Computing Shapley Values, Manipulating Value Division Schemes, and Checking Core Membership in Multi-Issue Domains

Vincent Conitzer and Tuomas Sandholm
{conitzer, sandholm}@cs.cmu.edu
Computer Science Department
Carnegie Mellon University
Pittsburgh, PA 15213

Abstract

Coalition formation is a key problem in automated negotiation among self-interested agents. In order for coalition formation to be successful, a key question that must be answered is how the gains from cooperation are to be distributed. Various solution concepts have been proposed, but the computational questions around these solution concepts have received little attention.

We study a concise representation of characteristic functions which allows for the agents to be concerned with a number of independent *issues* that each coalition of agents can address. For example, there may be a set of *tasks* that the capacity-unconstrained agents could undertake, where accomplishing a task generates a certain amount of value (possibly depending on how well the task is accomplished). Given this representation, we show how to quickly compute the *Shapley value*—a seminal value division scheme that distributes the gains from cooperation fairly in a certain sense. We then show that in (distributed) marginal-contribution based value division schemes, which are known to be vulnerable to manipulation of the order in which the agents are added to the coalition, this manipulation is NP-complete. Thus, computational complexity serves as a barrier to manipulating the joining order. Finally, we show that given a value division, determining whether some subcoalition has an incentive to break away (in which case we say the division is not in the *core*) is NP-complete. So, computational complexity serves to increase the stability of the coalition.

1. Introduction

Coalition formation is a key part of automated negotiation among self-interested agents. A coalition of agents can sometimes accomplish things that the individual agents cannot, or can do things more efficiently. Besides being of interest to the distributed AI / multiagent systems community, coalition formation has electronic commerce applications as well. For example, consider a large number of companies, some subsets of which could form profitable virtual organizations that can respond to larger or more diverse orders than the individual companies can.

In order for coalition formation to be successful, a key question that must be answered is how the gains from cooperation are to be distributed. This question has been studied extensively in cooperative game theory, and some of the resulting solution concepts have already been adopted in the multiagent systems literature (e.g., (Ketchpel 1994; Zlotkin & Rosenschein 1994; Shehory & Kraus 1996; 1998; Conitzer & Sandholm 2003a)). One objective that these solution concepts pursue is that of *fairness*. For instance, the *Shapley value* divides the value fairly in a certain sense. Another objective is that of *stability*. For instance, a value division is in the *core* if no subcoalition of agents has an incentive to break away and form their own coalition.

The computational questions around these solution concepts have received relatively little attention. (As exceptions, constructing solutions in the core has been studied under a concise representation relying on superadditivity (Conitzer & Sandholm 2003a), as well as for a routing game on graphs (Markakis & Saberi 2003), and a facility location game (Goemans & Skutella 2004).) When it comes to coalition formation among software agents (that represent real-world parties), these questions become increasingly explicit. Additionally, there are many potential commercial applications for methods that compute value divisions with certain properties. For instance, one possible application of being able to compute a value division in the core is to determine how much each employee of a company should be paid so that the company does not collapse as a result of a group of employees being bought away by another company.

One important source of computational complexity could be that each potential coalition has some hard optimization problem, making it difficult to ascertain a single coalition's value. For example, when the agents are carrier companies with their own trucks and delivery tasks, they can save costs by forming a coalition (pooling their trucks and tasks), but each potential coalition faces a hard optimization problem: a vehicle routing problem defined by the coalition's trucks and tasks. The effect of such hard optimization problems on coalition formation has been studied by Sandholm and Lesser (Sandholm & Lesser 1997), but the bulk of research on coalition formation (Aumann 1959; Charnes & Kortanek 1966; Shapley 1967; Kahan & Rapoport 1984; van der Linden & Verbeek 1985; Bernheim, Peleg, & Whinston 1987; Chatterjee *et al.* 1993; Ketchpel 1994; Moreno & Wooders 1996; Okada 1996; Ray 1996; Shehory & Kraus 1996; Milgrom & Roberts 1996; Evans 1997; Shehory & Kraus 1998; Conitzer & Sandholm 2003a) does not address this issue.

A second source of computational complexity, more directly related to the coalition formation process itself, is that even if we can compute each coalition's value, we still need a method to choose a value division among the agents that is consistent with the solution concept. Finding such a value division can be a nontrivial problem. How complex is it? There are other, related, important computational questions as well. For instance, how hard is it for an agent to manipulate (to its advantage) which of the consistent value divisions is chosen? Or, can we perhaps use a weaker notion of stability because it is computationally difficult to find a subcoalition that has an incentive to break away? In this paper we address all of these three questions.

We study the questions in a problem representation where the agents are concerned with a number of distinct *issues* that each coalition of agents can address. (That there can be multiple independent issues does not in any way restrict the settings that we can capture; in the worst case the problem is not decomposable, so there will be just one issue.) For example, there may be a set of *tasks* that the agents could undertake, where accomplishing a task generates a certain amount of value (possibly depending on how well the task is accomplished). Here, each coalition of agents would have a different collective skill set, and thereby achieve a different level of success on each task. We assume that each coalition's optimization problem for each individual issue is solved already (or easy to solve), thereby largely assuming away the first source of computational complexity. This paper belongs to the relatively new set of papers that study computational considerations directly related to value division among the agents—the second source of complexity.

The rest of the paper is organized as follows. In Section 2, we formalize our problem representation. In Section 3, we show how to efficiently compute the Shapley value. In Section 4, we show that manipulating marginal-contribution based value division schemes is hard. Finally, in Section 5, we show that it is hard to determine whether a given value division is stable—that is, belongs to the core.

2. Multi-issue characteristic function games

Characteristic function games

Value division in coalition formation is usually studied in *characteristic function games*, where each potential coalition S has a value $v(S)$ that it can obtain. This assumes that utility is transferable (e.g., payments are possible),[1] and that a coalition's value is independent of what nonmembers of the coalition do. In some settings, nonmembers' actions affect the coalition's value, for example, due to usage of shared limited resources. Such general games can be modeled in the characteristic function framework by either optimistically assuming that the nonmembers will do what maximizes the coalition's value, or pessimistically assuming that the nonmembers will do what minimizes the coalition's value. (In either case, the members of the coalition act to maximize the coalition's value.) The optimistic assumption yields stronger stability (in the sense of the core): if a coalition cannot beneficially deviate even if its value is maximized by the nonmembers, then it certainly cannot beneficially deviate.

Definition 1 *Given a set of agents A, a* characteristic function $v : 2^A \rightarrow \mathbb{R}$ *assigns a value to each coalition.*

Typically the function is increasing:

Definition 2 *v is* increasing *if $S_1 \subseteq S_2 \Rightarrow v(S_1) \leq v(S_2)$.*

The function being increasing entails that adding more agents to a coalition never hurts (in the worst case, they can sit on the side and do nothing). All of our results hold both with and without the assumption that v is increasing.

Another common assumption on characteristic functions is that they are *superadditive*:

Definition 3 *v is* superadditive *if for all disjoint coalitions $S_1, S_2 \subseteq A$, $v(S_1) + v(S_2) \leq v(S_1 \cup S_2)$.*

The motivation behind this is that, at worst, the agents can pretend that they are in two separate coalitions even though they are joined into a single one. However, superadditivity does not always hold, for any of several reasons: 1. There can be coordination overhead. A larger coalition may need to expend more effort in coordinating the agents in the coalition. 2. The problem of deciding how a coalition will handle its tasks can be a hard optimization problem, and the cost of solving it often increases superlinearly with the number of agents in the coalition (Sandholm & Lesser 1997). 3. There may be some penalty to collusion, for example, due to antitrust laws. 4. In games where a coalition's value can depend on what nonmembers do, and the characteristic function is derived using the optimistic assumption described above, the argument of pretending to be in two separate coalitions does not go through. This is because it would implicitly require all agents to act in the best interest of S_1, as well as in the best interest of S_2, which may not be possible.

Concise representation of multi-issue games

We are now ready to present our concise representation of characteristic functions, which involves a decomposition over a number of independent *issues*. Each issue has its own characteristic function.

Definition 4 *The vector of characteristic functions $(v_1, v_2, \ldots, v_T)$, with each $v_i : 2^A \rightarrow \mathbb{R}$, is a* decomposition over T issues *of characteristic function $v : 2^A \rightarrow \mathbb{R}$ if for any $S \subseteq A$, $v(S) = \sum_{i=1}^{T} v_i(S)$.*

The following lemmas show that if the functions into which the characteristic function decomposes are increasing or superadditive, then so is the characteristic function.

Lemma 1 *If $v = \sum_{i=1}^{T} v_i$ is a decomposition of v, and each v_i is increasing, then v is increasing.*

[1] In the general case where utility transfers are not necessarily possible, each coalition has a set of *utility possibility vectors*, each of which contains a utility for each agent. If utility is transferable, then the set of utility possibility vectors for a coalition is the set of all utility vectors for that coalition with utilities summing to at most the value of the coalition. We will only deal with the transferable utility case in this paper.

Proof: If $S_1 \subseteq S_2$, then $v(S_1) = \sum_{i=1}^{T} v_i(S_1) \leq \sum_{i=1}^{T} v_i(S_2) = v(S_2)$. ■

Lemma 2 *If $v = \sum_{i=1}^{T} v_i$ is a decomposition of v, and each v_i is superadditive, then v is superadditive.*

Proof: For disjoint $S_1, S_2 \subseteq A$, we have $v(S_1 \cup S_2) = \sum_{i=1}^{T} v_i(S_1 \cup S_2) \geq \sum_{i=1}^{T} v_i(S_1) + v_i(S_2) = \sum_{i=1}^{T} v_i(S_1) + \sum_{i=1}^{T} v_i(S_2) = v(S_1) + v(S_2)$. ■

The decomposition can lead to a more concise representation if the individual v_i are concisely representable. In this paper, we will study the case where each v_i only concerns a subset of the agents that are relevant to issue i. For instance, in a setting where the issues correspond to tasks, some of the agents may not have any skills relevant to a given task.

Definition 5 *We say that v_i* concerns only $C_i \subseteq A$ *if $v_i(S_1) = v_i(S_2)$ whenever $C_i \cap S_1 = C_i \cap S_2$. In this case, we only need to define v_i over 2^{C_i}.*

Our representation requires the specification of only $\sum_{i=1}^{T} 2^{|C_i|}$ values, exponentially fewer than the $2^{|A|}$ we need to specify in general—presuming the $|C_i|$ are small. We will conceive of the $|C_i|$ as being small (for example, a constant) throughout this paper.

3. Computing the Shapley value

We will now review a well-known value division scheme known as the *Shapley value* (Shapley 1953). The Shapley value aims to distribute the gains from cooperation in a fair manner. It has many equivalent characterizations; we will review one that gives a formula in closed form for it.

First consider a different value division scheme, which we will call the *marginal-contribution scheme*. It imposes an order π on the agents in A, and adds in the agents one by one in this order. An agent's payoff is its marginal contribution to the value of the coalition. This simple value division scheme has its advantages, not the least of which is its simplicity, and we will return to it later. A difficulty that it presents is that the value that an agent receives depends on the order, π, in which the agents join the coalition. The Shapley value resolves this by averaging each agent's payoff over *all* possible orderings.

Definition 6 *Given an ordering π of A, for any agent a, let $S(\pi, a)$ be the set of agents in A that appear before a in ordering π. Then the* Shapley value *for agent a is defined as* $Sh(A, a) = \frac{1}{|A|!} \sum_{\pi} (v(S(\pi, a) \cup \{a\}) - v(S(\pi, a)))$.

To operationalize a value division scheme, the scheme should be associated with an algorithm for finding a value division consistent with the scheme. In the rest of this section we derive a fast way of determining the Shapley value.

We first show that to compute the Shapley value in our representation, we can simply compute the Shapley value for each term v_i, and sum these. That is, if the characteristic function decomposes, then so does the Shapley value.

Lemma 3 *If $v = \sum_{i=1}^{T} v_i$ is a decomposition of v, then for any agent a we have $Sh(A, a) = \sum_{i=1}^{T} Sh_{v_i}(A, a)$, where Sh_{v_i} is the Shapley value computed with respect to characteristic function v_i.*

Proof: $Sh(A,a) = \frac{1}{|A|!} \sum_{\pi} (v(S(\pi,a) \cup \{a\}) - v(S(\pi,a))) = \frac{1}{|A|!} \sum_{\pi} (\sum_{i=1}^{T} v_i(S(\pi,a) \cup \{a\}) - \sum_{i=1}^{T} v_i(S(\pi,a))) = \frac{1}{|A|!} \sum_{\pi} \sum_{i=1}^{T} (v_i(S(\pi,a) \cup \{a\}) - v_i(S(\pi,a))) = \sum_{i=1}^{T} \frac{1}{|A|!} \sum_{\pi} (v_i(S(\pi,a) \cup \{a\}) - v_i(S(\pi,a))) = \sum_{i=1}^{T} Sh_{v_i}(A,a)$. ■

Next, we show that to compute the Shapley value of a function that only concerns a subset of the agents, we need to average over the orderings of only those agents.

Lemma 4 *If v_i only concerns $C_i \subseteq A$, then for any $a \in C_i$, $Sh_{v_i}(A,a) = Sh_{v_i}(C_i,a) = \frac{1}{|C_i|!} \sum_{\pi_{C_i}} (v_i(S(\pi_{C_i},a) \cup \{a\}) - v_i(S(\pi_{C_i},a)))$ (where the π_{C_i} are orderings of the agents in C_i only, and $S(\pi_{C_i},a)$ is the set of agents in C_i appearing before a in ordering π_{C_i}). For any $a \notin C_i$, $Sh_{v_i}(A,a) = 0$.*

Proof: Because $v_i(S) = v_i(S \cup \{a\})$ for any $S \subseteq A$ and $a \notin C_i$, the marginal contribution of an agent outside C_i is always 0, and it follows that its Shapley value is 0—proving the second part of the lemma. For any $S \subseteq A$ and $a \in C_i$, we have $v_i(S \cup \{a\}) - v_i(S) = v_i((S \cap C_i) \cup \{a\}) - v_i(S \cap C_i)$. Using the notation $\pi \Rightarrow \pi_{C_i}$ to indicate that π and π_{C_i} agree on the order of the elements in C_i, if $\pi \Rightarrow \pi_{C_i}$, it follows that $S(\pi,a) \cap C_i = S(\pi_{C_i},a)$. Combining this with our previous observation, we have $v_i(S(\pi,a) \cup \{a\}) - v_i(S(\pi,a)) = v_i(S(\pi_{C_i},a) \cup \{a\}) - v_i(S(\pi_{C_i},a))$. Then, $Sh_{v_i} = \frac{1}{|A|!} \sum_{\pi} (v_i(S(\pi,a) \cup \{a\}) - v_i(S(\pi,a))) = \frac{1}{|A|!} \sum_{\pi_{C_i}} \sum_{\pi:\pi\Rightarrow\pi_{C_i}} (v_i(S(\pi,a) \cup \{a\}) - v_i(S(\pi,a))) = \frac{1}{|A|!} \sum_{\pi_{C_i}} \sum_{\pi:\pi\Rightarrow\pi_{C_i}} (v_i(S(\pi_{C_i},a) \cup \{a\}) - v_i(S(\pi_{C_i},a))) = \frac{1}{|A|!} \sum_{\pi_{C_i}} \frac{|A|!}{|C_i|!} (v_i(S(\pi_{C_i},a) \cup \{a\}) - v_i(S(\pi_{C_i},a))) = \frac{1}{|C_i|!} \sum_{\pi_{C_i}} (v_i(S(\pi_{C_i},a) \cup \{a\}) - v_i(S(\pi_{C_i},a))) = Sh_{v_i}(C_i,a)$, proving the first part. ■

Finally, we show that we do not really need to sum over all possible orderings, but rather just over all possible subsets, if we add an appropriate weighting factor to each term.

Lemma 5 *We can write $Sh(A,a) = \frac{1}{|A|!} \sum_{S \subseteq A-\{a\}} |S|!(|A| - |S| - 1)!(v(S \cup \{a\}) - v(S))$.*

Similarly, if v_i only concerns $C_i \subseteq A$, then for any $a \in C_i$, we can write $Sh_{v_i}(C_i, a) = \frac{1}{|C_i|!} \sum_{S \subseteq C_i - \{a\}} |S|!(|C_i| - |S| - 1)!(v_i(S \cup \{a\}) - v_i(S))$.

Proof: We have $Sh(A, a) = \frac{1}{|A|!} \sum_{\pi} (v(S(\pi, a) \cup \{a\}) - v(S(\pi, a))) = \frac{1}{|A|!} \sum_{S \subseteq A - \{a\}} \sum_{\pi : S(\pi, a) = S} (v(S \cup \{a\}) - v(S)) = \frac{1}{|A|!} \sum_{S \subseteq A - \{a\}} |S|!(|A| - |S| - 1)!(v(S \cup \{a\}) - v(S))$. The proof for the v_i is exactly the same, because the formula has exactly the same structure. ∎

We can conclude that our representation allows for fast computation of the Shapley value, if the $|C_i|$ are small.

Theorem 1 *Suppose we are given a characteristic function with a decomposition $v = \sum_{i=1}^{T} v_i$, represented as follows. For each i with $1 \leq i \leq T$ we are given $C_i \subseteq A$, so that each v_i concerns only C_i. Each v_i is flatly represented over 2^{C_i}, that is, for each i with $1 \leq i \leq T$, we are given $v_i(S_i)$ explicitly for each $S_i \subseteq C_i$. Then (assuming that table lookups for the $v_i(S_i)$, as well computations of factorials, multiplications and subtractions take constant time), we can compute the Shapley value of v for any given agent in time $O(\sum_{i=1}^{T} 2^{|C_i|})$, or less precisely $O(T 2^{\max_i |C_i|})$. This holds whether or not the characteristic function is increasing, and whether or not it is superadditive.*

Proof: By Lemma 3, we can simply compute the agent's Shapley value for each individual issue, and then sum these together. By Lemmas 4 and 5, to compute the Shapley value of an individual issue, we only need to sum weighted marginal utilities over subsets of the agents that that issue concerns. The computation of each term in the latter summation only takes constant time, by assumptions made in the statement of the theorem. ∎

Thus, the Shapley value can be computed quickly when the $|C_i|$ are small (especially in the case where the $|C_i|$ are bounded by a small constant, as will be the case in the rest of the paper).

4. Manipulating marginal-contribution based value division schemes

We now return to marginal contribution schemes for value division where we do not average over all possible orders (unlike in the Shapley value scheme). In such schemes, we should be concerned that an agent may have some influence over which order is chosen, and will attempt to make the chosen order so that its marginal contribution is maximal when it joins. Choosing the order completely at random has been suggested as a solution to this (with the added bonus that the *expected* value to an agent is its Shapley value). This, however, requires either a trusted source of randomness, or a distributed cryptographic protocol—for instance, each agent could pick a permutation of the agents, submit an encryption of it to all the other agents, and then provide the decryption key once everybody has submitted an encrypted permutation. Then, we can choose the composition of all the permutations as the order with respect to which we compute the marginal contributions. For example, consider a 3-agent example where agent 1 submits the permutation π_1, where $\pi_1(1) = 2$, $\pi_1(2) = 1$, and $\pi_1(3) = 3$, agent 2 submits the permutation π_2, where $\pi_2(1) = 3$, $\pi_2(2) = 1$, and $\pi_2(3) = 2$, and agent 3 submits the permutation π_3, where $\pi_3(1) = 2$, $\pi_3(2) = 3$, and $\pi_3(3) = 1$. The final joining order would then be $\pi_3(\pi_2(\pi_1(1)))$, $\pi_3(\pi_2(\pi_1(2)))$, $\pi_3(\pi_2(\pi_1(3)))$, which is 2, 1, 3. Assuming that the decryption cannot be manipulated, if even one agent picks its order uniformly at random, then the final resulting order will be uniformly random. This is the approach suggested by Zlotkin and Rosenschein (Zlotkin & Rosenschein 1994). However, a problem with this approach remains that it may be possible for an agent to change its decryption key (and thus the plaintext of its submission) on the basis of the plaintexts of the other agents' permutations. This way the agent could again manipulate the joining order to its advantage.

We suggest a different approach to resolving this problem. *Even with perfect control over the (final) order chosen, it may be computationally hard for an agent to determine the order most beneficial to it. We want to use this computational complexity as the barrier to manipulation.*[2] The following problem captures the predicament of a manipulating agent with perfect control over the order chosen.

Definition 7 (MAX-MARGINAL-CONTRIBUTION) *We are given a characteristic function with a decomposition $v = \sum_{i=1}^{T} v_i$, represented as follows. For each i with $1 \leq i \leq T$ we are given $C_i \subseteq A$, so that each v_i concerns only C_i. Each v_i is flatly represented over 2^{C_i}, that is, for each i with $1 \leq i \leq T$, we are given $v_i(S_i)$ explicitly for each $S_i \subseteq C_i$. Additionally, we are given an agent $a \in A$, and a number k. We are asked if there is some $S \subseteq A - \{a\}$ such that $v(S \cup \{a\}) - v(S) \geq k$.*

We show this problem is NP-complete by reducing the MAX2SAT problem to it (Papadimitriou 1995).

Theorem 2 *MAX-MARGINAL-CONTRIBUTION is NP-complete, even when $|C_i| = 3$ for all i, v_i only takes on values in $\{0, 1, 2\}$ for all i, and all v_i (and thus, by Lemma 1, v) are increasing (but not necessarily superadditive).*

Proof: The problem is in NP because for a given $S \subseteq A - \{a\}$, we can easily compute the marginal contribution of a to this set. To show that it is NP-hard, we reduce an arbitrary MAX2SAT instance (given by a set of Boolean variables V and a set of clauses C, each of which contains 2 literals (a literal is a variable or its negation), corresponding to different variables; and a target number r of satisfied clauses) to the following MAX-MARGINAL-CONTRIBUTION instance.

[2] Using computational complexity as the barrier to manipulation has previously been studied in the context of the complexity of manipulating voting protocols (Bartholdi, Tovey, & Trick 1989; 1992; Bartholdi & Orlin 1991; Conitzer & Sandholm 2002; 2003b).

For each variable $v \in V$, there is an agent a_v; additionally, there is the agent a whose marginal contribution we seek to maximize. For every clause $c \in C$, there is an issue t_c (so that $T = |C|$). The set of agents that v_{t_c} concerns is $C_{t_c} = \{a\} \cup \{a_v : v \in c \vee -v \in c\}$ (exactly 3 agents because we are reducing from MAX2SAT). The characteristic function $v_{t_c} : 2^{C_{t_c}} \rightarrow \mathbb{R}$ is defined as follows. Let $P_{t_c} = \{a_v : v \in c\}$ and $N_{t_c} = \{a_v : -v \in c\}$. Then, for a given $S_{t_c} \subseteq C_{t_c}$, $v_{t_c}(S_{t_c}) = 0$ if at least one element of N_{t_c}, and a, are *not* in S_{t_c}; $v_{t_c}(S_{t_c}) = 1$ if at least one element of N_{t_c} is *not* in S_{t_c}, but a is in S_{t_c}; $v_{t_c}(S_{t_c}) = 1$ if $N_{t_c} \subseteq S_{t_c}$, and: either $a \notin S_{t_c}$, or no element of P_{t_c} is in S_{t_c}; $v_{t_c}(S_{t_c}) = 2$ if $N_{t_c} \subseteq S_{t_c}$, $a \in S_{t_c}$, and some element of P_{t_c} is in S_{t_c}. (It is easy to see that each v_{t_c} is increasing.) Finally, let $k = r$, that is, the two instances' target values are the same. We now show the instances are equivalent.

Suppose there is a solution to the MAX2SAT instance, that is, an assignment of truth values to the variables so that at least r clauses are satisfied. Let $V+$ be the variables set to *true*. Then, let $S = \{a_v : v \in V+\}$. Now, if $c \in C$ is satisfied in this assignment, either some variable v whose negation occurs in this clause is set to *false*; or, if this is not the case, some variable v that occurs in this clause (not negated) is set to *true*. In the former case, we have $a_v \in N_{t_c}$ and $a_v \notin S$; so that $v(S) = 0$ and $v(S \cup \{a\}) = 1$. In the latter case, we have $N_{t_c} \subseteq S_{t_c}$ (because the former case did not apply), $a_v \in P_{t_c}$, and $a_v \in S$; so that $v(S) = 1$ and $v(S \cup \{a\}) = 2$. In either case, the marginal contribution of a to this issue is at least 1, so that the total marginal contribution of a is at least $r = k$. Thus, S is a solution to the MAX-MARGINAL-CONTRIBUTION instance.

Now suppose there is a solution to the MAX-MARGINAL-CONTRIBUTION instance, that is, a set S so that $v(S \cup \{a\}) - v(S) \geq k = r$. Then, let our assignment be to set v to *true* if $a_v \in S$, and to *false* otherwise. The marginal contribution of a to an issue (relative to S) is either 0 or 1. If the marginal contribution of a to t_c is 1 (which by the previous has to be the case for at least r issues), then either some $a_v \in N_{t_c}$ is not in S, or we have that $N_{t_c} \subseteq S_{t_c}$ and some element a_v of P_{t_c} is in S. In the former case, the negation of v occurs in c, and we have set v to *false*, so c is satisfied. In the latter case, v occurs in c (not negated), and we have set v to *true*, so c is satisfied. It follows that the number of clauses satisfied by our assignment is at least r. Thus, our assignment solves the MAX2SAT instance. ∎

We observe that the MAX-MARGINAL-CONTRIBUTION problem is not necessarily hard if the characteristic function is known to have special structure. For instance, a characteristic function is *convex* if the marginal contribution of an agent is always increasing in the subset of agents to which the agent is added—that is, an agent always adds at least as much value to a coalition as it does to any subcoalition. In this case, it is easy to see that an agent always wants to be the last in the order.

An interesting aspect of convex games is that in such games the Shapley value and any value division stemming from a marginal-contribution based scheme are always in the core, so the resulting value division is stable (Shapley 1971; Mas-Colell, Whinston, & Green 1995; Osborne & Rubinstein 1994). So, these payoff schemes seem particularly desirable in that setting. It is thus frustrating that, as we discussed above, finding a beneficial manipulation of the joining order in convex games is easy!

The complexity of the MAX-MARGINAL-CONTRIBUTION problem remains open if the characteristic function is superadditive, but not convex.

It should also be observed that, even though the problem of maximizing an agent's marginal contribution is hard in the worst case, there may still exist effective heuristics for finding an order that makes the agent's marginal contribution at least relatively large (even if it is not the largest possible). For instance, if most other agents' skills can substitute for this agent's skills, then it is likely beneficial for the agent to be early in the order. On the other hand, with complementary skills, it is likely beneficial for the agent to be late in the order (convex games are an extreme example of this).

5. Checking core membership

We finally consider the best-known stability concept, the *core* (Gillies 1953; von Neumann & Morgenstein 1947). A value division is in the core if no subcoalition has an incentive to break away.

Definition 8 *A value division $d : A \rightarrow \mathbb{R}$ is* blocked *by coalition S if $v(S) > \sum_{a \in S} d(a)$. We say that d is in the* core *if it is not blocked by any coalition.*

Our next result shows that under the multi-issue representation, even checking whether a given value division is in the core is coNP-complete. We first define the problem.

Definition 9 (CHECK-IF-BLOCKED) *We are given a characteristic function with a decomposition $v = \sum_{i=1}^{T} v_i$, represented as follows. For each i with $1 \leq i \leq T$ we are given $C_i \subseteq A$, so that each v_i concerns only C_i. Each v_i is flatly represented over 2^{C_i}, that is, for each i with $1 \leq i \leq T$, we are given $v_i(S_i)$ explicitly for each $S_i \subseteq C_i$. Additionally, we are given a value division $d : A \rightarrow \mathbb{R}$.[3] We are asked whether d is outside of the core, that is, if there is some blocking coalition S with $v(S) > \sum_{a \in S} d(a)$.*

We show this problem is NP-complete by reducing the VERTEX-COVER problem to it (Papadimitriou 1995).

[3] We intentionally do not constrain d to be a *feasible* value division relative to the given charcteristic function (d is feasible if $\sum_{a \in A} d(a) \leq v(A)$). This omission is justified because it does not introduce any new instances of the computational problem: when d is not a feasible value division, it is always possible to increase the value of the grand coalition of all agents to the point where the value division is feasible, without changing the value of any other coalition. This new "valid" instance has the exact same strategic structure as the original instance. Apart from streamlining the definition, omitting the constraint that d is feasible also makes it easier to think about scenarios where there is an outside benefactor that gives some of the agents some additional value to prevent them from blocking the value division.

Theorem 3 *CHECK-IF-BLOCKED is NP-complete, even when $|C_i| = 3$ for all i, v_i only takes on values in $\{0, 1\}$ for all i, and all the v_i (and thus, by Lemmas 1 and 2, v) are increasing and superadditive.*

Proof: The problem is in NP because given a subset S, we can compute $v(S)$ and $\sum_{a \in S} d(a)$ in polynomial time, and check if the former is larger. To show that it is NP-hard, we reduce an arbitrary VERTEX-COVER instance (given by a graph $G = (V, E)$ ($|V| > 0$, $|E| > 0$) and a maximal number $r > 0$ of vertices to cover all the edges with) to the following CHECK-IF-BLOCKED instance. For every vertex $v \in V$, there is an agent a_v; there is one additional agent a_0. For every edge $e \in E$, there is an issue t_e (so that $T = |E|$). The set of agents that v_{t_e} concerns is $C_{t_e} = \{a_0\} \cup \{a_v : v \in e\}$ (we say $v \in e$ when one of e's endpoints is v). The characteristic function $v_{t_e} : 2^{C_{t_e}} \rightarrow \mathbb{R}$ is defined as follows: for a given $S_{t_e} \subseteq C_{t_e}$, $v_{t_e}(S_{t_e}) = 1$ if $a_0 \in S_{t_e}$ and $\{a_v : v \in e\} \cap S_{t_e}$ is nonempty, and $v_{t_e}(S_{t_e}) = 0$ otherwise. (It is easy to see that each v_{t_e} is increasing; each v_{t_e} is also superadditive, because when splitting a coalition into two disjoint subcoalitions, only one of them can have a_0 in it, and the other hence will have value 0.) Finally, for the value division, we have $d(a_0) = T - \frac{1}{2}$, and for any $v \in V$, $d(a_v) = \frac{1}{2(r+\frac{1}{2})}$. We now show the instances are equivalent.

Suppose there is a solution to the VERTEX-COVER instance, that is, a subset $W \subseteq V$ such that $|W| \leq r$ and for any $e \in E$, $\{v : v \in e\} \cap W$ is nonempty. Then consider the set $S = \{a_0\} \cup \{a_v : v \in W\}$. It is straightforward to check that $v_{t_e}(S) = 1$ for all issues, and thus $v(S) = T$. On the other hand, $\sum_{a \in S} d(a) = T - \frac{1}{2} + |W|\frac{1}{2(r+\frac{1}{2})} \leq T - \frac{1}{2} + r\frac{1}{2(r+\frac{1}{2})} < T - \frac{1}{2} + r\frac{1}{2r} = T = v(S)$. Thus, S is a blocking coalition, and there is a solution to the CHECK-IF-BLOCKED instance.

Now suppose there is a solution to the CHECK-IF-BLOCKED instance, that is, a subset $S \subseteq A$ such that $v(S) > \sum_{a \in S} d(a)$. We first observe that $a_0 \in S$, because otherwise we would have $v(S) = 0$ and S could not be a blocking coalition. Now consider the set $W = \{v : a_v \in S\}$ of the vertices corresponding to agents in the blocking coalition. Then, $\sum_{a \in S} d(a) = T - \frac{1}{2} + |W|\frac{1}{2(r+\frac{1}{2})}$. Combining this with $T \geq v(S) > \sum_{a \in S} d(a)$, and using the fact that $|W|$ is an integer, we can conclude $|W| \leq r$. Additionally, we have $v(S) > d(a_0) = T - \frac{1}{2}$. It follows that $v_{t_e}(S) = 1$ for every issue t_e, and thus, for any edge e, there is an agent $a_v \in S$ (and thus a vertex $v \in W$) with $v \in e$. It follows that W covers all the edges, and is thus a solution to the VERTEX-COVER instance. ∎

This result not only implies that it is difficult computationally to use the core as the solution concept, but also that the core may be an unnecessarily strong solution concept. If a value division is unstable in the sense that some subcoalition is motivated to break away, but nobody can *find* this coalition because it is too difficult computationally, then the value division is still stable in practice. Of course, NP-completeness is a worst-case measure of hardness, so it is still possible that in many instances, finding a subcoalition that could do better by breaking off is easy. Also, the computational hardness is not a significant barrier if the instances are small enough.

6. Conclusions and future research

Coalition formation is a key problem in automated negotiation among self-interested agents. A coalition of agents can sometimes accomplish things that the individual agents cannot, or can do things more efficiently. In order for coalition formation to be successful, a key question that must be answered is how the gains from cooperation are to be distributed. This question has been studied extensively in cooperative game theory, and some of the resulting solution concepts have already been adopted in the multiagent systems literature. However, the computational questions around these solution concepts have received relatively little attention. When it comes to coalition formation among software agents (that represent real-world parties), these questions become increasingly explicit.

We studied a concise representation of characteristic functions which allows for the agents to be concerned with a number of distinct independent *issues* that each coalition of agents can address. For example, there may be a set of *tasks* that the capacity-unconstrained agents could undertake, where accomplishing a task generates a certain amount of value (possibly depending on how well the task is accomplished). Here, each coalition of agents would have a different collective skill set, and thereby achieve a different level of success on each task. We assumed that each coalition's value determination problem (how it would handle the task/issue) for each individual issue is solved already (or easy to solve), and focused on the computational questions related to value division among the agents. To make our representation concise, we also assumed that each individual issue concerns only a small number of agents.

We showed how to quickly compute the *Shapley value*—a seminal value division scheme that distributes the gains from cooperation fairly in a certain sense. We then showed that in (distributed) marginal-contribution based value division schemes, which are known to be vulnerable to manipulation of the order in which the agents are added to the coalition, this manipulation is NP-complete. Thus, computational complexity serves as a barrier to manipulating the joining order. Finally, we showed that given a value division, determining whether some subcoalition has an incentive to break away (in which case we say the division is not in the *core*) is NP-complete. So, computational complexity serves to increase the stability of the coalition. These results yield a positive picture, where even fairly complex value divisions can be computed quickly, even distributed value division schemes are not too vulnerable to manipulation, and economic instability of the coalition is less of a worry.

For future research, an immediate extension would be to study the questions of this paper in settings where utility transfer is not always possible. More interestingly, is it

possible to design new value division schemes that are particularly hard to manipulate, for example, PSPACE-hard or average-case complete? Also, could one construct stability concepts that take into account the complexity of finding a beneficial deviation?

Acknowledgements

This material is based upon work supported by the National Science Foundation under CAREER Award IRI-9703122, Grant IIS-9800994, ITR IIS-0081246, and ITR IIS-0121678.

References

Aumann, R. 1959. Acceptable points in general cooperative n-person games. volume IV of *Contributions to the Theory of Games*. Princeton University Press.

Bartholdi, III, J. J., and Orlin, J. B. 1991. Single transferable vote resists strategic voting. *Social Choice and Welfare* 8(4):341–354.

Bartholdi, III, J. J.; Tovey, C. A.; and Trick, M. A. 1989. The computational difficulty of manipulating an election. *Social Choice and Welfare* 6(3):227–241.

Bartholdi, III, J. J.; Tovey, C. A.; and Trick, M. A. 1992. How hard is it to control an election? *Math. Comput. Modelling* 16(8-9):27–40. Formal theories of politics, II.

Bernheim, B. D.; Peleg, B.; and Whinston, M. D. 1987. Coalition-proof Nash equilibria: I concepts. *Journal of Economic Theory* 42(1):1–12.

Charnes, A., and Kortanek, K. O. 1966. On balanced sets, cores, and linear programming. Technical Report 12, Cornell Univ., Dept. of Industrial Eng. and Operations Res., Ithaca, NY.

Chatterjee, K.; Dutta, B.; Ray, D.; and Sengupta, K. 1993. A noncooperative theory of coalitional bargaining. *Review of Economic Studies* 60:463–477.

Conitzer, V., and Sandholm, T. 2002. Complexity of manipulating elections with few candidates. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, 314–319.

Conitzer, V., and Sandholm, T. 2003a. Complexity of determining nonemptiness of the core. In *Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence (IJCAI)*.

Conitzer, V., and Sandholm, T. 2003b. Universal voting protocol tweaks to make manipulation hard. In *Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence (IJCAI)*.

Evans, R. 1997. Coalitional bargaining with competition to make offers. *Games and Economic Behavior* 19:211–220.

Gillies, D. 1953. *Some theorems on n-person games*. Ph.D. Dissertation, Princeton University, Department of Mathematics.

Goemans, M., and Skutella, M. 2004. Cooperative facility location games. *Journal of Algorithms* 50:194–214. Early version: SODA 2000, 76–85.

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

Ketchpel, S. 1994. Forming coalitions in the face of uncertain rewards. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, 414–419.

Markakis, E., and Saberi, A. 2003. On the core of the multicommodity flow game. In *Proceedings of the ACM Conference on Electronic Commerce (ACM-EC)*, 93–97.

Mas-Colell, A.; Whinston, M.; and Green, J. R. 1995. *Microeconomic Theory*. Oxford University Press.

Milgrom, P., and Roberts, J. 1996. Coalition-proofness and correlation with arbitrary communication possibilities. *Games and Economic Behavior* 17:113–128.

Moreno, D., and Wooders, J. 1996. Coalition-proof equilibrium. *Games and Economic Behavior* 17:80–112.

Okada, A. 1996. A noncooperative coalitional bargaining game with random proposers. *Games and Economic Behavior* 16:97–108.

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

Papadimitriou, C. H. 1995. *Computational Complexity*. Addison-Wesley.

Ray, I. 1996. Coalition-proof correlated equilibrium: A definition. *Games and Economic Behavior* 17:56–79.

Sandholm, T., and Lesser, V. R. 1997. Coalitions among computationally bounded agents. *Artificial Intelligence* 94(1):99–137. Early version appeared at the International Joint Conference on Artificial Intelligence (IJCAI), pages 662–669, 1995.

Shapley, L. S. 1953. A value for n-person games. In Kuhn, H. W., and Tucker, A. W., eds., *Contributions to the Theory of Games*, volume 2 of *Annals of Mathematics Studies, 28*. Princeton University Press. 307–317.

Shapley, L. S. 1967. On balanced sets and cores. *Naval Research Logistics Quarterly* 14:453–460.

Shapley, L. S. 1971. Cores of convex games. *International Journal of Game Theory*.

Shehory, O., and Kraus, S. 1996. A kernel-oriented model for coalition-formation in general environments: Implemetation and results. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, 134–140.

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

van der Linden, W. J., and Verbeek, A. 1985. Coalition formation: A game-theoretic approach. In Wilke, H. A. M., ed., *Coalition Formation*, volume 24 of *Advances in Psychology*. North Holland.

von Neumann, J., and Morgenstein, O. 1947. *Theory of games and economic behavior*. Princeton University Press.

Zlotkin, G., and Rosenschein, J. S. 1994. Coalition, cryptography and stability: Mechanisms for coalition formation in task oriented domains. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, 432–437.