mathbf{G}[pfr])$;
3 **switch** *SCC* **do**
4 **case** $\mathbf{G}[pfr]$ *is strongly connected*
5 $(BC, ART\_NODES) \leftarrow$ `BICONNECTED-COMPONENTS` $(\mathsf{G}[pfr])$;
6 **forall** $(ag_i, Q, g) \in \mathcal{E}$ *s.t.* $ag_i \in ART\_NODES$ **do**
7 $BC' \leftarrow \{\langle \mathsf{V}', \mathsf{E}' \rangle \in BC \mid \{ag_i, ag_j\} \in \mathsf{E}' \text{ with } ag_j \in Q\}$;
8 $BC \leftarrow [BC \setminus BC'] \cup \{\bigcup BC'\}$;
9 **forall** $g \in Gl$ *and* $ag_i \in ART\_NODES$ *s.t.* $\exists (ag_j, \{ag_i\}, g) \in \mathcal{G}[pfr]$ **do**
10 $V_g = \{ag_j \mid (ag_j, \{ag_i\}, g) \in \mathcal{G}[pfr]\}$;
11 $BC' \leftarrow \{\langle \mathsf{V}, \mathsf{E} \rangle \in BC \mid \mathsf{V} \cap V_g \neq \emptyset\}$;
12 $BC \leftarrow [BC \setminus BC'] \cup \{\bigcup BC'\}$;
13 **if** $|BC| = 1$ **then**
14 $COMP\_PFR \leftarrow \{pfr\}$;
15 **forall** $(Q, g) \in pfr$ **do**
16 $pfr' \leftarrow pfr \setminus \{(Q, g)\}$;
17 $COMP\_PFR \leftarrow COMP\_PFR \cup$ `FIND` $(\mathcal{G}[pfr'])$;
18 **else**
19 **forall** $\langle \mathsf{V}, \mathsf{E} \rangle \in BC$ **do**
20 $\mathcal{E}' \leftarrow \{(ag_i, Q, g) \in \mathcal{E} \mid ag_i \in \mathsf{V} \wedge Q \subseteq \mathsf{V}\}$;
21 $\mathcal{V}' \leftarrow \mathsf{V}$;
22 $COMP\_PFR \leftarrow COMP\_PFR \cup$ `FIND` $(\langle \mathcal{V}', \mathcal{E}' \rangle)$;
23 **case** $\mathbf{G}[pfr]$ *has not strongly connected components*
24 $COMP\_PFR \leftarrow \{\emptyset\}$;
25 **otherwise**
26 **forall** $\langle \mathbf{V}, \mathbf{E} \rangle \in SCC$ **do**
27 $\mathcal{V}' = \mathbf{V}$;
28 $\mathcal{E}' = \{(ag_i, Q, g) \in \mathcal{E} \mid ag_i \in \mathbf{V} \wedge Q \subseteq \mathbf{V}\}$;
29 $COMP\_PFR \leftarrow COMP\_PFR \cup$ `FIND` $(\langle \mathcal{V}', \mathcal{E}' \rangle)$;
30 **return** *COMP_PFR*;

of $\mathsf{G}[pfr]$ and the set of articulation points *ART_NODES* are calculated, line 5. Algorithm 1 checks, for each articulation node ag_i, if there exists an AND-arc (ag_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, lines 6-8. In lines 9-12, the components in BC that have two agents desiring the fulfillment of the same commitment (Q, g) are calculated. Since for each goal at most one group of agents achieve it and two biconnected components share at most one node that is the articulation node ag_i, then Q can involve only of ag_i. Therefore, for all $g \in Gl$ and for all articulation nodes such that there exists an AND-arc in the form $(ag_j, \{ag_i\}, g)$, first the set V_g of all the nodes ag_j such that $(ag_j, \{ag_i\}, g) \in \mathcal{E}$ is calculated, then the biconnected components having a node in V_g are selected and replaced in BC with their union.

In the case $|BC| = 1$, then *pfr* satisfies the composition property and it is added to *COMP_PFR*. In lines 15-17, for all $(Q, g) \in pfr$, the Power Frame pfr' obtained removing (Q, g) from *pfr* is constructed, then `FIND` is called on $\mathcal{G}[pfr']$ and the output is added to *COMP_PFR*. If $|BC| > 1$, then *pfr* does not satisfy the composition property. Also all the subsets pfr' of *pfr* such that $\mathsf{G}[pfr']$ is not included in a component of BC cannot satisfy the composition property. Therefore, in lines 19-22, for all the components in BC, the maximal subgraph of $\mathcal{G}[pfr']$ *included* in this component is selected. `FIND` is recursively called on $\mathcal{G}[pfr']$ and the output is added to *COMP_PFR*.

Case 2: $\mathbf{G}[pfr]$ has not strongly connected components. Since strong connectivity is a necessary condition for the satisfaction of the composition property, no subset of *pfr* is a candidate for the composition property except for the empty set. Therefore, the empty set is assigned to *COMP_PFR*.

Case 3: $\mathbf{G}[pfr]$ is not strongly connected, but there exist some strongly connected components. Since a necessary condition for *pfr* to satisfy the composition property is that $\mathbf{G}[pfr]$ is strongly connected (Theorem 8), *pfr* does not satisfy the composition property. For this reason the subsets pfr' of *pfr* that could satisfy the composition property are only those such that the relative undirected graphs $\mathbf{G}[pfr']$ are subgraphs of a strongly connected component. Therefore, in lines 26-29, for each strongly connected component, the maximal tagged AND-graph $\langle \mathcal{V}', \mathcal{E}' \rangle$, *included* in the component, is constructed. The function `FIND` is recursively called on $\langle \mathcal{V}', \mathcal{E}' \rangle$ and its output is added to *COMP_PFR*.

Finally, *COMP_PFR* is returned, line 30.

3.10.2 Complexity of the algorithm

In this section we show the complexity of Algorithm 1. First of all, we shows that the problem to check if a singleton Power Frame satisfies the composition property is tractable. The algorithm `FIND` can be easily modified to simply check if a given singleton Power Frame satisfies the composition property. The simplest way is to replace lines 14-17 with an instruction that returns true, and lines 19-22, 24 and 26-29 with an instruction returning false. We denote with n the cardinality of *pfr*, with m the number of agents involved in *pfr* and with l the cardinality of the set of arcs in $\mathbf{G}[pfr]$. The procedure `STRONGLY-CONNECTED-COMPONENTS` takes a time proportional to l [CR90]. In the case $\mathbf{G}[pfr]$ is not strongly connected than *pfr* does not satisfies the composition property and the program return `false`. In the other case the procedure `BICONNECTED-COMPONENTS` is called on the undirected graph $\mathsf{G}[pfr]$.

Also `BICONNECTED-COMPONENTS` can be executed in a time that is proportional to $|\mathsf{E}|$ and, since $|\mathsf{E}| \leq l$, so far the algorithm has a complexity that is proportional to l. We have to consider now the complexity of the cycles corresponding to the lines 6-8 and 9-12. Since each AND-arc (ag_i, Q, g) outgoing a node corresponds to a commitment in *pfr*, the number of iterations of the cycle 6-8 is less than n. The instruction in line 7 has as upper bound m, assuming that, during the execution of `BICONNECTED-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. Therefore $n \cdot m$ is an upper bound for the cycle 6-8. Analogously it can be verified that also the cycle 9-12 has the same upper bound. This means that $l + (n \cdot m)$ is an upper bound for the problem to verify if a Power Frame satisfies the composition property.

With respect to the problem to find all the subsets of a singleton Power Frame that satisfy the composition property, consider the case of a singleton Power Frame containing only commitments of in the form $(\{ag_i\}, g)$ and such that no two agents desire the same goal. In this case we can represent *pfr* as a directed graph $\mathbf{G}[pfr]$, as we do not have AND-arcs, where each arc (u, v) univocally corresponds to a goal. The problem to find all the subsets that satisfy the composition property corresponds to find all the subgraphs that are the strongly connected subgraphs and such that the relative undirected graph, $\mathsf{G}[pfr]$, is biconnected. Since an hamiltonian cycle in *pfr*, if any exists, satisfies the previous two conditions we have to find a set of subgraphs that contains all the hamiltonian cycles of $\mathbf{G}[pfr]$. However, this problem is expo-

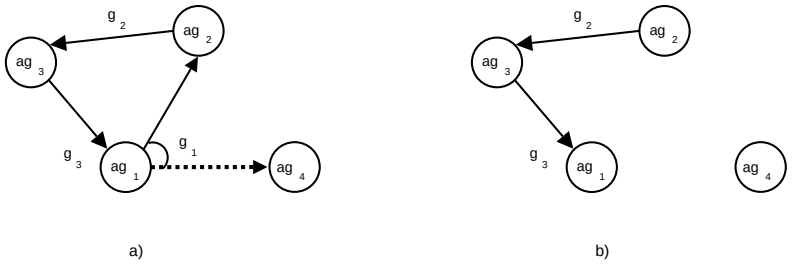

Figure 3.7: *The directed graph of the tagged AND-graph in (a) is dot strongly connected. By deleting the AND-arc relative to the goal g_1 the graph has no strongly-connected components (b).*

nential with respect to the number of arcs l. In contrary case, since checking if a subgraph of G is an hamiltonian cycle is linear with the cardinality of the nodes V, also the problem to find an hamiltonian cycle would be polynomial with respect to number of arcs.

In the case a Power Frame *pfr* can be represented by the corresponding directed graph $\mathbf{G}[pfr]$, the set of subgraphs to check is equal to 2^l. However, if *pfr* does not satisfy the composition property, then either $\mathbf{G}[pfr]$ is not strongly connected or $\mathsf{G}[pfr]$ has more than a component as calculated in the lines 5-12. In both case `FIND` is called directly on the subgraphs calculated in lines 20-21 and 27-28. So if there are k of these subgraphs, each of them with l_i AND-arcs, we have that the number of the graph which remain to be verified is $2^{l_1} + \cdots + 2^{l_k}$ instead of (approximately from below) $2^{l_1+\cdots+l_k} - 1$. If *pfr* does not satisfy the composition property, in the worst case $\mathbf{G}[pfr]$ is not strongly connected and it has one strongly component with $l-1$ arcs, as in Example 6. In this case 2^{l-1} graphs remain to be verified instead of $2^l - 1$.

We notice that this fact occurs not only once, but every time that a subset of *pfr* does not satisfy the composition property. Moreover, if the original graph is a proper AND-graph $\mathcal{G}$ 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 $\mathbf{G}$. The tagged AND-graph $\mathcal{G}[pfr]$ in Figure 3.7 (a) shows an example of this fact. The dotted arc is a bridge in the corresponding directed graph $\mathbf{G}[pfr]$, i.e. it is not contained in any strongly connected components, so *pfr* does not satisfy the composition property. The procedure `FIND` is then called on the tagged AND-graph obtained removing the AND-arc tagged with the goal g_1, Figure 3.7 (b), and this graph has not strongly connected components. This means that after 2 recursive calls,

and not $2^{|pfr|} = 8$, the procedure FIND returns the empty set as the only AND-graph satisfying the composition property.

Finally, Figure 3.8 shows the execution of the algorithm FIND on the tagged AND-graph relative to Example 10. First of all Example 10 satisfies all the assumption made in Section 3.10, therefore FIND can be directly applied. Each box in Figure 3.8 represents a recursive call of the algorithm FIND. The flow of the recursive calls is represented by the arrows connecting the boxes. As you can see the procedure FIND is called twice to the same input, corresponding to the box outlined by a bold dotted line. This leads to an overhead which slackens the algorithm with useless computation. A possible solution consists of standard memoizing technics [CR90], i.e when a recursive call of FIND terminates on a singleton Power Frame pfr, a key for pfr is stored in a data structure I, where I could be for example a balanced binary search tree [CR90]. This way, before executing a recursive call on a certain singleton Power Frame, its key is searched in I and, if it is found, then FIND has already checked it and its subsets. Both inserting and searching operations in balanced search trees are logarithmic with respect to the number of elements stored. As in our case the number of subsets of a Power Frame pfr is equal to $2^{|pfr|}$, then memoizing weights on each recursive call of the algorithm FIND an overhead that is proportional to the cardinality of pfr, and hence it does not modify considerably the complexity of a single call of FIND.

Each box in FIND consists of one, two or three parts, in the first part, on the left side of the box, the input $\mathcal{G}[pfr]$ of the recursive call is represented. For space reason the tags corresponding to the goals and the agents are omitted (Figure 3.4 shows a complete representation of Example 10). The strongly connected components, if there exist any, are represented in the middle part. If the directed graph $\mathbf{G}[pfr]$ relative to $\mathcal{G}[pfr]$ is strongly-connected, the third part of a box represents of components of the undirected graph $\mathsf{G}[pfr]$ calculated in lines 5-12. If only one component results, then pfr satisfies the composition property. Boxes outlined with a bold line corresponds to the Power Frames that satisfy the composition property. The relative inputs are the same represented in Figure 3.4 (b), (c), (d) and (e).

As the cardinality of the Power Frame in Example 10 (a) is 7, $2^7 = 128$ subsets of pfr should be checked by the algorithm FIND. However, assuming that our FIND algorithm uses memoization, the number of recursive calls corresponds to the number of the arrows connecting in Figure 3.4 the boxes. This number is equal to 16, i.e. only the 12,5% of the number of all subsets of pfr.

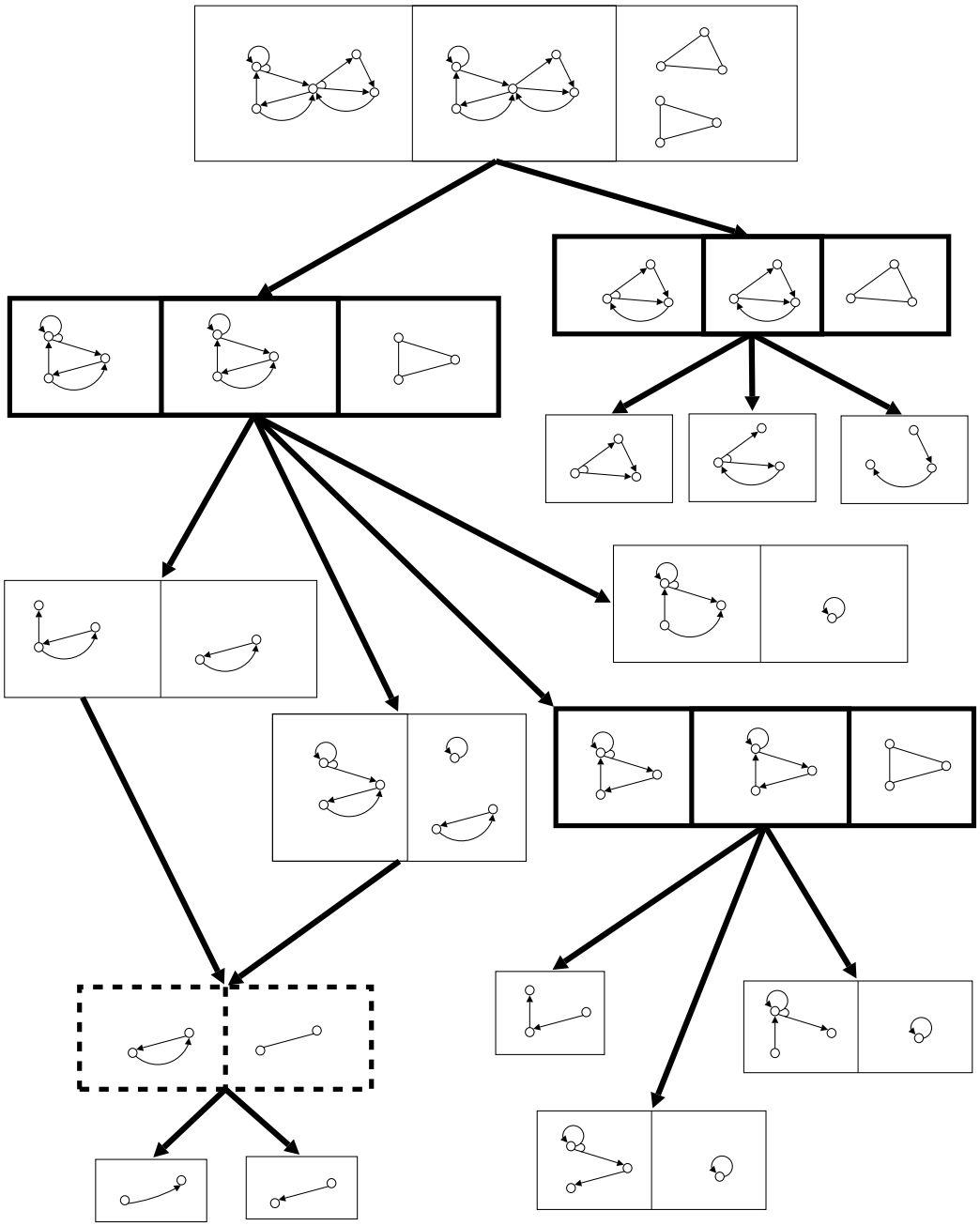

Figure 3.8: *Execution of the algorithm* `FIND` *on the Power Frames of Example 10.*

3.11 A final example

In this section we discuss a final example, this example is a simple version of the game described in Grosz et al. [GKT+04] describing a general problem of exchange of resources. This example emphasizes how the do-ut-des property and the composition property select a restrict set of all the Power Frames of a Power Structure.

Example 11 *There are four agents and each of them has two colored chips and two colored boxes, both the chips and the boxes can be colored as follows: red r, green g or blue b. Agents can maintain each of their own chips or give it to some other agent. Every agent has only one goal, to fill each of his boxes with a chip of the same color of the box. The initial configuration is:*

agents	*boxes*	*chips*
ag_1	$r\ r$	$r\ g$
ag_2	$r\ g$	$b\ b$
ag_3	$b\ g$	$r\ b$
ag_4	$b\ b$	$r\ g$

In the initial configuration all the agents are not self-sufficient to see to its own goal, so agents have to exchange chips with each other. Since chips can be given or maintained without any restriction, this means that the total of eight chips owned by the agents can be assigned to the four agents in any possible combination. Therefore, there are $8^4 = 4096$ combinations from which we want to find those that satisfy the do-ut-des property and the composition property. 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. Other configurations assign more that two goals to an agent, also these configurations are not plausible as they involve a waste of chips. The Power Structure can initially take into account these considerations in order to limit the search space.

The set of agents is $Ag = \{ag_1, ag_2, ag_3, ag_4\}$, 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 ag_2 and ag_3 has the power to achieve the goal bb of the agent ag_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. This interpretation of the notion of power seems reasonable because an agent can use its chips that are not directly useful for the achievement of its own goal in order to obtain a useful chip in exchange, therefore it does not want to

waste its chips in exchanges with agents that are not actually required for the achievement of its goals. We also assume that the relation power describes only the capability of groups of agents to actually achieve a goal, i.e. the relation power does not describe all the exchanges of the agents that does not achieve any goal, moreover since the goals can be always achieved individually we assume that all the commitments regards the achievement of a single goal, this entails that we consider singleton Power Frames.

For reason space, we show the relation pow 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
$\{ag_1, ag_3\}$, $\{ag_1, ag_4\}$, $\{ag_3, ag_4\}$	rr
$\{ag_1, ag_3\}$, $\{ag_1, ag_4\}$, $\{ag_1\}$, $\{ag_4\}$, $\{ag_1, ag_4\}$	rg
$\{ag_1, ag_2\}$, $\{ag_1, ag_3\}$, $\{ag_2, ag_4\}$, $\{ag_3, ag_4\}$	bg
$\{ag_2\}$, $\{ag_2, ag_3\}$	bb

The relation comp is defined by considering that each goal requires to use chips distinct from the ones used in the achievement of the another goals. Therefore a subset of pow is compatible, equivalently is a Power Frame pfr, if and only if 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 $(\{ag_3, ag_4\}, bg)$ and $(\{ag_3, ag_4\}, rr)$ is compatible because the agents use in the achievement of bg and rr different chips. On the contrary the commitments $(\{ag_1\}, rg)$ and $(\{ag_1, ag_3\}, rr)$ are incompatible as the total amount of red chips at their disposal is equal to two 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 ag_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.

Theorem 6 shows that if a Power Frame is do-ut-des, then there do not exist two groups in it achieving the same goal (no repeated goals). Therefore, we can restrict our search to the Power Frames which satisfy this condition. We find these Power Frames by solving a constraint satisfaction problem as follows: we select a group of agents that can achieve the goal rr, for example $\{ag_1, ag_3\}$, then we see which groups of agents has the power to achieve the goal rg compatibly with the fact that $\{ag_1, ag_3\}$ achieves rr, and so on up to the goal bb. The result of this constraint satisfaction problem is shown in

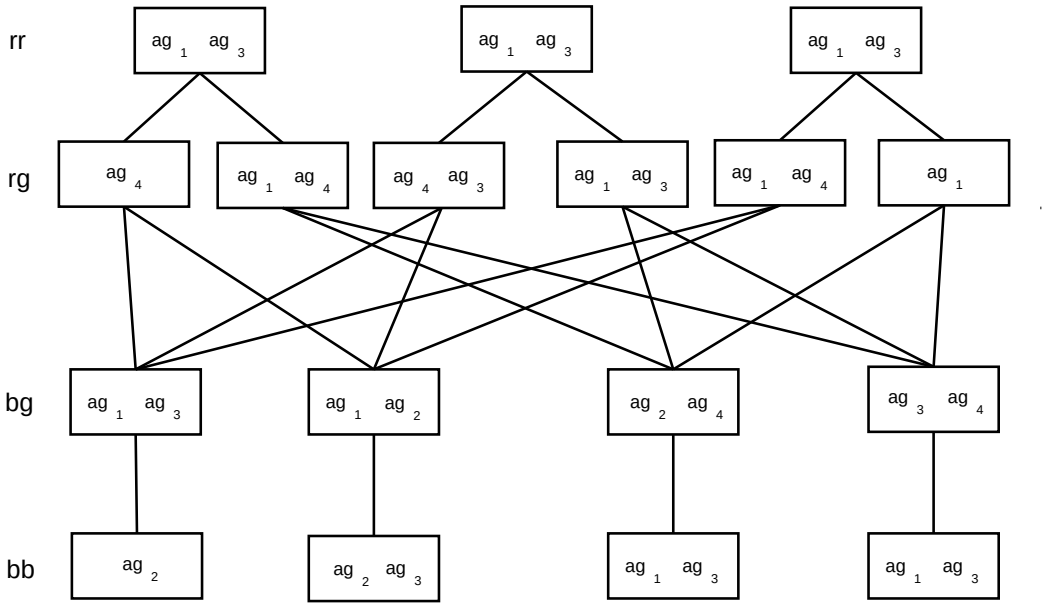

Figure 3.9: *The Power Frames in Example 11.*

Figure 3.9, every path starting from the top up to the bottom of the graph constitute a maximal Power Frame satisfying the condition of no repeated goals. A generic Power Frame with no repeated goals consists of a set of groups of agents lying in the same path and the corresponding goals. The number of such Power Frames is 157.

All these maximal Power Frame 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 Power Frame pfr_1 *composed by the commitments* $(\{ag_1, ag_3\}, rr)$ *and* $(\{ag_1, ag_3\}, bg)$ *is also do-ut-des. Analogously, another do-ut-des Power Frame* pfr_2 *is composed by the commitments* $(\{ag_2\}, bb)$ *and* $(\{ag_4\}, rg)$*. Applying the Theorem 6 it can be shown that these are the only Power Frames satisfying the do-ut-des property, therefore the number of Power Frames is equal to 14, i.e. about the 9% of the set of Power Frames satisfying the no repeated goals condition. For the composition property the do-ut-des power frame given by the union of* pfr_1 *and* pfr_2 *does not satisfy the composition property since* ag_1 *and* ag_3 *which are involved in* pfr_1 *are not interested in* pfr_2 *and vive versa* ag_2 *and* ag_4 *are not interested in* pfr_1*. All the other do-ut-des Power Frame satisfy the composition property.*

3.12 Summarizing

In this chapter we have defined an abstract structure, the Power Structure, which describes which goals different groups of agents can achieve collaborating. We have seen how different notions of power lead to Power Structures satisfying different properties. We have defined some of these properties (corresponding, for example, to the ones shown in Section 2.6.2) and we have shown which relations occur among them.

Starting from Power Structures as representation of a multiagent system we have defined the set of possible agreements that agents can reach, the Power Frames. Power Frames describe in our setting which coalitions can come out. The main aim of this thesis is to define some admissibility criteria that prune those coalition that cannot be formed under the assumption of self-interested agents. We have defined two admissibility criteria: the do-ut-des property and the composition property. The first property formalizes a notion of reciprocity stating that each agent takes part in the satisfaction of a goal only if this returns to it, directly or indirectly, the satisfaction of one of its goals. The second property the composition property is a refinement of the do-ut-des property which takes into account the fact 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 as the coalition formation processes can itself be costly and because, as we consider unanimously agreements, the more are the agents involved in a coalition the more is the risk that one of them gives up to join the coalition.

Both the two admissibility criteria have been defined as a no-dominance condition on some qualitative preferences of the agents. We think, as said in Section 1.4, that this formalization inspired by Game Theory improves the evidence of our admissibility criteria. However, we have provided a second characterization of these properties for a particular class of Power Frames, the singleton Power Frames. This characterization is based on the chains of exchanges which occurs in a Power Frame. Singleton Power Frames corresponds in our setting to the generalized exchanges considered in [CS02b]. In particular it is shown that the do-ut-des property is an extension of the notion of reciprocity defined in [CS02b] (see Definition 6) which takes into account that agents may obtain more than a single goal in a singleton Power Frame.

Finally, we have provided an algorithm to search the singleton Power Frames which satisfies the composition property. Even if this problem is not computationally tractable, we have shown that the problem to verify if a single Power Frame satisfies the composition property is tractable and, in several case, also the complexity of the first problem may decrease considerably

every time a singleton Power Frame which does not satisfy the composition property is found.

Chapter 4

Comparison with Game Theory

4.1 Introduction

In Chapter 3 we have defined two admissibility criteria establishing which coalitions can be proposed in a multiagent system populated by goal-directed agents. We have assumed that goals represent satisfaction conditions on different and not directly comparable objectives. This way, our admissibility criteria are intrinsically qualitative as they characterize the strategic behavior of agents which are qualitative decision makers. However, as our admissibility criteria have been founded on some notions of dominance, it is interesting to compare our approach to the game theoretical one for at least two reasons.

First, coalition formation has been one of the first research interests in modern Cooperative Game Theory [vNM44, Aum61]. Therefore, emphasizing some relationships with the results obtained in Cooperative Game Theory provide a further benchmark for our approach. Second, our admissibility criterion could be viewed as some qualitative reasoning to restrict the search space of quantitative admissibility criteria developed in Game Theory. This could be profitable as our criteria do not require, as in Cooperative Game Theory, to compare a coalition with all the other possible coalitions, but only with its subsets.

We restrict our analysis to a comparison between the do-ut-des property and a well known game theoretical criterion, the core. We have considered only the do-ut-des property because the notion of core is focussed only on the set of consequences that the grand coalition, i.e. the coalition formed by all the agents, can attain. The composition property is on the contrary inspired by the fact that the grand coalition could not be considered an admissible coalition if it is composed by independent sub-coalitions. So, the composition property seems to focus on an aspect that is not taken into account by the notion of core. In Cooperative Game Theory several solution criteria have been developed, such as stable sets, the kernel, the nucleolus. However, on one hand, the notion of stable sets is weaker than the notion of core as the core is a subset of each stable set [OR94]. On the other hand, the notions of kernel and nucleolus can been defined only in the case the utilities of the agents can been meaningfully compared. This fact occurs in case of games with transferable utilities, whereas, as we will see, we deal with games without transferable utilities.

Section 4.2 describes how in Game Theory collaborative behaviors of groups of agents are formalized starting from the strategies at the disposal of the individual agents. Section 4.3 concerns the definition of a cooperative game and the definition of the notion of core. In Section 4.4 we compare, for a given class of quantitative preferences of the agents, the do-ut-des property with the notion of core. Finally in Section 4.5 the preferences of the agents

are formalized be means of a cost-benefit analysis, and it is shown which class of cost-benefit analysis can be used to compare the do-ut-des property with the notion of core.

4.2 Representing collaborative behaviors in games

In Game Theory a game can be represented at different levels of abstraction. A large part of the game theoretical admissibility criteria, such as the Nash equilibrium, are defined by representing a game in the strategic form. The strategic form of a game abstracts from the description of the game rules as well as from the nature of the single strategies (whether they consist of a single move or sequence of moves). A game consists of a set of n agents Ag, a set of possible states St, for each agent ag_i a set of strategies Σ_i available to it and a function *outcome* that associates each strategy profile σ_{Ag} (i.e. a n-dimensional vector of strategies with each components corresponding to the strategy of a different agent) with a state which results when each agent acts according to σ_{Ag}.

Definition 29 (Game in strategic form) *A game represented in strategic form is a tuple:*

$$\langle Ag, \Sigma_1, \ldots, \Sigma_n, St, outcome \rangle$$

where Ag is a set of n agents, Σ_i is a nonempty set of strategies available to the agent ag_i, St is a set of states, and $outcome : \Sigma_1 \times \cdots \times \Sigma_n \to St$ is a surjective function that associates each strategy profile $\sigma_{Ag} \in \times_i \Sigma_i$ with a state in St.

A game in strategic form can be represented by a n-dimensional matrix corresponding to the values of the function *outcome*.

Example 12 *Consider a simple case where there are only two agents, ag_1 with available strategies L, C and R, and ag_2 with strategies T, M and B, a possible game in strategic form is represented in Figure 4.1. So, for example, the outcome of the strategy profile (R, M) is the state s_8 and the outcome of the strategy profile (L, T) is the state s_1.*

σ_Q denotes, with the obvious meaning, a strategy profile $(\sigma_i)_{ag_i \in Q}$ of the group of agents Q. Given a partition $Q_1, \ldots, Q_n$ of Ag we write $\sigma_{Q_1}, \ldots, \sigma_{Q_n}$ to indicate the strategy profile of σ_{Ag} such that, for each Q_i, its projection to Q_i is equal to σ_{Q_i}.

	L	C	R
T	s_1	s_2	s_5
M	s_3	s_4	s_8
B	s_3	s_6	s_7

Figure 4.1: An example of game in strategic form

Starting from the strategic representation of a game, we define the notion of α-ability and β-ability already described in Alternating Time Temporal Logic (see Section 2.6). Historically α and β-ability was one of the first formalizations of the effectivity function, which describes the sets of outcomes that groups of agents can force if they decide to collaborate [Aum61, Ros72, Pel98, Pau02a]. Informally, a group of agents Q is α-able to force the set of states $S \subseteq St$ if there exists a strategy profile of Q such that, if executed, no matter what the other agents do, the outcome is a state in S.

Definition 30 (α-ability) *Let $\langle Ag, \Sigma_1, \ldots, \Sigma_n, St, outcome \rangle$ be a game in strategic form, a group of agents Q is α-able to force the outcome to be in the set $S \subseteq St$ iff*

$$\exists \sigma_Q \forall \sigma_{Ag \setminus Q} \ [outcome(\sigma_Q, \sigma_{Ag \setminus Q}) \in S]$$

β-ability is weaker notion of effectivity, Q is β-able to force $S \subseteq St$ if for all the strategies of the other agents, there exists a strategy profile of Q that force the outcome to be in S.

Definition 31 (β-ability) *Let $\langle Ag, \Sigma_1, \ldots, \Sigma_n, St, outcome \rangle$ be a game in strategic form, a group of agents Q is β-able to force the outcome to be in the set $S \subseteq St$ iff*

$$\forall \sigma_{Ag \setminus Q} \exists \sigma_Q \ [outcome(\sigma_Q, \sigma_{Ag \setminus Q}) \in S]$$

Example 13 *Let consider again the example in Figure 4.1. The agent ag_1, by choosing a certain strategy, is α-able to force the outcome to be in S, for each S consisting of a triple of states in a certain column. So, for example, by choosing the strategy R, ag_1 is α-able to force the outcome to be one of s_5, s_8 or s_7.*

The agent ag_2 is β-able to force the outcome to be in S, for each S consisting of a triple of states such that the first state of the triple is the first row, the second in the second row and the third in the third row. For example, ag_2 is β-able to force the outcome to be one of s_1, s_4 and s_7. Indeed if ag_1 chooses the strategy T, ag_2 chooses the strategy L and the outcome is

	side 1	side 2
side 1	win ag_1	win ag_2
side 2	win ag_2	win ag_1

Figure 4.2: Matching pennies game

s_1. Analogously if ag_1 chooses the strategy M or D, ag_2 chooses respectively C and R obtaining s_4 or s_7.

α and β-ability are used in different scenarios. α-ability represents the effectivity of a group of agents when the member of Q cannot observe the strategies adopted by the other agents and behave consequently.

On the other hand β-ability is well suited when the agents in Q are able to observe in advance the behaviors of the others, but it cannot be used when the agents cannot observe each other as the following game shows:

Example 14 (Matching pennies) *There are two agents ag_1 and ag_2, each of them having a coin. At the same time they choose a side of its own penny, if the two pennies match ag_1 wins, otherwise he loses (see Figure 4.2).*

Note that each agent in the previous example is trivially β-able to win, so β-ability cannot describe the capabilities of the player, because they cannot both win. Here the problem is that even if each agent has, depending on the strategy adopted by the other agent, a set of counter-strategies to win, the intersection of these sets is empty, and it cannot foresee which of them to select because the sides of the pennies are selected at the same time.

The definitions of α and β-ability provide two different methods to calculate the effectivity of groups of agents starting from the strategic representation of a game. It is important to notice that the notion of effectivity is by no means in principle bound to these two notions, but a game can be directly described by means of the effectivity of groups of agents. This representation of a game, called the effectivity form of a game, is more abstract than the strategic representation, since it does not describe the possible choices available to the single agents. An effectivity function *eff* associates each group Q of agents with a set $\{S_1, \ldots, S_m\}$, where each S_i is a set of states. $S_i \in eff(Q)$ means that the agents in Q are able to collaborate in such a manner to force the outcome to be in S_i. The effectivity function is assumed to satisfy four conditions[1]. The first condition is that if no group decides to cooperate, then all the states in St are possible. The second condition is that if the agents

[1]The effectivity functions are studied at different extent in several works as [Ros72, Pel98, Aum61, Pau02a]. Here, for historical reasons, we report the definition of effectivity

decide to cooperate all together, then they can force any nonempty set of states. These two conditions means that, in a game, the states represent all the possibilities of the agents to modify the environment, therefore states that are not in any case available to the agents are not considered. The third and fourth conditions state that, however a group of agents Q decides to collaborate (or eventually decides to not collaborate), a state has to come out, and this state is included in the set of all the states St. In other words, the empty set cannot be effective for Q, whereas the set of all the states St is.

Definition 32 (Game in effectivity form) *A game in effectivity form is a tuple:*

$$\langle Ag, St, \mathit{eff} \rangle$$

where Ag is a set of agents, St is a set of states and $\mathit{eff} : 2^{Ag} \to 2^{2^{St}}$ satisfies the following conditions:

- $\mathit{eff}(\emptyset) = \{St\}$.
- $\mathit{eff}(Ag) = 2^{St} \setminus \emptyset$.
- *for all* $Q \subseteq Ag$, $\emptyset \notin \mathit{eff}(Q)$.
- *for all* $Q \subseteq Ag$, $St \in \mathit{eff}(Q)$.

As you can expect from Sections 2.6.2 and 3.3, the effectivity function of a game may satisfy the further conditions [Pel98, Pau02a]:
outcome monotonic: for all $Q \subseteq Ag$ and for all $S, S' \subseteq St$, if $S \in \mathit{eff}(Q)$ and $S \subseteq S'$, then $S' \in \mathit{eff}(Q)$.
coalitional monotonic: for all $Q, Q' \subseteq Ag$, if $Q \subseteq Q'$, then $\mathit{eff}(Q) \subseteq \mathit{eff}(Q')$.
super-additivity: For all $Q, Q' \subseteq Ag$ and for all $S, S' \subseteq St$, if $S \in \mathit{eff}(Q)$, $S' \in \mathit{eff}(Q')$ and $Q \cap Q' = \emptyset$, then $S \cap S' \in \mathit{eff}(Q \cup Q')$.

Under the conditions on the effectivity function described in Definition 32, these properties are not all independent, in particular it is straightforward to see that if an effectivity function is super-additive, then it is also coalitional monotonic.

Starting from the notion of α-ability an effectivity function eff_α can be defined as follows: $S \in \mathit{eff}_\alpha(Q)$ iff Q is α-able to force the outcome to be in S. In the same manner can be defined an effectivity function eff_β from the notion of β-ability. In particular, eff_α is super-additive and outcome monotonic, whereas eff_β is coalitional monotonic and outcome monotonic.

function due to Peleg [Pel98], however Pauly [Pau02a] presents a slightly different formalization of the notion of effectivity in which, in particular, the condition 2 of Definition 32 does not hold.

4.3 Admissibility criteria in Cooperative Game Theory

In Section 4.2 we have seen how to formalize the capabilities of groups of agents in terms of the states they can force collaborating. Nevertheless the main aim of Cooperative Game Theory regards the study of admissibility criteria in coalition formation.

An important distinction in Cooperative Game Theory regards whether the utilities that a group of agents is capable to attain can be redistributed among the members, *transferable utilities*, or not, *non-transferable utilities*. For our purposes we consider only games without transferable utilities. Following Osborne et al. [OR94] a cooperative game without transferable utilities is defined as follows:

Definition 33 (Cooperative games without transferable utilities) *A cooperative game without transferable utilities NTU is a tuple*

$$\langle Ag, Cs, att, \precsim_1, \ldots, \precsim_n \rangle$$

where Ag is a set of agents. Cs is a set of consequences. $att : 2^{Ag} \setminus \emptyset \rightarrow 2^{Cs}$ maps each nonempty group Q of agents with the set of consequences that are attainable by Q. For each agent ag_i, $\precsim_i$ is a preference relation, i.e. a total reflexive and transitive binary relation over Cs. Given two consequences $c_1, c_2 \in Cs$, with $c_1 \precsim_i c_2$ we intend that the agent ag_i prefers c_2 at least as c_1.

In many scenarios the preference relations over the consequences are represented by real-valued utility functions. In this case a cooperative game consists of a function, the *characteristic* function of the game, that maps a group of agents Q to a set of real-valued vectors (whose dimensions are equal to the cardinality of Q), which describe the attainable utility distributions for Q.

Several solution criteria have been proposed, as for example the notion of core, stable sets, nucleus. The notion of core is, as for the Nash equilibrium in Non-cooperative Game Theory, based on a dominance relation over the set of possible consequences attainable by the grand coalition Ag. $c \in att(Ag)$ is dominated if there exists a group of agents able to achieve a consequence which is strictly preferred to c by all its members.

Definition 34 (Core) *Let $\langle Ag, Cs, att, \precsim_1, \ldots, \precsim_n \rangle$ be a cooperative game without transferable utilities, the core is the set of consequences $c \in att(Ag)$ such that there does not exist a group of agents $Q \subseteq Ag$ and a consequence $c' \in att(Q)$ such that $c \prec_i c'$, for all $ag_i \in Q$.*

Given a game represented in the effectivity form $\langle Ag, St, \mathit{eff}\rangle$, the consequences Cs in Definition 33 are naturally represented by the nonempty sets of states $S \subseteq St$ and the function att is represented by the function eff. In order to define a cooperative game we need the preference relations $\precsim_i$ which describes when an agent ag_i prefers a set of states S_2 at least as another set of states S_1. However, the preferences of the agents are usually modeled as a relation $\preccurlyeq_i$ over the states, and not over sets of states. A way to calculate the $\precsim_i$ starting from the $\preccurlyeq_i$ derives from the notion of assurable utility [Mye97]. Given a set of states S we consider the set of states which are less preferred than all the states in S under $\preccurlyeq_i$. This is the set of states dominated by S, i.e. with respect to which S assures a better outcome for the agent ag_i. S_1 is preferred at most as S_2 for ag_i if the set of states dominated by S_1 is contained in the one dominated by S_2.

Definition 35 (Assurable preference relation) *Given a set of n agents Ag, a set of states St and for each $1 \leq i \leq n$ a preference relation $\preccurlyeq_i \subseteq St \times St$ of the agent ag_i, we denote with $\preccurlyeq_i[S]$, where $S \subseteq St$, the set of states $s' \in St$ such that $s' \preccurlyeq_i s$, for all $s \in S$. The assurable preference $\precsim_i$ relative to $\preccurlyeq_i$ is such that for all $S_1, S_2 \in St$, $S_1 \precsim_i S_2$ iff $\preccurlyeq_i[S_1] \subseteq \preccurlyeq_i[S_2]$.*

We finally notice that there exists a literature on how to describe cooperative behaviors directly in the normal form of a game, adopting as admissibility criteria some variation of the admissibility criteria developed in Non-cooperative Game Theory. These kinds of games are called claim games and are source of interest in order to see which relation occur between Cooperative and Non-cooperative game theory [DvdNT98, MMSSG98]. Generally the admissibility criterion used in this settings is the strong Nash equilibrium, i.e. a Nash equilibrium where there are considered as possible the deviations of entire groups of agents instead of the deviation of single agents.

4.4 Do-ut-des property and Cooperative games

In this section we study the relation between the do-ut-des property and the game theoretical notion of core. We show that do-ut-des property can be used in a cooperative game without transferable utilities as a qualitative method to reduce the set of coalitions on which to apply quantitative reasoning.

First of all we define a cooperative game relative to a Power Structure $PS = \langle Ag, Gl, \mathit{goals}, \mathit{pow}, \mathit{comp}\rangle$. In particular we have to describe the consequences a group Q of agents can attain if they decide to form a coalition. Wooldridge et al. in [WD04] formalize the consequences as sets of goals and, the consequences attainable by a group Q of agents as the sets of goals G

such that $(Q, G) \in pow$. This 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. In contrast we consider Power Frames as consequences. Since a Power Frame *pfr* with domain contained in Q represents a conflict free distribution of commitments internal to Q, the set of consequences that Q can attain are all the Power Frames such that the set of agents involved is contained in Q. Finally, for each agent ag_i a preference relation $\precsim_i$ enables to compare different consequences.

Definition 36 (NTU of a Power Structure) *Given a Power Structure* $PS = \langle Ag, Gl, pow, goals, comp\rangle$*, we say that* $\langle Ag, Cs, att, \precsim_1, \ldots, \precsim_n\rangle$ *is a cooperative game relative to PS, NTU*$[PS]$*, iff Cs is the set of all the Power Frames* $pfr \in comp$*, att maps for each set of agents* Q *the set of Power Frames* pfr *such that* $\bigcup_{(Q',G')\in pfr} Q' \subseteq Q$ *and the* $\precsim_i$ *are the preference relations over Cs of the agents in Ag.*

In particular each Power Frame $pfr \in comp$ is attainable by the grand coalition, so the notion of core takes into account all the possible coalitions that can be formed.

The previous definition does not provide any restriction to the preference relations $\precsim_i$ of the agents. Nevertheless the notion of do-ut-des Power Frames can be related to the quantitative notion of core only in the case the $\precsim_i$ are *compatible* with the do-ut-des preference relation (see Definition 18), i.e. for any agent ag_i, $\leq_i$ implies $\precsim_i$ and $<_i$ implies $\prec_i$.

Definition 37 (Do-ut-des compatible NTU) *Given a Power Structure PS, a relative cooperative game* $NTU[PS] = \langle Ag, Cs, att, \precsim_1, \ldots, \precsim_n\rangle$ *is do-ut-des compatible iff for all* $ag_i \in Ag$ *and for all Power Frames* pfr_1, pfr_2 *(1) if* $pfr_1 \leq_i pfr_2$*, then* $pfr_1 \precsim_i pfr_2$ *and (2) if* $pfr_1 <_i pfr_2$*, then* $pfr_1 \prec_i pfr_2$*.*

In Section 4.4 we will show that a reasonable class of preference relations is do-ut-des compatible.

In order to relate the qualitative do-ut-des property to the notion of core we consider a quantitative version of the do-ut-des property, we call it q-do-ut-des property. We use this property to relate the do-ut-des property to the notion of core, so we want that this property has a precise relationship with both of them. Roughly, the q-do-ut-des property uses the relation of dominance already used for the do-ut-des property (see Definition 19), where the do-ut-des preference relations $\leq_i$ are substituted with the total preference relations $\precsim_i$.

Definition 38 (Q-do-ut-des) *Let $NTU[PS] = \langle Ag, Cs, att, \precsim_1, \ldots, \precsim_n \rangle$ be the cooperative game of the Power Structure PS. A Power Frame pfr is q-do-ut-des iff there does not exist a Power Frame pfr' such that:*

1. *there exists an agent ag_i such that $pfr \prec_i pfr'$.*
2. *for all $ag_j \ltimes pfr'$, $pfr \precsim_j pfr'$.*

The following theorem shows that if a Power Frame *pfr* is q-do-ut-des, then it is in the core.

Theorem 10 *Let $NTU[PS] = \langle Ag, Cs, att, \precsim_1, \ldots, \precsim_n \rangle$ be a cooperative game of the Power Structure PS. If a Power Frame pfr is q-do-ut-des, then it is in the core.*

proof: If *pfr* is not in the core, then there exists an not empty group of agents Q and a Power Frame $pfr' \in att(Q)$ such that for all the agents $ag_i \in Q$, $pfr \prec_i pfr'$. Therefore, there exists an agent that strictly prefers pfr' to *pfr*. Moreover, since all the agents involved in pfr' are in Q, $pfr \precsim_i pfr'$ is true for all $ag_i \ltimes pfr'$. But this means that pfr' q-do-ut-des dominates *pfr*, and hence that *pfr* is not q-do-ut-des. □

Moreover, we show that, for do-ut-des compatible NTU, if a Power Frame *pfr* is q-do-ut-des, then it is a do-ut-des.

Theorem 11 *Let $NTU[PS] = \langle Ag, Cs, att, \precsim_1, \ldots, \precsim_n \rangle$ be a do-ut-des compatible cooperative game of the Power Structure PS, if a Power Frame pfr is q-do-ut-des, then it is do-ut-des.*

proof: Assume that *pfr* is q-do-ut-des, but not do-ut-des. By definition this means that there exists a Power Frame $pfr' \subset pfr$ and an agent $ag_i \ltimes pfr$ such that $pfr <_i pfr'$ and for all $ag_j \ltimes pfr'$, $pfr \leq_j pfr'$. Since $NTU[PS]$ is do-ut-des compatible, then it is also the case that $pfr \prec_i pfr'$ and for all $ag_j \ltimes pfr'$, $pfr \precsim_j pfr'$. But this means that pfr' q-do-ut-des dominates *pfr* against the hypothesis. □

The previous two theorems show that the set of q-do-ut-des Power Frames is contained in both the core and the do-ut-des Power Frames, so if a Power Frame *pfr* is not do-ut-des, then we are sure that there exists a subset of the core in which it cannot be included. Furthermore, we show that the Power Frames which do not satisfy the do-ut-des property are useless to see if another Power Frame is q-do-ut-des. This fact has also some relevance from a procedural point of view, indeed the definition of q-do-ut-des Power Frames requires to compare a Power Frame *pfr* with all the other Power Frames. Due to Theorem 11, we can restrict the set of Power Frames on

which to check for q-do-ut-des Power Frames to the set of do-ut-des Power Frames. However, for each do-ut-des Power Frame, we still should compare it with of all the other Power Frames. The following theorem shows that if *pfr* is not q-do-ut-des, then there exists a do-ut-des Power Frame that dominates it. So we do not need to compare a do-ut-des Power Frame with all the others Power Frames in order to see if it is q-do-ut-des, but only with the other do-ut-des Power Frames.

Theorem 12 *Given a do-ut-des compatible cooperative game NTU[PS] of a Power Structure PS, if a Power Frame pfr is not q-do-ut-des, then there exists a do-ut-des Power Frame pfr' such that pfr' q-do-ut-des dominates pfr.*

proof: Assume that *pfr* is not q-do-ut-des, and *per absurdum* all the Power Frames pfr' that q-do-ut-des dominates *pfr* are not do-ut-des. So, for all these pfr' by definition we have that

- pfr' q-do-ut-des dominates *pfr*: there exists an agent ag_i such that $pfr \prec_i pfr'$ and for all $ag_j \ltimes pfr'$, $pfr \precsim_j pfr'$.
- pfr' is not do-ut-des: there exists a $pfr'' \subset pfr'$ and an agent $ag_h \ltimes pfr'$ such that $pfr' <_h pfr''$ and for all $ag_k \ltimes pfr''$, $pfr' \leq_k pfr''$.

Since *NTU[PS]* is do-ut-des compatible, it is also the case that

- $pfr' \prec_h pfr''$ and hence, being $ag_h \ltimes pfr'$, $pfr \precsim_h pfr' \prec_h pfr''$.
- for all $ag_k \ltimes pfr''$, $pfr' \precsim_k pfr''$ and hence, being $pfr'' \subset pfr'$, $pfr \precsim_k pfr' \precsim_k pfr''$.

But this means that pfr'' q-do-ut-des dominates *pfr*. Now let consider a sequence of Power frames $pfr_1, \ldots, pfr_m$, such that (1) $pfr_m = pfr''$, (2) for all $1 \leq l \leq m-1$, pfr_l do-ut-des dominates pfr_{l+1} (3) pfr_1 is do-ut-des. Such a sequence certainly exists, indeed let consider the set of all the sequences $pfr_1, \ldots, pfr_m = pfr''$ such that for all $1 \leq m-1$, pfr_i do-ut-des dominates pfr_{i+1}, and denote this set with $DUD[pfr'']$. In the case there is no a Power Frame that do-ut-des dominates pfr'', then there exists only one sequence in $DUD[pfr'']$ and it is composed by pfr'' itself. In contrary case, there exists at least a sequence in $DUD[pfr'']$ of length equal to 2. Now if $pfr_1, \ldots, pfr_m \in DUD[pfr'']$ and pfr_1 is not do-ut-des, then there exists a sequence of length $m+1$ in $DUD[pfr'']$, but this means that if for all $pfr_1, \ldots, pfr_m \in DUD[pfr'']$, pfr_1 is not do-ut-des, then $DUD[pfr]$ should have a not finite cardinality, but this is impossible since there exists only a finite number of sequences of Power Frames $pfr_1, \ldots, pfr_m = pfr''$ such that for all $1 \leq l \leq m-1$ $pfr_l \subset pfr_{l+1}$.

Following the proof done for pfr'', for each $1 \leq l \leq n$, pfr_l q-do-ut-des dominates pfr, but this means that pfr_1 is a do-ut-des Power frame which q-do-ut-des dominates pfr. □

Theorems 10, 11 and 12 provide some further indication of the correctness of the do-ut-des property. The Power Frames which are not admissible for the do-ut-des property can be completely ignored in a quantitative analysis of their profitability. This analysis is provided by the notion of q-do-ut-des property, and, to see that this notion *makes sense*, we proved that it is even more restrictive of a the notion of core which is a well-known admissibility criteria developed in Cooperative Game Theory.

We also notice that, in some cases, there could be a Power Frame which is in the core but it is not do-ut-des. However, the do-ut-des property may provide a more effective method to establish the profitability of a Power Frame. The following example shows one of these cases.

Example 15 *Let PS be the following Power Structure:* $Ag = \{ag_1, ag_2, ag_3\}$, *each* ag_i *desires only one goal and no two agents desires the same goal. We denote with* g_i *the goal desired by the agent* ag_i. *The relation pow is given by the following table*

	Q	G
1	$\{ag_1\}$	$\{g_2\}$
2	$\{ag_1\}$	$\{g_3\}$
3	$\{ag_2\}$	$\{g_1\}$

and a relation $comp = 2^{pow}$. *pow is a Power Frame, but it is not do-ut-des because* ag_1 *provides the goal* g_3 *to* ag_3 *without having from* ag_3 *any goal in exchange (the subset composed by the first and the third rows do-ut-des dominates pow). Assume a NTU[PS] in which the preferences* $\precsim_i$ *of the agents are represented by means of utility functions* $utl_i(pfr) = |adv[pfr](ag_i)| - |obl[pfr](ag)|$. *It can be seen that the preference relation* $\precsim_i$ *so defined are do-ut-des compatible. In NTU[PS] pow is in the core because all the Power Frames pfr strictly preferred to pow by an agent* ag_i *always involve another agent* ag_j *which does not strictly prefer them to pow. So, for example,* ag_1 *strictly prefers the Power Frame* pfr' *composed by the first and the third rows to pow, but this Power Frame involves the agent* ag_2 *which does not strictly prefers* pfr' *to pow.*

4.5 Do-ut-des compatible cost-benefit analysis

The results shown in Section 4.4 hold under the hypothesis that the preferences relations of a cooperative game $NTU[PS]$ are do-ut-des compatible. So it is an issue if do-ut-des compatible preference relations constitute a somewhat *eccentric* kind of preferences or they cover plausible domains. In this section we show a class of preference relations which satisfy this condition and we provide a general example which suggests that do-ut-des compatible preferences are not oddball preferences.

Often the preference relation of an agent with respect to a set of consequences is represented by an utility function [Mye97, Aum61]. An utility function is a real valued function over the set of consequences Cs that represents, for each consequence, the profitability of that consequence. We say that an utility function $utl : Cs \to \mathbb{R}$ represents a preference relation $\precsim$ just in the case that for all $c_1, c_2 \in Cs$, $c_1 \precsim c_2$ if and only if $utl(c_1) \leq utl(c_2)$.

A way to effectively *calculate* the utility of a consequence is by means of a cost-benefit analysis [KR76, BDH99]. The idea underling the cost-benefit analysis is that any consequence represents a state of affairs that involves some benefits, but also it requires to sustain some costs in order to be achieved. In our case a consequence is a Power Frame pfr and, given an agent ag_i, the benefits of ag_i with respect to pfr are estimated as a real valued function bnf_i depending on the part of its goals that are satisfied in pfr, i.e. $adv[pfr](ag_i)$. The costs of ag_i with respect to pfr depends on many factors as, for example, the burdens it has to sustain in pfr, but also the costs implicit to the coalition formation process relative to pfr or to the fact that, binding itself to satisfy some goals, it puts some constraints to its future behaviors. For all these reasons we consider that the costs relative to a Power Frame are estimated as a real valued $cost_i$ depending on the Power Frame as a whole.

The utility of a Power Frame pfr, for the agent ag_i, is represented by a function blc_i that balances the estimation of the benefits with the estimation of the costs of pfr. Therefore, the utility of the agent ag_i relative to pfr, is given by the formula $utl_i(pfr) = blc_i(bnf_i(adv[pfr](ag_i)), cost_i(pfr))$.

Now we show some sufficient conditions utl_i has to satisfy in order to be do-ut-des compatible. The first condition says that blc_i is a function strictly increasing in the first argument and strictly decreasing in the second argument. The second condition says that the more are the burdens of an agent more are the costs it has to sustain. The third condition says that adding a desired goal to set of desired goals involves a not null increasing of

the benefit function.

Definition 39 (Do-ut-des compatible utility function) *The utility function of an agent ag_i, $utl_i(pfr) = blc_i(bnf_i(adv[pfr](ag_i)), cost_i(pfr))$, is said to be do-ut-des compatible iff*

1. *for all fixed $\hat{x} \in \mathbb{R}$ and $\hat{y} \in \mathbb{R}$, $blc_i(x, \hat{y})$ and $blc_i(\hat{x}, y)$ are respectively strictly increasing and strictly decreasing.*

2. *given two Power Frames pfr_1 and pfr_2, if $obl[pfr_1](ag_i) \subseteq obl[pfr_2](ag_i)$, then $cost_i(pfr_1) \leq cost_i(pfr_2)$ and if $obl[pfr_1](ag_i) \subset obl[pfr_2](ag_i)$, then $cost_i(pfr_1) < cost_i(pfr_2)$.*

3. *given two Power Frames pfr_1 and pfr_2, if $adv[pfr_1](ag_i) \subseteq adv[pfr_2](ag_i)$, then $bnf_i(adv[pfr_1](ag_i)) \leq bnf_i(adv[pfr_2](ag_i))$ and if $adv[pfr_1](ag_i) \subset adv[pfr_2](ag_i)$, then $bnf_i(adv[pfr_1](ag_i)) < bnf_i(adv[pfr_2](ag_i))$.*

The following example shows an utility function which is do-ut-des compatible.

Example 16 *Assume that the agents ag_i associates each goal $g \in goals(ag_i)$ with a strictly positive value v_g and estimate the benefits deriving the achievement of a set of its goals, $adv[pfr](ag_i)$, in a Power Frame pfr as the sum of the relative values:*

$$bnf_i(adv[pfr](ag_i)) = \sum_{g \in adv[pfr](ag_i)} v_g$$

We assume that the achievement of a goal $g \in Gl$ requires the use of a certain amount of resources. If m is the number of all the possible resources that can be used for the achievement of a goal g, we formalize the amount of resources needed for the satisfaction of a goal g as a not-negative vector $\mathbf{r}_g = (\mathbb{R}_0^+)^m$. We assume that each goal requires the use of at least one resource, therefore $\mathbf{r}_g$ cannot be the null vector. A strictly positive cost $c_j > 0$ is associated to each type of resource $1 \leq j \leq m$, establishing a monetary equivalent of a unit of resource. Finally, for simplicity, we assume that the total cost that a group Q of agents have to sustain to achieve a goal is equally divided among the members of Q (this assumption is not essential, it suffices that each agent involved in the achievement of a goal has to sustain a not null cost). So, the cost an agent ag_i has to sustain with respect to a Power Frame pfr is give by the following formula:

$$cost_i(pfr) = \sum_{(Q,G) \in obl_{[pfr](ag_i)}} \frac{1}{|Q|} \sum_{g \in G} \sum_{1 \leq j \leq m} c_j \cdot (\mathbf{r}_g)_j$$

Where $(\mathbf{r}_g)_j$ *is* j^{th} *projection of the vector* $\mathbf{r}_g$. *Finally the utility function* utl_i *simply subtract the cost from the benefits relative to a Power Frame,* $utl_i(pfr) = bnf_i(adv[pfr](ag_i)) - cost_i(pfr))$.

We show that the function utl_i *is do-ut-des compatible. First of all, being* $blc_i(x, y) = x - y$, *it is strictly increasing with respect to the first argument and strictly decreasing with respect to the second argument and hance the first condition of Definition 39 is satisfied. The costs of an agent relative to a Power Frame depends only on the obligations it is involved in, i.e.* $cost_i(pfr) = cost_i(obl[pfr](ag_i))$. *Thus, given two Power Frames* pfr_1 *and* pfr_2, *if* $obl[pfr_1](ag_i) = obl[pfr_2](ag_i)$, *then the value of* $cost_i$ *is the same for* pfr_1 *and* pfr_2. *If* $obl[pfr_1](ag_i) \subset obl[pfr_2](ag_i)$, *then*

$$cost_i(pfr_2) = cost_i(obl[pfr_1](ag_i)) + cost_i(obl[pfr_2](ag_i) \setminus obl[pfr_1](ag_i))$$

and since $cost(pfr) > 0$ *for each not empty* pfr, *then* $cost_i(pfr_2) > cost_i(pfr_1)$. *Finally, assumed that* $adv[pfr_1](ag_i) \subseteq adv[pfr_2](ag_i)$, $bnf_i(adv[pfr_2](ag_i))$ *is equal to:*

$$bnf_i(adv[pfr_1](ag_i)) + bnf_i([adv[pfr_2](ag_i) \setminus adv[pfr_1](ag_i)])$$

Since in general $bnf_i(adv[pfr](ag_i)) \geq 0$, *where the equality holds only in the case* $adv[pfr](ag_i) = \emptyset$, *it holds that*

- *if* $adv[pfr_1](ag_i) \subseteq adv[pfr_2](ag_i)$,
  then $bnf_i(adv[pfr_1](ag_i)) \leq bnf_i(adv[pfr_2](ag_i))$.
- *if* $adv[pfr_1](ag_i) \subset adv[pfr_2](ag_i)$,
  then $bnf_i(adv[pfr_1](ag_i)) < bnf_i(adv[pfr_2](ag_i))$.

The following theorem shows that if utl_i satisfies Definition 39, then it represents a do-ut-des compatible preference relation.

Theorem 13 *If the utility function* utl_i *is do-ut-des compatible, then it represents a preference relation* $\precsim_i$ *which is do-ut-des compatible.*

proof: Assume that the agent ag_i do-ut-des prefers pfr' to pfr ($pfr \leq_i pfr'$), then

$$adv[pfr](ag_i) \subseteq adv[pfr'](ag_i) \text{ and } obl[pfr'](ag_i) \subseteq obl[pfr](ag_i)$$

Being utl_i is do-ut-des compatible, we have that $cost_i(pfr') \leq cost_i(pfr)$ and $bnf_i(adv[pfr](ag_i)) \leq bnf_i(adv[pfr'](ag_i))$. As blc_i is a function strictly increasing in the first argument and strictly decreasing in the second argument, we have

$$\begin{array}{l} blc_i(bnf_i(adv[pfr](ag_i), cost_i(pfr)) \leq \\ blc_i(bnf_i(adv[pfr'](ag_i), cost_i(pfr)) \leq \\ blc_i(bnf_i(adv[pfr'](ag_i), cost_i(pfr')) \end{array}$$

Therefore, $utl_i(pfr) \leq utl_i(pfr')$ and hence $pfr \precsim_i pfr'$.

If $pfr <_i pfr'$, then the previous inequality holds and either $obl[pfr'](ag_i) \subset obl[pfr](ag_i)$, or $adv[pfr](ag_i) \subset adv[pfr'](ag_i)$. In the first case $cost_i(pfr') < cost_i(pfr)$, so

$$blc_i(bnf_i(adv[pfr'](ag_i), cost_i(pfr)) < blc_i(bnf_i(adv[pfr'](ag_i), cost_i(pfr'))$$

In the second case $bnf_i(adv[pfr](ag_i)) < bnf_i(adv[pfr'](ag_i))$, so

$$blc_i(bnf_i(adv[pfr](ag_i), cost_i(pfr)) < blc_i(bnf_i(adv[pfr'](ag_i), cost_i(pfr))$$

In both the cases $utl_i(pfr) < utl_i(pfr')$ and hence $pfr \prec_i pfr'$. □

4.6 Summarizing

In this chapter we have shown which relationship occurs between the do-ut-des property and the notion of core. In the first two sections we have provided a brief overview on Cooperative Game Theory. We have formalized a game at two different levels of abstractions, the strategic form and the effectivity form. The strategic form represents the individual strategies of the agents and the relative outcomes. The effectivity form represents directly which outcomes groups of agents can force collaborating. Therefore, the effectivity form of a game approximatively corresponds, in the game theoretical setting, to the level of abstraction we have formalized with the Power Structures. We have shown how the notions of α and β-ability can be used to represent a game in the effectivity form when it is initially represented in the strategic form. The effectivity form of a game enables to naturally define a cooperative game without transferable utilities and a well-known admissibility criterion, the core.

In Section 4.4 we have defined the cooperative game $NTU[PS]$ relative to a Power Structure PS and we have shown the relation between the do-ut-des property and the core. In particular, in order to relate the do-ut-des property to the notion of core, we have defined an admissibility criterion, the q-do-ut-des property, which is more restrictive of the notion of core. Under the hypothesis that the quantitative preference relations in $NTU[PS]$ are compatible with the qualitative do-ut-des preference relation, we have shown that it is possible to restrict the search for q-do-ut-des coalitions only to those coalition which satisfies the do-ut-des property. Therefore, the do-ut-des property is compatible with a quantitative criterion that implies the notion of core.

In several cases the preference relations are calculated by means of a cost-benefit analysis. So, in Section 4.5, we have shown some sufficient conditions assuring that the cost-benefit analysis of an agent represents a do-ut-des compatible preference relation. These conditions are not so restrictive and a not abstruse class of cost-benefit analyses are actually do-ut-des compatible.

Chapter 5

A logic based formalization of Power

5.1 Introduction

In Section 3.2 we have defined a representation of a multiagent system, the Power Structure, which abstracts from the representation of the individual agents in terms of their capabilities, and which does not describe how agents have to join their efforts in order to assure the achievement of their goals. Power Structures directly describe which groups of agents can see to the achievement of which goals.

However, usually multiagent systems are not directly represented in terms of the power of groups of agents. This kind of information has to be derived, for example, by reasoning about the possible coordination policies of the agents.

Thus, in order to make Power Structures operative, the notion of power has to be *constructively* defined. In recent year a formalism has received particular attention: Alternating-time Temporal Logic (see Section 2.6). Another closely related formalism is the Coalition Logic for Propositional Control (CL-PC, [vdHW05]), where the states of affairs a group of agents can force depend on the propositional atoms under the control of the members of that group. In these approaches central is the notion of so-called α-ability: the capability of a group of agents to enforce a certain state of affairs, no matter what actions the other agents take. Following [vdHW05], we call such logics *Cooperation Logics*.

Notice that in all of these approaches, the ability of groups of agents to obtain a state of affairs is modeled without an explicit representation of *how* these agents obtain them. In other words, these logics do not represent actions and plans in the object language, even though in some cases, the concepts of action or choices are present in the underlying semantics. However, an explicit representation of actions in the object language is important in many domains. For example, certain actions may be more costly then others, and the costs may be distributed among the agents helping to achieve the goal according to the actions they perform. Also, using such a language for verification of a multiagent system, would not only provide the user — or indeed, the agents themselves— with a proof *that* a desired goal can be achieved, but it would also explicitly capture a *plan* of how to achieve it. Finally, in some situations, it seems a good heuristic to formulate a plan before organizing a group to execute it [WJ94, SD01].

In this chapter[1], we are interested in how the ability to obtain a state of affairs relates to the ability to execute certain, possibly complex, and possibly collective, actions. We develop a logic that combines operators from

[1]Sections 1-3 of this Chapter are based on the work [SGvdHW05].

Cooperation Logics with an explicit representation of actions. We achieve this by dividing the problem in two parts: a logic for reasoning about actions and their effects (*à la* Dynamic Logic) together with a variant of CL-PC to reason about groups of agents and the actions that they can perform. In more detail, we study multiagent systems that consist of two separate parts: an *environment module* that describes actions and their effects, and an *agents module* describing how the agents can interact with the environment by performing certain actions. We use the agents module to represent the ability of a group Q to *execute* certain type of actions. Where β is a joint action (i.e., it denotes a set of actions to be executed concurrently), we write

$$\langle\langle Q \rangle\rangle \beta$$

to indicate that group of agents Q can execute the joint action β. In the environment, then, we can reason about the effect of actions using our environment logic, and write, for example:

$$[\beta]\phi$$

for 'every successful execution of β results in state where ϕ is true'.

The cooperation modality of our logic expresses the ability of a group to obtain a certain state of affairs (note that ϕ is a proposition, not a joint action)

$$\langle\langle Q \rangle\rangle \phi$$

This kind of expression can now be analyzed using the two modules; the basic observation that connects the two modules with this operator is that a group of agents has the ability to obtain a state of affairs ϕ just in case there exists a plan β that this group of agents is able to execute which is guaranteed to result in ϕ.

Such an approach presumes that the effect of actions can be modeled more or less independently from the behavior of the agents that perform these actions. We think that as a general schema, it is applicable to many real life scenarios. For example, an environment may be a communication network via which agents send and receive messages. Or, the environment may be a Database Management System, defining actions and constraints in a general way, and the agents are users querying and updating the database.

Separating actions from the agents that perform them is not only practical from an operative viewpoint, but from a logical standpoint as well. It allows us to break up the problem of reasoning about planning in a multi-agent environment into two sub-problems, and to study the logic of each of these 'modules' in isolation before putting them together. This approach is also

interesting, because it allows us to 'reuse' existing logics and exploit them as 'modules' in a richer system.

In Section 5.2 we will present the two building blocks for our framework. We first give a logic for the environment, in which one can reason about the effect of executing joint actions. Then we define our agents module, in which the α-ability of groups of agents is expressed not with respect to state of affairs that can be reached, but for joint actions that can be guaranteed.

Section 5.2 puts these blocks together in a module called a multiagent system. Here, we define $\langle\langle Q \rangle\rangle \phi$, the α-ability of a group Q to bring about state of affairs ϕ, in terms of $\langle\langle Q \rangle\rangle \beta$ and $[\beta]\phi$.

Section 5.3 is devoted to an example, illustrating how the logic can be used to represent properties of a multi-agent system. The example is a scenario in which agents can collectively set variables which they are unable to change on their own. Finally, Section 5.4 indicates how to instantiate a Power Structure starting from the our formalization of α-ability, and Section 5.5 shows two no-additive notions of power requiring that all the agents involved in the achievement of a set of goals have to be respectively necessary or useful.

5.2 Formalizing Action and Cooperation

In our setting, a multiagent system is composed of two *modules*: one module describes the *agents* in terms of the actions they can execute, another describes the *environment* that is populated by these agents, and in particular the effects of performing certain actions.

5.2.1 The environment module

There are many ways of modeling an environment. One very general way to model actions and their effects is by using a Labeled Transition System (LTS) [HKT84, CE82, McM92].

An LTS is a directed graph in which the arcs are labeled by elements from a set of atomic actions and the nodes are labeled with sets of propositional variables. Arcs are called transitions, the nodes are called states. The idea is that if a state s is connected to another state s' by a transition labeled with an action a, this means that s' is a possible result of executing action a in state s. The sets of variables at each end of the transition tell us how the performance of the action changes the state of the environment, in terms of the propositions that change value.

One of the drawbacks of this basic LTS model is that it does not allow for modeling concurrency directly: there is no straightforward way to model

the fact that performing certain actions together may have a very different effect than performing each of them separately.

In the present context, we are interested in the way that the choice of actions of different agents interact, so we propose to model the effect of actions being performed concurrently directly in the model. We do this by labeling the transitions not with single actions, but with *sets* of actions. Intuitively, a transition labeled with a set of actions A represents the effect of executing exactly those actions together, and no others. This approach to deal with concurrency is similar to that taken in [GMS98].

Definition 40 (Environment module) *An environment module Env is a Set-Labeled Transition System (SLTS), i.e. it is a tuple* $\langle St, Ac, (\rightarrow)_{A \subseteq Ac}, P, \pi \rangle$ *where:*

1. *St is a set of states.*
2. *Ac is a finite set of actions.*
3. *For each* $A \subseteq Ac$, $\rightarrow_A \subseteq St \times St$.
4. *P is a set of propositional variables.*
5. $\pi : (St \times P) \rightarrow \{1, 0\}$ *is an interpretation function that assigns to each state and each propositional variable a truth value.*

We require that the accessibility relations $\rightarrow_A$ *are serial for each set of actions A, i.e., that for each state s there is always an* s' *with* $s \rightarrow_A s'$. □

A state s is characterized by the propositional variables that are true and false in it, as given by $\pi(s, p)$, and by the possible effects of executing combinations of actions A concurrently, as defined by the accessibility relations $\rightarrow_A$. We will refer to a set of actions A as a *concurrent action*. Actions may be *non-deterministic*, in the sense that the outcome of performing a set of actions may not always determine a unique result.

To reason about actions and their effect in an SLTS, we use a language that is similar to that of Propositional Dynamic Logic (PDL) . PDL has a language that consists of two parts: an action language to describe actions, and a 'static' language to describe properties of states. However, rather than directly incorporating the program constructs of PDL, we use *Boolean combinations* of atomic actions to reason about sets of actions.

Definition 41 (Action expressions) *Given the set of atomic actions Ac, the set of action expressions is defined by the grammar:*

$$\alpha ::= a \mid \alpha \wedge \alpha \mid \neg\alpha$$

where $a \in Ac$.

To interpret this action language, we define a relation $\models^p$ between subsets of Ac and action expressions. $A \models^p \alpha$ is to be read as 'concurrent action A is of type α', or 'α is true of A'.

Definition 42 (Interpretation of action expressions) *Truth of action expressions is inductively defined as follows:*

1. $A \models^p a$ *iff* $a \in A$
2. $A \models^p \neg\alpha$ *iff* $A \not\models^p \alpha$
3. $A \models^p \alpha \wedge \beta$ *iff* $A \models^p \alpha$ *and* $A \models^p \beta$.

So, for example, the concurrent action $\{a, b\}$ is of type $a \wedge \neg c$, but it is not of type $\neg b$. Formally, the action language is simply classical Boolean propositional logic, with Boolean variables corresponding to atomic actions. We can define additional connectives, such as $\vee$ or $\rightarrow$, in the usual way.

Definition 43 (Language of environment logic) *Given a set of actions Ac and a set of propositional variables P, the language of environment logic is given by:*

$$\phi ::= p \mid \phi \wedge \phi \mid \neg\phi \mid [\alpha]\phi$$

where $p \in P$, and α is an expression of the action language.

In effect, the resulting language is that of Boolean Modal Logic [GP90]. The truth of sentences of environment logic is defined as follows.

Definition 44 (Semantics of environment logic) *Let $Env = \langle St, Ac, (\rightarrow)_{A \subseteq Ac}, P, \pi \rangle$ be a SLTS, and s is a state in St. Truth of sentences of the language of environment logic is defined inductively as follows:*

1. $Env, s \models^e p$ *iff* $\pi(s, p) = 1$
2. $Env, s \models^e \phi_1 \wedge \phi_2$ *iff* $Env, s \models^e \phi_1$ *and* $Env, s \models^e \phi_2$.
3. $Env, s \models^e \neg\phi$ *iff* $Env, s \not\models^e \phi$
4. $Env, s \models^e [\alpha]\phi$ *iff for all* $A \subseteq Ac$ *and* s', *if* $A \models^e \alpha$ *and* $s \rightarrow_A s'$, *then* $Env, s' \models^e \phi$.

We write $Env \models^e \phi$ *in the case for all the states* s, $Env, s \models^e \phi$.

So, for example, if a is an atomic action, then $[a]\phi$ is true just in case executing a will always result in a state where ϕ is true, no matter what other actions are performed concurrently.

A sentence $[\neg a]\phi$ expresses that *not* doing a will always result in a state where ϕ is true, i.e. performing any concurrent action that does not involve a will results in a state where ϕ is true.

In a precise sense, the formula $[\alpha]\phi$ expresses that being of type α is a sufficient condition for a set of actions to guarantee that ϕ be true; dually, the formula $[\neg\alpha]\neg\phi$ expresses that doing an action of type α is a *necessary* condition for reaching a state where ϕ is true (if the action is not of type α, for all the states in which the system will result ϕ is false).

The logic that results from this semantics is given by the following set of axioms and rules.

Definition 45 (environment logic) *The environment logic ($\vdash^e$) has the following axioms:*

1. *All axioms of propositional logic.*
2. $[\alpha](\phi \to \psi) \to ([\alpha]\phi \to [\alpha]\psi)$ *for each* α *(distribution).*
3. *All axioms of the form* $[\alpha]\phi \to [\beta]\phi$*, if* $\beta \to \alpha$ *in propositional logic.*
4. $([\alpha]\phi \wedge [\beta]\phi) \to [\alpha \vee \beta]\phi$ *(additivity).*
5. $\neg[\alpha]\neg\top$ *if* α *is consistent (seriality).*

Rules:

1. *from* $\vdash^e \phi$ *and* $\vdash^e \phi \to \psi$*, derive* $\vdash^e \psi$ *(modus ponens).*
2. *from* $\vdash^e \phi$*, derive* $\vdash^e [\alpha]\phi$ *(necessitation for each action expression* α*).*

We write $\Sigma \vdash^e \phi$ *if* ϕ *can be derived from the set of sentences* Σ *using these rules and axioms.*

Axiom 2 and the rule of necessitation state that the $[\alpha]$-operators are normal modal operators. Axiom 3 says that if β implies α in propositional logic, then, if an α-action always results in a ϕ state, then a β-action will as well (since any β-action is an α-action). Axiom 4 says that if all α-action and all β-actions result in a state where ϕ is true, then also all $\alpha \vee \beta$-actions will. Finally, the seriality axiom reflects that any combination of actions will lead to some resulting state.

This logic is sound and complete with respect to the semantics.

Theorem 14 (Soundness and completeness) *Environment logic is sound and complete with respect to environment models.*

$$\vdash^e \phi \text{ iff } Env \models^e \phi \text{ for each environment model } Env$$

proof:[2] Construct a canonical model $\langle W, (\rightarrow_A), V\rangle$ as follows. Its domain W consists of all maximal consistent sets Σ. We set $\Sigma \rightarrow_A \Gamma$ iff for all ϕ and α, if $A \models \alpha$ and $[\alpha]\phi \in \Sigma$, then $\phi \in \Gamma$. Set $V(\Sigma)(p) = 1$ iff $p \in \Sigma$.

The following *truth lemma* holds:

$$\phi \in \Sigma \text{ iff } \Sigma \models \phi \text{ in the canonical model}$$

This can be proved by an induction on the structure of ϕ, where the only interesting case is that of the modal operators.

So, suppose $[\alpha]\psi \in \Sigma$. Assume that $\Sigma \rightarrow_A \Gamma$ for some A such that $A \models \alpha$. Then $\psi \in \Gamma$ by definition of the canonical model, and by induction hypothesis, $\Gamma \models \psi$. Since Γ was arbitrary, it follows that $\Sigma \models [\alpha]\psi$.

For the other direction. Before proving the general case, we first consider the case where our action expression is of the form $\alpha_A = \bigwedge A \wedge \bigwedge\{\neg a \mid a \in Ac$ and $a \notin A\}$ for some set of actions A.

Suppose $[\alpha_A]\psi \notin \Sigma$ for some A. With axiom 2 and the consistency of Σ, this means that $\{\chi \mid [\alpha_A]\chi \in \Sigma\} \cup \{\neg\psi\}$ is consistent. This set can be extended to a maximal consistent Γ. We show that $\Sigma \rightarrow_A \Gamma$. To see this, take an arbitrary $[\alpha]\chi \in \Sigma$ such that $A \models \alpha$. Then, in propositional logic, $\alpha_A \models \alpha$, and by axiom 3, it must hold that $[\alpha_A]\chi \in \Sigma$. But then, by construction of Γ, $\chi \in \Gamma$. Putting everything together, we have that $\Sigma \rightarrow_A \Gamma$, $A \models \alpha_A$, and, by induction hypothesis, $\Gamma \not\models \psi$. We conclude that $\Sigma \not\models [\alpha_A]\psi$.

Now we move to the general case. Suppose that $\Sigma \models [\alpha]\psi$ for arbitrary α. The sentence α is equivalent in propositional logic to the sentence $\alpha^\vee = \bigvee\{\alpha_A \mid A \models \alpha\}$. Clearly, we have that $\Sigma \models [\alpha^\vee]\psi$; and also that, for axiom 3, $\Sigma \models [\alpha_A]\psi$ for each A such that $A \models \alpha$. By the previous argument, we have that $[\alpha_A]\psi \in \Sigma$ for each A such that $A \models \alpha$. With axiom 4, it follows then that $[\alpha^\vee]\psi \in \Sigma$. We can now use axiom 3 and the fact that $\alpha^\vee \rightarrow \alpha$, and conclude that $[\alpha]\psi \in \Sigma$. □

So, what we have now is a dynamic logic in which we can reason about *concurrent* actions. That is, we have a way of reasoning about the effect of performing *combinations* of actions, as well as about the effect of *not* performing certain actions. This dynamic logic will serve as the environment module of our multi-agent system.

[2] I thank Jelle Gerbrandy for the proofs of Theorems 14, 15 and 17

5.2.2 Agents Module

Agents are identified with the set of actions that they can perform. We assume that a set of actions is assigned to each agent which can perform any combination of these actions.

Definition 46 (Agents module) *An agents module is a tuple $\langle Ag, Ac, act\rangle$, where Ag is a set of agents. Ac is a set of actions which can be executed by the agents. act is a function that assigns to each agent ag_i a subset $act(ag_i)$ of actions from Ac, $act : Ag \to 2^{Ac}$.*

We require that $\bigcup_{ag_i \in Ag} act(ag_i) = Ac$.

We will abuse notation somewhat, and write 'act' as a *pars pro toto* for $\langle Ag, Ac, act\rangle$. We denote with $act(Q)$ the set $\bigcup_{ag_i \in Q} act(ag_i)$ of actions that the group Q of agents can perform.

To represent the ability of agents to perform actions of a certain type, we will use some notation from Cooperation Logics. For an action expression α, a sentence of the form $\langle\langle Q\rangle\rangle\alpha$ expresses the fact that Q is able to bring it about that a set of actions of type α will be performed[3]. It is important to note that α here is an action type: we are dealing with the ability to achieve a complex action, not the ability to bring about some state of affairs.

A group Q is able to enforce α exactly if there exists a subset from the actions under their control that, no matter what actions the other agents choose, is such that the resulting concurrent action is of type α.

The way the semantics is defined is similar to that of the Coalition Logic of Propositional Control of [vdHW05], except that we explicitly allow for different agents to 'control' the same action.

Definition 47 (Ability for actions) *Let $\langle Ag, Ac, act\rangle$ be an agents module, let $Q \subseteq Ag$ be a set of agents.*

$$\langle Ag, Ac, act\rangle \models^a \langle\langle Q\rangle\rangle\alpha \text{ iff there exists an } A \subseteq act(Q) \text{ such that for all } B \subseteq act(Ag \setminus Q) \text{ it holds that } A \cup B \models^p \alpha$$

So, for example, if act is an agents module in which an agent ag_i can execute action a, then $act \models^a \langle\langle\{ag_i\}\rangle\rangle a$: ag_i can enforce a. However, ag_i can enforce $\neg a$ just in the case it is the only one that can execute a: $act \models^a \langle\langle\{ag_i\}\rangle\rangle\neg a$ only if $a \notin act(ag_j)$ for any ag_j other than ag_i. Similarly, $act \models^a \langle\langle Q\rangle\rangle(a \wedge \neg b)$ exactly when at least on agent in Q can execute a, while no agent outside of Q can do b.

[3]More precisely, Q is α-able to bring it about that a set of actions of type α will be performed. However, we omit the prefix α- to avoid confusion with action symbols.

Definition 48 (Cooperation logic for actions) *The cooperation logic for actions is given by the following axioms, together with all axiom schemes of propositional logic, and modus ponens. We denote derivability in this logic with* $\vdash^a$.

1. $\langle\langle Q\rangle\rangle\top$, *for all* $Q \subseteq Ag$
2. $\langle\langle Q\rangle\rangle\alpha \leftrightarrow \neg\langle\langle Ag \setminus Q\rangle\rangle\neg\alpha$
3. $\langle\langle Q\rangle\rangle\alpha \rightarrow \langle\langle Q\rangle\rangle\beta$ *if* $\alpha \rightarrow \beta$ *in propositional logic.*
4. $(\langle\langle Q_1\rangle\rangle\alpha \wedge \langle\langle Q_2\rangle\rangle\beta) \rightarrow \langle\langle Q_1 \cup Q_2\rangle\rangle(\alpha \wedge \beta)$ *for all* $Q_1, Q_2 \subseteq Ag$ *such that* $Q_1 \cap Q_2 = \emptyset$
5. $\langle\langle Q\rangle\rangle a \rightarrow \bigvee_{ag_i \in Q} \langle\langle \{ag_i\}\rangle\rangle a$, *for all* $Q \subseteq Ag$ *and atomic* $a \in Ac$
6. $\langle\langle Q\rangle\rangle\alpha \wedge \langle\langle Q\rangle\rangle\beta) \rightarrow \langle\langle Q\rangle\rangle(\alpha \wedge \beta)$ *if* α *and* β *have no atomic actions in common.*
7. $\langle\langle Q\rangle\rangle\neg a \rightarrow \langle\langle Q\rangle\rangle a$ *for atomic* $a \in Ac$.
8. $\langle\langle Q\rangle\rangle\alpha \rightarrow \bigvee\{\langle\langle Q\rangle\rangle \bigwedge \Phi \mid \Phi$ *is a set of literals such that* $\bigwedge \Phi \rightarrow \alpha\}$

An explanation of some of the axioms is in order.

The first axiom states that any group of agents is able to enforce some trivial action. Note that it give us a kind of necessitation, even for actions: If α is a classical tautology, then we also have $\vdash^a \langle\langle Q\rangle\rangle\alpha$. Axiom 2 is typical of Cooperation Logics, and states that a group Q is able to enforce an action α if and only if the agents outside that group cannot enforce that an action that is not of type α will happen. This follows directly from the definitions.

Axiom 3 says that if a group of agents is able to enforce an α-action, and all α-actions are also of type β, then Q is also able to enforce a β-action.

Axiom 4 states that if two independent groups of agents have the ability to enforce, respectively, actions of type α and of type β, then they can combine forces to enforce an $\alpha \wedge \beta$-action. This holds only if the two groups are disjoint: if this is not the case, and the two groups have an agent in common, say agent ag_i, then it may be that ag_i needs to execute action a to (help) enforce α, while it needs to refrain from doing a to enforce β: there is no single choice of actions that guarantees that $\alpha \wedge \beta$ will happen.

Axiom 6 says that if a group can enforce actions α and β, and α and β involve completely different atomic actions, then the group can also enforce $\alpha \wedge \beta$ (by choosing the actions needed to guarantee α and performing them together with those that guarantee β).

The validity of axiom 7 is a consequence of the assumption that any atomic action can be executed by at least one agent. It says that if a group of agents can guarantee that an *atomic* action is *not* performed, then one of the agents in that group must be able to execute that action.

Axiom 8 says that if a group of agents Q can perform an α-action, then there must be a conjunction of literals (that is, formulas of the form a or $\neg a$) that implies α and can be enforced by Q. In other words, there must be a 'recipe' that tells them which actions to execute and which not to execute, that guarantees that α results.

Theorem 15 (Soundness and completeness) *Cooperation logic for actions is sound and complete with respect to agent models.*

proof: We start with a useful observation:
observation 1: Let $A \subseteq act(Q)$. Define $\Phi(A, Q)$ to be the set of literals $A \cup \{\neg a \mid a \in act(Q), a \notin A \text{ and } a \notin act(Ag \setminus Q)\}$. It holds that

$$\{A \cup B \mid B \subseteq act(Ag \setminus Q)\} = \{C \mid C \models \bigwedge \Phi(A, Q)\}$$

Soundness is a matter of checking the axioms one by one. We only consider the less obvious case of axiom 8. Suppose $act \models \langle\langle Q \rangle\rangle \alpha$. Then there is a set $A \subseteq act(Q)$ such that for all $B \subseteq act(Ag \setminus Q)$: $A \cup B \models \alpha$. Consider the set $\Phi(A, Q)$. Using observation 1 above, it follows that $\bigwedge \Phi(A, Q) \models \alpha$, and that $act \models \langle\langle Q \rangle\rangle \bigwedge \Phi(A, Q)$. A forteriori, we have that $act \models \bigvee \{\langle\langle Q \rangle\rangle \bigwedge \Phi \mid \Phi$ is a set of literals such that $\bigwedge \Phi \rightarrow \alpha\}$.

For completeness, let Σ be a maximal consistent set of sentences. We define act by setting $act(ag_i) = \{a \mid \langle\langle \{ag_i\} \rangle\rangle a \in \Sigma\}$. Note that with axioms 4 and 5, we have for atomic actions a that $a \in act(Q)$ iff $\langle\langle Q \rangle\rangle a \in \Sigma$, a fact that we will use in the proof.

We need to show that for each ϕ:

$$act \models \phi \text{ iff } \phi \in \Sigma$$

We prove this by induction on ϕ, where the only interesting case is the one where ϕ is of the form $\langle\langle Q \rangle\rangle \alpha$.

Suppose first that $act \models \langle\langle Q \rangle\rangle \alpha$. That means that there is an $A \subseteq act(Q)$ such that for all $B \subseteq act(Ag \setminus Q) : A \cup B \models \alpha$. Combining this with observation 1 gives us that $\bigwedge \Phi(A, Q) \vdash \alpha$ in propositional logic.

By definition of act, it holds that $\langle\langle Q \rangle\rangle a \in \Sigma$ for each $a \in A$. Similarly, we have for each $a \notin act(Ag \setminus Q)$ that $\langle\langle Ag \setminus Q \rangle\rangle a \notin \Sigma$, from which it follows with maximality and axiom 2 that $\langle\langle Q \rangle\rangle \neg a \in \Sigma$. A number of applications of

axiom 6 allow us to conclude that $\langle\langle Q\rangle\rangle \bigwedge \Phi(A, Q) \in \Sigma$, and by the monotony axiom 3 it follows that $\langle\langle Q\rangle\rangle\alpha \in \Sigma$.

For the other direction, assume that $\langle\langle Q\rangle\rangle\alpha \in \Sigma$. We need to show that $act \models \langle\langle Q\rangle\rangle\alpha$. Since Σ satisfies axiom 8, there must be a set of literals Φ such that $\bigwedge \Phi \vdash \alpha$ and $\langle\langle Q\rangle\rangle \bigwedge \Phi \in \Sigma$. With axiom 3, we know that for each of the literals l in Φ it holds that $\langle\langle Q\rangle\rangle l \in \Sigma$.

By construction of act, for positive literals $l \sim a$ we have $\langle\langle Q\rangle\rangle l$ implies that $a \in act(Q)$.

For negative literals $l \sim \neg a$, we have with axiom 7 that $\langle\langle Q\rangle\rangle\neg a \in \Sigma$ implies that $\langle\langle Q\rangle\rangle a \in \Sigma$, and so $a \in act(Q)$. Moreover, for the negative literals we can use axiom 2 and conclude that $\neg\langle\langle Ag \setminus Q\rangle\rangle a \in \Sigma$, and therefore that $a \notin act(Ag \setminus Q)$. But this means that $act \models \langle\langle Q\rangle\rangle \bigwedge \Phi$, and by axiom 3, that $act \models \langle\langle Q\rangle\rangle\alpha$. □

5.2.3 The multiagent system: agents acting in an environment

In the previous two subsections we defined a module describing an environment, and a separate module describing the actions that agents can choose to perform, either by themselves or together. These two modules can be related by identifying the set of actions in the two respective modules. Such a combination provides us with a semantics for reasoning about agents that act in an environment by way of performing certain actions.

Formally, a multiagent system is an environment together with an agent module that shares its action repertoire.

Definition 49 (Multiagent system) *A multiagent system MaS is a tuple*

$$\langle St, Ac, (\rightarrow)_{A\subseteq Ac}, P, \pi, Ag, act\rangle$$

where:

1. $\langle St, (\rightarrow_{A'})_{A\subseteq Ac}, P, \pi\rangle$ *is an environment (in the sense of Definition 40).*
2. $\langle Ac, Ag, act\rangle$ *is an agent module (in the sense of Definition 46).*

Of course, everything that we could express with the languages we used to talk about the environment and about the capabilities of the agents remains interesting in the combined system as well, so a logic for reasoning about a multiagent system should include both. But in the new, more complex system, there are other notions of interest that cannot be expressed in the separate modules.

In particular, we are interested in the more standard operators of Cooperation Logics that reason not about ability to enforce complex *actions*, but ability to enforce certain *results*. We will overload the operators $\langle\langle Q \rangle\rangle$ for this purpose, and add sentences of the form $\langle\langle Q \rangle\rangle \phi$ to the language, where ϕ can be any sentence. Intuitively, $\langle\langle Q \rangle\rangle \phi$ means that the agents in Q have the power to obtain a state where ϕ is true.

This leads to the following definition of a language for multiagent systems.

Definition 50 (Language of the multiagent system) *Given a multiagent system MaS, the language of our cooperation logic with actions is given by the following grammar:*

$$\phi ::= p \mid \phi \wedge \phi \mid \neg\phi \mid [\alpha]\phi \mid \langle\langle Q \rangle\rangle \alpha \mid \langle\langle Q \rangle\rangle \phi$$

where $Q \subseteq Ag$, $p \in P$, and α is an expression of the action language, as in Definition 41.

The satisfaction of formulae of the type $\langle\langle Q \rangle\rangle \phi$ is defined as follows.

Definition 51 (Ability) *$MaS, s \models^m \langle\langle Q \rangle\rangle \phi$ iff there is a set of actions $A \subseteq act(Q)$ such that for all $B \subseteq act(Ag \setminus Q)$ and for all states t, if $s \rightarrow_{A \cup B} t$, then $t \models^m \phi$.*

We write $MaS \models^m \phi$ in the case for all the states s, $MaS, s \models^m \phi$.

So, $\langle\langle Q \rangle\rangle \phi$ is true just in case the agents in Q are able to execute a concurrent action that guarantees, independently from what the other agents do, the truth of ϕ.

The following proposition provides a precise link between the ability to obtain a certain state of affairs, and the ability to execute an action that is guaranteed to result in such a state. It states that a group has the ability to enforce the truth of a sentence just in case it is able to enforce a concurrent action that is guaranteed to result in a state in which that sentence is true:

Theorem 16 *Given a multiagent system MaS and a state s of its environment:*

$MaS, s \models^m \langle\langle Q \rangle\rangle \phi$ iff there exists an action expression α with $MaS, s \models^m \langle\langle Q \rangle\rangle \alpha$ and $MaS, s \models^m [\alpha]\phi$

proof:
$\Longrightarrow$

Assume that $MaS, s \models^m \langle\langle Q \rangle\rangle \phi$, and let A be the set of actions in $act(Q)$ such that for all $B \subseteq act(Ag \setminus Q)$ and for all states t, if $s \rightarrow_{A \cup B} t$, then $t \models^m \phi$. The action expression $\bigwedge \Phi(A, Q)$ (see the proof of Theorem 15) is

the witness we are looking for: it holds that $MaS, s \models^m \langle\langle Q \rangle\rangle \bigwedge \Phi(A, Q)$, and that $MaS, s \models^m [\bigwedge \Phi(A, Q)]\phi$.
$\Longleftarrow$

Let α be such that $MaS, s \models^m \langle\langle Q \rangle\rangle \alpha$ and $MaS, s \models^m [\alpha]\phi$. By definition it follows from $MaS, s \models^m \langle\langle Q \rangle\rangle \alpha$ that there must be a set of actions $A \in act(Q)$ such that for each $B \subseteq act(Ag \setminus Q)$, $A \cup B \models \alpha$. Then, since $MaS, s \models^m [\alpha]\phi$, for all t such that $s \rightarrow_{A \cup B} t$, it holds that $t \models \phi$. So $MaS, s \models^m \langle\langle Q \rangle\rangle \phi$. □

The logic for the multiagent system obviously contains all axioms we already had from the two separate models. To obtain a complete axiom system, it suffices to add just two axioms that relate the ability to obtain a state of affairs with the ability to execute concurrent actions.

Definition 52 (Cooperation logic with actions) *The cooperation logic with actions, $\vdash^m$, is given by:*

1. *All axioms and rules from the environment logic of section 5.2.1*
2. *All axioms and rules from the agent logic of section 5.2.2*
3. $(\langle\langle Q \rangle\rangle \alpha \wedge [\alpha]\phi) \rightarrow \langle\langle Q \rangle\rangle \phi$ *for each* α
4. $\langle\langle Q \rangle\rangle \phi \rightarrow \bigvee \{ \langle\langle Q \rangle\rangle \alpha \wedge [\alpha]\phi \mid \alpha$ *is the conjunction of a set of action literals* $\}$

where 'action literals' are atomic actions a or their negations $\neg a$.

The two new axioms relate the ability to execute actions with the ability to enforce the truth of propositions, in the same way as is done in Theorem 16.

Theorem 17 (Soundness and completeness) *Cooperation logic with actions is sound and complete with respect to action models:*

$$\vdash^m \phi \text{ iff } MaS \models^m \phi \text{ for all models } MaS$$

proof: Soundness follows from the soundness of the two modules and Theorem 16.

Completeness is relatively straightforward with the completeness results of sections 5.2.1 and 5.2.2. We first construct a model MaS combining the constructions of the previous proofs in the straightforward way.

We need to prove a truth lemma stating that $MaS, \Sigma \models^m \phi$ iff $\phi \in \Sigma$. This is proven by induction on ϕ, leaving the proofs of Theorems 15 and 14 practically unchanged. The only interesting case is the new operator. So suppose $MaS, \Sigma \models^m \langle\langle Q \rangle\rangle \psi$. With Theorem 16, there is an action expression

α such that $\Sigma \models^m \langle\langle Q \rangle\rangle \alpha$ and $\Sigma \models^m [\alpha]\psi$. By induction hypothesis and the previous two cases, we have that $\langle\langle Q \rangle\rangle \alpha \in \Sigma$ and $[\alpha]\phi \in \Sigma$, and therefore, with our axiom 3 and the fact that Σ is maximal, that $\langle\langle Q \rangle\rangle \phi \in \Sigma$.

For the other direction, assume that $\langle\langle Q \rangle\rangle \psi \in \Sigma$. Then, with the axiom 4 and maximality of Σ, there is a set of action literals Φ such that $[\bigwedge \Phi]\psi \in \Sigma$ and that $\langle\langle Q \rangle\rangle \bigwedge \Phi \in \Sigma$. We can conclude (by induction hypothesis and the previous cases) that $\Sigma \models^m [\bigwedge \Phi]\psi$ and $\Sigma \models^m \langle\langle Q \rangle\rangle \bigwedge \Phi$. We then use Theorem 16 again to conclude that $\Sigma \models^m \langle\langle Q \rangle\rangle \psi$. □

So, we have defined an environment as a Set-Labelled Transition System, and defined a PDL-like logic for representing the performance of complex actions in such an environment. We then defined agents as actors that can choose to do certain sets of actions, and defined a sound and complete logic for reasoning about the ability of groups of agents to enforce that a certain type of concurrent action is executed. We then have combined these two modules in a single description of a multiagent system and we have defined the collaborative ability to force a certain state of affairs. This *modularization* of a multiagent system paid off: it turned out to be relatively easy to find a sound and complete logic for the combination of these two logics.

5.3 An example

Often in a multiagent system, certain variables can be changed only when two or more agents collaborate, while other variables can be changed by the agents acting by themselves. For example, a database may contain crucial data that can be changed only with the consent of all operators, while data of a lesser importance can be changed by any agent. For another example, a website may accept a request for information without further ado, but for accepting a transaction with a credit card, it will need the consent of both the customer and the credit card company. Similarly, a website such as Ebay will accept advertisements that are submitted by a single agent acting alone, but will not accept a transaction if this is not confirmed by all parties involved.

All of these examples are instances of a case in which certain variables are under control of single agents, while other variables can be changed by groups of agents acting together. We will consider an instance of this schema, in which two agents control two variables. To make the example concrete, we will talk about prison guards that control two doors (see Figure 5.1). A prison is composed of two interiors, a dormitory and a courtyard. One door opens to the outside, and it is important that it is not opened by accident. It can therefore only be opened if both guards act together. The other door separates the cell block from the courtyard, and each guard can open that

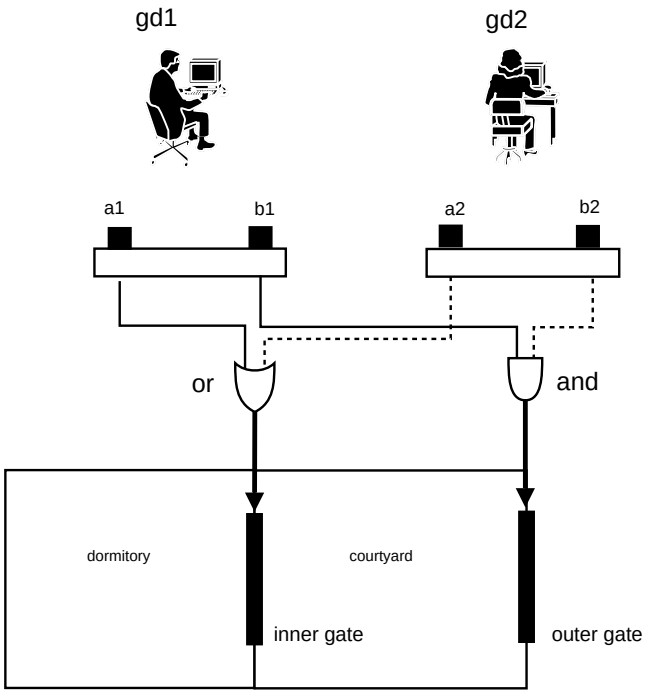

Figure 5.1: *The prison example.*

door by himself. The fact that the inner gate is open is expressed by a proposition in_open; if the outer gate is open, out_open is true.

The state of the gates is ruled by four buttons: a_1, b_1, a_2 and b_2. The gate in_open toggles its state if and only if at least one of a_1 and a_2 is pressed. The outer gate out_open toggles its state if and only if both b_1 and b_2 are pressed.

This description can be captured by an environment model *Env* with four states determined by the fact whether the gates are open or not. For example, we will write 01 for the state in which the inner door is closed and the outer door is open. Each of the buttons corresponds to an atomic action in the model, and the transitions are as described (so, for example, it will hold that $01 \rightarrow_{\{a_2,b_2\}} 11$).

With our environment language, we can express properties of the environment such that the fact that the action $a_1 \wedge b_1 \wedge b_2$ ensures in the state 00 that in the next state both the doors will be open, $Env, 00 \models^e [a_1 \wedge b_1 \wedge b_2](\mathsf{in\_open} \wedge \mathsf{out\_open})$; or the fact that if the outer door is closed, the action expression $b_1 \wedge b_2$ is both a sufficient and necessary condition to open the outer door, which is expressed by the observation that $Env \models^e$

$\neg\mathsf{out\_open} \rightarrow ([b_1 \wedge b_2]\mathsf{out\_open} \wedge [\neg(b_1 \wedge b_2)]\neg\mathsf{out\_open})$.

Two guards, gd_1 and gd_2, control the buttons. The first guard gd_1 has access to the buttons a_1 and b_1, whereas gd_2 has access to a_2 and b_2. This situation can be captured by an agents model with the same set of atomic actions as the environment, and a function act such that $act(gd_1) = \{a_1, b_1\}$ and $act(gd_2) = \{a_2, b_2\}$.

We can express that the action b_1 can be executed only by the guard gd_1 as $\langle\langle gd_1\rangle\rangle b_1 \wedge \neg\langle\langle gd_2\rangle\rangle b_1$ (strictly speaking, we should have written $\langle\langle\{gd_1\}\rangle\rangle$ instead of $\langle\langle gd_1\rangle\rangle$, but we are omitting the brackets for readability).

Together, the agents can push both b_1 and b_2, that is, $act \models^a \langle\langle gd_1, gd_2\rangle\rangle b_1 \wedge b_2$. Analogously, both of them can, individually, execute the disjunction of a_1 and a_2: $act \models^a \langle\langle gd_1\rangle\rangle(a_1 \vee a_2) \wedge \langle\langle gd_2\rangle\rangle(a_1 \vee a_2)$.

Combining the environment and the agents modules in multiagent system MaS, we obtain a model of the guards controlling the prison.

For example in a state where the outer door is closed, since $b_1 \wedge b_2$ is a necessary condition to open the outer gate, and each guard gd_i can prevent the execution of b_i, it follows that both guards are needed to open the outer gate, i.e. $MaS \models^m \neg\mathsf{out\_open} \rightarrow (\neg\langle\langle gd_1\rangle\rangle\mathsf{out\_open} \wedge \neg\langle\langle gd_2\rangle\rangle\mathsf{out\_open}))$ and $MaS \models^m \neg\mathsf{out\_open} \rightarrow \langle\langle gd_1, gd_2\rangle\rangle\mathsf{out\_open}$.

On the other hand, we want to be sure that when either one of the guards notes that there's a fire in the dormitory, he can open the inner gate to let the prisoners go in the courtyard. Since both the guards are able to execute at least one of the actions a_1 and a_2, $act \models^a (\langle\langle gd_1\rangle\rangle a_1 \vee a_2) \wedge (\langle\langle gd_2\rangle\rangle a_1 \vee a_2)$, and an action of type $a_1 \vee a_2$ opens the inner gate, $Env \models^e \neg\mathsf{in\_open} \rightarrow [a_1 \vee a_2]\mathsf{in\_open}$, each of the guards can open the inner gate if required, $MaS \models^m \neg\mathsf{in\_open} \rightarrow (\langle\langle gd_1\rangle\rangle\mathsf{in\_open} \wedge \langle\langle gd_1\rangle\rangle\mathsf{in\_open})$.

5.4 Constructing the Power Structure

In Definition 51 we have formalized a notion of the so-called α-ability as in Definition 30. α-ability is a *robust* notion of power as it guarantees that a group of agents can achieve a state of affairs independently from the behavior of the other agents and, in particular, without requiring for that group to foreknow the behavior of the other agents and react accordingly, as for example in the definition of β-ability (Definition 31). Thus, Definition 51 could be used in order to instantiate a Power Structure.

We assume that the multiagent system MaS is in a given state s and that the goals of the agents are described by sentences ϕ of our cooperation logic. In particular, a goal ϕ is satisfied if and only if the environment is in a state t satisfying the formula ϕ. A straightforward way to model the notion of

power is the following: a group of agents has the power to achieve a set of goals $G = \{\phi_1, \ldots, \phi_m\}$ if and only if it is α-able to satisfy all the formulas ϕ_i at the same time. A subset $(Q_1, G_1), \ldots, (Q_n, G_n)$ of *pow* is in *comp* if and only if there exists a set of consistent concurrent actions $\beta_1 \ldots \beta_n$ such that each β_i assures ϕ_i and Q_i is α-able to execute β_i.

Definition 53 *Given a multiagent system MaS, a state s, a set of state formulas Gl representing the goals of agents in MaS and function goals associating each agent with a set of goals, a Power Structure* $PS = \langle Ag, Gl, goals: Ag \rightarrow 2^{Gl}, pow \subseteq 2^{Ag} \times 2^{Gl}, comp \subseteq 2^{pow} \rangle$ *can be instantiated as follows. Given* $G \subseteq Gl$ *and* $Q \subseteq Ag$, $(Q, G) \in pow$ *iff* $MaS, s \models^m \langle\langle Q \rangle\rangle \bigwedge_{\phi \in G} \phi$, *where if* $G = \emptyset$, *then* $\bigwedge_{\phi \in G} \phi \equiv \top$. $\{(Q_1, G_1), \ldots, (Q_n, G_n)\} \subseteq pow$ *is in comp iff there exist n of concurrent actions* $\beta_1, \ldots, \beta_n$ *such that:*

1. $\beta_1 \wedge \cdots \wedge \beta_n \not\models^p \bot$
2. *for all* $1 \leq i \leq n$, $MaS, s \models^m \langle\langle Q_i \rangle\rangle \beta_i \wedge [\beta_i] \bigwedge_{\phi \in G_i} \phi$

It is straightforward to see that the conditions in Definition 13 are satisfied when the relations *pow* and *comp* are instantiated as in Definition 53. Moreover *pow* is goal-monotonic, super-additive and satisfy the property-1 in Definition 14 as proved in the following theorem.

Theorem 18 *The relation pow instantiated as in Definition 53 is goal-monotonic, super-additive and satisfies property-1 of Definition 14.*

proof: Assume that $(Q, G) \in pow$, this means that $MaS, s \models^m \langle\langle Q \rangle\rangle \bigwedge_{\phi \in G} \phi$, and hence, due to axiom 4 of Definition 52, there exists a concurrent action β such that $MaS, s \models^m \langle\langle Q \rangle\rangle \beta \wedge [\beta]\phi$. As $[\beta]$-operators are normal modal operator, given two generic state formulas ϕ_1 and ϕ_2, if $MaS, s \models^m [\beta]\phi_1$ and $\phi_1 \rightarrow \phi_2$, then $MaS, s \models^m [\beta]\phi_2$. Since for all $G' \subseteq G$, $(\bigwedge_{\phi \in G} \phi) \rightarrow (\bigwedge_{\phi \in G'} \phi)$, then $MaS, s \models^m [\beta] \bigwedge_{\phi \in G'} \phi$ and hence $MaS, s \models^m \langle\langle Q \rangle\rangle \beta \wedge [\beta] \bigwedge_{\phi \in G'} \phi$. For axiom 3 of Definition 52 this means that $MaS, s \models^m \langle\langle Q \rangle\rangle \bigwedge_{\phi \in G'} \phi$ and hence $(Q, G') \in pow$.

Now we prove super-additivity, assume that $(Q_1, G_1) \in pow$, $(Q_2, G_2) \in pow$ and $Q_1 \cap Q_2 = \emptyset$, this means for Definition 51 that there exists a set of actions $A_1 \subseteq act(Q_1)$ such that for all $B \subseteq act(Ag \setminus Q_1)$ and for all states t, if $s \rightarrow_{A_1 \cup B} t$, then $t \models \bigwedge_{\phi \in G_1} \phi$. In the same manner, there exists a set of actions $A_2 \subseteq act(Q_2)$ such that for all $B' \subseteq act(Ag \setminus Q_2)$ and for all the states t, if $s \rightarrow_{A_2 \cup B'} t$, then $t \models \bigwedge_{\phi \in G_2} \phi$. Consider a set of actions $A_1 \cup A_2 \cup B''$, where $B'' \in act(Ag \setminus [Q_1 \cup Q_2])$. Since $Q_1 \cap Q_2 = \emptyset$, then $A_2 \cup B'' \in act(Ag \setminus Q_1)$ and $A_1 \cup B'' \in act(Ag \setminus Q_2)$, but this means that for all the states t, if $s \rightarrow_{A_1 \cup A_2 \cup B''} t$, then $t \models \bigwedge_{\phi \in G_1} \phi$ and $t \models \bigwedge_{\phi \in G_2} \phi$, or equivalently that

$t \models \bigwedge_{\phi \in G_1 \cup G_2} \phi$. Therefore, $MaS, s \models^m \langle\langle Q_1 \cup Q_2 \rangle\rangle \bigwedge_{\phi \in G_1 \cup G_2} \phi$ and hence $(Q_1 \cup Q_2, G_1 \cup G_2) \in pow$.

Finally we prove property-1 of Definition 14, i.e. for all $Q \subseteq Ag$, $(Q, \emptyset) \in pow$. By definition, $\bigwedge_{\phi \in \emptyset} \phi \equiv \top$. Therefore, we have, by the necessitation rule of Definition 45, $MaS, s \models^m [\top]\top$ and for axiom 1 in Definition 48 $act \models \langle\langle Q \rangle\rangle \top$. Then, for item 3 of Definition 52, $MaS, s \models^m \langle\langle Q \rangle\rangle \top$ and hence for all $Q \subseteq Ag$, $(Q, \emptyset) \in pow$. □

In particular, as the relation *pow* of Definition 53 is super-additive and satisfies the property-1, we derive from Theorem 1 that *pow* is also coalitional monotonic.

5.5 Beyond α-ability

As seen in the previous section, the notion of power in Definition 53 is coalitional monotonic. Therefore, if a group of agents has the power to achieve a state formula ϕ, then by adding other agents the new group has (at least) the same power. We have seen in Section 3.5 that coalition monotonicity can invalidate the do-ut-des property as its significance decreases when, in the achievement of a set of goals, agents that do not play any role are included. For this reason in this section we formalize two other notions of power which, although deriving from the notion of α-ability, do not admit the presence of unnecessary or useless agents. We say that an agent ag_i is not necessary in the satisfaction of a state formula ϕ with respect of a group of agents Q if the group without ag_i is still α-able to achieve ϕ. Thus, starting from our formalization of α-ability, the condition that all the agents have to be necessary in the achievement of a set of goals G is formalized by imposing that all the subsets of Q are not α-able to achieve G.

Definition 54 (Power structure with only necessary agents) *Given a multiagent system MaS, a state s, a set of state formulas Gl representing the goals of agents in MaS and function goals associating each agent with a set of goals, a Power Structure* $PS = \langle Ag, Gl, goals : Ag \rightarrow 2^{Gl}, pow \subseteq 2^{Ag} \times 2^{Gl}, comp \subseteq 2^{pow} \rangle$ *can be instantiated as follows. Given* $G \in Gl$ *and* $Q \subseteq Ag$, $(Q, G) \in pow$ *iff* $MaS, s \models^m \langle\langle Q \rangle\rangle \bigwedge_{\phi \in G} \phi$ *and for all* $Q' \subset Q$, $MaS, s \not\models^m \langle\langle Q' \rangle\rangle \bigwedge_{\phi \in G} \phi$. *comp is instantiated as in Definition 53.*

We say that an agent ag_i is useful in the achievement of a set of goals G with respect to a group of agents Q if and only if there exists a concurrent action β such that

- β assures the achievement of all the goals in G.

- Q is able to execute β.
- Leaving ag_i the group Q, the remaining group is no more able to execute β.

The condition of useful agents is in particular the one adopted in the social reasoning mechanism developed in [SD01] as well as in distributed problem solving framework in [WJ94]. In these works an agent first reason about a plan to achieve a given goal, then it select all the actions of the plan which it is not able to execute and looks for those agents that can execute them.

The condition of useful agents considers a concurrent action β assuring a set of goals G, $MaS, s \models^m [\beta] \bigwedge_{\phi \in G} \phi$, and such that Q is able to execute β, $MaS \models^m \langle\langle Q \rangle\rangle \beta$, whereas all its subsets are not.

Definition 55 (Power structure with only useful agents) *Given a multiagent system MaS, a state s, a set of state formulas Gl representing the goals of agents in MaS and function goals associating each agent with a set of goals, a Power Structure* $PS = \langle Ag, Gl, goals : Ag \rightarrow 2^{Gl}, pow \subseteq 2^{Ag} \times 2^{Gl}, comp \subseteq 2^{pow} \rangle$ *can be instantiated as follows. Given* $G \subseteq Gl$ *and* $Q \subseteq Ag$, $(Q, G) \in pow$ *iff there exists a concurrent action* β *such that* $MaS, s \models^m \langle\langle Q \rangle\rangle \beta \wedge [\beta] \bigwedge_{\phi \in G} \phi$ *and for all* $Q' \subset Q$, $MaS, s \not\models^m \langle\langle Q' \rangle\rangle \beta$.

A set of commitments $\{(Q_1, G_1), \ldots, (Q_n, G_n)\} \subseteq pow$ *is in comp iff there exist n of concurrent actions* $\beta_1, \ldots, \beta_n$ *such that:*

1. $\beta_1 \wedge \cdots \wedge \beta_n \not\models^p \bot$.
2. *for all* $1 \leq i \leq n$, $MaS, s \models^m \langle\langle Q_i \rangle\rangle \beta_i \wedge [\beta_i] \bigwedge_{\phi \in G_i} \phi$.
3. *for all* $1 \leq i \leq n$ *and for all* $Q'_i \subseteq Q_i$, $MaS, s \not\models^m \langle\langle Q'_i \rangle\rangle \beta_i$.

By definition, the notion of power defined in Definition 54 satisfies the group minimality property, whereas the one defined in Definition 55 may be neither coalition monotonic nor group minimal.

5.6 Summarizing

In this chapter we have represented a multiagent system at a more detailed level of abstraction of the Power Structure and we have developed a logic in order to describe the capabilities of groups of agents to force a state of affairs independently from the behavior of the other agents (α-ability).

This research issue is not new in the field of modal logics, we have called the logics which focus on it *Cooperation Logics* [Pau02a, Pau02b, AHK02,

AHM+98, GvDar, vdHW03, vdHW05]. With respect to the other Cooperation Logics we have adopted a modular approach in which the multiagent system is composed by an environment module and an agents module. In Section 5.2.1, the environment module models the world in which the agents *live* as a transition system. In particular, as we are considering an environment populated by several agents which act at the same time, the transitions describe how the environment evolves from one state to another one when a set of actions are performed concurrently. We have developed a variant of Boolean Modal Logic to reason about combinations of actions and their effects. In Section 5.2.2, the agents module describes the agents populating the multiagent system by means of the actions each of them is able to execute. We have assumed that if an agent is able to execute a set of actions, then it can execute each subset of these actions concurrently. Also for this module we have developed a logic that formalize the capability of a group of agents to perform a concurrent action of a certain type, no matter which actions the other agents perform. In other words, we have formalized a notion of α-ability for actions similar to CL-PC, but with the difference that we admit that two agents may execute the same action. In Section 5.2.3 have combined the two modules in a module describing the whole multiagent system. Combining the previous logics in a straightforward way, we have obtained our cooperation logic which models the α-ability of groups of agents. In Section 5.3 we describe a concrete scenario and we model it with our cooperation logic.

In Section 5.4 we describe how to formalize a Power Structure by means of the notion of α-ability. Briefly, goals are formalized as formulas of our cooperation logic and a group of agents has the power to achieve a set of goals G if it is α-able to satisfy all the goals in G at the same time. Formalizing the notion of power as the α-ability of groups of agents has the drawback that it is coalitional monotonic, which we know to be not a good property to reason about possible coalitions. For this reason we have formalized other two notions of power in Section 5.5. The first notion starts from α-ability, but it requires also that if a group of agents Q has the power to achieve, for example, a certain goal g, then all the agents in Q have to be necessary, i.e. no subgroups of Q have the same power. The second notion of power states, instead, that all the agents in Q have to be useful, i.e. there exists a concurrent action that assures the achievement of g such that Q is α-able to execute this concurrent action, whereas all its subgroups are not. We notice that since this last notion of power explicitly refer to a concurrent action, it cannot naturally be expressed in those cooperation logics in which actions are implicitly present in the underlying semantic but not explicitly referred in the logic language.

Chapter 6

Conclusions

6.1 An overview

In the field of Multiagent Systems an important issue is how agents can help each other when they are not self-sufficient in the achievement of their own goals, or when collaboration allows better results. In this thesis we have studied the formation of coalitions among goal-directed agents. We started from a theory of social power and dependence developed in the field of sociology [Eme62, Gou93, YC93] and introduced in the field of multiagent systems by Castelfranchi et al. [Cas03, Cas00, SD01]. In this context a coalition is intended as a group of agents which agree 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 exchange can be considered but not all of them are realizable. For this reason, we have developed two criteria of admissibility establishing which coalition cannot be formed under the assumption that the agents are self-interest. The first admissibility criterion, the do-ut-des property, is a generalization of the notion of reciprocity defined by Conte et al. [CS02b]. There, the notion of reciprocity has been formulated under the hypothesis that each agent gains only one goal in an exchange, we consider the case where agents have multiple goals and may obtain more than a single goal in an exchange. The second admissibility criterion, the composition property, describes the attitude of the agents to deal separately coalition formation processes when these processes does not effect each other. This attitude can be justified by considering that a coalition formation process usually becomes more costly and has less chance to succeed when the number of the agents involved in it increases.

We have considered coalition formation processes as collusive behavior supported by unanimous and enforced agreements. In particular we mean with collusive behaviors that the agents have the possibility to decide if to join or not a coalition without imposition by anyone. An agreement is unanimous when, to be considered effective, all the agents involved have to sign it, whereas it is enforced when agents cannot deviate from what established in the agreement, once it has been stipulated. These assumptions give us the possibility to overcome some issues in the study of the coalition formation processes that constitute stand-alone research areas. In particular, the assumption of unanimity restrict a type of interaction protocol in the coalition formation process. Several other protocols could be considered, one for all the majority voting, however a large class of coalition formation processes, as the ones stipulated by means of contracts, uses unanimous approvals.

The assumption of enforced agreements, instead overcomes the problem

to explicitly consider in our framework the level of mutual trust among the agents. Mutual trust may depend on several factors as the reputation of the agents, the normative status of a coalition formation process, i.e. if it is supported by some institutional fact, the way agents contribute with their own burdens in a coalition, for example, concurrently or in turn.

We also have defined a logic based framework which enables to formalize how groups of agents have to collaborate in order to achieve one or more goals. This framework can be a tool to constructively instantiate the power of groups of agents which are the input data used to *calculate* the admissible coalitions.

6.2 Focussing on the right level of abstraction

In this thesis we have developed a framework to characterize the admissibility of a coalition among goal-directed agents. First of all we have studied which representation of the multiagent system would have been suitable to define our admissibility criteria. This representation is the Power Structure. Power Structures do not directly describe *how* a group of agents have to coordinate in order to achieve some goals, they simply describe which groups of agents have the power to achieve which sets of goals and which of these powers are compatible with each other. Focussing on this level of abstraction give us the possibility to neglect some details that do not play a role in the definition of the do-ut-des property and the composition property. Furthermore, this level of abstraction makes also our approach more general, as different frameworks formalizing distributed planning can be used to instantiate a Power Structure. We have assumed that the notion of power could be described by more simple structures than the Power Structure, for example by considering that groups of agents can always achieve goals individually and not in case only in *bunches*. However, we have compared the Power Structures with these alternative structures and we have shown how some examples are correctly described only by Power Structures.

The definition of a Power Structure has an analogous in the studies on coalition formation in the field of Game Theory and, in particular, in the notion of effectivity [Ros72, Pel98]. The main difference is that the notion of effectivity regards the capability of groups of agents to restrict the set of possible outcomes, whereas our notion of power is expressed in terms of the capability of a group of agents to achieve one or more goals.

We have defined different properties that a Power Structure can satisfy depending on the application domain and of the possible meaning we give to the notion of power. Some of these properties, as for example super-additivity

or goal-monotonicity, are a reformulation of properties already studied for the effectivity functions, some others, as the notion of group-minimality, derives from the study of some particular domains, as the one shown in Example 11.

6.3 Admissibility criteria for coalition formation

A Power Structure is a way to describe a multiagent system in such manner to abstract from the details that are not required in our formalizations of admissible coalitions, the do-ut-des property and the composition property. Once again we borrowed from Game Theory, and in particular from Cooperative Game Theory [vNM44, Aum61, OR94] the idea of formalizing a notion of dominance among different coalitions and to define our admissibility criteria as no-dominance criteria.

However, 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.

First, we model the preferences of an agent simply comparing, by means of the inclusion relation, the set of goals it obtains and provides in two different coalitions. Roughly, an agent prefers a coalition to another one if the set of goals obtained in the first coalition contains the set of goals obtained in the second one, whereas the set of goals it provides in the first coalition is contained in the one it provides in the second coalition. This simple principle does not compare directly the single goals or sets of goals containing different goals, under the principle that goals are not directly comparable. This is a crucial difference with the preferences modeled in Game Theory where each outcome can be compared with others. Putting it in mathematical terms, our preference relations are partial pre-orders among possible coalitions whereas preference relations in Game Theory are total pre-orders.

The second difference among our approach is that both the definitions of the do-ut-des property and of the composition 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.

By formulating the do-ut-des property and the composition property by means of a no-dominance criterion, we provided a characterization of these

criteria in terms of the qualitative preferences of the agents. However, the notion of reciprocity is given as a property of nets of exchanges. So, we have reformulated our admissibility criteria in these terms for a particular class of Power Frames called singleton Power Frames. These reformulations not only clarify the meaning of our criteria, they also provide some insight in order to design an algorithm to select the set of admissible coalitions among all the potential coalitions. We have proposed an algorithm that finds all the coalitions contained in a coalition which satisfy the composition property. It has been shown that this problem is, in general, NP-hard. However, it has been also shown that, whenever a coalition is found which does not satisfy the composition property, the complexity of the problem is at least halved. Therefore, in the case several subcoalitions of a coalition do not satisfy the composition property, then the complexity of this problem is drastically reduced.

Finally, we have characterized a certain class of cost-benefit analyses which is compatible with the qualitative preference relation used to define the do-ut-des property and we have compared the do-ut-des property with the notion of core. We have shown not only that these two admissibility criteria are not in contradiction, but also that the do-ut-des property can be used as a qualitative methods to restrict the search space of a notion which is even more restrictive than the notion of core. This result *a posteriori* is not totally surprising as, even if we use qualitative preferences, the domination relation we use in the do-ut-des property is more restrictive than the one used in the notion of core.

6.4 Formalization of the notion of power

Many logic-based formalisms have been developed in order to represent the power of groups of agents to achieve certain states of affairs.

Pauly [Pau02a] provides a modal logic, Coalition Logic, which axiomatizes α-ability, a notion of power initially developed in Game Theory. Informally, a group of agents Q is α-able for a state of affairs ϕ if and only if there exists a joint strategy for Q which, no matter what the other agents do, ensures the satisfaction of ϕ. Another well-know logic to reason about the α-ability is ATL, in which the expressivity of Coalition Logic is enriched with CTL temporal formulas [AHK02].

All these approaches directly describe the power of groups of agents to see to a certain state of affairs ϕ without explicitly representing what they actually do in order to achieve ϕ. This fact has two drawbacks. First, it is not easy to reason about some meta-information as the costs required to achieve

a certain state of affairs or how these costs are distributed. Second, even if these logics correctly describe if a group of agents has the power achieve a state of affairs, they do not provide, as required in cooperative problem solving or social reasoning mechanism, any clue about which group of agents has the power to achieve a desired state of affairs.

Thus, inspired by the works on cooperative problem solving [WJ94] and social reasoning mechanism [SD01], we have formalized the notion of α-ability by means of two modules: in the first module, the environment module, we have addressed the problem to represent a causal model of an environment in which several agents act at the same time. In the environment module it is possible to describe which plans, intended as concurrent actions, can achieve a certain state of affairs, independently from which other actions can in case be executed. In the second module, the agents module, the capabilities of the agents are modeled and in particular the possibility for a group of agents to execute a plan, no matter what the other agents do. Combining these two modules in a single module, we can express the fact that a group of agents is α-able to achieve a state of affairs ϕ when there exists a plan α assuring ϕ and the group can execute α without being obstructed by the other agents.

We have provided a complete axiomatization for the two separated modules as well as for their combination. We noted that the axiomatization of the combined module is straightforward, given the completeness of the two logics of the underlying modules. We see this as indicative of the success of the modularization approach we have adopted.

Finally, we have seen how it is possible to define three different notions of power with our logic-based framework. The first notion of power simply states that a group of agents Q has the power to achieve a state of affairs ϕ if it is α-able to achieve it. The second notion of power adds a condition of minimality on the group of agents Q by imposing that all the subgroups of Q have not the same power to achieve ϕ. This means that all the agents in Q are necessary for the satisfaction of ϕ. The third notion of power relaxes the minimality condition of the second one by imposing the existence of a shared plan which provides the satisfaction of ϕ and such that all the agents in Q are required for its execution. In other words, all the agents in Q have to be useful for the satisfaction of ϕ. We notice that if the first two notion of power can be defined also with the other approaches [Pau02a, AHK02, vdHW05], the last requires an explicit description of the plan used to satisfy a state of affairs ϕ.

6.5 Future works

In this section we outline possible improvements and extensions of our work. We have developed our research essentially on two different levels of abstraction and we have used for each of them two different formalisms. On one hand, we have used a modal logic to formalize coordination policies and the notion of power. On the other hand, we have used elementary set theory to develop admissibility criteria for collaboration and coalition formation. Finally, we have used graph theory to develop the relative algorithms. A possible improvement of our framework regards the possibility to use a unique formalism and the relative technics to address all these aspects. For example, logic based formalisms could be used to describe admissibility criteria and emphasize the axioms characterizing the meaning of admissible coalitions. This would provide another test bench to discuss about the correctness of our admissibility criteria.

Apart from issues about formalisms, each of the two levels of abstractions offers different starting points for further researches. With respect to the Power Structure we said that a Power Structure describes a single perspective of a multiagent system, for example the perspective of an external agent, a central manager of the agents community, or the perspective of an internal agent. In the first case the coalition formation processes are centralized in the sense that agents may not know each other, they simply inscribe themselves to the community describing their goals and personal capabilities, then the manager calculates the power of groups of agents, eventually using our modal logic, and it uses, for example, the composition property to propose admissible coalitions to the other agents. In the second case, we can consider distributed coalition formation processes similar to the direct task allocations described in [Fer99]. In this case each agent has its own view of the multiagent system, represented by a Power Structure, and it uses the composition property in order to propose the formation of admissible coalitions to the other agents.

However, as described in [SD01], when agents have different information on the multiagent system, they may detect different opportunities of coalition formation. This way, another kind of power can be the object of exchanges in the multiagent system: knowledge. For example, it could be the case that an agent ag_1 knows that another agent ag_3 has the power to achieve a certain goal g of a third agent ag_2. Now, in the perspective of open multiagent systems, ag_2 may not be informed about the existence of ag_3, and asking to ag_1 if it knows someone able to achieve g, ag_1 can use this information profitably. A first opportunity for ag_1 is to exchange this information for another information or for the achievement of one of its goals. A second

opportunity is to be delegated by ag_2 for the achievement of g and asking to ag_2 in exchange, for example, the achievement of one of its own goals. When ag_1 has been delegated for the achievement of g then it deals with ag_3 privately, exchanging one of the goals of ag_3 for g. This way, not revealing who actually has the power to achieve g, ag_1 maintains ag_2 dependent on it as it does not permit to ag_2 to use by itself this information in its own future exchanges.

This power of informing [Cas03] requires to include in our framework the possibility for the agents to reason to their own believes as well as on the believes of the others. Furthermore, exchanges of information involve an updating process on the Power Structures of the agents.

Also with respect to our cooperation logic, several aspects can be improved in future works. First, our logic enables to reason only about concurrent actions and their effects, but not on sequence of concurrent actions. Thus, we consider a simple one-step world which transits from an initial state to a final state. Nevertheless, several logics enables to reason about plans composed by sequences or cycles of actions and the corresponding sequences of states. In the area of Cooperation Logics, in particular, ATL [AHK02] enables to describe the power of a group of agents to achieve a state of affairs, for example, arbitrarily far in the future, or in the next step, or always in the future. Leaving the environment module as it is, we can enrich the language to allow us to express more complex plans and temporal properties. A natural choice is to introduce in our language operators borrowed from Propositional Dynamic Logic (PDL) [HTK00] as the concatenation operator, tests, arbitrarily repetition. With these tools we can express complex plans, then, we expect to be able to express, for example, that a plan is guaranteed to result sometimes in the future in a state where ϕ is true. Somewhat disappointingly, PDL does not provide us with a way of doing that: the expression $[\alpha]\phi$ captures that all halting executions end up in a ϕ-state. The problem is that α might not halt at all, in which case ϕ may not become true either, even if $[\alpha]\phi$ is. How this problem can be solved is object of future works.

Finally, another possible improvement of our cooperation logic concerns the description of the capabilities of the agents. Now the agent module describes, for each agent, the actions that it can execute by assuming, on one hand, that these actions do not depend on the particular state in which the environment is, on the other hand that an agent is able to execute any combination of its actions. However, in several cases both these assumptions are false. For example, in the game PAC-MAN the set of possible actions depends on the position of the PAC-MAN in the walls-pills world. On the other hand, the set of actions that a pianist may execute corresponds to all the notes in the piano, but not all the possible combinations of notes can be

concurrently performed.

Bibliography

[AGPS04] L. Ardissono, A. Goy, G. Petrone, and M. Segnan. Interaction with web services in the adaptiveweb. In *Adaptive Hypermedia and Adaptive Web-Based Systems, Third International Conference (AH 2004)*, volume 3137 of *Lecture Notes in Computer Science*, pages 14–23, Eindhoven, The Netherlands, 2004.

[AHK02] R. Alur, T.A. Henzinger, and O. Kupferman. Alternating-time temporal logic. *Journal of ACM*, 49(5):672–713, 2002.

[AHM+98] 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.

[Alo98] E. Alonso. How individuals negotiate societies. In *Proceedings of the 3rd International Conference on Multi-Agent Systems (ICMAS'98)*, pages 18–25, Paris, France, 1998.

[Aum61] R.J. Aumann. The core of a cooperative game without side payments. *Transaction of the American Mathematical Society*, 98:539–552, 1961.

[BDH99] C. Boutilier, T. Dean, and S. Hanks. Decision-theoretic planning: Structural assumption and computational leverage. *Journal of Artificial Intelligence Reasearch*, 11:1–94, 1999.

[BL02] G. Boella and L. Lesmo. A game theoretic approach to norms and agents. *Cognitive Science Quarterly*, pages 492–512, 2002.

[Bou94] C. Boutilier. Toward a logic for qualitative decision theory. In *Proceedings of the fourth International Conference on Principles of Knowlwdge Representation and Reasoning (KR'94)*, Bonn, Germany, 1994.

[BS99] S. Brainov and T. Sandholm. Power, dependence and stability in multiagent plans. In *Procs. of the Sixteenth National Conference on Artificial Intelligence (AAAI'99)*, pages 11–16, Orlando, Florida, 1999.

[BSvdT04a] G. Boella, L. Sauro, and L. van der Torre. Abstraction from Power to Coalition Structures. In *Proceedings of the 16th Eureopean Conference on Artificial Intelligence (ECAI'04)*, pages 965–966, 2004.

[BSvdT04b] G. Boella, L. Sauro, and L. van der Torre. Power and dependence relations in groups of agents. In *Procs. of International Conference on Intelligent Agent Technology (IAT'04)*, Beijin, Cina, 2004.

[BvdT03] G. Boella and L. van der Torre. Attributing mental attitudes to normative systems. In *Proceedings of The Second International Joint Conference on Autonomous Agents & Multiagent Systems (AAMAS'03)*, pages 942–943, Melbourne, Australia, 2003.

[BvdT04] G. Boella and L. van der Torre. Contracts as legal institutions in organizations of autonomous agents. In *Proceedings of The Third International Joint Conference on Autonomous Agents & Multiagent Systems (AAMAS'04)*, pages 948–955, New York, USA, 2004.

[BvdTar] G. Boella and L. van der Torre. A game theoretic approach to contracts in multiagent systems. *IEEE Trans. SMC, Part C*, to appear.

[Cas00] 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.

[Cas01] C. Castelfranchi. The theory of social functions: challenges for computational social science and multi-agent learning. *Journal of Cognitive Systems Research*, 2:5–38, 2001.

[Cas03] C. Castelfranchi. The Micro-Macro Constitution of Power. *ProtoSociology*, 18-19, 2003.

[CE82] E. M. Clarke and E. A. Emerson. Design and synthesis of synchronization skeletons using branching-time temporal logic. In *Logic of Programs, Workshop*, 1982.

[Che80] B. F. Chellas. *Modal Logic: an introduction*. Cambridge University Press, 1980.

[CKB04a] D. Cornforth, M. Kirley, and T. Bossomaier. Agent heterogeneity and coalition formation : Investigating the effects of diversity in a multi agent system. In *Proceedings of The Third International Joint Conference on Autonomous Agents & Multiagent Systems (AAMAS'04)*, pages 556–563, New York, USA, 2004.

[CKB04b] D. Cornforth, M. Kirley, and T. Bossomaier. 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.

[CKM+03] F. Curbera, R. Khalaf, N. Mukhi, S. Tai, and S. Weerawarana. The next step in web services. *Communications of the ACM*, 46(10):29–34, 2003.

[CL90] P. R. Cohen and H. J. Levesque. Intention is choice with commitment. *Artificial Intelligence*, 42:213–261, 1990.

[CR90] T. Cormen and C. Leiserson R. Rivest. *Introduction to Algorithms*. MIT Press, 1990.

[CS02a] 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.

[CS02b] R. Conte and J. S. Sichman. Dependence graphs: Dependence within and between groups. *Computational & Mathematical Organization Theory*, 8(2):87–112, 2002.

[dL96] 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.

[DR94] 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.

[DvdNT98] B. Dutta, A. van den Nouweland, and S. Tijs. Link formation in cooperative situations. *International Journal of Game Theory*, 27(2):245–256, 1998.

[DW04] 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.

[Eme62] R. M. Emerson. Power-dependence relations. *American Sociological Review*, 27(1):31–41, 1962.

[Fer99] J. Ferber. *Multi-Agent Systems, An introdiction to Distributed Atificial Intelligence*. Addison-Wesley, 1999.

[Fos02] I. Foster. What is the grid? a three point checklist. *GRIDtoday*, 1(6), 2002.

[Gib85] A. Gibbons. *Algorithmic Graph Theory*. Cambridge University Press, 1985.

[GKT+04] 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 & Multi-agent Systems (AAMAS'04)*, pages 782–789, New York, USA, 2004.

[GMS98] L. Giordano, A. Martelli, and C. Schwind. Dealing with concurrent actions in modal action logic. In *Proceedings of the 13th European Conference on Artificial Intelligence (ECAI'98)*, pages 537–541, Brighton, UK, 1998.

[Gou93] R. V. Gould. Collective action and network structure. *American Sociological Review*, 58(2):182–196, 1993.

[GP90] G. Gargov and S. Passy. A note on boolean modal logic. In P. Petkov, editor, *Mathematical Logic, Proceedings of Heyting88*, pages 311–321. Plenum Press, 1990.

[GvDar] V. Goranko and G. van Drimmelen. Decidability and complete axiomatization of the alternating-time temporal logic. *Theoretical Computer Science*, to appear.

[HKT84] 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.

[HTK00] David Harel, Jerzy Tiuryn, and Dexter Kozen. *Dynamic Logic*. MIT Press, Cambridge, MA, USA, 2000.

[KP00] S. Kraus and T. Plotkin. Algorithms of distributed task allocation for cooperative agents. *Theoretical Computer Science*, 242(1-2):1–27, 2000.

[KR76] R.L. Keeney and H. Raiffa. *Decisions with Multiple Objective*. John Wiley & Sons, Inc., 1976.

[KS96] M. Klusch and O. Shehory. A polynomial kernel-oriented coalition algorithm for rational information agents. In *Proceedings of Second Intenrational Conference on Multiagent Systems (ICMAS'96)*, pages 157–164, Kyoto, Japan, 1996.

[KS03] S. Kraus and O. Schechter. Strategic Negotiation for Sharing a Resource between Two Agents. *Computational Intelligence*, 19:9–41, 2003.

[McM92] K.L. McMillan. *Symbolic Model Checking*. PhD thesis, CMU University, 1992.

[MGS05] D. Manini, R. Gaeta, and M. Sereno. Performance modeling of p2p file sharing applications. In *FIRB-Perf Workshop on Techniques, Methodologies and Tools for Perfomance Evaluation of Complex Systems*, Torino, Italy, 2005.

[MMSSG98] A. Meca-Martinez, J. Sanchez-Soriano, and I. García. Strong equilibria in claim games corresponding to convex games. *International Journal of Game Theory*, 27(2):211–217, 1998.

[Mye97] R. B. Myerson. *Game Theory*. Harvard University Press, 1997.

[Nas50] J. Nash. The bargaining problem. *Econometrica*, 28:155–162, 1950.

[New80] Allen Newell. The knowledge level (presidential address). *AI Magazine*, 2(2):1–20, 33, 1980.

[OR94] M. Osborne and A. Rubinstein. *A Course in Game Theory*. MIT Press, 1994.

[Pau02a] M. Pauly. A Modal Logic for Coalitional Power in Games. *Journal of Logic and Computation*, 12:146–166, 2002.

[Pau02b] M. Pauly. On the Complexity of Coalitional Reasoning. *International Game Theory Review*, 4(3):237–254, 2002.

[Pel98] B. Peleg. Effectivity functions, game forms, games, and rights. *Social Choice Theory*, 15:67–80, 1998.

[Por70] I. Porn. *The Logic of Power*. Basil Blackwell, 1970.

[RG95] A. S. Rao and M. P. Georgeff. Bdi agents: From theory to practice. In *Proceedings of the 1st International Conference on Multiagent Systems (ICMAS'95)*, pages 312–319, San Francisco, California, USA, 1995.

[Ros72] R. W. Rosental. Cooperative games in effectiveness form. *Journal of Economic Theory*, 5:88–101, 1972.

[SCCD94] J. S. Sichman, R. Conte, C. Castelfranchi, and Y. Demazeau. A social reasoning mechanism based on dependence networks. In *Proceedings of the Eleventh European Conference on Artificial Intelligence (ECAI'94)*, pages 188–192, Amsterdam, The Netherlands, 1994.

[SD01] J. S. Sichman and Y. Demazeau. On social reasoning in multiagent systems. *Revista Iberoamericana de Inteligencia Artificial*, 13:68–84, 2001.

[SGvdHW05] L. Sauro, J. Gerbrandy, W. van der Hoek, and M. Woodridge. Reasoning about action and cooperation. Submitted to The Fifth International Joint Conference on Autonomous Agents & Multiagent Systems (AAMAS'06), 2005.

[Sim55] H.A. Simon. A behavioral model of rational choice. *Quarterly Journal of Economics*, 69:99–118, 1955.

[SK98] O. Shehory and S. Kraus. Methods for Task Allocation via Agent Coalition Formation. *Artificial Intelligence*, 101(1-2):165–200, 1998.

[SL97] T. Sandholm and V. Lesser. Coalitions among computationally bounded agents. *Artificial Intelligence*, 94(1):99–137, 1997.

[SL02] T. Sandholm and V. Lesser. Leveled-commitment contracting. *AI Magazine*, 23(3):89–100, 2002.

[SLA+99] 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.

[SS69] L.S. Shapley and M. Shubik. Pure competition, coalitional power and fair division. *International Economic Review*, 10(3):337–362, 1969.

[Tam97] M. Tambe. Towards flexible teamworks. *Journal of Artificial Intelligence Research*, 7:83–124, 1997.

[Tar55] A. Tarski. A lattice-theoretical fixpoint theorem and its applications. *Pacific Journal of Mathematics*, 5:285–309, 1955.

[Tar72] R. Tarjan. Depth-first search and linear graph algorithms. *SIAM Journal of Computation*, 1(2):146–160, 1972.

[vdHW03] 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.

[vdHW05] W. van der Hoek and M. Woodridge. On the logic of cooperation and propositional control. *Artificial Intelligence, 64:1-2, pp. 81–119.*, 64(1-2):81–119, 2005.

[vdTDB05] L. van der Torre, M. Dastani, and J. Broersen. How to decide what to do? *European Jurnal of Operations Research*, 160(3):762–784, 2005.

[vNM44] J. von Neumann and O. Morgensten. *Theory of Games and Economic Behavior*. John Wiley and Sons, 1944.

[WD04] M. Wooldridge and P.E. Dunne. On the computational complexity of qualitative coalitional games. *Artificial Intelligence*, 158(1):27–73, 2004.

[WJ94] 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.

[Woo02] M. Wooldridge. *An Introduction to MultiAgent Systems*. John Wiley & Sons, 2002.

[YC93] T. Yamagishi and K. S. Cook. Generalized exchange and social dilemmas. *Social Psycology Quarterly*, 56(4):235–248, 1993.

[ZR89] G. Zlotkin and J. S. Rosenschein. Negotiation and task sharing among autonomous agents in cooperative domains. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence (IJCAI'89)*, pages 912–917, Detroit, Michigan, 1989.

[ZR96a] G. Zlotkin and J. S. Rosenschein. Compromise in negotiation: Exploiting worth functions over states. *Artificial Intelligence*, 84(1-2):151–176, 1996.

[ZR96b] 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.

List of frequent symbols

$\leq_i$, do-ut-des preference relation of the agent ag_i (see Definition 18, p. 59).
$\preccurlyeq_i$, c-p preference relation of the agent ag_i (see Definition 26, p.75).
$\precsim_i$, game theoretic preference relation of the agent ag_i (see Definition 33, p. 105).
$\models^a$, see Definition 47, p. 124.
$\vdash^a$, see Definition 48, p. 125.
$\models^e$, see Definition 44, p. 121.
$\vdash^e$, see Definition 45, p. 122.
$\models^m$, see Definition 51, p. 128.
$\vdash^m$, see Definition 52, p. 129.
$\models^p$, see Definition 42, p. 121.

Ag, the set of all the agents in a multiagent system (see Definitions 13 and 49, p. 50 and p. 127).
Ac, the set of actions which the agents of a multiagent system can execute (see Definition 46, p. 124).
Cs, the set of all consequences (see Definition 33, p. 105).
Env, the environment in which agents *live* (see Definition 40, p. 120).
$\mathcal{G}[pfr]$, the AND-graph relative to a Power Frame (Definition 25, p. 70).
Gl, the set of all the goals of the agents (see Definition 13, p. 50).
MaS, a multiagent system (see Definition 49, p. 127).
NTU, a cooperative game without transferable payoffs (see Definition 33, p. 105).
PS, a Power Structure (see Definition 13, p. 50).
St, the set of all the states (see Definitions 29 and 40, p. 101 and p. 120).
att, see Definition 33, p 105.
act, see Definition 46, p. 124.
adv, see Definition 16, p. 59.
ag_i, an agent.
blc, a balance function of a cost-benefit analysis (see Definition 39, p. 119).
bnf, a function which quantifies the benefits is a cost-benefit analysis (see

Definition 39, p. 119).
comp, see Definition 13, p. 50.
cost, a function which quantifies the costs in a cost-benefit analysis (see Definition 39, p. 119).
duf, see Definition 23, p. 67.
goals, see Definition 13, p. 50.
obl, see Definition 17, p. 59.
outcome, see Definition 29, p. 101.
pfr, a Power Frame (Definition 15, p. 58).
uf, see Definition 24, p. 68.
utl_i, the utility function of an agent ag_i (see Definition 39, p. 119).