Lecture notes on Coalitional Games*

Stéphane Airiau
Institue for Logic, Language and Computation
University of Amsterdam

Abstract

Forming coalitions is a generic means for cooperation: people, robots, web services, resources, firms can improve their performance by joining forces. The use of coalitions has been successful in domains such as task allocations, sensor networks, and electronic marketplaces. Forming efficient coalitions requires identifying matching synergies between different entities (finding complementary partners, or similar partners, or partners who add diversity). In addition, the different parties must negotiate a fair repartition of the worth created by the coalition. We first review the different game theoretic stability criteria for coalitions and present additional insight from a multiagent perspective. We then survey the different scenarios and the key issues addressed by current research on coalition formation.

1 Introduction

Coalition formation is an important tool for enabling cooperation in agent societies. Social scientists and economists have studied situations where individuals and businesses benefits from joining forces. Whether we consider a set of people in an economy, a set of self-interested agents, a set of tasks, any subset is a coalition with a potential for serving individual or group interests. The process of coalition formation has been studied extensively in game theory, and has produced a set of stability criteria under which the players do not have any incentive to change coalitions. Unfortunately, there is no unique and accepted solution to share the payoff generated by the coalitions. Over the last decade, coalition formation has received increased attention in the multiagent system community: forming dynamic coalitions may lead to more efficient agent societies. Joining a coalition may be beneficial for an agent: the use of other members' resources may facilitate or enable the solution of a problem. An agent, however, may also want to change its coalition to achieve better performance, or it may demand a higher payoff than its current payoff. Game theory prescribes ways to share a payoff obtained by coalition to ensure stability, but it does not describe how to form efficient coalitions.

The coalition formation problem can be decomposed into two subproblems. The first problem is to select the agents in a coalition. Depending on the problem, agents that share complementary or similar expertise are needed. A typical assumption is that the coalition as a whole receives reward or payoff from the environment, and hence, the second problem involves sharing the coalition's reward between its constituents in a fair manner. Multiagent researchers must deal with these two subproblems at the same time [85].

Another problem faced by a multiagent community arises when agents are cooperative and want to collectively optimize an objective function (for example, maximize social welfare). This may require the entire community to be partitioned into groups or coalitions to achieve different subtasks that contribute to the overall goal of the community. It turns out that the search space for the best organization of the agent population into coalitions is large, and multiple search algorithms have been proposed to effectively search this space [27, 54, 81, 85, 89]. Multiple scenarios, for example in task oriented domains [10, 12, 50, 51, 60, 84, 91] or in the electronic marketplace [5, 26, 58, 59, 86, 87, 98, 99] have inspired many coalition formation protocols, many of them based on game theoretic stability criteria, which guarantees a fair distribution of the reward to the coalition members. These scenarios have also brought to light many issues and constraints that classical game theory did not address. One issue is the complexity of computing the stability criteria, which

*Notes prepared for a tutorial given at the 11th European Agent Systems Summer School (EASSS-2009), held in Torino, Italy, September 3rd 2009. Feedback welcome: s.airiau@uva.nl.

is $\mathcal{NP}$-complete in many cases. In addition, the communication complexity can also limit the use of game theoretic concepts. Other issues are related with dynamic environments: agents can enter and leave the system at any time, new tasks may appear in the environment, the environment may be uncertain (uncertainty about the value of the coalitions, about the competence of other agents, etc.). Safety and robustness issues should also be taken into account to guarantee a stable agent society. In addition, researchers must design protocols that are secure to prevent the possibility of manipulation or infiltration by agents or external forces.

The goal of the paper is to survey existing coalition formation protocols and results. We first start by presenting game theoretical stability concepts that provide the basis of many coalition formation protocols in Section 2. In Section 3 we survey the cooperative case where agents search for the optimal coalition structure. Then, in Section 4, we survey the non-cooperative case: we first present the different applications that are studied before describing the issues that these scenarios present. Finally, we present some elements of solution to these problems and discuss future areas of research.

2 A Static View: Game Theoretical Approach

The game theory community has extensively studied the coalition formation problem [43, 69]. The literature is devided into two main cases depending whether utility can be transfered between individuals or not. In a so-called non-transferable utility game (or NTU game), an agent has a preference over the different coalitions in which it is a member. In a transferable utility game (or TU game), it is assumed that agents can compare their utility and that a common scale of utility does exist. In this case, the value of a coalition is the worth the coalition can achieve through cooperation. Both frameworks share one important issue: how to maintain stability? We start with presenting the case of TU games, and then, we will briefly present the NTU games.

2.1 Transferable Utility Games (TU games)

The traditional setting for this study involves a set of players and a characteristic function that provides a reward for each possible coalition or subsets of agents. Traditionally, the characteristic function is common knowledge, and the reward of a coalition depends only on the other players present in its coalition. In general, it is not possible to satisfy the interests of all players at the same time, and unfortunately, there is no single criteria for characterizing an acceptable coalition. The focus of the game theoretical approach is on how to distribute the reward to individual members to maintain stable coalitions.

2.1.1 Notations and types of TU games

In the following, we use a utility-based approach and we assume that "every thing has a price": each agent has a utility function that is expressed in currency units. The use of a common currency enables the agents to directly compare alternative outcomes, and also enables side payments.

We consider a set N of n agents. A *coalition* is a non-empty subset of N. The set N is also known as the *grand coalition*. The set of all coalitions is $\mathscr{C}$ and its cardinality is 2^n. A *coalition structure (CS)* $\mathcal{S} = \{\mathcal{C}_1, \cdots, \mathcal{C}_m\}$ is a partition of N: each set $\mathcal{C}_i$ is a coalition with $\cup_{i=1}^{m}\mathcal{C}_i = N$ and $i \neq j \Rightarrow \mathcal{C}_i \cap \mathcal{C}_j = \emptyset$. The set of all coalition structures is $\mathscr{S}$ and its size is of the order $O(n^n)$ and $\omega(n^{\frac{n}{2}})$ [85]. The *characteristic function* (or *valuation function*) $v : 2^N \mapsto \mathbb{R}$ provides the worth or utility of a coalition. Note that in general, it is assumed that the valuation of a coalition $\mathcal{C}$ does not depend on the other coalitions present in the population, and we will return to this issue later. $\mathfrak{I}$ is the set of all characteristic functions whose value depends only on the coalition's members. A TU game is written as (N, v) where N is the set of agents, and $v \in \mathfrak{I}$.

One example of a TU game is the majority game. Assume that the number of agents $|N|$ is odd and that they decide between two alternatives using a majority vote. Also assume that no agent is indifferent. We model this by assigning to a "winning coalition" the value 1 and to the other ones the value 0, i.e.,

$$v(|\mathcal{C}|) = \begin{cases} 1 \text{ when } |\mathcal{C}| > \frac{|N|}{2} \\ 0 \text{ otherwise} \end{cases}$$

We now describes some types of valuation functions.

Additive (or inessential): $\forall \mathcal{C}_1, \mathcal{C}_2 \subseteq N \mid \mathcal{C}_1 \cap \mathcal{C}_2 = \emptyset$, $v(\mathcal{C}_1 \cup \mathcal{C}_2) = v(\mathcal{C}_1) + v(\mathcal{C}_2)$. When a TU-game is additive, $v(\mathcal{C}) = \sum_{i \in \mathcal{C}} v(i)$, i.e., the worth of each coalition is the same whether its members cooperate or not: there is no gain in cooperation or any synergies between coalitions, which explains the alternative name (inessential) used for such games.

Superadditive: $\forall \mathcal{C}_1, \mathcal{C}_2 \subseteq N \mid \mathcal{C}_1 \cap \mathcal{C}_2 = \emptyset$, $v(\mathcal{C}_1 \cup \mathcal{C}_2) \geq v(\mathcal{C}_1) + v(\mathcal{C}_2)$, in other words, any pair of coalitions is best of by merging into one. In such environment, social welfare is maximized by forming the grand coalition.

Subadditive: $\forall \mathcal{C}_1, \mathcal{C}_2 \subseteq N \mid \mathcal{C}_1 \cap \mathcal{C}_2 = \emptyset$, $v(\mathcal{C}_1 \cup \mathcal{C}_2) \leq v(\mathcal{C}_1) + v(\mathcal{C}_2)$: the agents are best of when they are on their own,i.e. cooperation not desirable.

Convex games. First let us call $v(\mathcal{C} \cup \{i\}) - v(\mathcal{C})$ the marginal contribution of a player i to coalition $\mathcal{C}$, i.e., it is the increase of value of coalition $\mathcal{C}$ due to the presence of agent i. We call then a valuation convex if for all $\mathcal{C} \subseteq T$ and $i \notin T$ $v(\mathcal{C} \cup \{i\}) - v(\mathcal{C}) \leq v(T \cup \{i\}) - v(T)$. So a valuation function is convex when the marginal contribution of each player increases with the size of the coalition he joins. Convex valuation functions are super-additive.

Unconstrained. The valuation function can be superadditive for some coalitions, and subadditive for others: some coalitions should merge when others should remain separated. This is the most difficult and interesting environment.

The valuation function provides a value to a set of agents, not to individual agent. The *payoff distribution* $x = \{x_1, \cdots, x_n\}$ describes how the worth of the coalition is shared between the agents, where x_i is the payoff of agent i. We also use the notation $x(\mathcal{C}) = \sum_{i \in \mathcal{C}} x(i)$. A *p*ayoff configuration (PC) is a pair $(\mathcal{S}, x)$ where $\mathcal{S} \in \mathscr{S}$ is a CS and x is a payoff distribution. $\mathscr{P}$ denotes the set of all PCs. In a TU game (N, v), the valuation function is an input, and the main question is what PC will form: what are the coalitions that will form and how to distribute the worth of the coalition. We are now going to present some rationality concepts for PCs, which describes good properties that a solution of the coalition formation should have.

2.1.2 Rationality concepts

We now discuss different rationality concepts for payoff distributions, i.e. some properties that link the coalition values to the agents' individual payoff.

Efficiency: $x(N) = v(N)$ the payoff distribution is an allocation of the whole worth of the grand coalition to all the players. In other words, no utility is lost at the level of the population.

Individual rationality : An agent i will be a member of a coalition only when $x_i \geq v(\{i\})$, i.e., to be part of a coalition, a player must be better off than when it is on its own

Group rationality : $\forall \mathcal{C} \subseteq N$, $x(\mathcal{C}) \geq v(\mathcal{C})$

Pareto optimal payoff distribution It may be desirable to have a payoff distribution where no agent can improve its payoff without lowering the payoff of another agent. More formally, a payoff distribution x is Pareto optimal iff

$$\nexists y \in \mathbb{R}^n \mid \exists i \in N \mid \{y_i > x_i \text{ and } \forall j \neq i, \, y_j \geq x_j\}.$$

Two notions will be helpful to discuss some solution concepts. The first is the notion of *imputation*, which is a payoff distribution with the minimal acceptable constraints.

Definition 2.1 (Imputation)**.** *An* imputation *is a payoff distribution that is efficient and individually rational for all agents.*

An imputation is a solution candidate for a payoff distribution, it can also be used to object a payoff distribution. The second notion is the *excess* which measures the improvement due to a change of coalition in a CS.

Definition 2.2 (Excess)**.** *The* excess *related to a coalition $\mathcal{C}$ given a payoff distribution x is* $e(\mathcal{C}, x) = v(\mathcal{C}) - x(\mathcal{C})$.

The excess measures the total amount that the players would gain or lose if they were to form coalition $\mathcal{C}$. When $e(\mathcal{C}, x) > 0$, the agents have an incentive to form the new coalition $\mathcal{C}$. Many coalitions can have positive excess. Some stability concepts (the kernel and the nucleolus) are based on the excess of coalitions. The core can also be defined using the notion of excess.

The solution of the coalition formation problem is a payoff configuration $(\mathcal{S}, x)$. The problems of finding the CS (i.e., finding which coalitions are formed) and of finding a payoff distribution (i.e., sharing the value of the coalitions between the members) cannot generally be separated. In the following, we are going to present different solution concepts proposed in the literature. Each has pros and cons, and none is clearly better than all others.

2.1.3 The Stable Set

The concept of dominance between outcomes in non-cooperative game theory can be applied to the context of coalition formation. Let $(\mathcal{R}, x) \in \mathscr{P}$ and $(\mathcal{S}, x) \in \mathscr{P}$ be two PCs. $(\mathcal{R}, x)$ dominates $(\mathcal{S}, y)$ iff $\exists T \subset N$ such that $\forall i \in T$, $x_i > y_i$ and $v(T) \geq x(T)$ and we note $(x, \mathcal{R}) \succ (y, \mathcal{S})$. In other words, there exists a coalition T where each member prefers the allocation x over y, and they can obtain this utility. There may or may not be dominance between any two PCs. Also, dominance may not be transitive.

One way to characterize fairness is to ensure, for all agents, that there is no other PC that dominates the currently used PC. The stable set $V \in \mathcal{P}$ is a set of PCs that satisfies the following conditions: 1) Internal Stability: $\forall x \in V$, $\nexists y \in V$ such that $y \succ x$ and 2) External stability: $\forall z \in \mathcal{P} \setminus V$, $\exists y \in V$ such that $y \succ z$. In other words, the internal stability ensures that no PC in the stable set dominates any other PC in the stable set. The external stability ensures that for any PC that is not in the stable set, there exists one PC in the stable set that dominates this PC. Hence, the stable set represents a set of acceptable PCs from a global point of view, which is akin to the Pareto Optimality concept of non-cooperative game theory: individual player can prefer some PCs over others in the stable set, but not all the players will have the same preference. Just as in non-cooperative game theory Pareto Optimality is accepted as a desirable equilibrium criteria, the stable set can be viewed a desirable property of a solution PC. Though in many situations, the stable set is guaranteed to be nonempty, it is not always the case.

2.1.4 The Core

The core was first introduced in [37] and is the most attractive and natural way to define stability: if a payoff distribution is in the core, no agent has any incentive to be in a different coalition. This is akin to the Nash equilibrium concept of non cooperative games. The core actually presents a stronger condition: no set of agents can benefit by forming a new coalition, which corresponds to the group rationality assumption.

Definition 2.3. *A payoff distribution $x \in \mathbb{R}^n$ is in the* core *of a game (N, v) iff x is an imputation that is group rational, i.e.* $Core(N, v) = \{x \in \mathbb{R}^n \mid \sum_{i \in N} x_i = v(N) \ \wedge \ \forall \mathcal{C} \subseteq N \ x(\mathcal{C}) \geq v(\mathcal{C})\}$

A payoff distribution is in the core when no group of agents has any interest in rejecting it, i.e., no group of agent can gain by forming a different coalition. Note that this condition has to be true for all subsets of N, in particular, this ensures individual rationality. The core can thus be defined as a payoff structure that satisfies weak linear inequalities. The core is therefore closed and convex. Another way to define the core is in terms of excess: a payoff configuration is in the core when there exists no coalition that has a positive excess.

Definition 2.4. *The core is the set of payoff distribution $x \in \mathbb{R}^n$, such that $\forall R \subset N$, $e(R, x) \leq 0$*

This definition is attractive as it shows that no coalition has any complaint: each coalition's demand can be granted.

There are, however, multiple concerns associated with using the notion of the core. First, the core can be empty: the conflicts captured by the characteristic function cannot satisfy all the players simultaneously. When the core is empty, at least one player is dissatisfied by the utility allocation and therefore blocks the coalition. Let us consider the following example from [43]: $v(\{A, B\}) = 90$, $v(\{A, C\}) = 80$, $v(\{B, C\}) = 70$, and $v(N) = 120$. In this case, the core is the PC where the grand coalition forms and the associated payoff distribution is $(50, 40, 30)$. If $v(N)$ is increased, the size of the core also increases. But if $v(N)$ decreases, the core becomes empty. The other issue for adopting the core as stability concept concerns computational complexity. Checking whether a payoff distribution is in the core is $\mathcal{NP}$-hard [23]. Additionally, determining the non-emptiness of the core, even for a superadditive game, is $\mathcal{NP}$-hard [22]. There exists a transfer scheme to converge to the core [101]. In addition, [31] introduces a process that leads to a core allocation in non-superadditive games.

There are few extensions to the concept of the core. As discussed above, one main issue of the core is that it can be empty. In particular, a member of a coalition may block the formation so as to gain a very small payoff. When cost to build the coalition are considered, it can be argued that it is not worth blocking a coalition for a small utility gain. The strong and weak ϵ-core concepts model this possibility. The constraints defining the strong (respectively the weak) ϵ-core become $\forall T \subseteq N, x(T) \geq v(T) - \epsilon$, (respectively $\forall T \subseteq N, x(T) \geq v(T) - |T| \cdot \epsilon$). In the weak core, the minimum amount of utility required to block a coalition is per player, whereas for the strong core, it is a fixed amount.

In most traditional work in game theory, the superadditivity of the valuation function is not explicitly stated, but it is implicitly assumed in when the core is defined. In particular, this assumption ensures that the grand coalition always emerge. That is one of the reason that the efficiency is defined with respect to the grand coalition. In case of an unconstrained valuation function, the grand coalition may not form, a different coalition structure may emerge. Aumann and Drèze discuss in [7] why the coalition formation process generates a CS. One reason may be that the valuation is not superadditive (and they provide some discussion about why it may be the case). Another reason is that a coalition structure may "reflect considerations that are excluded from the formal description of the game by necessity (impossibility to measure or communicate) or by choice".

We can define the Core for CS, we borrow the definitions from [20], but the definitions are similar to [31, 84]. A payoff distribution x is *efficient* with respect to a CS S when $\forall \mathcal{C} \in S$, $\sum_{i \in \mathcal{C}} x_j = v(\mathcal{C})$. A payoff distribution is an *imputation* when it is efficient (with respect to the current CS) and individually rational (i.e. $\forall i \in N$, $x_i \geq v(\{i\})$). The set of all imputations for a CS S is denoted by $I(S)$. We can now state the definition of the core:

Definition 2.5 (Core)**.** *The core of a game (N, v) is the set of all payoff configuration (S, x) such that $x \in I(CS)$ and $\forall \mathcal{C} \subseteq N$, $\sum_{i \in \mathcal{C}} x_j \geq v(\mathcal{C})$.*

2.1.5 The Kernel

The Kernel was first introduced by Davis in [28]: in the Kernel, the strength of the players is measured by the maximum excess the agent can obtain by forming a new coalition with different agents. An agent can consider a payoff distribution to be acceptable by comparing its own 'strength' with the 'strength' of other members of its coalition. When both agents have equal strength, they do not have any incentive to leave the coalition. Although its definition is not as intuitive as the core, the Kernel exists and is always non-empty, and hence, its use as a solution concept has been popular in multiagent research, for example in [10, 12, 27, 48, 51, 53, 92].

Definition

We recall that the *excess* related to coalition $\mathcal{C}$ for a payoff distribution x is defined as $e(\mathcal{C}, x) = v(\mathcal{C}) - x(\mathcal{C})$. For two agents k and l, the *maximum surplus* $s_{k,l}$ of agent k over agent l with respect to x is $\max_{c \in \mathscr{C} | k \in c, \, l \notin c} e(c, x)$. This maximum surplus can be used by agent k to shows its strength over agent l: assuming it is positive and that the agent can claim all of it, agent k can argue that it will be better off without agent l and hence should be compensated more utility for staying in the current coalition. Two agents that are in the same coalition are in equilibrium when any one of the following condition hold:

- $s_{k,l} = s_{l,k}$: agents have equal power
- $s_{k,l} > s_{l,k}$ and $x_l = v(l)$: k dominates l, but l has the minimum utility it can get (individual rationality), hence k cannot claim more utility from player l
- $s_{k,l} < s_{l,k}$ and $x_k = v(k)$: l dominates k and k has the minimum utility.

A payoff distribution is in the Kernel when all the agents in the coalition are in equilibrium. An approximation of the Kernel is the ϵ-Kernel where the equality $s_{k,l} = s_{l,k}$ above is replaced by $|s_{k,l} - s_{l,k}| \leq \epsilon$. Notice that this definition requires considering all pairs of agents in the coalition. The Kernel is always non-empty [69]. One property of the Kernel is that agents with the same maximum surplus, i.e., symmetric agents, will receive equal payoff. For ensuring fairness, this property is important.

Computational Issues

One method for computing the Kernel is the Stearns method [96]. The idea is to build a sequence of side-payments between agents to decrease the difference of surpluses between the agents. At each step of the sequence, the agents with the largest maximum surplus difference exchange utility so as to decrease their surplus: the agent with smaller surplus makes a payment to an agent with higher surplus so as to decrease their surplus difference. After each side-payment, the maximum surplus over all agents decreases. In the limit, the process converges to an element in the Kernel. Computing an element in the Kernel may require an infinite number of steps as the side payments can become arbitrarily small, and the use of the ϵ-Kernel can alleviate this issue. A criteria to terminate Stearns method is proposed in [92], and we present the corresponding algorithm in Algorithm 1.

Algorithm 1: Transfer scheme to converge to a ϵ-Kernel-stable payoff distribution for the CS S

```
compute-ε-Kernel(ε, S)
repeat
    for each coalition C ∈ S do
        for each member i ∈ C do
            for each member j ∈ C, j ≠ i, do // compute the surplus for two members of a coalition in S
                s_ij ← max_{R∈𝒞|(i∈R, j∉R)} v(R) − x(R)
    δ ← max_{(i,j)∈N²} |s_ij − s_ji|;
    (i*, j*) ← argmax_{(i,j)∈N²} s_ij − s_ji;
    if (x_j* − v({j}) < δ/2) then          // payment should be individually rational
        d ← x_j* − v({j*});
    else
        d ← δ/2;
    x_i* ← x_i* + d;
    x_j* ← x_j* − d;
until δ/v(S) > ε ;
```

Computing a Kernel distribution is of exponential complexity. In Algorithm 1, computing the surpluses is expensive as we need to search through all coalitions that contains a particular agent and does not contain another agent. Note that when a side-payment is performed, it is necessary to recompute the maximum surpluses. The derivation of the complexity of the Stearns method to compute a payoff in the ϵ-Kernel can be found in [48, 92], and the complexity for one side-payment is $O(n \cdot 2^n)$. Of course, the number of side-payments depend on the precision ϵ and on the initial payoff distribution. They derive an upper bound for the number of iterations: converging to an element of the ϵ-Kernel requires $n \log_2(\frac{\delta_0}{\epsilon \cdot v(S)})$, where δ_0 is the maximum surplus difference in the initial payoff distribution. To derive a polynomial algorithm, the number of coalitions must be bounded. The solution used in [48, 92] is to only consider coalitions whose size is bounded in the interval K_1, K_2. The complexity of the truncated algorithm is $O(n^2 \cdot n_{coalitions})$ where $n_{coalitions}$ is the number of coalitions with a size between K_1 and K_2, which is a polynomial of order K_2.

Fuzzy Kernel

In order to take into account the uncertainty in the knowledge of the utility function, a fuzzy version of stability concept can be used. Blankenburg et al. consider a coalition to be Kernel-stable with a degree of certainty [13]. This work also presents a side-payment scheme and shows that the complexity is similar to the crisp Kernel, and that they can use the idea proposed in [48] to derive a polynomial coalition formation algorithm. This approach assumes a linear relationship of the membership and coalition values.

Fuzzy coalitions can also allow agents to be members of multiple coalitions at the same time, with possibly different degrees of involvement [11]. It can be mutually beneficial for an agent to be in two different coalitions. It may be beneficial for the agent to be in both coalition. In addition, the two coalitions may need the competence of the same agent, though the coalitions do not have any incentive to merge as they may not have anything to do with each other. This solution may allow to form coalitions that involve only agents that need to work together. In the previous example, without the possibility of being member of multiple coalitions, the two coalitions should merge to benefit from the agent participation, and agents that would not need to be in the same coalition are forced to be in the same coalition. In [11], the degree of involvement of an agent in different coalition is a function of the risk involved in being in that coalition. The risk is quantified using financial risk measures. This work presents a definition of the Kernel based on partial membership in coalitions and introduces a coalition formation protocol that runs in polynomial time.

2.1.6 The Nucleolus

The nucleolus is also based on the notion of excess and has been introduced by Schmeidler [88]. The excess measures the amount of "complaints" of a coalition for a payoff distribution. The goal of the Kernel is to minimize the excess between any two agents. In the nucleolus, the goal is to reduce the excess in the sense of the lexicographical ordering[1]. For a given payoff distribution, we can compute the excess for each coalition and order the excesses from larger to smaller values. The vector obtained is an ordered vector of complaints for a payoff distribution. To compare two payoffs distribution, we can use the lexicographic order. A payoff distribution is in the nucleolus when it yields the "least problematic" sequence of complaints, i.e. when no other payoff distribution is better.

We now provide a more formal account. First, let $e(x)$ be the vector containing all the excess of all the possible coalitions of the n agents for a payoff distribution x, i.e. $e(x) = \{e(R,x)_{R \in 2^N \setminus \{N,\emptyset\}}\}$. Let $e^{>}(x)$ be the values of $e(x)$ sorted in decreasing order. In other word, the first entry of $e^{>}(x)$ is the maximum excess: the agents involved in the corresponding coalition have the largest incentive to leave their current coalition and form a new one. Put another way, the agents involved in that coalition have the most valid complaint.

To compare two PCs $(x, \mathcal{R})$ and $(y, \mathcal{C})$, we can use the lexicographic ordering $\succ$ defined as

$$e(x) \succ e(y) \Leftrightarrow \left(\exists i^* \in 2^{N-2} \mid \left\{\forall (i < i^*)\; e_i^{>}(x) = e_i^{>}(y) \text{ and } e_{i^*}^{>}(x) > e_{i^*}^{>}(y)\right\}\right).$$

For a given CS $\mathcal{S}$, the nucleolus of the game is a set of payoff distributions such that the corresponding vector of excess $e^{>}(x)$ is minimal. The nucleolus tries to minimize the possible complaints (or minimize the incentives to create a new coalition) over all possible payoff distributions.

Definition 2.6. *The* nucleolus *is defined as the following set:* $\{x \in \mathbb{R}^n \mid (\nexists y \mid e(y) \succ e(x))\}$.

The nucleolus is guaranteed to be nonempty and it is unique. Moreover it is related to other solution concepts. When the core is nonempty, the nucleolus is in the core. The nucleolus is also contained in the Kernel. The nucleolus can be computed using a sequence of linear programs of decreasing dimensions. Each these group size is, however, exponential. In some special cases, the nucleolus can be computed in polynomial time ([52, 29]), but in the general case, computing the nucleolus is not guaranteed to be polynomial. Only a few papers in the multiagent community used the nucleolus [103].

2.1.7 Shapley Value

The Shapley value is designed to provide a fair payoff distribution in a coalition [90]. The Shapley value and its variants have been used in multiagent system [44, 47, 105]. First we present a set of axioms that defines the Shapley value. We then present a different interpretation of the Shapley value which is based on marginal surplus. Finally, we present computational issues and a value that is cheaper to compute which derived from the Shapley value.

An Axiomatic Characterization

The Shapley value is uniquely defined by the following three axioms:

Axiom 1 ("Dummy actions") : It is not rational for an agent i to enter a coalition if it loses utility, hence an agent should obtain at least the worth $v(\{i\})$ it receives when it forms a singleton.

[1] the lexicographic ordering is the ordering used to order names on a phonebook.

If the presence of agent i does not improve the worth of a coalition by more than $v(\{i\})$, the agent does not bring anything to the coalition and can be considered as a dummy. Therefore, if $v(\mathcal{C} \cup i) - v(\mathcal{C}) = v(i)$ for all $\mathcal{C} \in N$, $i \notin \mathcal{C}$, then $x_i = v(\{i\})$.

Axiom 2 ("Symmetry") : When two agents have the same contributions, they should be rewarded equally: for $i \neq j$ and $\forall S \in N$ such that $i \notin S$ and $j \notin S$, if $v(\mathcal{C} \cup i) = v(\mathcal{C} \cup j)$, then $x_i = x_j$.

Axiom 3 ("Additivity") : For any two characteristic functions v and w, if the agents get payoff profiles x and y, respectively, then they should get the payoff $x+y$ if the characteristic function is $v + w$. As stated in [69], this axiom is hard to motivate: "the structure of $v + w$ may induce behavior that is unrelated to that induced by v or w separately".

The Shapley value is the unique value that is budget balanced and that satisfies axioms 1, 2, and 3 [69]. Unlike the core, this value always exists and is unique. Note that other axiomatisations have been proposed (by H.P. Young [104] or R. Myerson [66]).

Ordinal Marginal Contribution

Another interpretation of the Shapley value is based on the notion of ordered marginal contribution. Consider a coalition $\mathcal{C}$ and few agents that want to join this coalition. One way to compare the contribution of different agents is to compare the marginal surplus of each agent i to this coalition: $\Delta_i(\mathcal{C}) = v(\mathcal{C} \cup i) - v(\mathcal{C})$: the greater the surplus, the more wealth will be distributed to the agents. The Shapley value is the average marginal surplus over all possible join order of the agents.

Let us consider that a coalition $\mathcal{C}$ is built incrementally with one agent at a time entering the coalition. Also consider that the payoff of each agent i in the coalition $\mathcal{C}$ is its marginal contribution. In this case, the value of each agent would depend on the order in which the agents enter the coalition, which may not be fair. For example, if agents form a coalition to take advantages of price reduction when buying large quantities: agents that start the coalition may have to spend large setup cost. To alleviate this issue, the Shapley value averages each agents' payoff over all possible orderings: the value of agent i in coalition $\mathcal{C}$ is the average marginal value over all possible orders in which the agents may join the coalition:

$$Shapley(\mathcal{C}, i) = \sum_{i \notin S \subset \mathcal{C}} \frac{|S|!(|\mathcal{C}| - |S| - 1)!}{|\mathcal{C}|!} \left(v(S \cup \{i\}) - v(S)\right).$$

Other properties

At noted before, the Shapley value always exist and is unique. When the valuation function is superadditive, the Shapley value is individually rational, i.e., it is an imputation. When the core is non-empty, the Shapley value may not be in the core. However, when the valuation function is convex, the Shapley value is also group rational, hence, it is in the core.

Computational Issues

The nature of the Shapley value is combinatorial as all possible orderings to form a coalition needs to be considered. This computational complexity can sometimes be an advantage as agents cannot benefit from manipulation. For example, it is $\mathcal{NP}$-complete to determine whether an agent can benefit from false names [102]. Nevertheless, the structure of the characteristic function can be exploited to rapidly compute the Shapley value. Conitzer and Sandholm consider characteristic functions that can be decomposed over multiple issues that interest only a subset of the agents [23]. When few agents are concerned by each issue, and the number of issues is small, fast computation of the Shapley value is possible. In [56], the characteristic function is represented by a set of rules and the computation of the Shapley value can be performed in time linear in the number of rules. We now provide more details about these approaches.

Conitzer and Sandholm analyze the case where the agents are concerned with multiple independent issues that a coalition can address. For example, performing a task may require multiple abilities, and a coalition may gather agents that work on the same task, but with limited or no interactions between them. A characteristic function v can be decomposed over T issues when it is of the form $v(\mathcal{C}) = \sum_{t=1}^{T} v_t(\mathcal{C})$). In that case, the Shapley value $Shapley(\mathcal{C}, i)$ for agent i for the characteristic function v is the sum of the Shapley value $Shapley_t(\mathcal{C}, i)$ for agent i over the t different issues: $Shapley(\mathcal{C}, i) = \sum_{t=1}^{T} Shapley_t(\mathcal{C}, i)$. If only a small number of agents is concerned about an issue, computing the Shapley value for the particular issue can be cheap. For an issue

t, the characteristic function v_t concerns only the agents in $\mathcal{I}_t$ when $\forall \mathcal{C}_1 \in \mathscr{C}, \mathcal{C}_2 \in \mathscr{C}$ such that $\mathcal{I}_t \cap \mathcal{C}_1 = \mathcal{I}_t \cap \mathcal{C}_2 \Rightarrow v_t(\mathcal{C}_1) = v_t(\mathcal{C}_2)$. When the characteristic function v is decomposed over T issues and when $|\mathcal{I}_t|$ agents are concerned about each issue $t \in [1...T]$, computing the Shapley value takes $O(\sum_{t=1}^{T} 2^{|\mathcal{I}_t|})$.

The characteristic function can also be represented by a set of "rules" [56]. Each "rule" associates a pattern and a value. The pattern is a conjunction of agents. A coalition matches the pattern if it is a superset of the pattern. The value of a coalition is the sum over the values of all the rules that apply to the coalition. The pattern can be extended to positive or negative literals for representational efficiency: a positive literal represents the presence of an agent in a coalition, whereas a negative literal represents the absence of an agent in the coalition. This representation can be much more concise than the traditional representation of the characteristic function for certain game. Under this representation, the algorithm for computing the Shapley value runs in linear time of the input.

Bilateral Shapley Value

In order to reduce the combinatorial complexity of the computation of the Shapley value, Ketchpel introduces the Bilateral Shapley Value (BSV) [44]. The idea is to consider the formation of a coalition as a succession of merging between two coalitions. Two disjoint coalitions $\mathcal{C}_1$ and $\mathcal{C}_2$ with $\mathcal{C}_1 \cap \mathcal{C}_2 = \emptyset$, may merge when $v(\mathcal{C}_1 \cup \mathcal{C}_2) \geq v(\mathcal{C}_1) + v(\mathcal{C}_2)$. When they merge, the two coalitions, called founders of the new coalition $\mathcal{C}_1 \cup \mathcal{C}_2$, share the marginal utility as follows: $BSV(\mathcal{C}_1) = \frac{1}{2}v(\mathcal{C}_1) + \frac{1}{2}\left(v(\mathcal{C}_1 \cup \mathcal{C}_2) - v(\mathcal{C}_2)\right)$ and $BSV(\mathcal{C}_2) = \frac{1}{2}v(\mathcal{C}_2) + \frac{1}{2}\left(v(\mathcal{C}_1 \cup \mathcal{C}_2) - v(\mathcal{C}_1)\right)$. This is the expression of the Shapley value in the case of an environment with two agents. In $\mathcal{C}_1 \cup \mathcal{C}_2$, each of the founder gets half of its 'local' contribution, and half of the marginal utility of the other founder. Given this distribution of the marginal utility, it is rational for $\mathcal{C}_1$ and $\mathcal{C}_2$ to merge if $\forall i \in \{1,2\} v(\mathcal{C}_i) \leq BSV(\mathcal{C}_i)$. Note that symmetric founders get equal payoff, i.e., for $\mathcal{C}_1, \mathcal{C}_2, \mathcal{C}$ such that $\mathcal{C}_1 \cap \mathcal{C}_2 = \mathcal{C}_1 \cap \mathcal{C} = \mathcal{C}_2 \cap \mathcal{C} = \emptyset$, $v(\mathcal{C} \cup \mathcal{C}_1) = v(\mathcal{C} \cup \mathcal{C}_2) \Rightarrow BSV(\mathcal{C} \cup \mathcal{C}_1) = BSV(\mathcal{C} \cup \mathcal{C}_2)$. Given a sequence of successive merges from the states where each agent is in a singleton coalition, we can use a backward induction to compute a stable payoff distribution [47]. Though the computation of the Shapley value requires looking at all of the permutations, the value obtained by using backtracking and the BSV only focuses on a particular set of permutations, but the computation is significantly cheaper.

2.2 Voting Games

The formation of coalition is usual in parliaments or assemblies that take some decisions. It is therefore interesting to consider a particular class of coalitional games that models voting in an assembly. For example, we can represent an election between two candidates as a voting game where the winning coalitions are the coalitions of size at least equal to the half the number of voters. The formal definition follows.

Definition 2.7 (voting game)**.** *A game* (N, v) *is a* voting game *when*

- *the valuation function takes only two values: 1 for the winning coalitions, 0 otherwise.*
- v *satisfies* unanimity*:* $v(N) = 1$
- v *satisfies* monotonicity*:* $S \subseteq T \subseteq N \Rightarrow v(S) \leq v(T)$.

Unanimity and monotonicity are natural assumptions in most cases. Unanimity reflects the facts that all agents agree, hence the coalition should be winning. Monotonicity tells that the addition of agents in the coalition cannot turn a winning coalition into a losing one. A variant of a voting game is a weighted voting game where each agent has a weight and a coalition needs to achieve a threshold or quota to be winning. The European Union uses a combination of weighted voting games (a decision is accepted when it is supported by 55% of Member States, including at least fifteen of them, representing at the same time at least 65% of the Union's population).

Definition 2.8 (weighted voting game)**.** *A game* (N, v, q, w) *is a* weighted voting game *when*

- $w = (w_1, w_2 \ldots, w_{|N|})$ *is a vector of weights, one for each voter*
- *A coalition* $\mathcal{C}$ *is winning (i.e.* $(v(\mathcal{C}) = 1)$ *iff* $\sum_{i \in \mathcal{C}} w_i \geq q$*, it is losing otherwise (i.e.* $(v(\mathcal{C}) = 0)$
- v *satisfies monotonicity:* $S \subseteq T \subseteq N \Rightarrow v(S) \leq v(T)$.

The computational complexity of such games have been studied in [30, 35]. For example, the problem of determining whether the core is empty is polynomial. When the core is non-empty, the problem of computing the nucleolus is also polynomial, otherwise, it is an NP-hard problem.

Another interesting topic is to measure the power of a voter in a game. For example, how much power has a voter with a given weight in a weighted voting game? Does it have any power at all if the weight is small? If the weight is large, what does it mean? Multiple indices have been proposed to answer these questions, and we now present few of them. We say that a voter i is pivotal for a coalition $\mathcal{C}$ when it turns it from a losing to a wining coalition, i.e. $v(\mathcal{C}) = 0$ and $v(\mathcal{C} \cup \{i\}) = 1$. Let w be the number of winning coalitions. For a voter i, let η_i be the number of coalition for which i is pivotal, i.e., $\eta_i = \sum_{S \subseteq N \setminus \{i\}} v(S \cup \{i\}) - v(S)$.

Shapley-Shubik index: is the Shapley value of the voting game, its interpretation in this context is the percentage of the permutations of all players in which i is pivotal. One issue is that the voters do not trade the value of the coalition, though the decision that the voters vote about is likely to affect the entire population.

Banzhaf index $\beta_i = \frac{\eta_i}{2^{n-1}}$ is the probability that agent i is pivotal.

Coleman index: Coleman defines three indices [21]: the power of the collectivity to act $A = \frac{w}{2^n}$ (A is the probability of a winning vote occurring); the power to prevent action $P_i = \frac{\eta_i}{w}$ (it is the ability of a voter to change the outcome from winning to losing by changing its vote); the power to initiate action $I_i = \frac{\eta_i}{2^n - w}$(it is the ability of a voter to change the outcome from losing to winning by changing its vote, the numerator is the same as in P, but the denominator is the number of losing coalitions, i.e., the complement of the one of P)

There is a slight difference in the probability model between the Banzhaf β_i and Coleman index P_i: in Banzhaf, all the voters but i votes randomly whereas in Coleman, the asssumption of random voting also applies to the voter i. Hence the Banzhaf index can be written as $\beta_i = 2P_i \cdot A = 2I_i \cdot (1-A)$

2.3 Game with Coalition Structure

In the description of the core and the Shapley value, an underlying assumption was that the agents were forming the grand coalition (this is due to the fact that they were initially defined for super-additive games). For example in the core, the imputation was supposed to be efficient for the grand coalition, not efficient for any other CS. The game with coalition structure extends a TU games by assuming that a certain CS is already formed. The CS may model affinities among the players, which can have external causes (e.g., the affinities can be based on location: each agent may come from the same country or university, or trust relations, etc). Given a TU game (N, v) and a CS S, the problem is to find a payoff distribution that reflects these affinities.

Definition 2.9 (Game with coalition structure)**.** *A* game with coalition structure *is a triplet* (N, v, S)*, where* (N, v) *is a TU game, and* S *is a particular CS. In addition, transfer of utility is only permitted within (not between) the coalitions of* S*, i.e.* $\forall \mathcal{C} \in S$*,* $x(\mathcal{C}) = v(\mathcal{C})$*.*

The agents in a coalition share the value of that coalition. In order to discuss their payoffs, the agents can refer to the value of coalitions with agents outside of their coalition (i.e., opportunities that agents would have outside of their coalition). Aumann and Drèze extend the definition of the core and the Shapley value as well as other stability concepts (Nucleolus, Bargaining set, Kernel) [7]. Other values were proposed [39, 70].

The next subsection considers the case where all possible affinities between the agents are considered, i.e., the agents are not constraint to forming a single CS.

2.4 Game with externalities

A traditional assumption in the literature of coalition formation is to consider that the value of a coalition does not depend on non-members actions. In general, this may not be true: some externalities (positive or negative) can create a dependency between the value of a coalition and the actions of non-members. Sandholm attributes these externalities to the presence of shared resources (if a coalition uses some resource, they will not be available to other coalitions), or when there are conflicting goals: non members can move the world farther from a coalition's goal state [84]. Ray and Vohra states that a "recipe for generating characteristic functions is a minimax argument": the value of a coalition $\mathcal{C}$ is the value $\mathcal{C}$ gets when the non-members respond optimally so as to minimize the payoff of $\mathcal{C}$ [83]. This formulation acknowledges that the presence of other coalitions in the population may affect the payoff of the coalition $\mathcal{C}$. As in [40, 83], we can study the interactions between different coalitions in the population: decisions about joining forces or splitting a coalition can depend on the way the competitors are organized. For example, when different companies are competing for the same market niche, a small company might survive against a competition of multiple similar individual small companies. However, if some of these small companies form a viable coalition, the competition significantly changes: the other small companies may now decide to form another coalition to be able to successfully compete against the existing coalition. Another such example is a bargaining situation where agents need to negotiate over the same issues: when agents form a coalition, they can have a better bargaining position, as they have more leverage, and because the other party needs to convince all the members of the coalition. If the other parties also form coalition, the bargaining power of the first coalition may decrease.

We consider that the valuation of a coalition depends on the overall CS: the characteristic function assigns a value for a coalition in each CS it is member of, i.e. $v : 2^N \times \mathscr{S} \mapsto \mathbb{R}$, and $\mathfrak{D}$ denote the set of all characteristic functions whose value depends on the CS. Compared to a characteristic function that does not depend on a CS, i.e., a characteristic function in $\mathfrak{I}$, a coalition may have a different value for every CS where it is present and the input space needed to describe it is exponentially larger.

Given this function, the question remains which coalitions will form and will be stable. In [84], Sandholm proposes to study this case using Normal Form Games. The possible solution concepts are Nash Equilibrium, strong Nash equilibrium (that may be too strong as it may not exist), and Coalitional Proof Nash equilibrium.

In [83], Ray and Vohra study a protocol that incrementally builds coalitions: each coalition is formed after all members accept the proposal from an agent. Each time a coalition is formed, the agents that formed the coalition cannot receive any further proposals, and the process continues with the remaining agents until all agents are in some coalition. The proposition contains the description of all payoff allocations for every CS that finally forms. For all possible coalitions that have already formed, an agent must know how to answer a proposal, and what proposal to make. These decisions can be probabilistic. An equilibrium strategy is a collection of strategies where no agent benefits from a deviation from its strategy. The authors study the case where all agents are similar and only the number of agents in a coalition matters and provide necessary conditions for the existence of an equilibrium. They also provide some insights for the general case.

We believe that forming coalitions in domains where the value depends on the CS is an important problem which deserve more attention. Because of the dependencies of the value and the CS, it is also a harder problem. Recently the topic has raised interest in AI. Rahwan et al. [79] consider the problem of coalition structure generation in this case (we will present this problem later). Michalak et al. [65] tackle the problem of representing such games (use a more compact description, still allowing efficient computation). Elkind et al. [34] consider the restriction for weighted voting games.

2.5 Non Transferable Utility Games (NTU games)

The underlying assumptions behind a TU game is that agents have a common scale to measure the worth of a coalition. Such a scale may not exist in every situation, which leads to the study of games where the utility is non transferable. The definition follows.

Definition 2.10 (NTU Game). *A non transferable utility game (NTU Game) $(N, X, V, (\succ_i)_{i\in N}$ is defined by*

- *a set of agents N;*
- *a set of outcomes X;*
- *a function $V : 2^N \to 2^X$ that describes the outcomes $V(\mathcal{C}) \subseteq X$ that can be brought about by coalition $\mathcal{C}$;*
- *a preference relation $\succ_i$ (transitive and complete) over the set of outcomes for each agent i.*

Intuitively, $V(\mathcal{C})$ is the set of outcomes that $\mathcal{C}$ can bring about by means of its join action. The agents have a preference relation over the outcome.

First, we can note that the definition of the core can easily be modified in the case of NTU games.

Definition 2.11. $Core(V) = \{x \in V(N) \mid \nexists \mathcal{C} \subset N, \nexists y \in V(\mathcal{C}), \forall i \in \mathcal{C}\ y \succ_i x\}$

An outcome $x \in X$ is blocked by a coalition $\mathcal{C}$ when there is another outcome $y \in X$ that is preferred by all the members of $\mathcal{C}$. An outcome is then in the core when it can be achieved by the grand coalition and it is not blocked by any coalition. As is the case for TU game, it is possible that the core of an NTU game is empty.

Another model called *Hedonic games* considers that agents have preferences over the coalitions, i.e. agents value the company of the other members of the coalition, hence the name. In this case, the set of outcomes is the set of coalitions. We first start to give the definition of stability concepts adapted from [14]. Each agent i is then modeled by a preference order $\succsim_i$ over the set of coalitions.

Definition 2.12. *A coalition structure s is **core stable** iff* $\nexists C \subset N \mid \forall i \in C, C \succ_i s(i)$.

Definition 2.13. *A coalition structure s is **Nash stable*** $(\forall i \in N)$ $(\forall C \in s \cup \{\emptyset\})$ $s(i) \succsim_i C \cup \{i\}$

Definition 2.14. *A coalition structure s is **individually stable** iff* $(\nexists i \in N)$ $(\nexists C \in \mathscr{C} \cup \{\emptyset\}) \mid (C \cup \{i\} \succ_i s(i))$ *and* $(\forall j \in C,\ C \cup \{i\} \succsim_j C)$

Definition 2.15. *A coalition structure s is **contractually individually stable** iff* $(\nexists i \in N)$ $(\nexists C \in \mathscr{C} \cup \{\emptyset\}) \mid (C \cup \{i\} \succ_i s(i))$ *and* $(\forall j \in C,\ C \cup \{i\} \succsim_j C)$ *and* $(\forall j \in s(i) \setminus \{i\},\ s(i) \setminus \{i\} \succsim_j s(i))$

If a CS is core stable, no subset of agents has incentive to leave their respective coalition to form a new one. In a Nash stable CS s, no single agent i has an incentive to leave its coalition $s(i)$ to join an existing coalition in s or create the singleton coalition $\{i\}$. The two other criteria add a constraint on the members of the coalition joined or left by the agent. For an individually stable CS, there is no agent that can change coalition from s(i) to $\mathscr{S}$ yielding better payoff for itself, and the members of $scrS$ should not lose utility. The contractually individual stability requires that in addition, the members of s(i), the coalition left by the agent, should not lose utility.

The definition of Nash, individually and contractually individually stability can be extended to the case where the value of a coalition depends on the CS. Another criterion for a rational agent to be a member of a coalition is individual rationality: an agent i would consider joining a coalition only when it is beneficial for itself. The agent compares the situation when it is on its own and when it is a member of a coalition. However, the payoff the agent gets when it is by itself depends on the CS. The minimum payoff that agent i can guarantee on its own is $r_i = \min_{s \in \mathscr{S}, \{i\} \in s} v(s, i)$ [31] (the minimum is over all the CSs where agent i forms a coalition on its own). An agent is individual rational when its payoff in a coalition with other agents is greater than the minimum payoff it can get on its own.

For some coalition formation problem, it is possible that no CS satisfies any of these stability criteria. Satisfying the individually or contractually individually stability criteria may depend on the protocol used by the agents to form coalition. We can consider that agents in a coalition have the power to veto the entrance of a newcomer, but cannot prevent a member from leaving a coalition. For example, an academic can freely leaves its department to join a new one, provided that no member of the new department will suffer from its presence. In some cases, the coalition left is allowed to demand compensation. For example, as pointed out in [32], a player of a soccer team can join another club, but its former club can receive a compensation for the transfer.

The literature in game theory focuses on finding conditions for the existence of the core. In the AI litereature, Elkind and Wooldridge have proposed a succinct representation of Hedonic games [36].

3 The Cooperative Case: Sharing the Computation of Coalition values and Searching of the Optimal Coalition Structure

In the previous section, the focus was on the agents: when the agents are individually rational or group rational, what coalition should form and how to distribute payoff. The issues of a system designer may be different: the total value obtained by the agents may be of importance. The value obtained by the agents can be correlated to the level of utilization of system resources, which the system designer wants to be maximized. From the point of view of individual agents, in some situations, it also makes sense to form a social welfare maximizing CS, as there will be more utility to share between all agents. In addition, in some environments, the goal of the agents is to maximize the reward of the entire multiagent society: for example, robots in a rescue operation, rovers exploring a new planet, agents monitoring the health of a person or the activity of a complex system share the same goal: optimizing an objective function that depends the entire multiagent system. In that sense, the agents should cooperate to achieve a high value of the objective function.

From the point of view of a system designer, or a set of cooperative agents that wants to form an optimal CS, there are two issues. The first is to generate the value for each coalition. In the game theoretic approach, these values were input, but for use in a real application, these values have to be computed. Sandholm and Lesser [84] provide an example where computing the value of a coalition is NP-complete. The second is to search though these values to locate the optimal CS. Sandholm et al. show that once the coalition values are known, the finding the optimal CS is an $\mathcal{NP}$-complete problem [85].

We consider a TU game (N, v), and we recall that a coalition structure (CS) $s = \{\mathcal{S}_1, \cdots, \mathcal{S}_m\}$ is a partition of N, where $\mathcal{S}_i$ is the i^{th} coalition of agents, and $i \neq j \Rightarrow \mathcal{S}_i \cap \mathcal{S}_j = \emptyset$ and $\cup_{i\in[1..m]}\mathcal{S}_i = N$. $\mathscr{S}$ denotes the set of all CSs and can be represented by a lattice. An example for a population of four agents is provided in Figure 1. The first level of the lattice consists only of the CS corresponding to the grand coalition $N = \{1, 2, 3, 4\}$, the last level of the lattice contains CS containing singletons only, i.e., coalitions containing a single member. Level i contains all the CSs with exactly i coalitions. The number of CSs at level i is $\mathcal{S}(|N|, i)$, where $\mathcal{S}$ is the Stirling Number of the Second Kind[2]. The Bell number, $\mathscr{B}(n)$, represents the total number of CSs with n agents, $\mathscr{B}(n) = \sum_{i=0}^{n} \mathcal{S}(n, k)$. This number grows exponentially, as shown in Figure 2, and is $O(n^n)$ and $\omega(n^{\frac{n}{2}})$ [85]. When the number of agents is relatively large, e.g., $n \geq 20$, exhaustive enumeration may not be feasible.

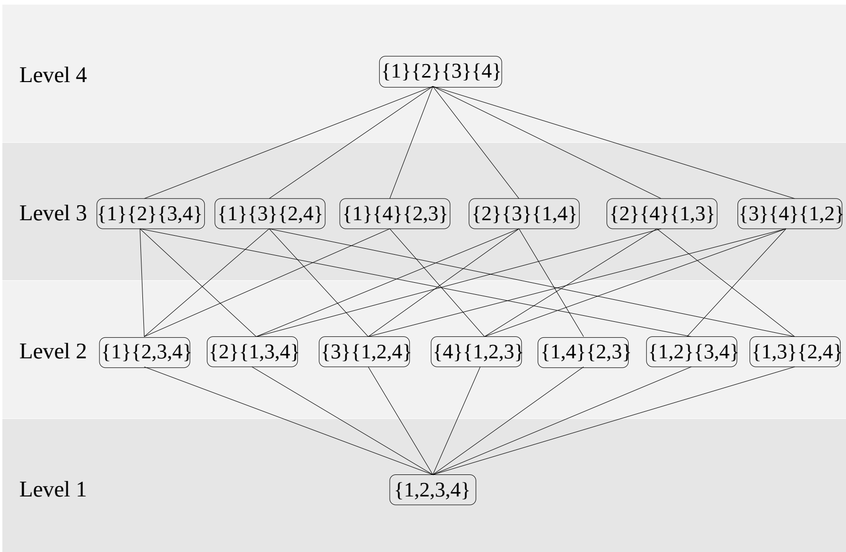

Figure 1: Set of coalition structures for 4 agents.

3.1 Sharing the computation of the coalition values

Shehory and Kraus were the first to propose an algorithm to share the computation of the coalition values (see [91]). In their algorithm, the agents negotiate which computation is performed by which agent. Rahwan and Jennings later proposed an algorithm called DCVC [76] that performs much better. One key observation is that in general, it should take longer to compute the value of a large coalition compared to a small coalition (i.e. the computational complexity of the computation increases with the size of the coalition). Their method improves the balance of the loads by distributing coalitions of the same size to all agents. By knowing the number of agents n participating in the computation an index number (i.e. an integer in the range $\{0..n\}$), the agents determine for

[2] $\mathcal{S}(n, m)$ is the number of ways of partitioning a set of n elements into m nonempty sets.

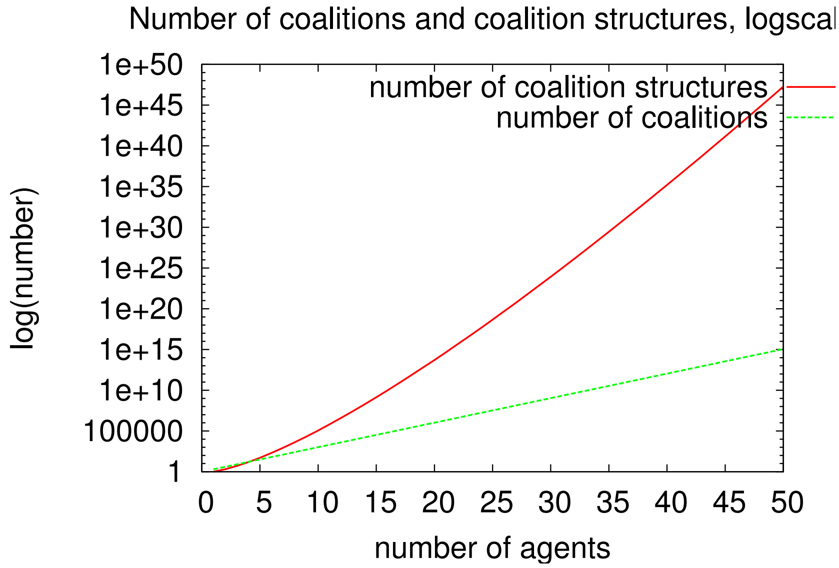

Figure 2: Number of Coalition Structures in a population of n agents.

each coalition size which coalition values to compute. The algorithm can also be adapted when the agents have different known computational speed.

3.2 Searching for the optimal coalition

The difficulty of searching for the optimal CS is the large search space as recognized by existing algorithms [27, 54, 80, 81, 85, 89]. Some algorithms [27, 54, 80, 85] guarantee finding CSs within a bound from the optimum when an incomplete search is performed and when the value of a coalition is independent of non-members. Sen and Dutta [89] propose to use genetic algorithms to search for the optimal CS in the case where there is an underlying pattern in the characteristic function. This algorithm may be faster, but there is no guarantee that the solution found is optimal.

3.2.1 Games with no externalities

Sandholm et al. [85] propose an algorithm that search through a lattice as presented in Figure 1. Their algorithm guarantees that the CS found, s, is within a bound from the optimal $s^\star$ when there is no externalities (i.e. the value of a coalition depends only on its members). The bound considered is $\frac{v(s)}{V(\mathfrak{s}^\star)} \leq \mathcal{K}$. They prove that to ensure a bound, it is necessary to visit a least 2^{n-1} CSs (Theorem 1 and 3 in [85]) which corresponds to the first two levels of the lattice, i.e., the algorithm needs to visit the grand coalition and all the CSs composed of 2 coalitions. The bound improves each time a new level is visited. An empirical study of different strategies for visiting the other levels are presented in [54]. Three different algorithms are empirically tested over characteristic functions with different properties: values of coalitions are uniform in $[0, 1]$ or in $[0, |\mathcal{S}|]$ (where $|\mathcal{S}|$ is the size of the coalition), subadditive and superadditive characteristic functions. The algorithms have different rankings for these cases, demonstrating the importance of the properties of the characteristic function in the performance of the search algorithm. The algorithm by Dang and Jennings [27] outperform the one of Sandholm et al. for low bounds from the optimal. For large bounds, both algorithms visit the first two level of the lattice. Then, when the algorithm by Sandholm et al. continues by searching each level of the lattice, the algorithm of Dang and Jennings only searches specific subset of each level to decrease the bound faster. This algorithm is anytime, but its complexity is not polynomial.

The previous algorithms were based on a lattice as the one presented in Figure 1 where a CS in level i contains exactly i coalitions. The best algorithm to date has been developed by Rahwan et al. and uses a different representation called integer-partition (IP) of the search space. It is an anytime algorithm that has been improved over a series of paper: [80, 81, 77, 78, 82]. In this representation the CSs are grouped according to the sizes of the coalitions they contain, which is called configuration. For example, for a population of four agents, the configuration $\{1, 3\}$ represents CSs that contain a coalition with a singleton and a coalition with three agents. A smart scan of the input allows to search the CSs with two coalitions the grand coalition and the CS containing singletons only. In addition, during the scan, the algorithm computes the average and maximum value for each coalition size. The maximum values can be used to prune the search space (when constructing a configuration, the use of the maximum values of a coalition for each size permits the computation of an upper bound of the value of a CS that follows that configuration, and if the value is not greater than the current best CS, it is not necessary to search through the CSs with that configuration, which prunes the search tree). Then, the algorithm searches the remaining configurations, starting with the most promising ones. During the search of a configuration, a branch and bound technique is used. In addition, during the search, no CS is evaluated twice. Empirical evaluation shows that the algorithm outperform any other current approaches over different distributions used to generate the values of the coalitions.

3.2.2 Game with externalities

The previous algorithm explicitly uses the fact that the valuation function only depends on the members of the coalition, i.e. has no externalities. When it is not the case, i.e., the valuation function depends on the CS, it is still possible to use some algorithms, e.g. the one proposed in [54], but the guarantee of being within a bound from the optimal is no longer valid. Sen and Dutta use genetic algorithms techniques [89] to perform the search. The use of these techniques only assumes that there exists some patterns in the characteristic function. When such patterns exist, the genetic search makes much faster improvement in locating higher valued CS compared to the level-by-level search approach. One downside of the genetic algorithm approach is that there is no optimality guarantee. Empirical evaluation, however, shows that the genetic algorithm does not take much longer to find a solution when the value of a coalition does depend on other coalitions.

More recently, Rahwan et al. and Michalak et al consider the problem for some class of externalities and modify the IP algorithm for the games with externalities [64, 79].

4 The Non-Cooperative Case: Applications and Issues for MAS

Game theory provides great tools for computing stable outcomes, but it does not describe how to form the coalitions except in the two cases that we will present in Section 5.1. In addition, forming coalition of software agents necessitate paying attention to many issues and in particular to the dynamic aspect of the environments [46]. For example, agents have limitations in computational power [84], may have time constraints, may enter or leave the environment at any time, etc. We review application domains where coalition formation has been used in the context of multiagent systems. These applications highlight the important issues of coalition formation for real world applications, e.g, complexity, uncertainty, security, trust, etc.

4.1 Task Allocation Problem

The task allocation problem is well suited to be solved by coalitions of agents. A task may require multiple agents to be performed due to the following reasons:

- All the agents have the required ability or expertise to perform a task, but they do not have enough resource on their own to perform the task. For example, robots have the ability to move objects in a plant, but multiple robots are required to move a heavy box [3, 91].
- Complementary expertise may be required to perform a complex task, and many approaches assume that no agent has all the required expertise to perform a complex task on its own [50, 51, 60, 91]. In the general case, a task can be decomposed into subtasks, and the agents are able to perform a subset of all possible subtasks.
- A variant of the previous case is when an agent coalition can perform a complex task, but another coalition may be better suited to perform the task because of the synergy between members. For example, in a transportation domain [10, 12, 84], a coalition of transporters

is able to deliver goods in a particular regions at a cheaper price than any other coalition. Another example is in [8] where each agent has a set of tasks, a private volume of resource and a private cost to use the resource. In that case, exchanging tasks to reduce cost and improve utility can be beneficial for the society of agents.

All these cases can be generalized into a generic problem: a coalition of agents is formed to perform a complex task and each agent in the coalition plays a role in the completion of the task (they can all have the same or complementary roles). The completion of a task is rewarded by a payoff. The cost associated with the task completion depends on the coalition members. The value of the coalition is the net benefit (payoff minus cost) of completing the task. Hence, the task allocation problem is well modeled by a coalition formation problem where the characteristic function is in $\mathfrak{I}$, i.e. the valuation of a coalition depends only on its members. Of course, it is possible to solve the problem in a different way, e.g., use a market where agents buy and sell their capabilities to perform the task [26], but the literature shows that this problem can be efficiently solved using agent coalitions. Note that an important criteria to assess the quality of a solution is social welfare: the population of agents as a whole is trying to maximize the total utility. Agents are not self-interested, but group rational. A taxonomy is proposed to distinguish different classes of the problem [55] based on three factors: (1) Is the same task likely to be offered again? (2) Does the multiagent system have more than enough/just enough/not enough resources to performing a set of tasks? (3) Is the reward intrinsic to the task, or does it only depend on the members performing the task? They show that some combinations of factors lead to polynomial problems, and other combinations have exponential complexity.

The task allocation problem is in general a hard problem: when agents are limited to perform a single task, the coalition problem resembles the set partitioning problem. When agents are able to perform multiple tasks, the allocation problem gets closer to the set covering problem. In both cases, these problems are $\mathcal{NP}$-complete [91]. The allocation problem as stated earlier has been tackled in [91]: the agents are collectively trying to find a solution that maximizes the social welfare. The paper proposes greedy algorithms in the case of coalitions that can or cannot overlap and with or without precedence order. To ensure polynomial complexity, this paper assumes that coalitions of smaller sizes are preferred, and hence, the search is limited to a coalition of a size smaller than a constant k. This assumption is motivated by the fact that negotiation with a lot of partners can become costly, and hence, keeping the number of partners low ensures a lower cost of negotiation.

Task allocation problems may be even more complex. First, the tasks may be dependent. For example in [91], there is a partial precedence order between the tasks. This assumption is of particular importance in the transportation domain. The existence of task dependency may promote cooperation between the agents as advocated in [4]: the dependence between the tasks may translate into a certain form of dependence between the agents. If agents realize it, they may reciprocally help each other: agent A may help agent B to perform a task needed for the completion of an important task for agent B, and vice versa. Another issue is to have agents being able to participate in multiple coalitions (for example, in a transportation domain, if each item is mapped to a coalition, a vehicle that carries multiple items will be a member of multiple coalitions) [91].

Another model consists of using an auctioneer that proposes to sell complex tasks (or contracts) that can be decomposed into subtasks to agent coalitions [50, 51]. The auction resembles a Dutch auction: at each round, the auctioneer reduces the reward to perform the task. This forces the agents to decide rapidly to form a coalition to take advantage of the high reward. The first coalition that accepts the contract gets it and if multiple coalitions agree, one coalition is chosen at random. Agents that are only capable to perform a subset of the subtasks must propose or join a coalition. At each round, they can propose a coalition or accept to be part of one. When performing a task, each agent incurs a cost, and the interesting point of this work is that this cost is private, other agents may only have an estimate of it (mean). In [50] the focus is on two main heuristics to form a coalition: one based on maximizing the expected surplus (*marginal heuristic*), the other based on choosing agents in the coalition that have non-overlapping expertise (*expert heuristic*). In this paper, reward is distributed equally to the group. Other distribution methods, e.g. proportional and Kernel-based, are investigated in [51]. Maniterski et al. formalize the task allocation problem by an integer program [60]. To find an efficient solution for the cooperative case, they propose a centralized algorithm that is exponential in the number of tasks, but polynomial in the number of agents and subtasks. This solution is based on finding a minimum weighted perfect matching in a bipartite graph. On one side of the graph is the set of agents and on the other side is the set of subtasks. The goal is to find the matching that optimizes utility. The problem is to make sure that all subtasks of a task are completed. To ensure that, they iterate through each element of the power set of the set of tasks. They also provide some impossibility results: in the allocation problem defined, no efficient protocol exists that is individually rational and budget balanced, even in Bayesian Nash equilibrium. In addition no protocol achieving the efficient solution can exist for real cost environments that is individually rational and budget balanced, not even in a Bayesian Nash equilibrium. Unlike in [91] where all the tasks are known in advance, in these works, a coalition is formed incrementally for each task. The order of the tasks may play an important role in the overall payoff of the agents. If the tasks arrive with some pattern, it may be efficient to form similar coalition for similar tasks. Abdallah and Lesser assume the existence of a hierarchy of agents in [1]. When an agent gets a task for which it does not have the necessary resources, it can ask the agent above it in the hierarchy to take care of the task. If agents placed below it can solve subtasks of the task, the agent can decompose the task and assigns it to the agents below in the hierarchy. Learning can be used to choose which agent can perform the task. They show that learning allows for faster and better task assignments.

Belmonte et al. uses a different model [8]: agents are assigned tasks and they can exchange task to maximize social welfare. Their model assumes each agent has a fixed, limited capacity (an agent i can perform k_i task units for a private cost c_i) and an initial assignment of tasks. To improve social welfare, the agent can exchange task, but this exchange is costly. Unlike in [86], the cost to form a coalition is negligible. Belmonte et al. use a particular characteristic function which is superadditive and for which they prove that the core is non-empty and present a payment vector that yield a payoff distribution is in the core. They show that computing the payments has a polynomial complexity. To find an optimal allocation of the tasks, they use a linear programming formulation which can be solved in polynomial time.

Dutta and Sen also consider that agents can improve their performance by mutual help [33]. They compare the performance of agents that are willing to cooperate and selfish agents and show that cooperative agents with complementary expertise that cooperates can develop self-sustaining coalitions by exchanging help, i.e. by performing tasks for other agents in their coalition.

4.2 Electronic Marketplace

Coalition formation has also been used to model firms or agents in the electronic marketplace [5, 26, 58, 59, 86, 87, 98, 99]. The common underlying theme of this body of work originates from [98] where it is shown that consumer agents can form a coalition, or a buying group, to benefit from the quantity discount provided by sellers. From the point of view of a system designer, the problem is again to form a CS, and each coalition is forming a buying group. Desirable property of the CS formed include to be Pareto Optimality, i.e., no other CS gives more to a consumer without giving less to another one [5] and social welfare maximization, which provides the most revenue to the sellers.

In [5], each consumer has a private valuation of the buying groups, and the valuation depends on the preference of the consumer and on the number of agents in the buying group. They consider that there is no side-payment (non-transferable utility), although it is recognized that side payments could allow for more efficient outcome. The goal of Asselin and Chaib-Draa is to define protocols that finds a Pareto Optimal solutions as they consider only non-transferable utilities. They propose a centralized solution. Sarne and Kraus are concerned with equilibrium strategy in an environment where the cost to search for other coalition members, e.g., spend effort to advertise presence, to look for partners, to negotiate price and payment is expensive [86]. The goal of the agents is to increase the size of the coalition so that the benefit from forming a coalition is worth the effort. The dilemma is about executing the task with the current configuration or starting a costly search to find new partners. The paper analyses the equilibrium strategies of the agent. In the following models, all the agents try to obtain an item at the lowest price (hence, the utility is the same for all agents), which corresponds to the transferable utility assumption.

Yamamoto and Sycara propose an auction mechanism where buyers can form buying groups and sellers can bid discount prices to sell large volume of items. Unlike in [98], a buying group is not for a particular item: each buyer agent can have a list of single items or a disjunction of items. Each buying group is managed by an agent that has to solve two problems: (1) given the requests from the buyer agents, the manager agent chooses the sellers and buys the appropriate items, (2) the manager agent chooses the price paid by each buyer agent. To address the first problem, the proposed algorithm performs a greedy search. To answer the second problem, Yamamoto and

Sycara use a surplus sharing rule that ensures a payoff distribution which is in the core. In [59], the agents can bid in combinatorial auctions: agents bid a reservation value for a bundle of items. This makes the problem even more complex since a winner determination problem has to be solved and a stable payoff distribution must be found. The mechanism design aspect of this problem can be found in [58]. Li and Sycara present an algorithm that computes an optimal coalition and a payoff division in the core in [59], but it is not guaranteed to be of polynomial complexity. Hence, they also present an approximation algorithm that is polynomial.

The goal of [57] is to form a coalition formation mechanism that has low communication and computation cost. The model assumes that when an agent randomly meets another agent or a coalition that wants to buy the same item, it is beneficial for both entities to join and form a bigger coalition. Agents are also allowed to leave a coalition to join another one. The protocol they present is not very practical: with some communication, agents should be able to meet agents looking for the same item and they could easily form an efficient buying group. The main contribution of the paper is a mathematical model using first order differential equations that describes the dynamics of the coalitions and allows for computation of a steady state equilibrium. They show that a steady state equilibrium always exists and yields higher utility gain compared to the case where agents are buying on their own, or when leaving a coalition is not allowed. The algorithm does not guarantee Pareto Optimality.

Finally, Vassileva et al. addresses long term coalitions [15, 99]: in many other papers, a coalition is formed to complete a given task, and the coalition is disbanded when the task is accomplished. In contrast, the goal is to form a coalition of agents that will collaborate for a long period of time. The decision to leave a coalition and join a new one should also be a function of the trust put in another agents, i.e., the belief that they will have successful interaction in the future.

Another application of coalition formation in the context of an electronic marketplace is application in service oriented computing. A large number of services are offered on the Internet, offering different services at different price with different quality of service. Blankenburg et al. propose in [11] the use of service Request Agent that can request one (potentially) complex task and a Service Provider Agent that can provide a service. The later can also, given a task and a set of service advertisements, compose services to form a plan that implements the task. The service requester agents only pay the service provider agent if the task is performed on time. The service provider agents must evaluate the risk involved with accepting a request. In addition, a service provider may be involved in more than one coalition, i.e., it can have multiple clients at the same time. Blankenburg et al. propose the use of Fuzzy coalitions to allow agents to be member of multiple coalitions. The agents use a measures of risk from the finance literature and accept a proposal if the risk is below a threshold. To distribute the payoff, Blankenburg et al. define the Kernel for their fuzzy coalition and use Stearns method to converge to a payoff distribution in the Kernel.

4.3 Other Domains

Coalitions of agents have also been used for information seeking agents [47, 48]. An agent is associated with a local database. To answer a query, an agent may require other agents (database). When the agents form a coalition, all agents in the coalition must cooperate: the agents in a coalition must share some of their private data, e.g., dependency information. If an agent does not cooperate, it will not have access to some information schema that are available to members of the coalition. The coalition formation process assumes an utilitarian mechanism, and each agent tries to maximize its expected utility. The bilateral Shapley value is used to determine the payoff distribution in [47]. A Kernel oriented solution is proposed in [48] for the same domain.

Coalitions have been used to track a moving target using a sensor network, a problem introduced in [42]. The problem is to ensure that at least three agents are sensing the target at the same time to perform triangularization. The problem becomes complex as the target is moving and sensors and communication can be faulty. In [93], the goal is for the agents to self-organize and form an appropriate coalition to track the target. The paper used a variant of the contract net protocol to negotiate a coalition that will be used throughout the tracking. Two valuation functions are studied (local and social utility) and different protocols are empirically tested. Soh et al. also solve a real-time tracking problem in [95]. An initiator agent starts the coalition formation process by contacting the neighboring agents that are most suitable for the particular task and engages in negotiation with each of them. Case based reasoning is used to choose the most promising negotiation protocol. In addition, reinforcement learning is used to estimate the utility of a coalition. The coalition formation process may or may not succeed.

A coalition can be used to solve a classification problem: each agent analyses the data with a different algorithm and they form a coalition to decide on the classification of the data [2]. In [73], agents can form committees (which can be viewed as a coalition of agents) to classify a new sponge. Each agent has its own expertise, a set of cases, and uses case based reasoning for the classification problem. In their work, Plaza and Ontañón show how to decide when a committee is needed and how to select the agents to form a committee for a new sponge.

Coalitions of agents have been used in the context of distribution and planning of infrastructure for power transportation [24, 25, 74]. Yeung and Poon model the trading process between firms that generate, transmit or distribute power using agents [74]. Agents rank other agents by possible gains and send the Bilateral Shapley Value of the potential partner when it makes an offer. If both agents send requests to each other, it is beneficial for them to work together and they form a single entity. The process iterates until no further improvement is possible. In the power transmission domain, the problem is to decide whether or not to create a new line or a new plant, and if so, how to share the cost between the different parties involved. [24, 25] uses similar solutions as Yeung and Poon.

Coalitions of agents have been used in the context of planning and scheduling. For example, Pěchouček et al. tackle the problem of planning humanitarian relief operations in [71] and the problem of production planning in [72]. For the humanitarian relief operation scenario, different organizations can form coalitions to be more efficient and provide optimal help to the people. However, the different groups, that have different capabilities, can also have different goals, and hence, they might not want to disclose all available information. In that context, the authors propose a formation of alliances: provided some public information, the agents seek to form groups of agents with the same kind of goals. These alliances can be viewed as long-term agreements between agents, and alliances define a partition of the agents. Unlike alliances, coalitions are viewed as short-term agreements to perform a specific task, and to reduce the search space, coalitions can form within an alliance. In case of impossibility of forming coalitions within an alliance, agents from different alliances can be used. First, different classes of neighborhoods are defined: total (awareness of the other agents), social (share some information, and reason about each other), and cooperation neighborhood (agents that are committed to achieve some goal(s)). The authors are interesting in the amount of information agents have to disclose: when it sends a request, an agent may reveal private or semi-private information (strong information disclosure). This can occur when an agent asks an agent of a different alliance to perform a task (revealing that neither it nor its alliance can complete the task). An agent can also decide to disclose private information when it wants to inform other agents, for instance, when they form alliances (weak information disclosure).

In the context of production planning, instead of using a centralized planning approach, Pěchouček et al. want to use local coalition formation to execute tasks in an efficient manner. One requirement is that agents know their possible collaborators well in order to minimize the communication effort, e.g., agents have a knowledge of the status of surrounding agents, so an agent may ask help from another agent if it knows the agent is not busy. Caillou et al. use the scenario of scheduling classes in a university [17], where a coalition is a schedule. This work considers non-transferable utility. Caillou et al. propose a protocol where a set of acceptable coalitions is passed from agents to agents, and each time, agents can add coalitions or remove coalitions that are not acceptable. The result of the protocol is a Pareto Optimal schedule. The protocol also considers re-using existing solutions to compute a solution to a modified problem (i.g., when a class is removed from the schedule, or a professor is coming, previous solutions of the problem can be used to accommodate these changes).

4.4 Issues

We now highlight issues that have emerged from the applications presented in the last subsection. The protocols and algorithms we cited there presented some solutions to these issues. Some additional issues remained unsolved, for example, dealing with agents that can enter and leave the environment at any time in an open, dynamic environment. None of the current protocols can handle these issues without re-starting computation, and only few approaches consider how to re-use the already computed solution [9, 17].

4.4.1 Stability and Dynamic Environments

Real-world scenarios often present dynamic environments. Agents can enter and leave the environment at any time, the characteristics of the agents may change with time, the knowledge of the agents about the other agents may change, etc.

The game theoretic stability criteria are defined for a fixed population of agents and the introduction of a new agent in the environment requires significant computation to update a stable payoff distribution. For example, for the kernel, all the agents need to check whether any coalition that includes the new agent changes the value of the maximum surplus, which requires re-evaluating $O(2^n)$ coalitions. Given the complexity of the stability concept, one challenge that is faced by the multiagent community is to develop stability concepts that can be easily updated when an agent enters or leave the environment.

In addition, agents that may drop during the negotiation may cause problem for the remaining agents. For example, a protocol that guarantees a Kernel stable payoff distribution is shown not to be 'safe' when the population of agents is changing: if an agent i leaves the formation process without notifying other agents, the other agents may complete the protocol and find a solution to a situation that does not match the reality. Each time a new agent enters or leaves the population, a new process needs to be restarted [12].

In an open environments, manipulations will be impossible to detect: agents may use multiple identifiers (or false names) to pretend to be multiple agents, or the other way around, multiple agents may collude and pretend to be a single agents, or agents can hide some of their skills. Hence, it is important to propose solution concepts that are robust against such manipulations. We will come back later to some of the solution that have been proposed: the anonymity-proof core [102] and anonymity-proof Shapley value [68].

4.4.2 Uncertainty about Knowledge and Task

In real-world scenario, agents will be required to handle some uncertainty: agents may not know some tasks [12] or the value of some coalitions. In such cases, the agents play a different coalitional game that may reduce the payoff of some agents compared to the solution of the true game. Another example can be found in [50] where each agent knows the cost it incurs to perform a given task, but this information is considered private: an agent does not know the cost incurred by other agents and may only estimate these costs. This work assumes the presence of a trusted agent that plays the role of an auctioneer, and they assume that agents truthfully reveal their cost functions. In [45], Ketchpel proposes an auction based protocol to distribute the payoff when there is uncertainty about the valuation of a coalition. In [95], agents also have uncertain and incomplete knowledge, and their approach is to use satisficing rather than optimal solution. Knowing the exact values of all coalitions is quite a strong assumption, and to relax it, [13] uses Fuzzy sets to represent the coalition value. Another approach is to consider expected values of coalitions [18], and agents can have different expectations for a coalition value. This can also be the case when the value function is computationally hard to compute. In [84], computing a value for a coalition requires solving a version of the traveling salesman problem and approximations are used to solve that problem. In addition, when the agents do not use the same algorithm to compute the value of a coalition, all agents may not share the same value for each coalition.

4.4.3 Safety and Robustness

It is also important that the coalition formation process is robust. For instance, communication links may fail during the negotiation phase and hence, some agents may miss some components of the negotiation stages. This possibility is studied in [12] for the KCA protocol [48]: coalition negotiations are not safe when some agents become unavailable (intentionally or otherwise). In particular, the payoff distribution is not guaranteed to be Kernel-stable. [9] empirically studies the robustness of the use of a central algorithm introduced in [8]: the cost to compute a task allocation and payoff distribution in the core is polynomial, but it can still be expensive. In the case of agent failure, the computation needs to be repeated. Belmonte et al. propose an alternative payoff division model that avoids such a re-computation, but the solution is no longer guaranteed to be in the core, it is only close to the core. There is a tradeoff between computational efficiency and the utility obtained by the agent. They conclude that when the number of agents is small, the loss of utility compared to the optimal is small, and hence, the improvement of the computational efficiency can be justified. For a larger number of agents, however, the loss of utility cannot not justify the improvement in computational cost.

4.4.4 Protocol Manipulation

When agents send requests to search for members of a coalition or when they accept to form a coalition, the protocol may require that they disclose some private information [71]. When the agents reveal some of their information, the mechanism must ensure that there is no information asymmetry that can be exploited by some agents [10]. To protect a private value, some protocol [12] may allow the addition of a constant offset to the private value, as long as this addition does not impact the outcome of the negotiation.

Belmonte et al. study the effect of deception and manipulation of their model in [9]. They show that some agents can benefit from falsely reporting their cost. In some other approaches [12, 23], even if it is theoretically possible to manipulate the protocol, it is not possible in practice as the computational complexity required to ensure higher outcome to the malevolent agent is too high. For example, Conitzer shows that manipulating marginal-contribution based value division scheme is $\mathcal{NP}$-hard (except when the valuation function has other properties, such as being convex) [23].

Other possible protocol manipulations include hiding skills, using false names, colluding, etc. The traditional solution concepts can be vulnerable to false names and to collusion [102]. To address this problems, it is beneficial to define the valuation function in terms of the required skills instead of defining it over the agents: only skills, not agents, should be rewarded by the characteristic function. In that case, the solution concept is robust to false names, collusion, and their combination. But the agents can have incentive to hide skills. A straight naive decomposition of the skills will increase the size of the characteristic function, and [103] proposes a compact representation in this case.

4.4.5 Communication

While one purpose of better negotiation techniques may be to improve the quality of the outcome for the agents, other goals may include decreasing the time and the number of message required to reach an agreement. For example, learning is used to decrease negotiation time in [94]. The motivation Lerman's work in [57] is to develop a coalition formation mechanism that has low communication and computation cost. In another work, the communication cost are represented in the characteristic function [97]. It is important to consider the communication complexity for software agents. The complexity of some protocols has been derived. For instance, the exponential protocol in [92] and the coalition algorithm for forming Bilateral Shapley Value Stable coalition in [47] have communication complexity of $O(n^2)$, the negotiation based protocol in [92] is $O(n^2 n_{coalitions})$, and it is $O(n^k)$ for the protocol in [91] (where k is the maximum size of a coalition). The goal of [75] is to analyze the communication complexity of computing the payoff of a player with different stability concepts: they find that it is $\Theta(n)$ when the Shapley value, the nucleolus, or the core is used.

4.4.6 Scalability

When the population of heterogeneous agent is large, discovering the relevant agents to perform a task may be difficult. In addition, if all agents are involved in the coalition formation process, the cost in time and computation will be large. To alleviate this scalability issue, a hierarchy of agents can be used [1]. When an agent discovers a task that can be addressed by agents below this agent in the hierarchy, the agent picks the best of them to perform the task. If the agents below cannot perform the task, the agent passes the task to the agent above it in the hierarchy and the process repeats. The notion of clans [38] and congregations [16], where agents gather together for a long period have been proposed to restrict the search space by considering only a subset of the agents (see Section 4.4.7).

Another issue is the computational cost of the protocols for coalition formation. The nature of the problem is combinatorial: the size of the input representing a characteristic function in $\mathfrak{I}$ is at most $O(2^n)$ but when it is in $\mathfrak{D}$, the input is even larger as it is $\Omega(n^{\frac{n}{2}})$. For large number of agents, it is not feasible to compute payoff distribution that satisfies a stability criteria like the Shapley value or a Kernel-stable payoff distribution. By restricting the size of the coalitions, Kernel oriented coalition formation can be computed in polynomial time [48]. The use of bilateral Shapley value is also polynomial.

4.4.7 Long Term Vs Short Term

In general, a coalition is a short-lived entity that is *"formed with a purpose in mind and dissolve when that need no longer exists, the coalition ceases to suit its designed purpose, or critical mass is lost as agents depart"* [41]. It can be beneficial to consider the formation of long term coalitions, or the process of repeated coalition formation involving the same agents. The work by Vassileva and Breban explicitly study long term coalitions [99], and in particular the importance of trust in this content. Brooks and Durphee refer to a long term coalition as a congregation [16]. The purpose of a congregation is to reduce the number of candidates for a successful interaction: instead of searching the entire population, agents will only search in the congregation. The goal of a congregation is to gather agents, with similar or complementary expertise to perform well in an environment in the long run, which is not very different from a coalition. The only difference is that group rationality is not expected in a congregation. The notion of congregation is similar to the notion of clans [38]: agents gather not for a specific purpose, but for a long-term commitment. The notion of trust is paramount in the clans, and sharing information is seen as another way to improve performance.

4.4.8 Overlapping Coalitions

It is typically assumed that an agent belongs to a single coalition, however, there are some applications where agents can be members of multiple coalitions. As explained in the task allocation domain (see Section 4.1), the expertise of an agent may be required by different coalitions at the same time, and the agent can have enough resources to be part of two or more coalitions. In a traditional setting, the use of the same agent i by two coalitions $\mathcal{C}_1$ and $\mathcal{C}_2$ would require a merge of the two coalitions. This larger coalition U is potentially harder to manage, and a priori, there would not be much interaction between the agents in $\mathcal{C}_1$ and C_2, except for agent i. Another application that requires the use of overlapping coalition is tracking targets using a sensor networks [100]. In this work, a coalition is defined for a target, and as agents can track multiple targets at the same time, they can

The traditional stability concepts do not consider this issue. One possibility is for the agent to be considered as two different agents, but this representation is not satisfactory as it does not capture the real power of this agent. Shehory and Kraus propose a setting with overlapping coalition [91]: Each agent has a capacity, and performing a task may use only a fraction of the agent's capacity. Each time an agent commits to a task, the possible coalitions that can perform a given task can change. A mapping to a set covering problem allows to find the coalition. However, the study of the stability is not considered. Another approach is the use of fuzzy coalition [11]: agents can be member of a coalition with a certain degree that represents the risk associated with being in that coalition. Other work considers that the agents have different degree of membership, and their payoff depends on this degree [6, 61, 67]. The protocols in [55] also allows overlapping coalitions.

More recently, Chalkiadakis et al. [20] have studied the notion of the Core in overlapping coalition formation. In their model, each agent has one resource and the agent contributes a fraction of that resource to each coalition it participates in. The valuation function v is then $[0,1]^n \to \mathbb{R}$. A coalition structure is no longer a partition of the agents: a coalition structure S is a finite list of vectors, one for each 'partial' coalition, i.e., $S = (r^1, \dots, r^k)$. The size of S is the number of coalition, i.e. k. The support of $r^{\mathcal{C}} \in S$ (i.e. the set of indices $i \in N$ such that $r_i^{\mathcal{C}} \neq 0$) is the set of agents forming coalition $\mathcal{C}$. For all $i \in N$ and all coalition $\mathcal{C} \in S$, $r_i^{\mathcal{C}} \in [0,1]^n$ represents the fraction of resource that agent i contributes to coalition $\mathcal{C}$, and hence, $\sum_{\mathcal{C}\in S} r_i^{\mathcal{C}} \leq 1$ (i.e. agent i cannot contributes more than 100% of its resource). A payoff distribution for a CS S of size k is defined by a k-tuple $x = (x^1, \dots, x^k)$ where $x^{\mathcal{C}}$ is the payoff distribution that the agents obtain for coalition $\mathcal{C}$. If an agent is not in the coalition, it must not receive any payoff for this coalition, hence $(r_i^{\mathcal{C}} = 0) \Rightarrow (x_i^{\mathcal{C}} = 0)$. The total payoff of agent i is the sum of its payoff over all coalitions $p_i(CS, x) = \sum_{\mathcal{C}=1}^{k} x_i^{\mathcal{C}}$. The efficiency criterion becomes $\forall r^{\mathcal{C}} \in S, \sum_{i\in N} x_i^{\mathcal{C}} = v(r^{\mathcal{C}})$. An imputation is an efficient payoff distribution that is also individually rational. We denote by $I(S)$ the set of all imputations for the coalition structure S. We are now ready to define the overlapping core: a pair (S, x) is in the *overlapping core* when for any set of agents $\mathcal{C} \subseteq N$, any coalition structure S', any imputation $y \in I(S')$ we have $\exists i \in N, p_i(S', y) \leq p_i(S, x)$. The work in [20] provides characterization of the core under some (mild) conditions for the utility function)

5 Some Elements of Solutions

The applications presented in Section 4 showcase a variety of scenarios and inspired different solutions. In case of dynamic environments, unless the stability criteria can be computed expediently, using game theoretic stability concept may not be feasible. The game theory literature has only few approaches that address dynamic coalition formation. We review two such examples and then present some solutions that may address the issue of dynamic coalition formation. The first solution uses trust to reduce the number of agents considered for coalition formation. Another solution may be to use learning to quickly select the potential members or the negotiation protocol to be used.

5.1 Dynamic Coalition Formation in Game Theory

Some work in game theory focuses in the dynamic formation of coalitions. From the multiagent perspective, these research provide some feasibility results, but may not be practical.

Konishi and Ray propose a model for coalition formation [49]. The model defines a set of finite states that can be interpreted as the description of a CS as well as a vector of payoff to each player. For each state x and a coalition $\mathcal{S}$, there is a set of states that are achievable from x by $\mathcal{S}$. Each player has a payoff function that provides the utility for each state and a discount factor δ. The goal of an agent is to maximize the expected discounted sum of payoffs received over an infinite sequence of states. The process of coalition formation is modeled by a transition probability $p(\mathcal{S}_0, \mathcal{S}_1)$ for transition from state $\mathcal{S}_0$ to state $\mathcal{S}_1$. The transitions are induced by coalitions that will benefit from such move: a transition is allowed only if all members of a coalition $\mathcal{S}$ agree on the transition as they cannot find any strictly better alternative state. If there exists a strictly better state for all agents in a coalition, that move must be made. The main result of the paper is that an equilibrium process always exists. Unfortunately, the theorem is not constructive and does not prescribe how to compute the equilibrium process.The uncertainty in the process arise from multiple sources: for a given state, multiple coalitions can benefit by a move, and randomization can occur to choose the coalition that will perform the move. Moreover, there may be multiple equally beneficial states, and a coalition may need to randomize over them. The existence of cycles in the process is possible. In the restricted case, where the process of coalition formation is deterministic (for all states x, y, $p(x, y) \in \{0, 1\}$), it is possible to derive the following results: (1) if there exists a state x^* in the core, there exists a discount factor and a deterministic equilibrium process with x^* as its unique limit and (2) if the limit is not unique, and there are no cycle, the absorbing states may no longer be in the core.

A different model of dynamic coalition formation is presented in [31]. Unlike the previous model, the payoff of each agent is a CS is not defined: the characteristic function provides a worth for the coalition, and each agent claims a share of this worth. If the claims can be met, each agents gets its. If it is not possible to meet the claims, each agent gets the worth it would get if it were to form a singleton coalition. More precisely, for a coalition $\mathcal{C}$, given the demand d_i of agent $i \in \mathcal{C}$, agent i receive a payoff

$$x_i = \begin{cases} d_i & \text{if } \sum_{i \in \mathcal{C}} d_i \leq v(\mathcal{C}) \\ v(i) & \text{otherwise} \end{cases} .$$

The coalition formation process goes as follows: at each time step, an agent is picked using a given distribution random draw from a Bernoulli trial, and is given the opportunity to take an action. The actions available to this agent is either to stay in the current coalition, form a singleton, or to change coalition to an existing one. The agent computes its new demand: assuming other agents' demands remain fixed, the agent claims the maximum surplus it can get by changing coalition. Hence, at each time step, an agent tries to myopically maximize its payoff. The authors show that this process will be trapped in an absorbing set of states, which may not be composed of core allocations. To alleviate this issue, agents are allowed to 'experiment': an agent can take a suboptimal action (an action that does not maximize its payoff), and can trigger a chain of actions yielding a better allocation. When an agent sees a dominated coalition, it can try to move from it. Using experimentation, the process converges to a core allocation in the limit, i.e., there is a positive probability to reach an absorbing state that is in the core, and once reached, the agent will remain in this state. The authors also claim that this process can converge to a core allocation even when the characteristic function is in $\mathfrak{D}$.

5.2 Trust

The notion of trust can be an important metric to determine whom to interact with. This is particularly important when the a coalition is expected to live for a long term. In [10], an agent computes a probability of success of a coalition, based on a notion of trust which can be used to eliminate some agents from future consideration. This probability is used to estimate the value of different coalitions and help the agent in deciding which coalition to join or form. In [99], the decision to leave or join a coalition is function of the trust put in other agents. In this paper, the concept of trust is defined as a belief that agents will have successful interaction in the future, hence, trust is used to consider a subset of the entire population of agents for the formation of future coalitions. Trust is used to compute coalitions, but they do not compute a payoff distribution. Another work that emphases trust is [38] which introduces the concept of clans. A clan is formed by agents that trust each other with long-term commitments. Given the trust and an estimate of local gain, agents can accept to join a clan. The idea behind this work is that agents that trust each other will be collaborative. Moreover, when an agent needs to form a coalition of agents, it will only search partners in the clan, which reduces the search space. Trust can therefore be very effective for scaling up in large society of agents.

5.3 Learning

When the agents have to repeatedly form coalitions in the presence of the same set of agents, learning can be used to improve performance of the coalition formation process both in terms of speed of the process and in terms of better valuation.

A basic model of iteratively playing many coalitional games is presented in [62]: at each time step, a task is offered to agents that are already organized into coalitions. The task is awarded to the best coalition. The model is made richer in [63] where the agents can estimate the value of a coalition and have a richer set of actions: as the agents can fire members from a coalition, join a different coalition, or leave a coalition to replace some agents in a different coalition. However in both work, the agents are not learning, they have a set of static strategies. Empirical experiments compare the results over populations using either the same strategy or a mix of strategies.

Chalkiadakis and Boutilier also consider a repeated coalition formation problem [18, 19]. The setting is a task allocation problem where agents know their own types (i.e. skill to perform some type of tasks), but do not know the ones of other agents in the population. Each time a coalition is formed, the agents will receive a value for that coalition. From the observation of this value, the agents can update a belief about the type of other agents. When an agent is reasoning about which coalition to form, it uses his beliefs to estimate the value of the coalition. This problem can be formulated using a POMPD (Paritally observable Markov Decision Process) where the agents are maximizing the long-term value of their decision over the repetition of the coalition formation process. Solving a POMPD is a difficult task, and the POMPD for the coalition formation problem grows exponentially with the number of agents. In [18], a myopic approach is proposed. More recently, Chalkiadakis and Boutilier propose additional algorithms to solve that POMPD, and empirically compare the solutions [19].

6 Conclusion

As shown in this survey paper, many scenarios of coalition formations have been investigated by the multiagent community. Given a particular problem, it is difficult to choose a protocol among the ones we presented: no protocol has been proven to be clearly better than another one. Each protocol has its pros and cons: the protocols can be based upon different game theoretic stability criteria, and may or may not handle each issue we presented. We categorized research on coalition formation by issues so as to facilitate the choice of a protocol given the description of a problem. Future protocols should try to handle as many issues as possible.

In addition, some issues have not been successfully treated. For example, in a dynamic environment, most of the current protocol would require a restart each time an agent leave or enter the environment. Another important issue is the scale up properties of the current protocol. By nature, the formation of coalitions is a computationally expensive problem as the number of coalitions in a population of n agents is 2^n. Although some protocols are polynomial [11, 48, 59, 92], they may still require too much time to be useful in real-world application. In addition, the computation of game theoretic stability concept may need to consider an exponential number of coalitions to ensure stability, which may not be feasible. For example, the Kernel is used by many protocols. When the size of the coalition is bounded, it is possible to compute a Kernel stable distribution in polynomial time. When this assumption cannot be met, it may not be feasible to use the Kernel and other approximations are needed.

In many other multiagent applications, agents are only self-interested, and in particular, they may not be group rational. In that case, even if the payoff distribution guarantees to be Kernel stable, agents may still have an incentive to change coalition. Nash equilibrium can be used as a stability criteria in such cases. Hence, a promising line of research is to develop protocols for forming coalition of self interested agents.

Finally, most of the research we surveyed makes the classical assumption that the value of a coalition does not depend on non-members. As we argued, in some real-world situations, often the value of a coalition depends on the behavior of other agents in the environment. We believe that more attention should be given to developing effective coalitions and CSs in these scenarios.

References

[1] S. Abdallah and V. Lesser. Organization-based cooperative coalition formation. In *IAT 04*, 2004.

[2] S. Aknine and P. Caillou. Agreements without disagreements. In *Proceedings of the 16th Eureopean Conference on Artificial Intelligence, ECAI'2004*, pages 3–7, August 2004.

[3] S. Aknine, S. Pinson, and M. F. Shakun. A multi-agent coalition formation method based on preference models. *Group Decision and Negotiation*, 13(6):513–538, November 2004.

[4] S. Aknine and O. Shehory. Coalition Formation: Concessions, Task Relationships and Complexity Reduction. *ArXiv Computer Science e-prints*, Feb. 2005.

[5] F. Asselin and B. Chaib-draa. Performance of software agents in non-transferable payoff group buying. *Journal of Experimental and Theoretical Artificial Intelligence*, 18(1):1–32, mar 2006.

[6] J.-P. Aubin. *Mathematical Methods of Game and Economic Theory*. North-Holland, 1979.

[7] R. J. Aumann and J. H. Drèze. Cooperative games with coalition structures. *International Journal of Game Theory*, 3(4):217–237, 1974.

[8] M.-V. Belmonte, R. Conejo, J.-L. P. de-la Cruz, and F. T. Ruiz. A stable and feasible payoff division for coalition formation in a class of task oriented domains. In *ATAL '01: Revised Papers from the 8th International Workshop on Intelligent Agents VIII*, pages 324–334, London, UK, 2002. Springer-Verlag.

[9] M.-V. Belmonte, R. Conejo, J.-L. P. de-la Cruz, and F. T. Ruiz. A robust deception-free coalition formation model. In *Proceedings of the 2004 ACM symposium on Applied computing (SAC '04)*, pages 469–473, New York, NY, USA, 2004. ACM Press.

[10] B. Blankenburg, R. K. Dash, S. D. Ramchurn, M. Klusch, and N. R. Jennings. Trusted kernel-based coalition formation. In *Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems*, pages 989–996, New York, NY, USA, 2005. ACM Press.

[11] B. Blankenburg, M. He, M. Klusch, and N. R. Jennings. Risk-bounded formation of fuzzy coalitions among service agents. In *Proceedings of 10th International Workshop on Cooperative Information Agents*, 2006.

[12] B. Blankenburg and M. Klusch. On safe kernel stable coalition forming among agents. In *Proceedings of the third International Joint Conference on Autonomous Agents and Multiagent Systems*, 2004.

[13] B. Blankenburg, M. Klusch, and O. Shehory. Fuzzy kernel-stable coalitions between rational agents. In *Proceedings of the second international joint conference on Autonomous agents and multiagent systems (AAMAS-03)*. ACM Press, 2003.

[14] A. Bogomolnaia and M. O. Jackson. The stability of hedonic coalition structures. *Games and Economic Behavior*, 38(2):201–230, 2002.

[15] S. Breban and J. Vassileva. A coalition formation mechanism based on inter-agent trust relationships. In *AAMAS '02: Proceedings of the first international joint conference on Autonomous agents and multiagent systems*, pages 306–307, New York, NY, USA, 2002. ACM Press.

[16] C. H. Brooks and E. H. Durfee. Congregation formation in multiagent systems. *Autonomous Agents and Multi-Agent Systems*, 7(1-2):145–170, 2003.

[17] P. Caillou, S. Aknine, and S. Pinson. A multi-agent method for forming and dynamic restructuring of pareto optimal coalitions. In *AAMAS '02: Proceedings of the first international joint conference on Autonomous agents and multiagent systems*, pages 1074–1081, New York, NY, USA, 2002. ACM Press.

[18] G. Chalkiadakis and C. Boutilier. Bayesian reinforcement learning for coalition formation under uncertainty. In *Proceedings of the third International Joint Conference on Autonomous Agents and Multiagent Systems(AAMAS'04)*, 2004.

[19] G. Chalkiadakis and C. Boutilier. Sequential decision making in repeated coalition formation under uncertainty. In *Proc. of the 7th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS-08)*, May 2008.

[20] G. Chalkiadakis, E. Elkind, E. Markakis, and N. R. Jennings. Overlapping coalition formation. In *Proceeedings of the 4th International Workshop on Internet and Network Economics (WINE2008)*, pages 307–321, 2008.

[21] J. S. Coleman. The benefits of coalition. *Public Choice*, 8:45–61, 1970.

[22] V. Conitzer and T. Sandholm. Complexity of determining nonemptiness of the core. In *Proceedings of the International Joint Conference on Artificial Intell igence (IJCAI)*, 2003.

[23] V. Conitzer and T. Sandholm. Computing shapley values, manipulating value division schemes, and checking core membership in multi-issue domains. In *Proceedings of the 19th National Conference on Artificial Intelligence (AAAI-04)*, pages 219–225, 2004.

[24] J. Contreras, M. Klusch, O. Shehory, and F. Wu. Coalition formation in a power transmission planning environment. In *Proceeding of the Second International Conference on Practical Applications of Multi-Agent Systems, PAAM97*, April 1997.

[25] J. Contreras, M. Klusch, and J. Yen. Multi-agent coalition formation in power transmission planning: a bilateral shapley value approach. In *Proceedings of the Fourth International Conference on Artificial Intelligence Planning Systems*, 1998.

[26] D. Cornforth, M. Kirley, and T. Bossomaier. Agent heterogeneity and coalition formation: Investigating market-based cooperative problem solving. In *Proceedings of the third International Joint Conference on Autonomous Agents and Multiagent Systems(AAMAS'04)*, 2004.

[27] V. D. Dang and N. R. Jennings. Generating coalition structures with finite bound from the optimal guarantees. In *Proceedings of the third International Joint Conference on Autonomous Agents and Multiagent Systems(AAMAS'04)*, 2004.

[28] M. Davis and M. Maschler. The kernel of a cooperative game. *Naval Research Logistics Quarterly*, 12, 1965.

[29] X. Deng, Q. Fang, and X. Sun. Finding nucleolus of flow game. In *SODA '06: Proceedings of the seventeenth annual ACM-SIAM symposium on Discrete algorithm*, pages 124–131, New York, NY, USA, 2006. ACM Press.

[30] X. Deng and C. H. Papadimitriou. On the complexity of cooperative solution concetps. *Mathematical Operation Research*, 19(2):257–266, 1994.

[31] T. Dieckmann and U. Schwalbe. Dynamic coalition formation and the core. *Journal of Economic Behavior & Organization*, 49(3):363–380, November 2002.

[32] J. H. Drèze and J. Greenberg. Hedonic coalitions: Optimality and stability. *Econometrica*, 48(4):987–1003, 1980.

[33] P. S. Dutta and S. Sen. Emergence of stable coalitions via task exchanges. In *AAMAS '02: Proceedings of the first international joint conference on Autonomous agents and multiagent systems*, pages 312–313, New York, NY, USA, 2002. ACM Press.

[34] E. Elkind, G. Chalkiadakis, and N. R. Jennings. Coalition structures in weighted voting games. In *Proceedings of the 18th European Conference on Artificial Intelligence (ECAI-08)*, pages 393–397, 2008.

[35] E. Elkind, L. A. Goldberg, P. Goldberg, and M. Wooldridge. Computational complexity of weighted threshold games. In *Proceedings of the Twenty-Second AAAI Conference on Artificial Intelligence (AAAI-07)*, pages 718–723, 2007.

[36] E. Elkind and M. Wooldridge. Hedonic coalition nets. In *Proc. of the 8th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS-09)*, pages 417–424, May 2009.

[37] D. B. Gillies. *Some theorems on n-person games*. PhD thesis, Department of Mathematics, Princeton University, Princeton, N.J., 1953.

[38] N. Griffiths and M. Luck. Coalition formation through motivation and trust. In *Proceedings of the second international joint conference on Autonomous agents and multiagent systems (AAMAS03)*, New York, NY, USA, 2003. ACM Press.

[39] G. Hamiache. A value for games with coalition structures. *Social Choice and Welfare*, 26(1), January 2006.

[40] S. Hart and M. Kurz. Endogenous formation of coalitions. *Econometrica*, 51(4), July 1983.

[41] B. Horling and V. Lesser. A survey of multi-agent organizational paradigms. *The Knowledge Engineering Review*, 19:281–316, 2004.

[42] B. Horling, R. Vincent, R. Mailler, J. Shen, R. Becker, K. Rawlins, and V. Lesser. Distributed sensor network for real time tracking. In *Proceedings of the fifth international conference on Autonomous agents(Agents'01)*, pages 417–424, New York, NY, USA, 2001. ACM Press.

[43] J. P. Kahan and A. Rapoport. *Theories of Coalition Formation*. Lawrence Erlbaum Associates, Publishers, 1984.

[44] S. P. Ketchpel. The formation of coalitions among self-interested agents. In *Proceedings of the Eleventh National Conference on Artificial Intelligence*, pages 414–419, August 1994.

[45] S. P. Ketchpel. Forming coalitions in the face of uncertain rewards. In *Proceedings of the Eleventh National Conference on Artificial Intelligence*, pages 414–419, August 1994.

[46] M. Klusch and A. Gerber. Issues of dynamic coalition formation among rational agents. In *Proceedings of the Second International Conference on Knowledge Systems for Coalition Operations*, pages 91–102, 2002.

[47] M. Klusch and O. Shehory. Coalition formation among rational information agents. In R. van Hoe, editor, *Seventh European Workshop on Modelling Autonomous Agents in a Multi-Agent World*, Eindhoven, The Netherlands, 1996.

[48] M. Klusch and O. Shehory. A polynomial kernel-oriented coalition algorithm for rational information agents. In *Proceedings of the Second International Conference on Multi-Agent Systems*, pages 157 – 164. AAAI Press, December 1996.

[49] H. Konishi and D. Ray. Coalition formation as a dynamic process. *Journal of Economic Theory*, 110(1):1–41, May 2003.

[50] S. Kraus, O. Shehory, and G. Taase. Coalition formation with uncertain heterogeneous information. In *Proceedings of the second international joint conference on Autonomous agents and multiagent systems*, pages 1–8. ACM Press, 2003.

[51] S. Kraus, O. Shehory, and G. Taase. The advantages of compromising in coalition formation with incomplete information. In *Proceedings of the third International Joint Conference on Autonomous Agents and Multiagent Systems(AAMAS'04)*, 2004.

[52] J. Kuipers, U. Faigle, and W. Kern. On the computation of the nucleolus of a cooperative game. *International Journal of Game Theory*, 30(1):79–98, 2001.

[53] K. S. Larson and T. W. Sandholm. Anytime coalition structure generation: an average case study. In *Proceedings of the Third Annual Conference on Autonomous Agents*, pages 40–47. ACM Press, 1999.

[54] K. S. Larson and T. W. Sandholm. Anytime coalition structure generation: an average case study. *Journal of Experimental & Theoretical Artificial Intelligence*, 12(1):23–42, 2000.

[55] H. C. Lau and L. Zhang. Task allocation via multi-agent coalition formation: Taxonomy, algorithms and complexity. In *15th IEEE International Conference on Tools with Artificial Intelligence (ICTAI 2003)*, pages 346–350, November 2003.

[56] S. Leong and Y. Shoham. Marginal contribution nets: A compact representation scheme for coalitional games. In *Proceedings of EC'05*, 2005.

[57] K. Lerman and O. Shehory. Coalition formation for large-scale electronic markets. In *Proceedings of the Fourth International Conference on MultiAgent Systems (ICMAS-2000)*. IEEE Computer Society, July 2000.

[58] C. Li, U. Rajan, S. Chawla, and K. Sycara. Mechanisms for coalition formation and cost sharing in an electronic marketplace. In *ICEC '03: Proceedings of the 5th international conference on Electronic commerce*, pages 68–77, New York, NY, USA, 2003. ACM Press.

[59] C. Li and K. Sycara. Algorithm for combinatorial coalition formation and payoff division in an electronic marketplace. In *Proceedings of the first international joint conference on Autonomous agents and multiagent systems(AAMAS '02)*, pages 120–127, New York, NY, USA, 2002. ACM Press.

[60] E. Manisterski, E. David, S. Kraus, and N. R. Jennings. Forming efficient agent groups for completing complex tasks. In *Proceedings of the Fifth International Joint Conference on Autonomous Agents and Multiagent Systems AAMAS-06*, May 2006.

[61] M. Mares. Fuzzy cooperative games: cooperation with vague expectations. *Studies in fuzziness and soft computing*, 72, 2001.

[62] C. Mérida-Campos and S. Willmott. Modelling coalition formation over time for iterative coalition games. In *Proceedings of the third International Joint Conference on Autonomous Agents and Multiagent Systems(AAMAS'04)*, 2004.

[63] C. Mérida-Campos and S. Willmott. The effect of heterogeneity on coalition formation in iterated request for proposal scenarios. In *Proceedings of the Forth European Workshop on Multi-Agent Systems (EUMAS 06)*, 2006.

[64] T. Michalak, A. Dowell, P. McBurney, and M. Wooldridge. Optimal coalition structure generation in partition function games. In *Proceeding of the 2008 conference on ECAI 2008*, pages 388–392, Amsterdam, The Netherlands, The Netherlands, 2008. IOS Press.

[65] T. Michalak, T. Rahwan, J. Sroka, A. Dowell, M. Wooldridge, P. McBurney, and N. R. Jennings. On representing coalitional games with externalities. In *Proceedings of the 10th ACM conference on Electronic Commerce 09 (EC'09)*, 2009.

[66] R. B. Myerson. Graphs and cooperation in games. *Mathematics of Operations Research*, 2:225–229, 1977.

[67] I. Nishizaki and M. S. Masatoshi. Fuzzy and multiobjective games for conflict resolution. *Studies in Fuzziness and Soft Computing*, 64, 2001.

[68] N. Ohta, V. Conitzer, Y. Satoh, A. Iwasaki, and M. Yokoo. Anonimity-proof shapley value: Extending shapley value for coalitional games in open environments. In *Proc. of the 8th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS-09)*, May 2009.

[69] M. J. Osborne and A. Rubinstein. *A Course in Game Theory*. The MIT Press, 1994.

[70] G. Owen. Values of games with a priori unions. In M. O. Hein R, editor, *Essays in mathematical economics and game theory*. Springer, New York, 1977.

[71] M. Pěchouček, V. Mařík, and J. Bárta. A knowledge-based approach to coalition formation. *IEEE Intelligent Systems*, 17(3):17–25, 2002.

[72] M. Pěchouček, V. Mařík, and O. čStépánková. Coalition formation in manufacturing multi-agent systems. In *Proceedings of the 11th International Workshop on Database and Expert Systems Applications (DEXA'00)*, 2000.

[73] E. Plaza and S. O. nón. Learning collaboration strategies for committees of learning agents. *Journal of Autonomous Agents and Multi-Agent Systems*, 13(3):429–461, 2006.

[74] A. S. Y. Poon, C. S. K. Yeung, and F. F. Wu. Game theoretical multi-agent modeling of coalition formation for multilateral trades. *IEEE Transactions on Power Systems*, 4(3), August 1999.

[75] A. D. Procaccia and J. S. Rosenschein. The communication complexity of coalition formation among autonomous agents. In *Proceedings of the Fifth International Joint Conference on Autonomous Agents and Multiagent Systems AAMAS-06*, May 2006.

[76] T. Rahwan and N. R. Jennings. An algorithm for distributing coalitional value calculations among cooperating agents. *Artificial Intelligence*, 171(8-9):535–567, 2007.

[77] T. Rahwan and N. R. Jennings. Coalition structure generation: Dynamic programming meets anytime optimization. In *Proceedings of the 23rd conference on artificial intelligence (AAAI-08)*, pages 156–161, 2008.

[78] T. Rahwan and N. R. Jennings. An improved dynamic programming algorithm for coalition structure generation. In *Proceedings of the 7th international conference on Autonomous Agents and Multi-Agent Systems (AAMAS-08)*, 2008.

[79] T. Rahwan, T. Michalak, N. R. Jennings, M. Wooldridge, and P. McBurney. Coalition structure generation in multi-agent systems with positive and negative externalities. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence (IJCAI-09)*, 2009.

[80] T. Rahwan, S. D. Ramchurn, V. D. Dang, A. Giovannucci, and N. R. Jennings. Anytime optimal coalition structure generation. In *Proceedings of the Twenty-Second Conference on Artificial Intelligence (AAAI-07)*, pages 1184–1190, 2007.

[81] T. Rahwan, S. D. Ramchurn, V. D. Dang, and N. R. Jennings. Near-optimal anytime coalition structure generation. In *Proceedings of the Twentieth International Joint Conference on Artificial Intelligence (IJCAI'07)*, pages 2365–2371, January 2007.

[82] T. Rahwan, S. D. Ramchurn, N. R. Jennings, and A. Giovannucci. An anytime algorithm for optimal coalition structure generation. *Journal of Artificial Intelligence Research*, 34:521–567, 2009.

[83] D. Ray and R. Vohra. A theory of endogenous coalition structures. *Games and Economic Behavior*, 26:286–336, 1999.

[84] T. Sandholm and V. R. Lesser. Coalitions among computationally bounded agents. *AI Journal*, 94(1–2):99–137, 1997.

[85] T. W. Sandholm, K. S. Larson, M. Andersson, O. Shehory, and F. Tohmé. Coalition structure generation with worst case guarantees. *Artificial Intelligence*, 111(1–2):209–238, 1999.

[86] D. Sarne and S. Kraus. The search for coalition formation in costly environments. In *Cooperative Information Agents VII, 7th International Workshop, CIA 2003, Helsinki, Finland, August 27-29, 2003, Proceedings*, volume 2782 of *Lecture Notes in Computer Science*. Springer Berlin / Heidelberg, August 2003.

[87] D. Sarne and S. Kraus. Cooperative exploration in the electronic marketplace. In *Proceedings of the Twentieth National Conference on Artificial Intelligence*, pages 158–163. AAAI Press AAAI Press / The MIT Press, 2005.

[88] D. Schmeidler. The nucleolus of a characteristic function game. *SIAM Journal of applied mathematics*, 17, 1969.

[89] S. Sen and P. S. Dutta. Searching for optimal coalition structures. In *ICMAS '00: Proceedings of the Fourth International Conference on MultiAgent Systems (ICMAS-2000)*, page 287, Washington, DC, USA, 2000. IEEE Computer Society.

[90] L. Shapley. A value for n-person games. In H. Kuhn and A. Tucker, editors, *Contributions to the Theory of Games*, volume 2. Princeton University Press, Princeton, NJ, 1953.

[91] O. Shehory and S. Kraus. Methods for task allocation via agent coalition formation. *Artificial Intelligence*, 101(1-2):165–200, May 1998.

[92] O. Shehory and S. Kraus. Feasible formation of coalitions among autonomous agents in non-superadditve environments. *Computational Intelligence*, 15:218–251, 1999.

[93] M. Sims, C. V. Goldman, and V. Lesser. Self-organization through bottom-up coalition formation. In *Proceedings of the second international joint conference on Autonomous agents and multiagent systems (AAMAS03)*, pages 867–874, New York, NY, USA, 2003. ACM Press.

[94] L.-K. Soh and C. Tsatsoulis. Satisficing coalition formation among agents. In *AAMAS '02: Proceedings of the first international joint conference on Autonomous agents and multiagent systems*, pages 1062–1063, New York, NY, USA, 2002. ACM Press.

[95] L.-K. Soh, C. Tsatsoulis, and H. Sevay. A satisficing, negotiated, and learning coalition formation architecture. *Distributed Sensor Networks: A Multiagent Perspective*, pages 109–138, 2003.

[96] R. E. Stearns. Convergent transfer schemes for n-person games. *Transactions of the American Mathematical Society*, 134(3):449–459, December 1968.

[97] F. Tohmé and T. Sandholm. Coalition formation processes with belief revision among bounded-rational self-interested agents. *Journal of Logic and Computation*, 9(6):793–815, 1999.

[98] M. Tsvetovat, K. P. Sycara, Y. Chen, and J. Ying. Customer coalitions in electronic markets. In *Agent-Mediated Electronic Commerce III, Current Issues in Agent-Based Electronic Commerce Systems (includes revised papers from AMEC 2000 Workshop)*, pages 121–138, London, UK, 2001. Springer-Verlag.

[99] J. Vassileva, S. Breban, and M. Horsch. Agent reasoning mechanism for making long-term coalitions based on decision making and trust. *Computational Intelligence*, 18(4):583–595, 2002.

[100] A. R. Viet Dung Dang, Rajdeep K. Dash and N. R. Jennings. Overlapping coalition formation for efficient data fusion in multi-sensor networks. In *Proceedings of the Twenty-First Conference on Artificial Intelligence (AAAI-06)*, pages 635–640, 2007.

[101] L. S.-Y. Wu. A dynamic theory for the class of games with nonempty cores. *SIAM Journal on Applied Mathematics*, 32(2):328–338, March 1977.

[102] M. Yokoo, V. Conitzer, T. Sandholm, N. Ohta, and A. Iwasaki. Coalitional games in open anonymous environments. In *Proceedings of the Twentieth National Conference on Artificial Intelligence*, pages 509–515. AAAI Press AAAI Press / The MIT Press, 2005.

[103] M. Yokoo, V. Conitzer, T. Sandholm, N. Ohta, and A. Iwasaki. A compact representation scheme for coalitional games in open anonymous environments. In *Proceedings of the Twenty First National Conference on Artificial Intelligence*, pages –. AAAI Press AAAI Press / The MIT Press, 2006.

[104] H. P. Young. Monotonic solutions of cooperative games. *International Journal of Game Theory*, 14:65–72, 1985.

[105] G. Zlotkin and J. S. Rosenschein. Coalition, cryptography, and stability: mechanisms for coalition formation in task oriented domains. In *AAAI '94: Proceedings of the twelfth national conference on Artificial intelligence (vol. 1)*, pages 432–437, Menlo Park, CA, USA, 1994. American Association for Artificial Intelligence.